

Mémoire

Auteur : Wachtelaer, Thibault

Promoteur(s) : Damanet, François

Faculté : Faculté des Sciences

Diplôme : Master en sciences physiques, à finalité approfondie

Année académique : 2024-2025

URI/URL : <http://hdl.handle.net/2268.2/22984>

Avertissement à l'attention des usagers :

Tous les documents placés en accès ouvert sur le site le site MatheO sont protégés par le droit d'auteur. Conformément aux principes énoncés par la "Budapest Open Access Initiative"(BOAI, 2002), l'utilisateur du site peut lire, télécharger, copier, transmettre, imprimer, chercher ou faire un lien vers le texte intégral de ces documents, les disséquer pour les indexer, s'en servir de données pour un logiciel, ou s'en servir à toute autre fin légale (ou prévue par la réglementation relative au droit d'auteur). Toute utilisation du document à des fins commerciales est strictement interdite.

Par ailleurs, l'utilisateur s'engage à respecter les droits moraux de l'auteur, principalement le droit à l'intégrité de l'oeuvre et le droit de paternité et ce dans toute utilisation que l'utilisateur entreprend. Ainsi, à titre d'exemple, lorsqu'il reproduira un document par extrait ou dans son intégralité, l'utilisateur citera de manière complète les sources telles que mentionnées ci-dessus. Toute utilisation non explicitement autorisée ci-avant (telle que par exemple, la modification du document ou son résumé) nécessite l'autorisation préalable et expresse des auteurs ou de leurs ayants droit.



LIÈGE université
Sciences

Faculty of Sciences - Department of Physics

A quantum algorithm for finding the steady states of non-Markovian open quantum systems

WACHTELAER Thibault

Supervisor:

Dr. François Damanet

Reading committee:

Dr. Éric OPSOMER

Prof. Thierry BASTIN

Prof. Ngoc Duy NGUYEN

Prof. Christophe GEUZAINÉ

Thesis presented in the partial fulfillment of the requirements for
the degree of Master of Science in Physics

Academic year 2024-2025

Acknowledgments

As I finalize this master's thesis, I extend my deepest gratitude to all those without whom this work would not have been possible.

First and foremost, I express my profound appreciation to my supervisor, Dr. François Damanet, for his invaluable guidance, unwavering encouragement throughout this year, and the considerable time he devoted to this project despite his demanding schedule. I am particularly grateful for the thesis topic he proposed, bridging open quantum systems and quantum computation, which made this research exceptionally rewarding.

I thank Baptiste Debecker for insightful discussions and advice during the early stages of this work. My sincere thanks also go to the members of the reading committee, Dr. Eric Opsomer, Prof. Thierry Bastin, Prof. Ngoc Duy Nguyen, and Prof. Christophe Geuzaine, for graciously accepting this role. I hope my efforts to make this thesis engaging and accessible are evident throughout.

To my family: My deepest thanks to my sister and her partner for patiently listening to my frustrations amidst their own challenges. I wish them tranquility and joy with my newborn niece (now six days old, whom I've yet to meet due to final thesis commitments, for which I do not thank her for being born at such a time). I thank my mother and brother for enduring my often-monologic tirades about this work. Finally, I thank my father for... well, his restraint.

Above all, I owe my deepest gratitude to Fanny for her unwavering love, support, and patience during this journey.

Contents

Acknowledgements	1
Introduction	1
1 Open quantum systems	6
1.1 Density operator formalism	7
1.1.1 Why ket vectors formalism is insufficient	7
1.1.2 Why should we use projectors ?	9
1.1.3 Definition and properties of the density operator	9
1.1.4 Quantum mechanics postulates in the density operator formalism	11
1.2 Dynamics of Markovian open quantum systems and steady state definition	14
1.2.1 The Markovian approximation and the GKSL master equation	15
1.2.2 Steady state definition	18
1.2.3 Choi–Jamiołkowski isomorphism	18
1.3 Finding the steady state of a simple open quantum system	20
1.4 Dynamics of non-Markovian open quantum systems	23
1.4.1 Markovian embedding	24
1.4.2 The HEOM method	28
1.4.3 Markovian embedding VS HEOM	30
1.5 Finding the steady state of the open quantum Rabi model using the HEOM method	33
1.6 Summary	36
2 Quantum information and computation	37
2.1 Fundamental concepts	37
2.1.1 Conventions and notations	37
2.1.2 Quantum circuits and frequently used quantum gates	39
2.2 Quantum Fourier Transform	42
2.2.1 Definitions and properties	42
2.2.2 Circuit implementation	45
2.2.3 Complexity analysis and further discussions	46
2.3 Quantum Phase Estimation	48
2.3.1 Overview	48
2.3.2 Circuit and detailed description	49
2.3.3 Further discussions	52
2.4 Suzuki-Trotter expansion	52
2.4.1 Theoretical background	53
2.4.2 Simple example	54
2.5 Summary	55

3	Quantum computation for open quantum systems	58
3.1	Ramusat and Savona (RS) algorithm	58
3.1.1	Overview	59
3.1.2	Preparation stage	61
3.1.3	Quantum phase estimation stage	62
3.1.4	Measurement stage	63
3.1.5	Complexity analysis	65
3.2	Direct application of the RS algorithm in the case of a simple open quantum system	67
3.2.1	Circuit simulations	68
3.2.2	Numerical predictions for the algorithm's outcome	69
3.2.3	Convergence analysis	71
3.3	Adapting the RS algorithm to non-Markovian systems using the HEOM method: the RS HEOM algorithm	73
3.3.1	Overview	73
3.3.2	Preparation stage	74
3.3.3	Quantum phase estimation stage	75
3.3.4	Measurement stage	78
3.3.5	Complexity analysis	79
3.4	Applications of the RS HEOM algorithm	80
3.4.1	Open quantum Rabi model	80
3.4.2	RS VS RS HEOM algorithm: open quantum Rabi model	83
3.4.3	RS VS RS HEOM algorithm: two-mode open quantum Rabi model	89
3.5	Summary	92
	Conclusion	92

Introduction

The last century has witnessed the birth of quantum mechanics, a new framework for physics where systems are described by *ket vectors* and evolve according to the renowned *Schrödinger equation* [1]. While this formalism has had its successes, it possesses a significant flaw: it is only valid in the case of *isolated quantum systems* [2]. While theoretically any system can be described as closed if one includes the entirety of the environment (i.e., the universe), such an approach is neither feasible numerically nor analytically. This realization led physicists to develop new formalisms, namely the *density operator* and *quantum master equation* formalism as part of the more general theory of *open quantum systems* [3]. These are quantum systems which are not isolated and interact with their environment. In fact, all quantum systems are open, as they cannot be truly isolated from their environment. Furthermore, even in the isolated framework, any measurement operation is intrinsically an interaction with an environment. This is the basis of the theory of *decoherence* [4], which explains how classical behavior emerges from the interactions between quantum systems and their environment.

In the case of open quantum systems weakly interacting with their environment, their dynamics depend usually only on their current states and not on their past states. Those systems have no memory and are said to be *Markovian*. They can accurately be described by the *Gorini-Kossakowski-Sudarshan-Lindblad* (GKSL) master equation [5], [6]. This equation eliminates the state of the environment from the dynamics, and the evolution is governed entirely by the state of the system. Despite the success of the GKSL equation for describing such systems, the Markovian hypothesis is, in fact, an approximation. Analogously, just as closed systems constitute an *idealized* framework where *decoherence* is neglected, Markovian systems simplify their dynamics by neglecting memory effects. These effects inherently stem from the interaction between a system and its environment: the system influences the state of the environment, which in turn may impact the state of the system. In many situations, memory effects cannot be disregarded, such as in quantum computing noise [7] or the protein folding problem in biochemistry [8]. Systems for which the Markovian approximation is invalid are termed *non-Markovian* systems.

The complexity of non-Markovian open quantum system dynamics has prompted chemists and physicists to develop new computational strategies. These can be loosely divided into two categories: *perturbative approaches* and *exact numerical methods*. Among the latter, *Markovian embedding techniques* are conceptually the simplest approach [9], [10]. The physical system is appended with *pseudo-modes* representing environmental degrees of freedom until the composite system becomes Markovian, at which point the GKSL master equation is applied. While conceptually straightforward, the computational cost rapidly becomes unmanageable for multi-mode systems. To maintain rigor while reducing numerical demands, the *hierarchical equations of motion* (HEOM) method, an *exact non-perturbative numerical* approach, was developed by Yoshitaka Tanimura and Ryogo Kubo in 1989 [11]. This technique introduces ancillary states that serve as memory kernels for the

system. Although HEOM offers significant computational advantages over Markovian embeddings, its cost can still become prohibitive for multi-mode systems.

The high computational cost of simulating open quantum systems is not inherent to non-Markovian approaches. Simulating the dynamics of 300 spins (i.e., two-level systems) without mean-field approximation requires a density matrix of size $2^{300} \times 2^{300}$ (2^{600} elements), a number larger than the number of atoms in the observable universe, which hinders simulations of their dynamics on classical computers. Such an exponential increase of the number of degrees of freedom to deal with as a function of the number of particles is a plague that significantly slows down the discovery of new useful materials (such as batteries, catalysts, photoelectric cells, ...) and drugs, as the quantum dynamics dictating the properties of even relatively small molecules containing a dozen of atoms are impossible to simulate exactly on classical computers.

A promising solution would be to employ *quantum computers*, which have been shown to perform specific computational tasks exponentially faster than classical computers. The concept of incorporating quantum properties into computers was first formulated by Paul Benioff in 1980 [12]. Soon after, Richard Feynman (1981) [13] and Yuri Manin (1980) [14], [15] independently suggested that quantum computers might efficiently simulate quantum systems. While introduced early in quantum computing’s history, the development of *quantum algorithms* for open quantum systems has expanded significantly only in recent years [16]–[24]. Furthermore, quantum algorithms for simulating the dynamics of non-Markovian systems have received limited attention [25]–[31]¹. Notably, Dan, Geva, and Batista [32] are the only researchers to have developed a quantum algorithm for simulating the dynamics of non-Markovian open quantum systems using the HEOM method.

The purpose and original contribution of this master’s thesis is to develop a quantum algorithm using the HEOM method for finding the *steady states* of non-Markovian open quantum systems. Finding the steady state of open quantum systems can be critically important for retrieving information about their dynamics and long-term behavior [33]. Furthermore, certain steady states may serve as resource states for quantum teleportation and quantum error correction [34]. To construct this algorithm, we draw significant inspiration from Ramusat and Savona’s work “A quantum algorithm for the direct estimation of the steady state of open quantum systems,” adapting their approach to non-Markovian systems using the HEOM generator.

The structure of this thesis is as follows: in Chapter 1, we introduce the reader to the field of open quantum systems and, in particular, the non-Markovian methods used in this work. In Chapter 2, the formalism for quantum information and computation will be briefly presented. Furthermore, the key *subroutines* used in the algorithm of this thesis will be extensively described. Finally, in Chapter 3, we will present the algorithm of Ramusat and Savona and explain how to adapt it to non-Markovian systems using the HEOM method. Additionally, we will benchmark our adaptation and prove that it provides a non-trivial advantage for multi-mode systems compared to the original algorithm used in the case of a Markovian embedding approach. The logic of this thesis is to provide the reader with the necessary tools to understand the algorithm developed in this thesis.

¹This source list is nearly exhaustive, furthermore, only three of these were published before the start of this master’s thesis.

Chapter 1

Open quantum systems

“Perhaps we should first define what an open system is. An open system is a system that is not closed.”

John Martin, first class of open quantum systems”

This chapter aims at introducing the reader to the field of open quantum systems. In particular, we will explain why the ket vector formalism is insufficient and introduce the density operator formalism. The quantum mechanics postulates for isolated systems will be restated using this formalism. We will then present the Gorini-Kossakowski-Sudarshan-Lindblad (GKSL) master equation, which describes the dynamics of Markovian open quantum systems. Afterwards, we will define what the steady state is. Following that, we will discuss how we can map the density matrix to a ket vector using the Choi–Jamiołkowski isomorphism, something that will be instrumental in the third chapter of this thesis. We then provide the reader with a simple example of an open quantum system and how to classically find its steady state. This example will also accompany us in the third chapter where we will apply the quantum algorithm of Ramusat and Savona to find the steady state as well. We will then proceed by presenting two methods to explore the dynamics of non-Markovian open quantum systems: the Markovian embedding approach and the HEOM method. The first method will serve as a benchmark for the other. Furthermore, we will also provide the reader with an example of a non-Markovian open quantum system, named the open quantum Rabi model. This example will also be reused in the later chapters in order to benchmark the quantum algorithm presented in this thesis. Most of this chapter originates from Orszag’s book *Quantum optics*[35], Breuer and Petruccione’s book *The theory of open quantum systems*[2], *Quantum Optics*’s book *Quantum Optics*[36] and Nielsen and Chuang’s book *Quantum Computation and Quantum Information*[15].

An *open quantum system* is a quantum system interacting with an *environment* or *bath* (i.e., not closed). This interaction leads to effects such as *decoherence*, *dissipation*, and *loss of quantum information*. Furthermore, open quantum systems experience *non-unitary evolution* unlike isolated quantum systems. Typically, an open quantum system S is associated with a *Hilbert space* \mathcal{H}_S , a *Hamiltonian* H_S , and its state is described by a *density operator* ρ_S . The system S is surrounded by an environment E associated with its own Hilbert space, Hamiltonian, and density operator. The total system $S + E$ contains an additional term in its own Hamiltonian, which is termed *interaction Hamiltonian*. These terms will be explained in more detail in the next sections, and the general

sketch of an open system is given in Fig. 1.1.

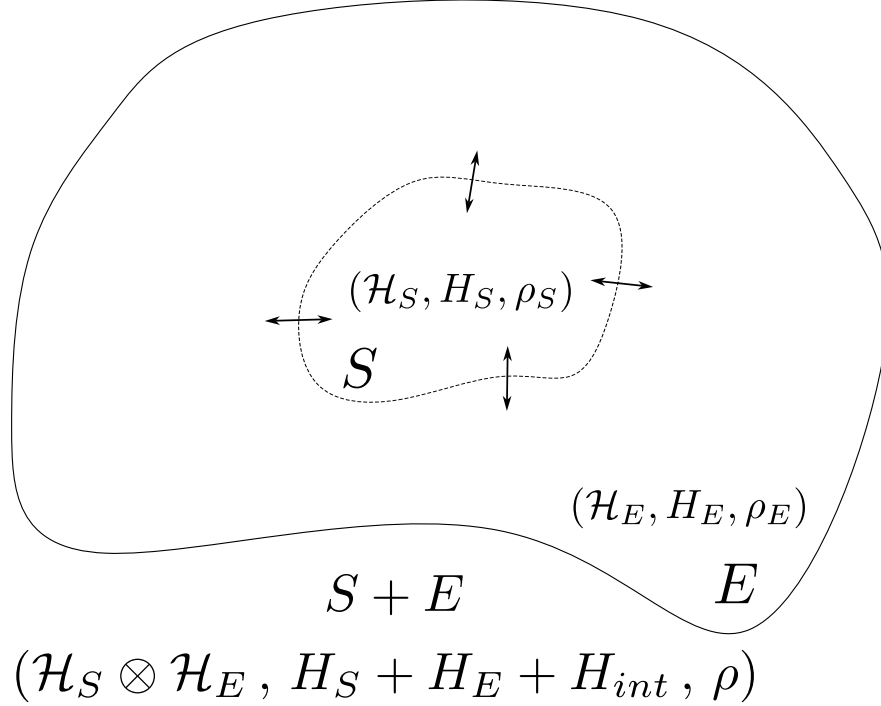


Figure 1.1: Sketch of a general open quantum system. To each subsystem is associated a density operator, Hamiltonian operator and Hilbert space. The Hamiltonian of the entire system composed of the physical system of interest + the environment, contains an additional term which is the interaction Hamiltonian.

1.1 Density operator formalism

1.1.1 Why ket vectors formalism is insufficient

In traditional quantum mechanics, the information concerning the state of a system is encoded in a *ket vector* $|\psi\rangle$ that belongs to the *Hilbert space* \mathcal{H} associated with the system. The Hilbert space is a *complete*¹ complex vector space equipped with an *inner product*. The ket vector evolves according to the *Schrödinger equation*

$$i\dot{|\psi\rangle} = H |\psi\rangle, \quad (1.1)$$

where H is the *Hamiltonian operator* of the system, which is potentially *time-dependent*. An *operator* is a *linear map* defined on the Hilbert space of the system. In this thesis, we chose to denote operators without the usual hat, i.e. $H \equiv \hat{H}$, and the nature of a quantum object will be explicitly stated if any ambiguity arises somewhere throughout this work. Furthermore, we also chose to work in natural units using $\hbar = c = 1$. The general solution of the Schrödinger equation in the case of a *time-independent* Hamiltonian is then given by

¹Complete means that every Cauchy sequence of the space converges within the space

$$|\psi(t)\rangle = e^{-iH(t-t_0)} |\psi(t_0)\rangle, \quad (1.2)$$

where the time dependence of the ket vector is explicitly written.

The two main problems that arise from this formalism are that first; the Schrödinger equation is only valid for *closed systems*² and thus, we need another equation to describe the evolution of open quantum systems. The second problem is that a ket vector can only describe what we call *pure states* which are ideal states representing the *maximal information* we can have about a quantum system.

To illustrate this, let us consider a system for which its current state is only partially known; for example, we can consider a two-level system which was just measured with a really defective measurement device. We consider that the two states of the system are $|0\rangle \equiv \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle \equiv \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ and we assume that we have measured $|0\rangle$ with a measurement device that is only 95% precise, which can be due to poor quality with the electronics for instance. We know that the current state of the system is either $|0\rangle$ with 95% certainty or $|1\rangle$ with 5% certainty. However we should emphasize that the wave function of the system has *collapsed* into one of the two states due to the measurement. The current state of the system is what we call a *mixed state*, which cannot be described using a single ket vector. At first glance, one could naively think that this situation could be described using a ket vector $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ with $|\alpha|^2 = 0.95$ and $|\beta|^2 = 0.05$. The problem with this notation, other than the fact that the relative phase between α and β is not known, is that it intrinsically corresponds to a *coherent superposition*. To explain this issue, let us consider that we now measure this state using the Pauli operator

$$\sigma_x \equiv X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad (1.3)$$

which corresponds to a *bit flip* because it transforms $|0\rangle$ into $|1\rangle$ and vice versa. In the case where our system collapsed into $|0\rangle$ we have

$$\langle 0 | \sigma_x | 0 \rangle = \langle 0 | 1 \rangle = 0,$$

and we find the same result using $|1\rangle$. However, in the case of $\alpha|0\rangle + \beta|1\rangle$, we find

$$\langle \psi | \sigma_x | \psi \rangle = (\alpha^* \langle 0 | + \beta^* \langle 1 |) (\alpha | 1 \rangle + \beta | 0 \rangle) = \alpha^* \beta + \alpha \beta^* = 2 \operatorname{Re}[\alpha^* \beta],$$

which in general, is non-zero unless α^* and β have a relative phase factor of $\pi/2$. However, in this case, we could also measure using the Pauli operator

$$\sigma_y \equiv Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad (1.4)$$

which corresponds to both a *bit flip* and a *sign flip* (for the phase). Considering once again that we are in the case where our system has collapsed into the state $|0\rangle$, we have

$$\langle 0 | \sigma_y | 0 \rangle = i \langle 0 | 1 \rangle = 0, \quad (1.5)$$

²It is purely unitary and as such, it cannot describe non-unitary evolutions.

and $|1\rangle$ also yields 0. Conversely, if we measure $\alpha|0\rangle + \beta|1\rangle$ we find

$$\langle\psi|\sigma_y|\psi\rangle = (\alpha^* \langle 0| + \beta^* \langle 1|) (-i\beta|0\rangle + i\alpha|1\rangle) = -i\alpha^*\beta + i\alpha\beta^* = 2\text{Re}[i\alpha\beta^*].$$

The last equality is also non-zero in general, unless, α and β^* have no relative factor or a relative factor of π . This means that there is no α and β satisfying the probabilities conditions and for which $\langle\sigma_x\rangle$ and $\langle\sigma_y\rangle$ are both equal to 0. As such, there exists no ket vectors that can accurately describe the state obtained after using a defective measurement device. This is what we mean when the state is in a coherent superposition, the system is *simultaneously* in multiple states and the relative phase factors between the states matter.

1.1.2 Why should we use projectors ?

In order to represent a mixed state, we need to give up on trying to use ket vectors and instead we must use operators and more precisely, projectors. Assuming our state is now represented by an operator ρ , we have

$$\begin{aligned}\rho &= p_0|0\rangle\langle 0| + p_1|1\rangle\langle 1| \\ &\equiv \begin{pmatrix} p_0 & 0 \\ 0 & p_1 \end{pmatrix}\end{aligned}$$

where p_i is the probability of being in the state $|i\rangle$ (which are 0.95 and 0.05 in our example) and $|i\rangle\langle i|$ is the projector associated with the state $|i\rangle$. The second line is the matrix representation of the operator in the eigenbasis $|0\rangle$ and $|1\rangle$. Defining the measurement operation for a measurement operator O , by $\text{Tr}[O \cdot]$ and using this definition to measure ρ with σ_x , we have

$$\begin{aligned}\langle\sigma_x\rangle &= \text{Tr}\left[\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} p_0 & 0 \\ 0 & p_1 \end{pmatrix}\right] \\ &= \text{Tr}\left[\begin{pmatrix} 0 & p_1 \\ p_0 & 0 \end{pmatrix}\right] \\ &= 0,\end{aligned}$$

which is exactly what we are supposed to have. If we measure using the Pauli operator

$$\sigma_z \equiv Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (1.6)$$

for which $|0\rangle$ and $|1\rangle$ are the *eigenstates* and which corresponds to a *sign flip* (for the phase). Since the expected value is interpreted as the averaged measurement outcome over an infinite number of measurements or *shots*, we should obtain $p_0 - p_1$ which is exactly what we would get by computing $\text{Tr}[\sigma_z\rho]$. The concept of using projectors to describe the state of a quantum system is what we call *density operator formalism* and ρ is precisely named *density operator*.

1.1.3 Definition and properties of the density operator

In this subsection, we will define the general density operator for a mixed state and then explain some of its main properties.

Density operator for mixed states

$$\rho = \sum_{i=1}^n p_i |\psi_i\rangle \langle \psi_i| \quad \text{with} \quad \sum_{i=1}^n p_i = 1 \quad \text{and} \quad p_i \geq 0, \quad (1.7)$$

where the states $|\psi_i\rangle$ are not necessarily orthogonal to one another, nor are they eigenstates of a specific basis; our only condition is that they can be written as ket vectors of \mathcal{H} . The p_i represent the probabilities of being in the states $|\psi_i\rangle$, respectively.

We should emphasize that while the density operator is constructed via projectors, it is not a projector in general. It is only the case when the state is *pure*, a notion that will be explained later.

Properties of the density operator

1. ρ is Hermitian

$$\rho = \rho^\dagger. \quad (1.8)$$

2. ρ is positive semi-definite

$$\langle \phi | \rho | \phi \rangle \geq 0 \quad \forall |\phi\rangle \in \mathcal{H}. \quad (1.9)$$

3. ρ is trace 1

$$\text{Tr}[\rho] = 1. \quad (1.10)$$

4. The trace of ρA corresponds to the weighted sum of the expected values of A with respect to each state $|\psi_i\rangle$, weighted by their corresponding probabilities

$$\text{Tr}[\rho A] = \sum_{i=1}^n p_i \langle \psi_i | A | \psi_i \rangle \quad \text{for any operator } A. \quad (1.11)$$

5. The trace of ρ^2 belongs to the interval $[0, 1]$

$$0 \leq \text{Tr}[\rho^2] \leq 1. \quad (1.12)$$

This expression can be used as a measurement for the *purity* of a state and when the equality $\text{Tr}[\rho^2] = 1$, there exists a pure state $|\psi\rangle$ such that ρ can be written as $|\psi\rangle \langle \psi|$.

Proofs. The first property comes from the fact that the p_i are real and that computing the Hermitian conjugate of two objects AB is equivalent to computing $B^\dagger A^\dagger$ and thus

$$\rho^\dagger = \left(\sum_{i=1}^n p_i |\psi_i\rangle \langle \psi_i| \right)^\dagger = \sum_{i=1}^n p_i |\psi_i\rangle \langle \psi_i| = \rho,$$

since $|\psi_i\rangle^\dagger = \langle \psi_i|$. As a consequence of that, we have that ρ is unitarily diagonalizable, i.e., there exists a unitary matrix U such that $\rho = U^\dagger \Sigma U$ with Σ a diagonal matrix. Physically, this means that we can always find a representation of ρ using $|\psi_i\rangle$ that form an orthogonal basis.

The second property can be straightforwardly obtained by developing

$$\langle \phi | \rho | \phi \rangle = \sum_{i=1}^n p_i \langle \phi | \psi_i \rangle \langle \psi_i | \phi \rangle = \sum_{i=1}^n p_i \underbrace{|\langle \phi | \psi_i \rangle|^2}_{\geq 0} \geq 0,$$

and the equality specifically occurs when the state $|\phi\rangle$ has no overlap with any of the state $|\psi_i\rangle$. The third property is due to the linearity of the trace, we have that

$$\text{Tr}[\rho] = \text{Tr} \left[\sum_{i=1}^n p_i |\psi_i\rangle \langle \psi_i| \right] = \sum_{i=1}^n p_i \text{Tr} [|\psi_i\rangle \langle \psi_i|] = \sum_i p_i = 1,$$

where we used the fact that $\text{Tr} [|\psi_i\rangle \langle \psi_i|] = 1, \forall |\psi_i\rangle \in \mathcal{H}$ which can be proven by choosing an orthogonal basis $\{|\phi_j\rangle\}$ of \mathcal{H} , with $j = 1, 2, \dots, d$ the dimension of the Hilbert space. We can then compute the trace by assuming that $|\psi_i\rangle$ admits a decomposition $\sum_k^d c_k |\phi_k\rangle$ into this basis with $\sum_k^d |c_k|^2 = 1$, which yields

$$\text{Tr} [|\psi_i\rangle \langle \psi_i|] = \sum_{j=1}^d \langle \phi_j | \left(|\psi_i\rangle \langle \psi_i| \right) | \phi_j \rangle = \sum_{j=1}^d \sum_{k=1}^d \sum_{k'=1}^d c_k c_{k'}^* \underbrace{\langle \phi_j | \phi_k \rangle}_{\delta_{jk}} \underbrace{\langle \phi_{k'} | \phi_j \rangle}_{\delta_{k'j}} = \sum_{k=1}^d |c_k|^2 = 1,$$

and which concludes this proof.

The fourth property is also obtained using our favorite basis of \mathcal{H} , as we have

$$\text{Tr} [|\psi_i\rangle \langle \psi_i| A] = \sum_{k=1}^d \underbrace{\langle \phi_k | \psi_i \rangle}_{c_k} \langle \psi_i | A | \phi_k \rangle = \langle \psi_i | A \sum_{k=1}^d c_k |\phi_k\rangle = \langle \psi_i | A | \psi_i \rangle,$$

where we used the definition of the trace then the definition of the c_k coefficients, then the linearity of the operator to make the decomposition of $|\psi_i\rangle$ appear explicitly. We can then use the linearity of the trace to develop

$$\text{Tr}[\rho A] = \sum_{i=1}^n p_i \text{Tr} [|\psi_i\rangle \langle \psi_i| A] = \sum_{i=1}^n p_i \langle \psi_i | A | \psi_i \rangle,$$

which concludes this proof.

For the last property, we can use the fourth property along with the last equality in the proof of the second property. This yields

$$\text{Tr}[\rho^2] = \sum_{i=1}^n p_i \langle \psi_i | \rho | \psi_i \rangle = \sum_{i=1}^n \sum_{j=1}^n \underbrace{p_i p_j |\langle \psi_i | \psi_j \rangle|^2}_{\geq 0} \leq \sum_{i=1}^n \sum_{j=1}^n p_i p_j = 1.$$

In particular, the equality can only occur if $|\langle \psi_i | \psi_j \rangle|^2$ is always equal to 1, which can only happen if all ket vectors $|\psi_i\rangle$ are the exact same. This means that we could rewrite ρ as $|\psi\rangle \langle \psi|$, and thus, ρ would be the density operator of a pure state. \square

1.1.4 Quantum mechanics postulates in the density operator formalism

The purpose of this section is to reformulate the well-known postulates of quantum mechanics for isolated systems using the density operator formalism. The postulates using the ket vector formalism are taken from Cohen-Tannoudji, Diu, and Laloë's book, *Quantum mechanics, volume 1* [37] while the postulates in density operator formalism are taken from Le Bellac's book *Quantum Physics* [38]. Furthermore, these are combined using the postulates for POVM (*Positive Operator Valued Measures*) taken from Orszag's book *Quantum optics* [35].

First postulate

At any time t , the state of an isolated quantum system associated with a Hilbert space \mathcal{H} , also called *state space*, is described through a density operator $\rho(t)$ acting on the state space, which is positive definite and trace one.

In the case of an isolated system, $\rho(t)$ can always be constructed using a single ket vector $|\psi(t)\rangle$.

Second postulate

For an isolated quantum system, the time evolution of the density operator $\rho(t)$ is described by a unitary transformation $U(t, t_0)$:

$$\rho(t) = U(t, t_0)\rho(t_0)U^\dagger(t, t_0), \quad (1.13)$$

where $U(t, t_0) = e^{-iH(t-t_0)}$. The dynamics are equivalently governed by the Liouville-von Neumann equation

$$\dot{\rho}(t) = -i[H, \rho(t)], \quad (1.14)$$

where H is the Hamiltonian Hermitian operator of the isolated system.

The Liouville-Von Neumann equation is strictly equivalent to the Schrödinger equation (1.1), moreover, (1.13) is also equivalent to the general solution of the Schrödinger equation (1.2). These can be proven by starting from the density operator for a pure state and computing its time derivative

$$\begin{aligned} \dot{\rho}(t) &= |\dot{\psi}(t)\rangle \langle \psi(t)| + |\psi(t)\rangle \langle \dot{\psi}(t)| \\ &= -iH |\psi(t)\rangle \langle \psi(t)| + i |\psi(t)\rangle \langle \psi(t)| H \\ &= -i[H, \rho(t)], \end{aligned}$$

where we simply replaced the time derivative of the bra and ket vectors using the Schrödinger equation (1.1) and then rewrote the expression using the commutator³. The other equivalence is effortlessly proven by replacing the bra and ket vectors of $\rho(t)$ using the expression (1.2). We have

$$\begin{aligned} \rho(t) &= |\psi(t)\rangle \langle \psi(t)| \\ &= e^{-iH(t-t_0)} |\psi(t_0)\rangle \langle \psi(t_0)| e^{iH(t-t_0)} \\ &= U(t, t_0)\rho(t_0)U^\dagger(t, t_0). \end{aligned}$$

³The commutator $[A, B]$ is defined as $AB - BA$.

Third postulate

Quantum measurements are described by a collection of M_j called *measurement operators* acting on the state space. The index j refers to the measurement outcome j . Furthermore, the measurement operators satisfy the *completeness relation*

$$\sum_j M_j^\dagger M_j = \mathbb{1}. \quad (1.15)$$

The probability p_j of measuring the outcome j is given by the *Hilbert-Schmidt inner product* between the density operator and the positive semi-definite operator $M_j^\dagger M_j$:

$$p_j = \text{Tr} \left[\rho M_j^\dagger M_j \right], \quad (1.16)$$

which is also referred to as the *Born rule*. If the outcome j was measured, the state of the system immediately collapses into the state

$$\rho_j = \frac{M_j \rho M_j^\dagger}{\text{Tr} \left[\rho M_j^\dagger M_j \right]}. \quad (1.17)$$

If a measurement is performed without recording the result, the state of the system is given by the following mixed state

$$\bar{\rho} = \sum_j p_j \rho_j = \sum_j M_j \rho M_j^\dagger. \quad (1.18)$$

A mixed state is also referred to as a *statistical mixture*.

The collection of positive semi-definite operators $\{M_j^\dagger M_j\}$, is precisely what we call a POVM. The $M_j^\dagger M_j$ are called *POVM elements* and are always Hermitian and positive. When the measurement operators M_j are orthogonal projectors ($M_j^2 = M_j$ and $M_j M_k = \delta_{jk} M_j$), the POVM reduces to the standard von Neumann projective measurement framework. POVMs generalize this by allowing non-orthogonal, non-commuting operators and can describe measurements with more outcomes than the dimension of the Hilbert space. The Hilbert-Schmidt inner product of two operators A and B acting on the same state space is defined by

$$\langle A, B \rangle_{\text{HS}} = \text{Tr} \left[A^\dagger B \right]. \quad (1.19)$$

One might think that in the case where a measurement is performed without recording the result, the resulting mixed state is in contradiction with the isolated system quantum mechanics framework. However, in order to perform the measurement, the system must inherently interact with its environment (the measurement device in this scenario) and is thus no longer isolated. This phenomenon is what we call *decoherence*, where the emergence of classicality from quantum mechanics is caused by interactions with the environment. In fact, all non-unitary evolutions of a physical system are due to its interactions with the environment and if we consider the enlarged system constituted of the physical system and the whole environment, the evolution remains unitary.

Fourth postulate

The state space \mathcal{H} of a composite quantum system is the *tensor product* of the Hilbert spaces of its individual subsystems. For N subsystems, this is expressed as

$$\mathcal{H} = \bigotimes_{j=1}^N \mathcal{H}_j \equiv \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \cdots \otimes \mathcal{H}_N. \quad (1.20)$$

If each subsystem j is *independently prepared* in the state ρ_j (with no correlations or entanglement between subsystems), the density operator of the composite system is the *product state*

$$\rho = \bigotimes_{j=1}^N \rho_j \equiv \rho_1 \otimes \rho_2 \otimes \cdots \otimes \rho_N. \quad (1.21)$$

This formalism assumes the subsystems are uncorrelated. For entangled or correlated states, ρ cannot be written as a simple tensor product and requires a more general density matrix.

As mentioned above, not all states of a composite physical system can be represented this way. In particular, *correlated states* are states that can be represented as statistical mixtures of product states

$$\rho_{\text{correlated}} = \sum_{i=1}^n p_i \bigotimes_{j=1}^N \rho_j^i, \quad (1.22)$$

which can be seen as some sort of mixed state for composite systems. Another kind of states are *entangled states*, which cannot be written by separating each subsystem with a tensor product. For example, the *Bell state*

$$\begin{aligned} \rho &= \frac{1}{\sqrt{2}} \left(|00\rangle + |11\rangle \right) \frac{1}{\sqrt{2}} \left(\langle 00| + \langle 11| \right) \\ &= \frac{1}{2} \left(|00\rangle \langle 00| + |00\rangle \langle 11| + |11\rangle \langle 00| + |11\rangle \langle 11| \right), \end{aligned} \quad (1.23)$$

cannot be factorized into two separate density operators or expressed as a correlated state⁴.

1.2 Dynamics of Markovian open quantum systems and steady state definition

In this section, we will introduce the concept of master equations for open quantum systems and the corresponding *Gorini-Kossakowski-Sudarshan-Lindblad* (GKSL) master equation along with its key hypothesis, the Markovian approximation.

The term *master equation* was first proposed in 1940 by Nordsieck, Lamb, and Uhlenbeck [39] to describe a governing equation for the time evolution of probabilities in stochastic systems. Their insight was that this equation could serve as a universal framework from which other key equations,

⁴This is not so easy to prove since there is no limit to the number of product states in a correlated state. However, one way to proceed is to prove that the state can be written as a pure state, which is almost done in the first line of (1.23), and then to argue that since it is a pure state, the total density operator cannot be written as a correlated state. Then, the only thing left is to prove that it cannot be factorized as $\rho = \rho_1 \otimes \rho_2$, which is already simpler to do.

such as those for moments (mean, variance, etc.) and equilibrium distributions (e.g., Boltzmann, Gibbs), could be derived. For example, summing over the master equation’s probabilities yields mean-field rate equations, while assuming detailed balance recovers equilibrium statistics. The idea was then generalized by *quantum master equations* in the context of open quantum systems, by including non-diagonal elements into the dynamics. A classical master equation is a system of differential equations for the set of probabilities; this corresponds to only the diagonal elements of the density matrix. Nowadays, the name “master equation” is used to indicate the unification of the deterministic (unitary) and stochastic (dissipative) dynamics under a single equation.

1.2.1 The Markovian approximation and the GKSL master equation

The entire derivation of the GKSL master equation will not be done in this work⁵ however, the general concepts will still be explained in a more superficial manner. The interested reader can read the papers of Gorini, Kossakowski, and Sudarshan [5] and Lindblad [6] alternatively, source [40] provides a pedagogical derivation for the unversed reader. The starting point is to consider the Liouville-von Neumann equation for the enlarged system containing both the environment and the system, which is thus assumed to be a closed system. Then we use the partial trace⁶ to eliminate the environment degrees of freedom. Afterwards, we use the interaction picture to obtain an integro-differential equation for $\dot{\rho}_S(t)$ depending only on the system density operator and the interaction Hamiltonian. This equation can then be restricted to second order by using our two major approximations.

The first one, which is called the *Born approximation*, involves considering a *weak coupling* between the system and the environment, which allows us to rewrite the total density operator as

$$\rho(t) \approx \rho_S(t) \otimes \rho_E. \quad (1.24)$$

From an intuitive point of view, this is similar to considering that the Hilbert space of the environment has so many degrees of freedom that it is barely affected by its interaction with the system. To give a classical analogy, this is similar to imagining a gas bottle in space; if we open the bottle, then no matter how much gas leaks, the space could still be considered empty. The gas would thus still flow out at a rate that only depends on how much gas is left inside, the size of the opening, and the nature of the gas, i.e., the current state of the bottle. Now of course, space is not completely empty and the leaked gas particles are wandering in space and affect the state of the environment (space). However, this change is negligible on a macroscopic scale and since this particular interaction depends on the macroscopic parameters of the environment (pressure, volume, temperature), its impact on the interaction is negligible. This is actually the crux of the Born approximation.

The *Markovian approximation* is our second major approximation and central to this thesis. It simplifies the dynamics by eliminating memory effects, transforming the integro-differential equation (derived in the interaction picture) into a *time-local* master equation. While often applied in tandem with the Born approximation, these are distinct assumptions. The Markovian approximation is equivalent to assuming that any perturbation in the environment caused by the system decays on a timescale shorter than any other meaningful system timescale. In contrast, the Born approximation considers that the interaction is so weak that it cannot significantly alter the state of the environment in a way that feeds back into the system dynamics. There is significant overlap

⁵This year, the reading committee will be spared from reading yet again a GKSL derivation.

⁶The partial trace consists in the operation $\text{Tr}_E[\rho] = \sum_i^{N_E} \langle \phi_i^E | \rho | \phi_i^E \rangle$, where the $\{|\phi_i^E\rangle\}$ form an orthogonal basis of \mathcal{H}_E , the Hilbert space of the environment.

between the two approximations, as a weak interaction inherently generates smaller perturbations in the environment, which thereby decay more rapidly. Conversely, faster decay of these perturbations reduces their influence on the system-environment interaction dynamics.

To continue our earlier classical analogy, imagine the gas bottle is now in a small compartment floating in space. This compartment has an opening that connects the inside with the outside; the size of this opening is directly related to the Markovian approximation. A greater size will cause the gas particles that escaped the gas bottle to leak out faster. The Born approximation is related to both the opening of the gas bottle and the size of the compartment. A smaller opening will imply that fewer particles escape within a given time period, while reducing the probability of an escaped particle returning near this opening. However, if the compartment opening is too small, particles will accumulate to the point where neglecting the probability of a particle returning to the bottle opening becomes invalid. Similarly, if the gas bottle opening is too large, the number of particles in the compartment will no longer decay quickly enough. These observations allow us to combine these two approximations into one, called the *Born-Markov approximation*. If we denote g as the *coupling strength* and κ as the *dissipation rate*, the Born-Markov approximation is equivalent to assuming that the dimensionless ⁷ ratio $g/\kappa \ll 1$. The classical gas bottle analogy is illustrated in Fig 1.2.

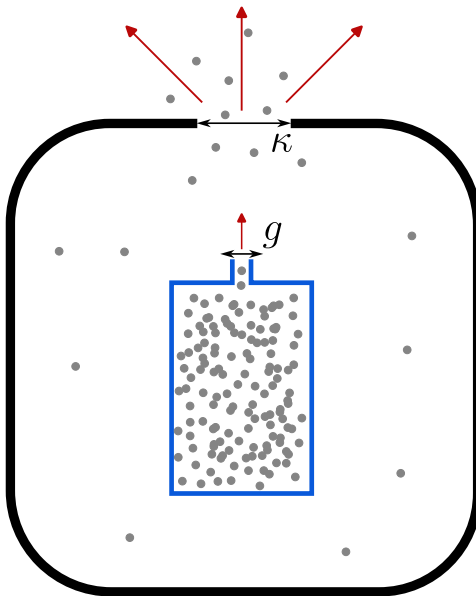


Figure 1.2: Sketch of the classical gas bottle analogy for the Born-Markov approximation. The opening of the gas bottle is too narrow to let particles leak at a significant rate. Conversely the opening of the compartment is excessively wide, leading to leakage rate too high for the particles to linger around. Here g can be interpreted as the particle exchange rate between the bottle and the compartment and κ is interpreted as the particle loss rate into space. In the case of the Born-Markov approximation, we have $g/\kappa \ll 1$.

The point of explaining the first part of the GKSL master equation derivation was to introduce the Markovian approximation and where it intervenes. The rest of the derivation is irrelevant to understand the remainder of this work and will thus be omitted. The following box introduces the general form of the GKSL master equation.

⁷With $\hbar = 1$, energies and frequencies are dimensionally equivalent.

Gorini-Kossakowski-Sudarshan-Lindblad master equation

The general GKSL master equation is given by

$$\dot{\rho} = -i[H, \rho] + \sum_{i,j} h_{ij} \left(A_i \rho A_j^\dagger - \frac{1}{2} \{A_j^\dagger A_i, \rho\} \right), \quad (1.25)$$

where ρ abusively designates the density operator of the system ρ_S and not the total density operator. The time-dependence of the density operator was also removed for readability and H designates $H_S + H_{LS}$, with H_{LS} the *Lamb shift Hamiltonian*^a. The second term is also referred to as the *dissipator*

$$\mathcal{D}(\cdot) = \sum_{i,j} h_{ij} \left(A_i \cdot A_j^\dagger - \frac{1}{2} \{A_j^\dagger A_i, \cdot\} \right), \quad (1.26)$$

where $\{A_k\}$ are arbitrary operators and h is a positive semi-definite matrix. In this general representation, there is no limit to the number of operators A_k , however it is always possible to unitarily diagonalize the h matrix such that

$$u^\dagger h u = \begin{pmatrix} \gamma_1 & 0 & \dots & 0 \\ 0 & \gamma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \gamma_{N_S^2-1} \end{pmatrix}, \quad (1.27)$$

with $\gamma_j \geq 0$ for $j = 1, 2, \dots, N_S^2 - 1$. And the master equation can then always be rewritten in the Lindblad form

$$\dot{\rho} = -i[H, \rho] + \sum_j^{N_S^2-1} \gamma_j \left(L_j \rho L_j^\dagger - \frac{1}{2} \{L_j^\dagger L_j, \rho\} \right), \quad (1.28)$$

with

$$L_j = \sum_k u_{kj} A_k, \quad (1.29)$$

the *Lindblad operators*, which form an orthonormal operator basis, and are now interpreted as *jump operators* and the γ_j are interpreted as *damping rates*. And when the γ_j are equal to zero, the equation reduces to the Liouville-von Neumann equation. The dissipator (in diagonalized form) and the Hamiltonian commutator, together form the *superoperator*^b

$$\mathcal{L}(\cdot) = -i[H, \cdot] + \mathcal{D}(\cdot), \quad (1.30)$$

which is called *Liouvillian* or equivalently *Lindbladian*.

^aThe Lamb shift Hamiltonian represents a correction to the system's energy levels due to its interaction with the environment. This renormalization mechanism is analogous to mass and charge corrections in quantum field theory but arises here at second order in perturbation theory within the Born-Markov framework. In practice, this contribution is often not considered, simply because the energy levels measured in experiments are always the corrected levels, while, for theorists, the energy levels are often parameters and not absolute physical constants. The interested reader may find more information about renormalization theory in this source [41].

^bIt is called a superoperator because it acts linearly on operators the same way operators act linearly on vectors. Note that the dissipator is also a superoperator.

The GKSL master equation is a cornerstone of open quantum system theory, providing a powerful

framework to model noise, decoherence, and dissipation. It allowed us to unify unitary dynamics with environment-induced stochasticity through the Lindblad operators. However, as we have seen, the GKSL equation relies on both the Born approximation and the Markovian approximation, which assume weak system-environment coupling and memoryless dynamics. This master equation is thus not suited to describe *non-Markovian systems* where the memory cannot be disregarded as well as systems under strong coupling where *dissipative phase transitions* (DPT) may occur.

1.2.2 Steady state definition

Now that we have introduced the concepts of master equation and Liouvillian, we can explain another key concept in this thesis.

Steady state definition

The *steady state* is a state that remains unchanged over time and is thus unaffected by the dynamics of the system. It is mathematically defined as

$$\dot{\rho}_{ss} = 0, \quad (1.31a)$$

$$\mathcal{L}(\rho_{ss}) = 0, \quad (1.31b)$$

where the use of the Liouvillian on the second line does not necessarily imply that we are studying Markovian systems as we will see later on.

Although there exists particular cases of systems for which no steady state exists. Throughout this thesis, we will assume that a steady state always exists. Furthermore, even though it is not necessarily unique, for all our intents and purposes, we will assume that it is the case in this work. Note that systems for which the steady state is not unique arise under very specific conditions: systems with subspaces immune to dissipation, symmetric Liouvillians, or dissipative phase transitions. The first kind of systems is not studied in this thesis, and it can even be argued that these only occur under isolated systems approximations. For the second kind, the steady state remains unique for a given initial condition, while for the third kind, the degeneracy of the steady-state space can only happen in the thermodynamic limit ($N, V \rightarrow \infty$). In any case, this thesis focuses on numerical approaches for which these situations do not occur and are thus irrelevant. In the literature, the steady state is often referred to as *non-equilibrium steady state* (NESS) in the case of *driven-dissipative* systems, to emphasize that the steady state is outside the usual equilibrium statistics distribution (Bose-Einstein, Fermi-Dirac for example). In this work, we chose to denote all steady states using the *ss* subscript in order to avoid introducing too many notations.

1.2.3 Choi–Jamiołkowski isomorphism

We have seen that in order to propagate the dynamics of open quantum systems, we have to use a superoperator called Liouvillian. And in the case of the GKSL master equation, the Liouvillian is equivalent to the Lindbladian. However, while we can represent explicitly the density operator as a matrix, it is difficult to give an explicit representation of a superoperator⁸. Something that is possible to do is to vectorize the density operator and use the Choi–Jamiołkowski isomorphism [42], [43] to map the Liouvillian to an operator which thus admits a matrix representation. What this isomorphism states is that any *quantum channel* can be mapped to a *quantum state*. A quantum channel is a *completely positive* (CP) map and the Liouvillian is a particular case called *completely*

⁸Since it is equivalent to a rank 4 tensor.

positive trace-preserving (CPTP) map. The quantum state of the isomorphism refers to a density operator state and not a ket vector state. As such, this isomorphism allows us to give a matrix representation for the Liouvillian superoperator. The density operator is mapped from an operator acting on the Hilbert space \mathcal{H} to a ket vector of the Hilbert space $\mathcal{H} \otimes \mathcal{H}$ through the following process

$$\rho = \sum_{ij} \rho_{ij} |i\rangle\langle j| \quad \rightarrow \quad |\rho\rangle = \sum_{ij} \rho_{ij} |i\rangle \otimes |j\rangle. \quad (1.32)$$

Here we chose to work with the row vectorization convention however, one has to be careful since the other convention, i.e. column vectorization is also used in the literature. The row vectorization of the density matrix is obtained by stacking the successive rows of ρ onto one another:

$$\begin{pmatrix} \rho_{11} & \rho_{12} & \dots & \rho_{1N} \\ \rho_{21} & \rho_{22} & \dots & \rho_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{N1} & \rho_{N2} & \dots & \rho_{NN} \end{pmatrix} \longrightarrow \begin{pmatrix} \rho_{11} \\ \rho_{12} \\ \vdots \\ \rho_{1N} \\ \rho_{21} \\ \vdots \\ \rho_{2N} \\ \vdots \\ \rho_{NN} \end{pmatrix}. \quad (1.33)$$

The dynamics of the vectorized density operator are then given by

$$|\dot{\rho}\rangle = \mathcal{L} |\rho\rangle \quad (1.34)$$

where \mathcal{L} without parenthesis designates the Liouvillian in matrix form. The definition of the steady state can be expressed in vectorized form as

$$\mathcal{L} |\rho_{ss}\rangle = 0. \quad (1.35)$$

This means that the steady state is the eigenvector of the Liouvillian corresponding to the eigenvalue 0. In the case of the GKSL master equation, the matrix representation of the Liouvillian can be explicitly written as

$$\mathcal{L} = -i (H \otimes \mathbb{1} - \mathbb{1} \otimes H^T) + \sum_j \gamma_j \left(L_j \otimes L_j^* - \frac{1}{2} L_j^\dagger L_j \otimes \mathbb{1} - \frac{1}{2} \mathbb{1} \otimes L_j^T L_j^* \right). \quad (1.36)$$

This can be formally proven by only using the relation

$$\text{vec}(A\rho B^T) = (A \otimes B) \text{vec}(\rho), \quad (1.37)$$

where the $\text{vec}(\cdot)$ denotes the vectorization procedure in (1.32). The vectorization procedure of the density operator serves two main purposes. The first one is that we can compute the dynamics, the eigenvalues and eigenvectors of the Liouvillian by using pre-existing algorithms for matrices. Moreover, the general solution for the dynamics is given by

$$|\rho(t)\rangle = e^{\mathcal{L}t} |\rho_0\rangle, \quad (1.38)$$

and the steady state is now equivalent to the eigenvector corresponding to the eigenvalue 0. The second purpose, and perhaps the most important, is that in order to implement the density operator on qubits (which are described by ket vectors) and the Liouvillian using quantum gates (which are thus operators), we have no other mean than to use the Choi–Jamiołkowski isomorphism.

1.3 Finding the steady state of a simple open quantum system

This example is taken from [19]. We will consider a particle of spin $\frac{1}{2}$ subject to an external magnetic field of intensity h_0 along the x -axis. The particle will also be affected by a decay process at a rate γ along the z -axis. The Hamiltonian of the system is then given by

$$H = h_0 \sigma_x. \quad (1.39)$$

Furthermore, the jump operator along the z -axis is defined as

$$\sigma_- = \frac{\sigma_x - i\sigma_y}{2} \quad (1.40)$$

$$= \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad (1.41)$$

so that the GKSL master equation reads

$$\dot{\rho} = -ih_0[\sigma_x, \rho] + \gamma \left(\sigma_- \rho \sigma_+ - \frac{1}{2} \sigma_+ \sigma_- \rho - \frac{1}{2} \rho \sigma_+ \sigma_- \right). \quad (1.42)$$

Technically, we could solve for the steady state using only the equation above. However, to illustrate the Choi–Jamiołkowski isomorphism presented in the previous section with an example, we will apply the vectorization procedure for this system. Since our Hilbert space is of dimension 2, this means that our density operator is of dimension 2×2 and thus, of dimension 4 in vectorized form. We then have

$$\rho = \begin{pmatrix} \rho_{00} & \rho_{01} \\ \rho_{10} & \rho_{11} \end{pmatrix} \rightarrow |\rho\rangle = \begin{pmatrix} \rho_{00} \\ \rho_{01} \\ \rho_{10} \\ \rho_{11} \end{pmatrix}. \quad (1.43)$$

Here we chose to work in the computational basis where

$$|0\rangle \otimes |0\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad |0\rangle \otimes |1\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad |1\rangle \otimes |0\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad |1\rangle \otimes |1\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

The Liouvillian of the system can then be rewritten after the vectorization of the density operator using Eq. (1.36) and we have

$$\mathcal{L} = -ih_0(\sigma_x \otimes \mathbb{1} - \mathbb{1} \otimes \sigma_x) + \gamma(\sigma_- \otimes \sigma_- - \frac{1}{2} \sigma_+ \sigma_- \otimes \mathbb{1} - \frac{1}{2} \mathbb{1} \otimes \sigma_+ \sigma_-), \quad (1.44)$$

where σ_+ is defined as

$$\sigma_+ = \frac{\sigma_x + i\sigma_y}{2} \quad (1.45)$$

$$= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad (1.46)$$

and we used the following equalities

$$\sigma_x = \sigma_x^T, \quad (1.47)$$

$$\sigma_{\pm} = \sigma_{\mp}^T, \quad (1.48)$$

$$\sigma_{\pm} = \sigma_{\pm}^*. \quad (1.49)$$

One way to proceed is to apply \mathcal{L} onto the computational basis to explicitly construct its matrix representation and we have

$$\mathcal{L}|0\rangle \otimes |0\rangle = -\gamma|0\rangle \otimes |0\rangle + ih_0|0\rangle \otimes |1\rangle - ih_0|1\rangle \otimes |0\rangle + \gamma|1\rangle \otimes |1\rangle, \quad (1.50)$$

$$\mathcal{L}|0\rangle \otimes |1\rangle = ih_0|0\rangle \otimes |0\rangle - \frac{1}{2}\gamma|0\rangle \otimes |1\rangle - ih_0|1\rangle \otimes |1\rangle, \quad (1.51)$$

$$\mathcal{L}|1\rangle \otimes |0\rangle = -ih_0|0\rangle \otimes |0\rangle - \frac{1}{2}\gamma|1\rangle \otimes |0\rangle + ih_0|1\rangle \otimes |1\rangle, \quad (1.52)$$

$$\mathcal{L}|1\rangle \otimes |1\rangle = -ih_0|0\rangle \otimes |1\rangle + ih_0|1\rangle \otimes |0\rangle, \quad (1.53)$$

which directly gives us that \mathcal{L} can be explicitly written as

$$\mathcal{L} = \begin{pmatrix} -\gamma & ih_0 & -ih_0 & 0 \\ ih_0 & -\frac{1}{2}\gamma & 0 & -ih_0 \\ -ih_0 & 0 & -\frac{1}{2}\gamma & ih_0 \\ \gamma & -ih_0 & ih_0 & 0 \end{pmatrix}. \quad (1.54)$$

Alternatively, we could have explicitly computed Eq. (1.44) using the Kronecker product, however it is only defined in the second chapter of this thesis and we thus chose not to. The steady state can then be found by finding the eigenvector associated to the zero-eigenvalue which is equivalent to solving the following system of equations

$$\begin{pmatrix} \dot{\rho}_{00} \\ \dot{\rho}_{01} \\ \dot{\rho}_{10} \\ \dot{\rho}_{11} \end{pmatrix} = \begin{pmatrix} -\gamma & ih_0 & -ih_0 & 0 \\ ih_0 & -\frac{1}{2}\gamma & 0 & -ih_0 \\ -ih_0 & 0 & -\frac{1}{2}\gamma & ih_0 \\ \gamma & -ih_0 & ih_0 & 0 \end{pmatrix} \begin{pmatrix} \rho_{00} \\ \rho_{01} \\ \rho_{10} \\ \rho_{11} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad (1.55)$$

Solving the set of equations by hand or politely asking Mathematica gives us the following solution

$$|\rho_{ss}\rangle_{\text{unnormalized}} = \left(\frac{4h_0^2}{\gamma^2 + 4h_0^2}, -\frac{2i\gamma h_0}{\gamma^2 + 4h_0^2}, \frac{2i\gamma h_0}{\gamma^2 + 4h_0^2}, 1 \right)^T, \quad (1.56)$$

which we then have to normalize by imposing the condition $\text{Tr}[\rho_{ss}] = 1$. We then obtain

$$|\rho_{ss}\rangle = \left(\frac{4h_0^2}{\gamma^2 + 8h_0^2}, -\frac{2i\gamma h_0}{\gamma^2 + 8h_0^2}, \frac{2i\gamma h_0}{\gamma^2 + 8h_0^2}, \frac{\gamma^2 + 4h_0^2}{\gamma^2 + 8h_0^2} \right)^T. \quad (1.57)$$

We can convert this back into a density operator

$$\rho_{ss} = \frac{1}{\gamma^2 + 8h_0^2} \begin{pmatrix} 4h_0^2 & -2ih_0\gamma \\ 2ih_0\gamma & \gamma^2 + 4h_0^2 \end{pmatrix}, \quad (1.58)$$

which can then be used to compute expected values of operators through the relation using the trace of the product. Following this we find the following expected values for σ_z and σ_y

$$\langle \sigma_y \rangle = \frac{4h_0\gamma}{\gamma^2 + 8h_0^2} = \frac{4\frac{h_0}{\gamma}}{1 + 8\frac{h_0^2}{\gamma^2}}, \quad (1.59a)$$

$$\langle \sigma_z \rangle = \frac{-\gamma^2}{\gamma^2 + 8h_0^2} = \frac{-1}{1 + 8\frac{h_0^2}{\gamma^2}}. \quad (1.59b)$$

As we can see, the ratio between the magnetic field intensity h_0 and the damping rate γ determines uniquely the expected values. As such, it is convenient to define the ratio of the two parameters as $h = h_0/\gamma$ such that the expected values can be rewritten as

$$\langle \sigma_y \rangle = \frac{4h}{1 + 8h^2}, \quad (1.60a)$$

$$\langle \sigma_z \rangle = \frac{-1}{1 + 8h^2}. \quad (1.60b)$$

Both equations are plotted in Fig. 1.3. Note that $\langle \sigma_x \rangle$ is always 0, as we can see easily from (1.58).

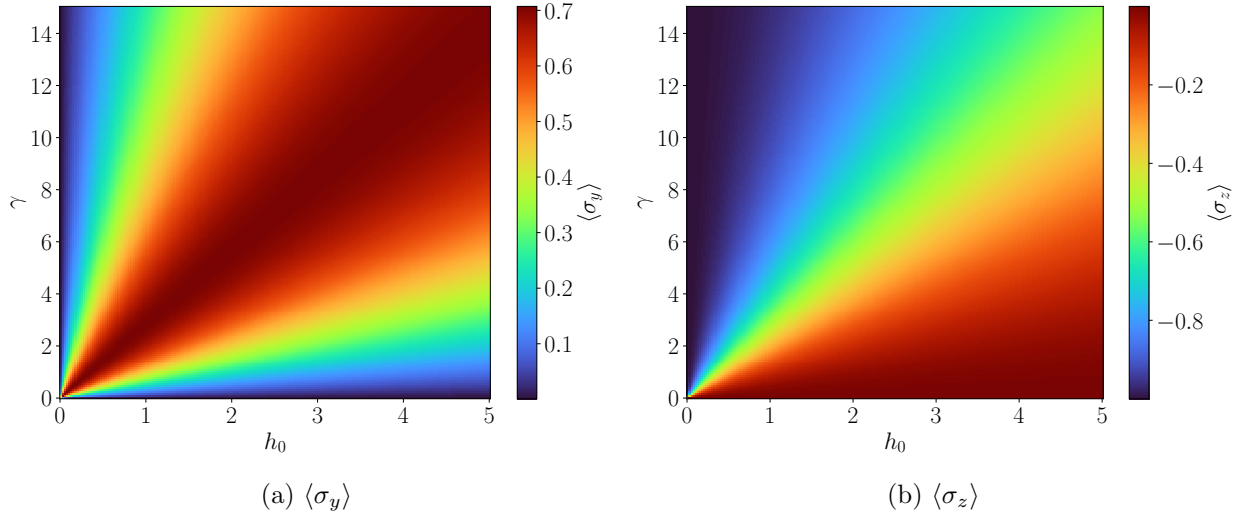


Figure 1.3: Density plots of $\langle \sigma_y \rangle$ (a) and $\langle \sigma_z \rangle$ (b) in terms of h_0 ranging 0 to 5.0 and γ varying from 0 to 15. These plots were obtained using eqs. (1.60a) and (1.60b). We see that points of the same color are on a straight line, which confirms that the values of $\langle \sigma_y \rangle$ and $\langle \sigma_z \rangle$ depend indeed only on the ratio h_0/γ .

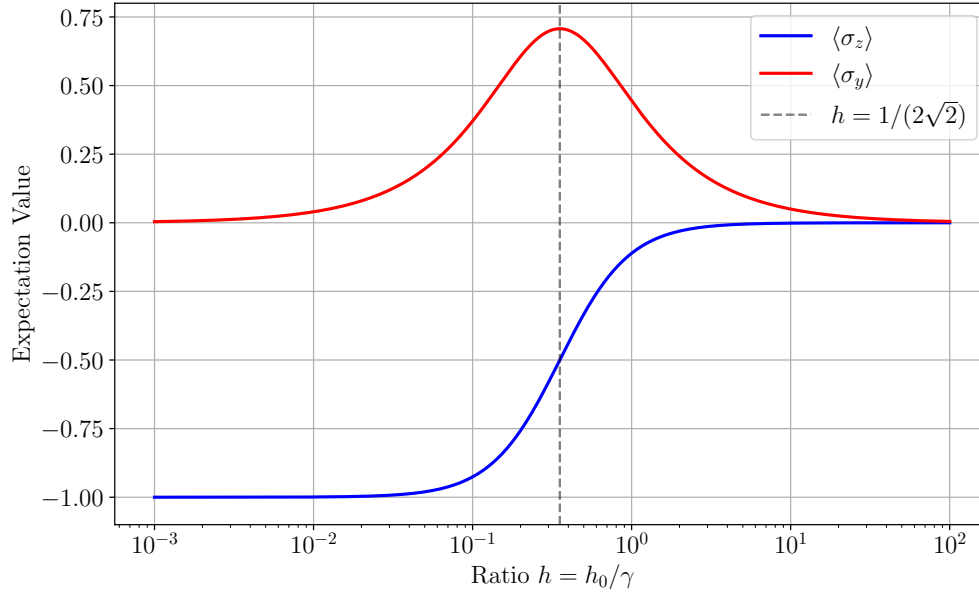


Figure 1.4: plot of $\langle \sigma_y \rangle$ and $\langle \sigma_z \rangle$ in terms of the ratio $h = h_0/\gamma$ from 10^{-3} to 10^2 . These plots were obtained using eqs. (1.60a) and (1.60b).

1.4 Dynamics of non-Markovian open quantum systems

In Section 1.2, we saw how to describe the dynamics of open quantum systems under the Markovian approximation, using the GKSL master equation. However, in systems with long correlation times (e.g., low-temperature baths or structured reservoirs), it becomes impossible to neglect memory effects when describing their dynamics; such systems are termed *non-Markovian*.

It is crucial to emphasize that the memory of an open quantum system resides entirely in the environment, and in fact, it is a byproduct of the interaction between the system and the environment. To illustrate this, recall our classical analogy: a gas bottle within a compartment floating in space. As we discussed earlier, if the opening of the compartment is narrow or the opening of the bottle is wide (or both), gas particles accumulate in the compartment before slowly escaping into space as shown in Fig. 1.5. Over time, the compartment's gas concentration reflects the bottle's prior states, acting as a memory buffer. Conversely, observing the bottle alone reveals no information about its past (we cannot infer the initial gas quantity). This is also the reason why the evolution due to dissipation is non-unitary: it is not possible to revert the evolution and go back in time.

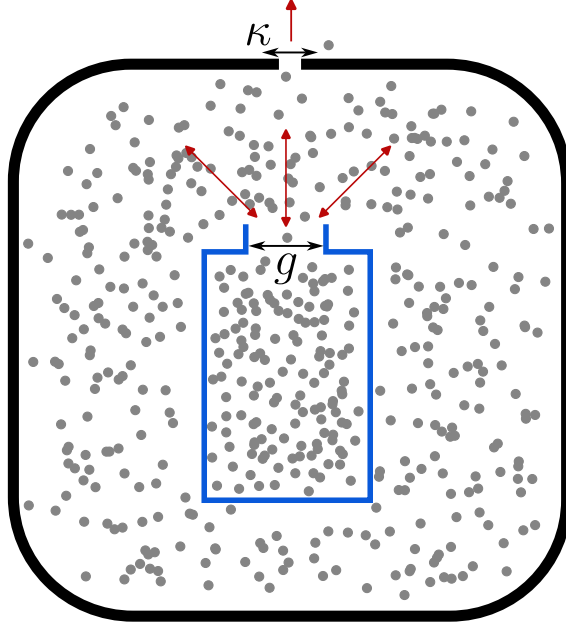


Figure 1.5: Sketch of the classical analogy of a gas bottle within a compartment outside the validity of the Born-Markov approximation. The combined effects of the opening of the gas bottle being now overly large and the opening of the compartment being too narrow leads to the accumulation of gas particles within the compartment. These particles now have a chance of returning inside the gas bottle, impacting the overall interaction between the gas bottle and the compartment. Here g can be interpreted as the particle exchange rate between the bottle and the compartment and κ is interpreted as the particle loss rate into space. For systems exhibiting a strong non-Markovianity, we thus have $g/\kappa \gg 1$.

To accurately describe the evolution of non-Markovian systems, we need to introduce new methods that take into account the past states of the systems. In this work, we focus on the *Hierarchical Equations of Motion* (HEOM) method, which encodes memory effects into auxiliary degrees of freedom, and benchmark it against *Markovian embedding* techniques.

1.4.1 Markovian embedding

The first method that we will introduce is in fact not novel: it consists of embedding the physical system of interest into a larger Markovian system and computing its dynamics using the GKSL master equation. This method is of course radical and not very efficient; instead of trying to integrate the memory of the systems in the dynamics, it simply takes the whole part of the environment that contains the memory and propagates its dynamics along with the system's. At the same time, it is also the simplest procedure which is why it will be used as a benchmarking tool. To connect with our classical example, this procedure would be equivalent to considering the entirety of the compartment along with the bottle, as our physical system, and the interaction between the space and the compartment could be safely approximated as Markovian due to the immensity of space.

More formally, if we consider a system coupled to a bosonic bath at zero temperature, the total Hamiltonian is given by [44]

$$H_{SE} = H_S + \underbrace{\sum_k \tilde{\omega}_k b_k^\dagger b_k}_{H_E} + \underbrace{\sum_k \left(\tilde{g}_k b_k L^\dagger + \tilde{g}_k^* b_k^\dagger L \right)}_{H_{int}}, \quad (1.61)$$

where H_S , H_E and H_{int} are the Hamiltonians of the system, the bath and the interaction respectively. b_k is the annihilation operator for the k^{th} mode of frequency $\tilde{\omega}_k$, L is an arbitrary system operator relative to the nature of the interaction, and \tilde{g}_k are the system-bath coupling strengths⁹. The *spectral density* $J(\omega)$ describes the strength and structure of the interaction between the system and the bath as a function of the energy ω : more specifically, it measures how easily an energy ω can be exchanged with the system (the larger $J(\omega)$ for a given ω , the easier the energy exchange). It is defined as

$$J(\omega) = \pi \sum_k |\tilde{g}_k|^2 \delta(\omega - \tilde{\omega}_k), \quad (1.62)$$

and is related to the *bath correlation function* (BCF) $\alpha(\tau)$ through a Fourier transform such that

$$\alpha(\tau) = \sum_k |\tilde{g}_k|^2 e^{-i\tilde{\omega}_k \tau} = \frac{1}{\pi} \int_0^\infty J(\omega) e^{-i\omega \tau} d\omega. \quad (1.63)$$

The BCF is interpreted as a *temporal correlation function* for the bath¹⁰ and can be rewritten as

$$\alpha(t) = \left\langle B(t) B^\dagger(0) \right\rangle_{\text{bath}}, \quad (1.64)$$

with $B(t)$ the *bath coupling operator* in the interaction picture¹¹ which in our situation (a bosonic bath) is given by

$$B = \sum_k \tilde{g}_k b_k. \quad (1.65)$$

In general, the spectral density has a non-trivial structure due to the continuum of modes k . However, if we consider the particular case where $J(\omega)$ is a Lorentzian of the form

$$J(\omega) = \frac{\kappa g'^2}{(\omega - \omega')^2 + \kappa^2}, \quad (1.66)$$

then the BCF would approximately takes the following form [45], [46]

$$\alpha(\tau) = g'^2 e^{-i\omega' \tau - \kappa |\tau|}. \quad (1.67)$$

This corresponds to a pseudo-mode of the bath. The decaying exponential factor represents how the excitation “disappears” in the environment due to the many degrees of freedom of the bath. Intuitively, for Markovian systems, κ is considered infinite so that any perturbation in the pseudo-mode of the bath immediately decays. This would correspond to a spectral density that is flat (i.e.,

⁹We chose to denote the coupling strengths, frequencies using a tilde and the annihilation operator with the letter b and not the usual a . This is to emphasize that these are different from those corresponding to the pseudo-modes. However, since in practice, we work with the pseudo-modes, it alleviates the notations to not have the tilde on them and not the other way around.

¹⁰As such, it can also be interpreted as a propagator.

¹¹In the interaction picture, operators evolve under the free bath Hamiltonian, in our case, we have $B(t) = e^{iH_E t} B(0) e^{-iH_E t}$.

constant). The purpose behind this discussion is that we can extend this idea to a spectral density which is a sum of multiple Lorentzians

$$J(\omega) = \frac{1}{\pi} \sum_{j=1}^M \frac{\kappa_j g_j^2}{(\omega - \omega_j)^2 + \kappa_j^2}, \quad (1.68)$$

with M being the total number of pseudo-modes. This in turn yields a BCF

$$\alpha(\tau) = \sum_{j=1}^M g_j^2 e^{-i\omega_j \tau - \kappa_j |\tau|}. \quad (1.69)$$

It has been proven that it is always possible to approximate the spectral density to a satisfying degree using a sum of Lorentzians [47]–[50]. This scheme is illustrated in Fig. 1.6

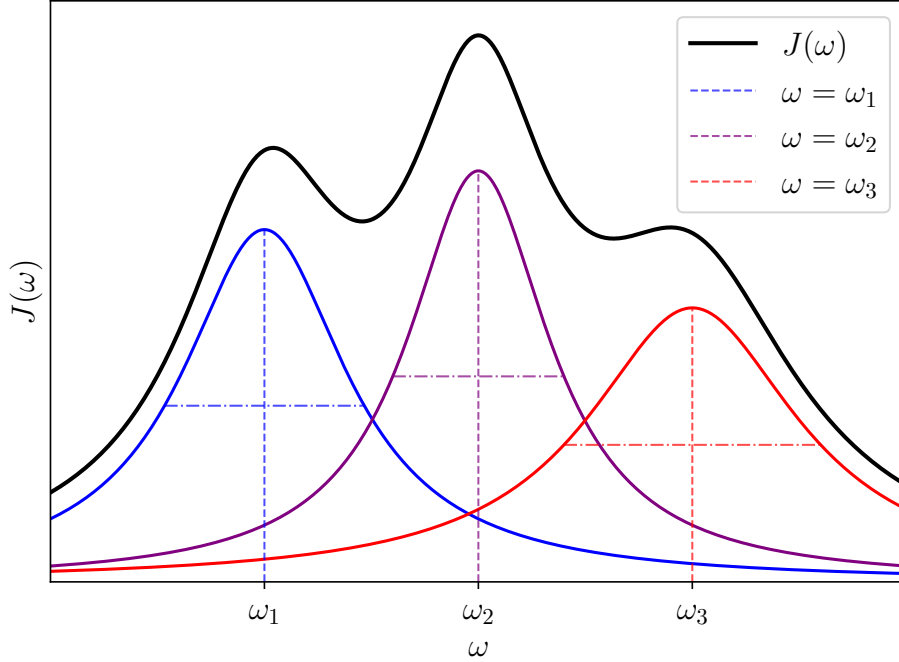


Figure 1.6: Sketch representing how a general spectral density $J(\omega)$ can be decomposed as a sum of Lorentzians. The frequency of the pseudo-mode is given by the center of the Lorentzian, the damping rate by the half-height width and the coupling strength is related to the height (it is not given by the height however).

Note that it has also been proven that the Lorentzian decomposition is not unique [51]. Performing this decomposition amounts to coupling the system to a set of pseudo-modes with frequencies ω_j which have damping rates κ_j and coupling strengths g_j . The Markovian embedding approach thus consists in incorporating the system with the non-Markovian part of the environment by adding pseudo-modes until the enlarged system can safely be considered Markovian. The total system then evolves according to the GKSL master equation

$$\dot{\rho}_{SB} = \mathcal{L}_M(\rho_{SB}) \equiv -i[H_{SB}, \rho_{SB}] + \mathcal{D}(\rho_{SB}) \quad (1.70)$$

where the subscript SB designates the enlarged system composed of the physical system of interest and pseudo-modes of the bath (the letter B only designates the pseudo-modes of the bath and not the bath itself.) and where \mathcal{L}_M designates the Liouvillian of the enlarged system with the Hamiltonian H_{SB} being given by

$$H_{SB} = H_S + \sum_{j=1}^M \omega_j a_j^\dagger a_j + \sum_{j=1}^M \left(g_j a_j L^\dagger + g_j^* a_j^\dagger L \right), \quad (1.71)$$

with a_j the bosonic annihilation operator of the pseudo-mode j . And finally, the dissipator is given by

$$\mathcal{D}(\cdot) = \sum_{j=1}^M \kappa_j \left(2a_j \cdot a_j^\dagger - \left\{ a_j^\dagger a_j, \cdot \right\} \right). \quad (1.72)$$

An example of a system interacting with the pseudo-mode of a bosonic bath is given in Sec. 1.5, where we consider the open quantum Rabi model. It represents an atom, assumed to be a two-level system inside an electromagnetic cavity.

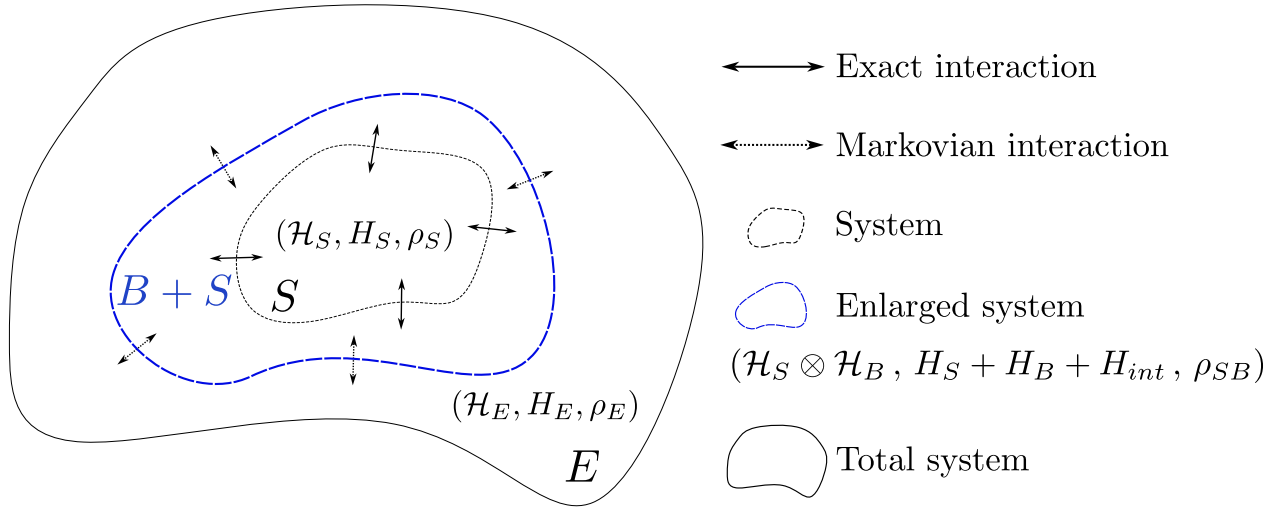


Figure 1.7: Representation of the Markovian embedding approach. The system is enlarged by adding pseudo-modes of the bath and now contains the exact interaction between the system and the bath. The enlarged system still interacts with the rest of the bath through Markovian interactions and its evolution is thus described by the GKSL master equation. However the non-Markovian part is now contained within the enlarged system.

We should remind the reader that the enlarged system does not contain the entirety of the bath but only the part relevant to the non-Markovian interaction. However, even if only a limited portion of the environment is taken into account, the corresponding Hilbert space has an infinite size (at each mode is associated a Fock space with an infinite size). This means that in order to numerically compute the dynamics of the system, we need to introduce a *cutoff parameter* \mathbf{N}_c . It is written in bold font to indicate that it is a M -dimensional integer vector $(N_c^1, N_c^2, \dots, N_c^M)$ such that the N_c^j is the number of Fock states associated to the pseudo-mode ω_j and thus the corresponding maximum

number of excitations is $N_c^j - 1$. However, in this thesis, we chose to use the same cutoff for all Fock spaces for simplicity, and we will denote it N_c ¹². If we suppose the size of the Hilbert space of the system to be N_S , the length of the Liouvillian (in matrix representation) of the enlarged system, that we will denote $\dim(\mathcal{L}_M)$ is given by

$$\dim(\mathcal{L}_M) = N_S^2 N_c^{2M}. \quad (1.73)$$

It is evident that this scaling becomes rapidly unmanageable for multi-mode systems and systems at critical coupling where the numbers of excitations in the Fock spaces increase sharply and thus require a larger cutoff. This is the underlying motivation for finding new methods that are numerically advantageous for this kind of systems. One such method is presented in the next subsection.

1.4.2 The HEOM method

In this subsection, we will introduce the primary method of this work, to compute the dynamics of non-Markovian systems, namely the hierarchical equations of motion (HEOM) method. The derivation will not be included in this thesis as it requires advanced techniques such as the path integral formalism; however, a physical intuition will be provided instead. This method was initially developed by Yoshitaka Tanimura and Ryogo Kubo in a paper published in 1989 [11] and was then polished multiple times by the former [52]–[55]. It is a *non-perturbative* numerical method to propagate the dynamics of non-Markovian open quantum systems. The form we use in this work stems from these sources [56], [57]. When the enlarged system is initially in the state $\rho(0) = \rho_S(0) \otimes \rho_B(0)$, the exact dynamics of the system can be described by the following set of equations

$$\begin{aligned} \dot{\rho}^{(\mathbf{n}, \mathbf{m})} = & -i \left[H_S, \rho^{(\mathbf{n}, \mathbf{m})} \right] - (\mathbf{w}^* \cdot \mathbf{n} + \mathbf{w} \cdot \mathbf{m}) \rho^{(\mathbf{n}, \mathbf{m})} \\ & + \sum_{j=1}^M \left(G_j n_j L \rho^{(\mathbf{n}_j^-, \mathbf{m})} + G_j^* m_j \rho^{(\mathbf{n}, \mathbf{m}_j^-)} L^\dagger + \left[\rho^{(\mathbf{n}_j^+, \mathbf{m})}, L^\dagger \right] + \left[L, \rho^{(\mathbf{n}, \mathbf{m}_j^+)} \right] \right), \end{aligned} \quad (1.74)$$

with $\mathbf{n} = (n_1, n_2, \dots, n_M)$, $\mathbf{m} = (m_1, m_2, \dots, m_M)$ vectorized sets of multi-indices in \mathbb{N}^M , $\mathbf{w} = (w_1, w_2, \dots, w_M) \in \mathbb{C}^M$, $w_j = \kappa_j + i\omega_j$, $\mathbf{a} \cdot \mathbf{b} = \sum_{j=1}^M a_j^* b_j$ the inner product on \mathbb{C}^M and $\mathbf{n}_j^\pm = (n_1, \dots, n_j \pm 1, \dots, n_M)$. For a bosonic bath, the G_j are real and simply equal to g_j^2 . At first glance, Eq. (1.74) seems a bit overwhelming however, this is simply a fancy set of linear coupled master equations. The notation $\rho^{(\mathbf{n}, \mathbf{m})}$ signifies that we use multiple density operators to accurately describe the dynamics. More precisely, $\rho^{(0,0)}$ corresponds to the density operator of the system and all ancillary operators serve as memory cells for the system. The memory is encoded by taking into account the build-up correlations between the system and the environment. As time evolves, the ancillary states will become gradually populated through the $G_j n_j L$ and $G_j^* m_j L^\dagger$ terms (the greater G_j , the faster this will happen). These populations will decay through the $(\mathbf{w}^* \cdot \mathbf{n} + \mathbf{w} \cdot \mathbf{m})$ terms (the greater κ_j , the faster this will happen), however at the same time, they will influence the ancillary states closer to the system density state through the L and L^\dagger commutators. In theory, the number of ancillary states is infinite, nonetheless, it can be truncated at higher n_j and m_j

¹²The earlier part was written to emphasize that using a single cutoff for all Fock spaces is a choice made to simplify the convergence analysis made latter in this work. Nonetheless, it is technically possible to choose asymmetric cutoffs and achieve similar convergences at the cost of having M degrees of liberty instead of 1.

by assuming that these states will remain too weak to influence the dynamics. The advantage of HEOM is that this truncation can be chosen to be triangular such that only states for which

$$\sum_{j=1}^M (n_j + m_j) \leq K_{max}, \quad (1.75)$$

are taken into account, where $K_{max} \in \mathbb{N}$ is the corresponding cutoff parameter. This implies that the total number of ancillary states (+ the system density state) is given by [58]

$$\#_{\rho} = \frac{(2M + K_{max})!}{(2M)!K_{max}!}. \quad (1.76)$$

We can also vectorize (1.74) using Choi–Jamiołkowski isomorphism however in practice, we have to assign an order for the ancillary states to construct the matrix representation of the HEOM Liouvillian, which can be chosen arbitrarily. The equation (1.74) in vectorized form reads

$$\begin{aligned} \left| \dot{\rho}^{(\mathbf{n}, \mathbf{m})} \right\rangle = & -i \left[H_S \otimes \mathbb{1} - \mathbb{1} \otimes H_S^T - (\mathbf{w}^* \cdot \mathbf{n} + \mathbf{w} \cdot \mathbf{m}) \right] \left| \rho^{(\mathbf{n}, \mathbf{m})} \right\rangle \\ & \sum_{j=1}^M \left(G_j n_j L \otimes \mathbb{1} \left| \rho^{(\mathbf{n}_j^-, \mathbf{m})} \right\rangle + G_j^* m_j \mathbb{1} \otimes L^* \left| \rho^{(\mathbf{n}, \mathbf{m}_j^-)} \right\rangle \right) \\ & \sum_{j=1}^M \left(\left(\mathbb{1} \otimes L^* - L^\dagger \otimes \mathbb{1} \right) \left| \rho^{(\mathbf{n}_j^+, \mathbf{m})} \right\rangle + (L^T \otimes \mathbb{1} - \mathbb{1} \otimes L) \left| \rho^{(\mathbf{n}, \mathbf{m}_j^+)} \right\rangle \right), \end{aligned} \quad (1.77)$$

which is independent of the ordering of the ancillary states. In order to provide the reader with a stronger intuition of the HEOM method, we will restrain equation (1.77) to only one mode, in this scenario we have [56]

$$\begin{aligned} \left| \dot{\rho}^{(n, m)} \right\rangle = & D_{nm} \left| \rho^{(n, m)} \right\rangle + A_n \left| \rho^{(n-1, m)} \right\rangle + B_m \left| \rho^{(n, m-1)} \right\rangle \\ & + C \left| \rho^{(n+1, m)} \right\rangle - C^\dagger \left| \rho^{(n, m+1)} \right\rangle, \end{aligned} \quad (1.78)$$

where the corresponding submatrices are defined as

$$D_{nm} \equiv -i (H_S \otimes \mathbb{1} - \mathbb{1} \otimes H_S^T) - ((n-m)i\omega + (n+m)\kappa) \mathbb{1} \otimes \mathbb{1}, \quad (1.79a)$$

$$A_n \equiv G n L \otimes \mathbb{1}, \quad (1.79b)$$

$$B_m \equiv G^* m \mathbb{1} \otimes L^*, \quad (1.79c)$$

$$C \equiv \mathbb{1} \otimes L - L^\dagger \otimes \mathbb{1}. \quad (1.79d)$$

To construct the matrix representation of the HEOM Liouvillian we order the density operators using the following logic, taking $K_{max} = 2$ as an example:

$$|\rho\rangle = \left(\left| \rho^{(0,0)} \right\rangle, \left| \rho^{(0,1)} \right\rangle, \left| \rho^{(0,2)} \right\rangle, \left| \rho^{(1,0)} \right\rangle, \left| \rho^{(1,1)} \right\rangle, \left| \rho^{(2,0)} \right\rangle \right)^T. \quad (1.80)$$

The matrix representation are given for $K_{max} = 1$ and $K_{max} = 2$ below

$$\mathcal{L}_{\text{HEOM}}(K_{max} = 1) = \begin{pmatrix} D_{00} & -C^\dagger & C \\ B_1 & D_{01} & 0 \\ A_1 & 0 & D_{10} \end{pmatrix}, \quad (1.81)$$

$$\mathcal{L}_{\text{HEOM}}(K_{max} = 2) = \begin{pmatrix} D_{00} & -C^\dagger & 0 & C & 0 & 0 \\ B_1 & D_{01} & -C^\dagger & 0 & C & 0 \\ 0 & B_2 & D_{02} & 0 & 0 & 0 \\ A_1 & 0 & 0 & D_{10} & -C^\dagger & C \\ 0 & A_1 & 0 & B_1 & D_{11} & 0 \\ 0 & 0 & 0 & A_2 & 0 & D_{20} \end{pmatrix}. \quad (1.82)$$

Figure 1.8 provides a diagram representing the interactions between each density operator up to $K_{max} = 3$. We intuitively understand how K_{max} relates to the memory of the system, as a greater K_{max} increases the memory depth to which an interaction can propagate and also allows density operators deeper in the memory to intervene in the total system dynamics. From this, we understand that the length of the Liouvillian for multi-modes systems, in terms of the cutoff K_{max} is given by

$$\dim(\mathcal{L}_{\text{HEOM}}) = N_S^2 \frac{(2M + K_{max})!}{(2M)!K_{max}!}. \quad (1.83)$$

1.4.3 Markovian embedding VS HEOM

Compared to the dimension of the enlarged system Liouvillian (1.73), it is not straightforward to show the advantage of the HEOM method. In this section, we only provide a rough argument for this by comparing the typical dimensions of the HEOM and Markovian embedding Liouvillians. Later on, we will provide quantitative results showing that indeed the HEOM Liouvillian gives rise to a given precision using less computational resources than the Markovian embedding Liouvillian in specific models and parameter regimes.

Now we will consider that $N_c - 1$ ¹³ and K_{max} scale similarly, which is justified by the fact that in order to measure the same correlations, both cutoffs should be equal [56]. From Eq. (1.83), the leading order term in K_{max} is given by $N_S^2(K_{max})^{2M}/(2M)!$. This means that the ratio between the dimensions of the HEOM Liouvillian and the Markovian embedding Liouvillian goes to $1/(2M)!$ as K_{max} and $N_c - 1$ go to infinity. In general, combining (1.83) and (1.73) and using $K_{max} = N_c - 1$ leads to

$$\frac{\dim(\mathcal{L}_{\text{HEOM}})}{\dim(\mathcal{L}_M)} = \frac{(2M + K_{max})!}{(2M)!K_{max}!(K_{max} + 1)^{2M}}, \quad (1.84)$$

and plotting this equation in terms of the cutoff K_{max} and the number of modes M yields the plots in Fig. 1.9. As we can notice, the HEOM method provides an extreme advantage for multi-mode systems. At only 4 modes and around $K_{max} = 10$, the ratio is already at the order of $1/4000$, this corresponds to $\dim(\mathcal{L}_{\text{HEOM}}) = 43758N_s^2$ and $\dim(\mathcal{L}_M) = 21435881N_s^2$. Furthermore, even if K_{max} and $N_c - 1$ did not scale similarly, Fig. 1.10 shows that for multiple modes, the HEOM method is often advantageous even when its cutoff is larger than the Markovian embedding cutoff. Furthermore, we see on the graph the white line indicating the relation K_{max} and N_c should have for

¹³We consider $N_c - 1$ because the argument considers the cutoff as the maximum number of photons and not the maximum number of Fock states.

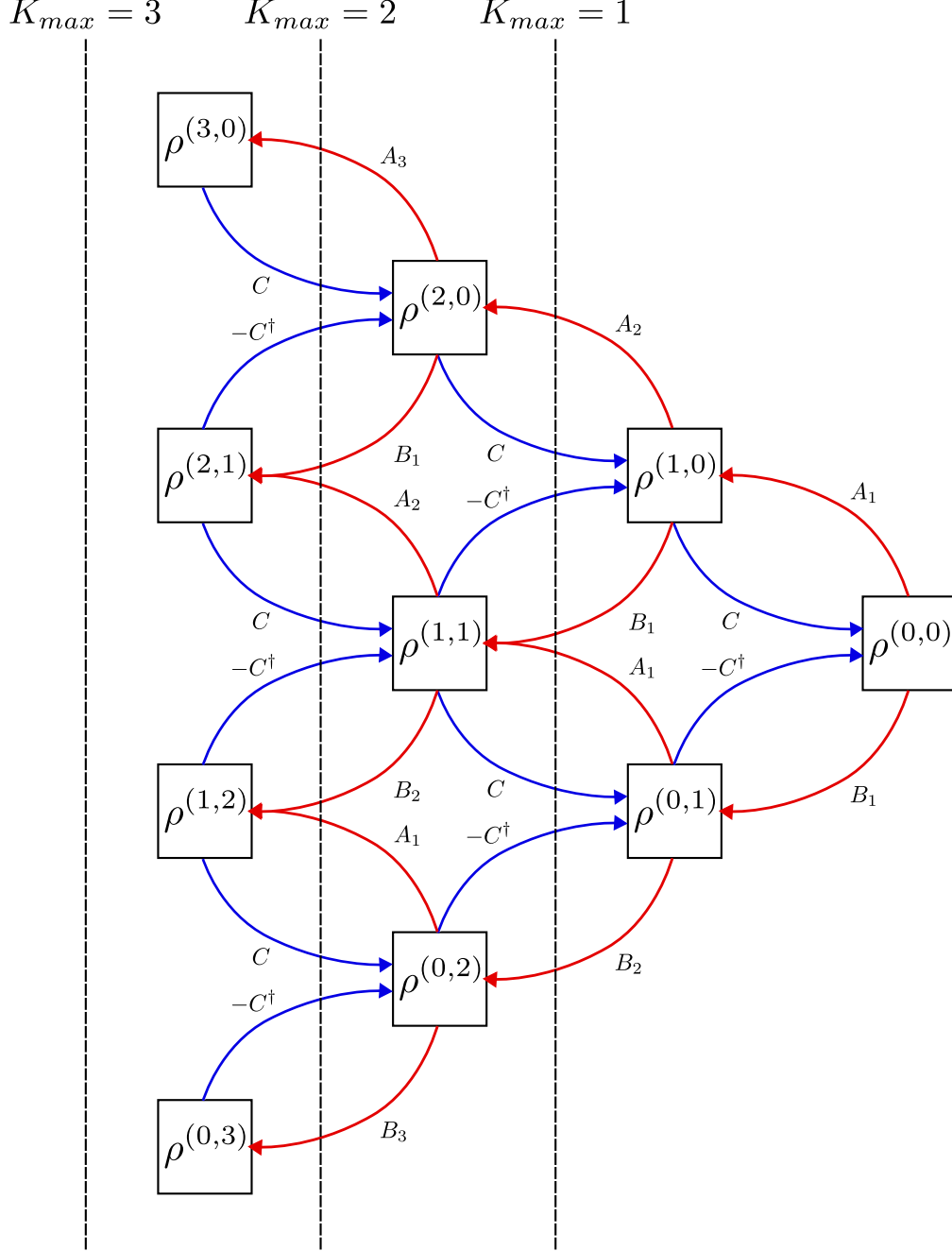


Figure 1.8: Representation of the one-mode HEOM method. The interactions between the density operators are shown by the blue and red arrows, where the red arrows represent the propagation of information deeper into the memory while the blue arrows represent the resurgence of information going to lower depths of the hierarchy. K_{max} indicates the depth of the hierarchy as a triangular truncation scheme with $n + m \leq K_{max}$. The corresponding interaction matrices are also shown, for example, the red arrow going from $\rho^{(0,2)}$ to $\rho^{(0,3)}$ indicates that $\dot{\rho}^{(0,3)} = B_3 \rho^{(0,2)} + (\dots)$. Although it is not shown on this figure, each density operator also evolves depending on their actual state.

their corresponding Liouvillians to have the same dimension. Furthermore, this line seems straight and its slope is roughly around $1/M$. This can be explained by computing the logarithm of the

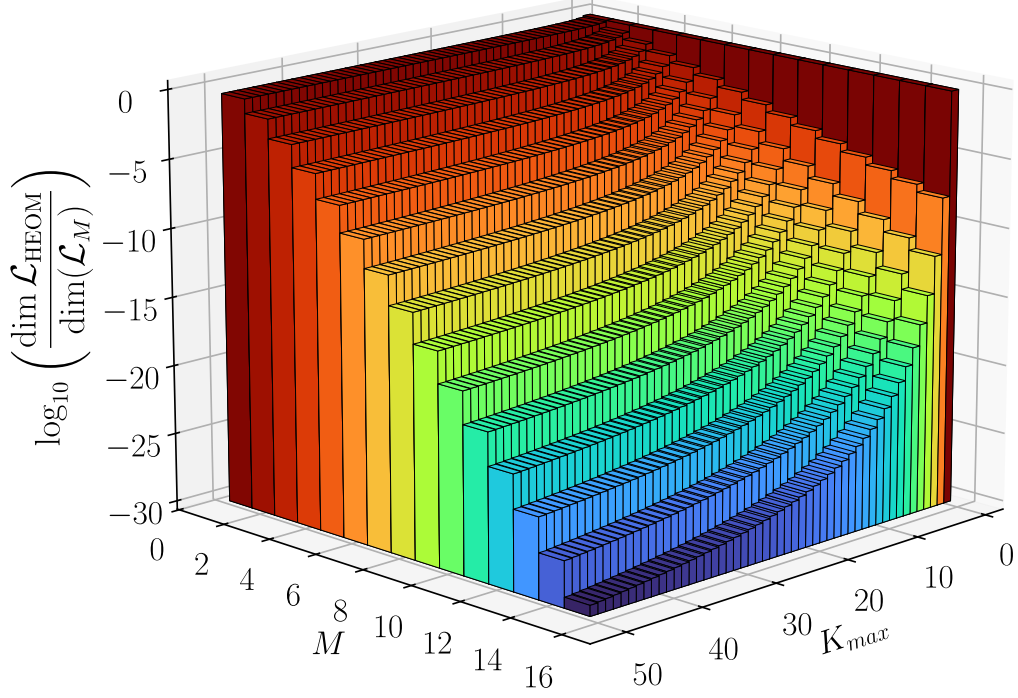


Figure 1.9: 3D bar plot of the base 10 logarithm of the ratio between the dimensions of both Liouvillian where we replace N_c by $K_{max} + 1$, in terms of the number of pseudo-modes M ranging from 1 to 15 and the cutoff K_{max} varying from 0 to 50.

ratio explicitly and solving for when it is equal to 0, i.e.,

$$\log \left(\frac{\dim \mathcal{L}_{\text{HEOM}}}{\dim(\mathcal{L}_M)} \right) = 0, \quad (1.85)$$

which is of course equivalent to solving

$$\frac{(2M + K_{max})!}{(2M)!K_{max}!} = N_c^{2M}. \quad (1.86)$$

Considering M large enough and $K_{max} \gg 2M$, we can then Use Stirling's approximation which yields

$$K_{max} \approx \frac{2M}{e} N_c, \quad (1.87)$$

which confirms our observations.

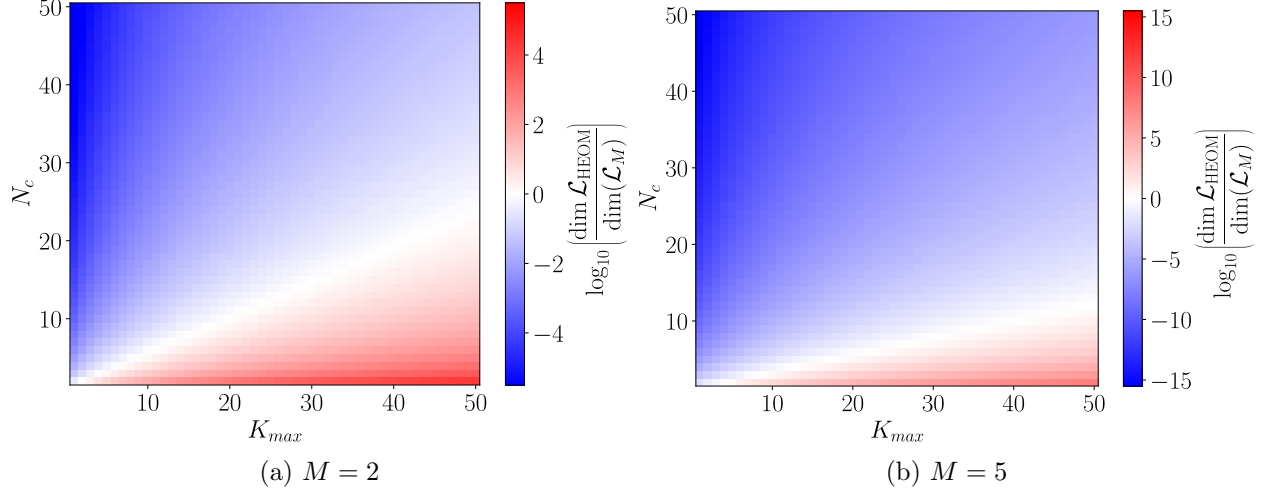


Figure 1.10: Density plots of the base 10 logarithm of the ratio between the dimensions of both Liouvillian for (a) $M = 2$ and (b) $M = 5$, in terms of both cutoffs each ranging from 1 to 50. The colormap is chosen as to highlight where one method is advantageous compared to the other. The white cells thus show where both Liouvillians have around the same size.

1.5 Finding the steady state of the open quantum Rabi model using the HEOM method

The Rabi model describes an atom assumed to be a two-level system coupled to a single-mode electromagnetic field inside a lossy cavity. In this scenario, the dynamics of the atom can be non-Markovian depending on the photon loss rate. Indeed, if the cavity loss rate is small, an emitted photon by the atom inside the cavity is likely to stay around for while and thus be susceptible to be reabsorbed by the atom. However, if one includes the cavity mode in the system description, the dynamics of the enlarged system is typically Markovian (as the coupling of the cavity mode to the outside electromagnetic field through the cavity mirror is well described within a Born-Markov approximation). The Hamiltonian of the enlarged system for such a Markovian embedding approach is given by

$$H_{\text{Rabi}} = \omega_0 \sigma_z + \omega a^\dagger a + g(a + a^\dagger) \sigma_x, \quad (1.88)$$

where H_{Rabi} corresponds to H_{SB} in the expression (1.71). As such, the first term, which corresponds to the atom Hamiltonian, is identified as H_S , the second term, which corresponds to the electromagnetic field Hamiltonian for the cavity, is identified as H_B and the last term, which represents the energy of interaction, is thus identified as H_{int} . Furthermore, we identify L as σ_x . The dissipator for this system is given by

$$\mathcal{D}(\cdot) = \kappa \left(2a \cdot a^\dagger - \left\{ a^\dagger a, \cdot \right\} \right), \quad (1.89)$$

with κ the damping rate of the cavity. This means that the bath is characterized by the correlation function

$$\alpha(\tau) = g^2 e^{-\kappa|\tau| - i\omega\tau}, \quad (1.90)$$

Having only a single-mode, the HEOM expression Eq. (1.78) and the Liouvillian takes the corresponding form (1.81) (1.82) for $K_{max} = 1, 2$. By following the same process used in Section 1.3, we can compute each of the submatrices in (1.79), we find

$$D_{nm} = \begin{pmatrix} d_{nm} & 0 & 0 & 0 \\ 0 & d_{nm} - 2i\omega_0 & 0 & 0 \\ 0 & 0 & d_{nm} + 2i\omega_0 & 0 \\ 0 & 0 & 0 & d_{nm} \end{pmatrix}, \quad (1.91)$$

$$d_{nm} = -((n - m)i\omega + (n + m)\kappa), \quad (1.92)$$

$$A_n = \begin{pmatrix} 0 & 0 & g^2 n & 0 \\ 0 & 0 & 0 & g^2 n \\ g^2 n & 0 & 0 & 0 \\ 0 & g^2 n & 0 & 0 \end{pmatrix}, \quad (1.93)$$

$$B_m = \begin{pmatrix} 0 & g^2 m & 0 & 0 \\ g^2 m & 0 & 0 & 0 \\ 0 & 0 & 0 & g^2 m \\ 0 & 0 & g^2 m & 0 \end{pmatrix}, \quad (1.94)$$

$$C = \begin{pmatrix} 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ -1 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 \end{pmatrix}. \quad (1.95)$$

Plugging these into (1.81) and (1.82) and then solving the homogeneous system gives the following expressions for the normalized steady state

$$|\rho_{ss}^{(0,0)}\rangle\rangle = \left(\frac{\kappa^2 + (\omega - 2\omega_0)^2}{2(\kappa^2 + \omega^2 + 4\omega_0^2)}, 0, 0, \frac{\kappa^2 + (\omega + 2\omega_0)^2}{2(\kappa^2 + \omega^2 + 4\omega_0^2)} \right)^T \quad (1.96)$$

for $K_{max} = 1$ and

$$|\rho_{ss}^{(0,0)}\rangle\rangle = \left(\frac{2g^2 + \kappa^2 + (\omega - 2\omega_0)^2}{2(2g^2 + \kappa^2 + \omega^2 + 4\omega_0^2)}, 0, 0, \frac{2g^2 + \kappa^2 + (\omega + 2\omega_0)^2}{2(2g^2 + \kappa^2 + \omega^2 + 4\omega_0^2)} \right)^T \quad (1.97)$$

for $K_{max} = 2$. If these were converted back into matrices, the off-diagonal elements would be 0, which means the steady state has expected values for σ_x and σ_y which are equal to 0. This is expected as the Rabi model possesses a \mathbb{Z}_2 symmetry which implies that the parity is conserved and σ_x and σ_y do not preserve the parity [59]. The expected values for σ_z are then given by

$$\langle\sigma_z\rangle = \frac{-4\omega\omega_0}{\kappa^2 + \omega^2 + 4\omega_0^2} \quad (1.98)$$

for $K_{max} = 1$ and

$$\langle\sigma_z\rangle = \frac{-4\omega\omega_0}{2g^2 + \kappa^2 + \omega^2 + 4\omega_0^2} \quad (1.99)$$

for $K_{max} = 2$. Here we can see that the expression for $K_{max} = 2$ only brings a g^2 correction in the denominator. Analytically solving for higher cutoffs is not very practical as it requires solving

linear systems of increasing size. Furthermore HEOM is a numerical method in essence. Instead, we can thus compute numerically for different values of g and K_{max} as shown in Figures 1.11 and 1.12

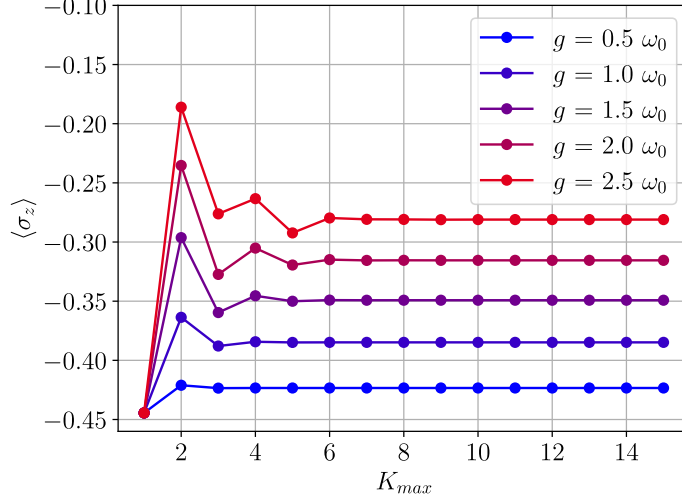


Figure 1.11: Expected value of σ_z in the case of the Rabi model for progressively larger values of K_{max} , where K_{max} ranges from 1 to 15, and for different values of g/ω_0 starting from 0.5 to 2.5 by increment of 0.5. The dimensionless parameters $\omega/\omega_0 = 1.0$ and $\kappa/\omega_0 = 2.0$ were used for the computations. The HEOM method requires a larger K_{max} for higher values of g to reach the desired precision.

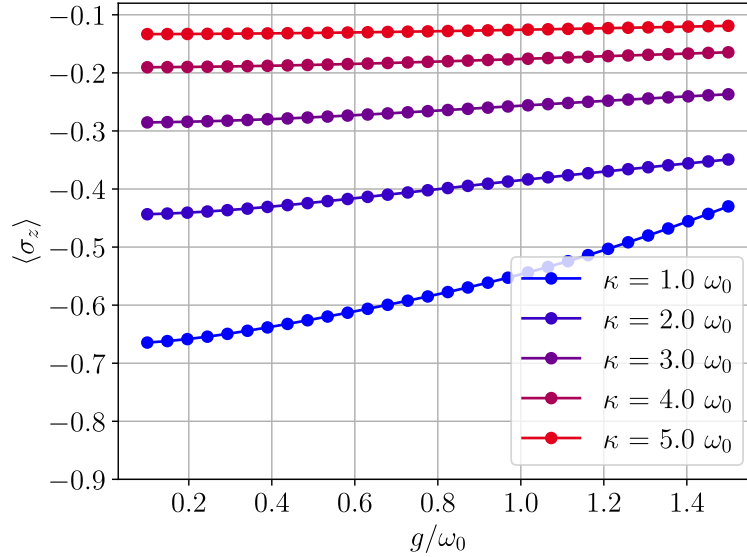


Figure 1.12: Expected value of σ_z in the case of the Rabi model for progressively larger values of g/ω_0 that ranges from 0.1 to 1.5 by increment of 0.05 for different values of κ/ω_0 , using $K_{max} = 15$ and $\omega/\omega_0 = 1.0$.

1.6 Summary

This chapter introduced the reader to open quantum systems. We explained why the ket vector formalism was insufficient and presented the density operator formalism. The quantum mechanics postulates for isolated systems were restated using this formalism. We then introduced the Gorini-Kossakowski-Sudarshan-Lindblad (GKSL) master equation governing Markovian open quantum systems. Subsequently, we defined the steady state. Following that, we discussed the Choi-Jamiołkowski isomorphism to map the density matrix to a ket vector. We provided a simple example of an open quantum system and demonstrated how to classically find its steady state; this same example will return in Chapter 3 for applying Ramusat and Savona’s quantum algorithm. Finally, we presented two methods for exploring non-Markovian dynamics: the Markovian embedding approach (used as a benchmark) and the HEOM method. The open quantum Rabi model was introduced as a non-Markovian example and will also be reused in Chapter 3 to benchmark this thesis’ quantum algorithm.

Chapter 2

Quantum information and computation

“Quantum mechanics is just completely strange and counterintuitive. We can’t believe that things can be here and there at the same time. And yet that’s a fundamental piece of quantum mechanics. So then the question is, life is dealing us weird lemons, can we make some weird lemonade from this?”

Seth Lloyd 2011

In this chapter, we will introduce the reader to the basics of quantum computation. In the first section, we will first explain the concepts of computational basis, quantum circuits and the general quantum gates. The next sections will be dedicated to presenting some well-known quantum algorithms which are frequently used as subroutines in general, namely the *quantum Fourier transform* (*QFT*) and the *quantum phase estimation* (QPE). These subroutines are precisely what we will use in the next chapter regarding the quantum algorithms for open quantum systems. The last section of this chapter is dedicated to the *Suzuki-Trotter expansion*, which is a method used to approximate specific quantum gates. The majority of this chapter originates from Nielsen and Chuang’s book, *Quantum Computation and Quantum Information* [15].

2.1 Fundamental concepts

2.1.1 Conventions and notations

We should mention that although we have spent the previous chapter of this thesis developing the formalism related to open quantum systems, in quantum computation, it is assumed that the state of the system is described using ket vectors and the isolated systems formalism¹. Furthermore, we work in what is called the *computational basis*, i.e., the eigenstates basis of n two-level systems,

¹Which is also why quantum computers have not yet realized their full potential. The hardware must be extremely well isolated for extended periods of time in order to produce reliable results.

which we call *qubits*. The corresponding states for a single qubit are

$$|0\rangle \equiv \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle \equiv \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad (2.1)$$

where $|0\rangle$ corresponds to the excited state and $|1\rangle$ is the ground state. In this context, the eigenstates for two qubits are given by

$$\begin{aligned} |0\rangle \otimes |0\rangle &\equiv |00\rangle \equiv \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, & |0\rangle \otimes |1\rangle &\equiv |01\rangle \equiv \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \\ |1\rangle \otimes |0\rangle &\equiv |10\rangle \equiv \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, & |1\rangle \otimes |1\rangle &\equiv |11\rangle \equiv \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \end{aligned}$$

where we use the vector and matrix representation of the eigenstates by using what we call the *Kronecker product*² which is simply the representation of the tensor product for *discrete bases*. It is defined for two 2×2 matrices by

$$\begin{aligned} A &= \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, & B &= \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}, \\ A \otimes B &\equiv \begin{pmatrix} a_{11}B & a_{12}B \\ a_{21}B & a_{22}B \end{pmatrix} = \begin{pmatrix} a_{11}b_{11} & a_{11}b_{12} & a_{12}b_{11} & a_{12}b_{12} \\ a_{11}b_{21} & a_{11}b_{22} & a_{12}b_{21} & a_{12}b_{22} \\ a_{21}b_{11} & a_{21}b_{12} & a_{22}b_{11} & a_{22}b_{12} \\ a_{21}b_{21} & a_{21}b_{22} & a_{22}b_{21} & a_{22}b_{22} \end{pmatrix}. \end{aligned} \quad (2.2)$$

Although we have only shown the explicit product of two 2×2 matrices for convenience, the product is valid for any combination of two matrices of any size (the two matrices do not have the same size in general). More specifically, the Kronecker product is the map $\otimes : \mathbb{C}^n \times \mathbb{C}^m \otimes \mathbb{C}^p \times \mathbb{C}^q \rightarrow \mathbb{C}^{n \times p} \times \mathbb{C}^{m \times q}$. We can see from this that the number of eigenstates available using n qubits is 2^n and that any operator acting on the total Hilbert space has a size $2^n \times 2^n$. We often use the notation $|j\rangle$, where j is an integer between 0 and $2^n - 1$, to represent the basis state $|j_0 j_1 \dots j_{n-1}\rangle \equiv |j_0\rangle \otimes |j_1\rangle \otimes \dots \otimes |j_{n-1}\rangle$, where $j_0 j_1 \dots j_{n-1}$ is the *binary representation* of j . For example, $|5\rangle \equiv |101\rangle \equiv |1\rangle \otimes |0\rangle \otimes |1\rangle$. If the system has more than 3 qubits, it is implicitly understood that the remaining qubits are in the state $|0\rangle$. This convention allows us to refer to a computational basis state using the *decimal representation* of the corresponding binary eigenstate.

Let us now introduce the notion of *Pauli strings* [60]. A Pauli string is defined as

$$P = \bigotimes_{j=0}^{N-1} \sigma^j, \quad (2.3)$$

with $\sigma^j \in \{\mathbb{1}_{2 \times 2}, \sigma_x, \sigma_y, \sigma_z\}$. These are operators that act on N qubits and form an *operator basis*, which means that any operator can be decomposed into this basis, using the Hilbert–Schmidt inner

²Although this product is called after Leopold Kronecker, the first person to define it was actually Johann Georg Zehfuss https://en.wikipedia.org/wiki/Kronecker_product.

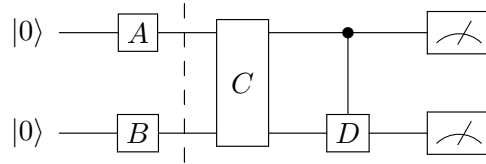
product³ defined as $\langle A|B\rangle_{HS} = \text{Tr}[A^*B]$. It is convenient to rewrite the Pauli matrices with the notations I, X, Y and Z and write any Pauli string in a compact form such that for example

$$\mathbb{1}_{2 \times 2} \otimes \sigma_x \otimes \sigma_y \otimes \sigma_z \equiv IXYZ, \quad (2.4)$$

which illustrates why they are called this way. It is used everywhere in quantum computing[61] notably, to efficiently implement any measurement operator on a quantum circuit. It is also an active field of research, as several individuals have recently proposed different algorithms to achieve a computational advantage for computing the decomposition more effectively [60], [62]. It allows to understand more intuitively how a gate is applied qubit-wise, and thus how to construct a specific operator in a more accessible way.

2.1.2 Quantum circuits and frequently used quantum gates

In this subsection, we will present the concept of *quantum circuits* and the different conventions employed. A quantum circuit is composed of horizontal lines (wires), each representing a qubit with the time going from left to right. At the initial time, the qubits, unless stated otherwise, are all assumed to be in the state $|0\rangle$. The circuit is appended with *quantum gates*, which are unitary operations⁴ that have matrix representations. Below, we give an example of a quantum circuit.



Here we explicitly indicate that the qubits are initially in the state $|0\rangle$, we apply the gate A on the first qubit and the gate B on the second one. There are two main conventions used in quantum computing, the first one, is to consider the lowermost qubit corresponding to the last qubit and the other convention, which is the one used in Qiskit⁵, is the exact opposite. In the first convention, the state obtained at the dashed line (which is simply used to visually separate different parts of a quantum circuit) would be $|\psi\rangle = A|0\rangle \otimes B|0\rangle$ ⁶. In the second convention (i.e., the Qiskit convention), we would have $|\psi\rangle = B|0\rangle \otimes A|0\rangle$. Unless stated otherwise, the first convention is what we use in this thesis. Then we apply the C gate which is a *two-qubit gate*. It is often implied that such gates cannot be factorized as a Kronecker product of two one-qubit gate (i.e., $C \neq C_1 \otimes C_2$)⁷. We then apply what we call a controlled-gate, where the black dot signifies that the first qubit is the “control” qubit and the second the “target” qubit, in the sense that the gate D will only be applied on the second qubit if the first qubit is in the state $|1\rangle$ (otherwise nothing happens). We have to emphasize that no destructive measurement is applied on the first qubit to determine whether the D gate is applied or not. If the state of the qubit before this gate is $|x\rangle \otimes |y\rangle = (x_0|0\rangle + x_1|1\rangle) \otimes |y\rangle$, then the state after the controlled- D gate is given by $x_0|0\rangle \otimes |y\rangle + x_1|1\rangle \otimes D|y\rangle$. The attentive reader

³Be careful that the bracket notation is used in the context of an inner product, these are operators and not ket vectors.

⁴The operations are unitary because in quantum mechanics, these are the operations which conserve the norm of a quantum state, which is interpreted as the total probability and must always be equal to 1.

⁵Qiskit is a library in Python to build and simulate quantum circuits <https://github.com/Qiskit/qiskit>. Although it was not done in this thesis, these circuits can also be sent on IBM quantum computers.

⁶From this, we understand that applying the gate A on the first qubit is equivalent to applying the operator $A \otimes \mathbb{1}$ on the whole circuit, where $\mathbb{1}$ is the identity operation that leaves unchanged the state of a qubit.

⁷However, it is always possible to find a succession of one-qubit gates such that $C = \prod_j C_{1j} \otimes C_{2j}$ when the C gate creates no entanglement.

will have understood that the controlled-gates are what we use to create entanglement in quantum circuits. The last gates, which are in fact not gates since they are not unitary, are measurement applied in the computational basis. For example, if we assume that our state before the measurement is $|\psi\rangle = \alpha|0\rangle \otimes |0\rangle + \beta|0\rangle \otimes |1\rangle + \gamma|1\rangle \otimes |1\rangle$ with $|\alpha|^2 + |\beta|^2 + |\gamma|^2 = 1$, then the measurement outcomes are 00, 01, 11, with probabilities $|\alpha|^2$, $|\beta|^2$ and $|\gamma|^2$, respectively. Furthermore, the state is then projected on the eigenspace associated to the measurement outcome.







Additionally, we use the following notation

=====

to designate a wire carrying a classical bit and

——/n——

to indicate that the wire carries n qubits. The table below presents the frequently used one-qubit gates along with their matrix representations:

Gate	Circuit	Matrix
Hadamard		$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$
Pauli-X (NOT gate)		$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
Pauli-Y		$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$
Pauli-Z		$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
Phase (S)		$\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$
$\pi/8$ (T)		$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$

Among the above gates, the Hadamard gate is one of the most important in quantum computing, owing to the fact that it is the gate required to create superpositions. Indeed, if we apply the gate on the states $|0\rangle$ and $|1\rangle$ we have

$$H|0\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle, \quad (2.5a)$$

$$H|1\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle, \quad (2.5b)$$

where we assume that the context is enough to distinguish whether H is used to designate the Hamiltonian or the Hadamard gate. From its matrix representation, we can effortlessly see that the

Hadamard gate is Hermitian, it is also unitary since

$$HH^\dagger = HH = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \mathbb{1}, \quad (2.6)$$

which means that applying H on (2.5a) and (2.5b) will give $|0\rangle$ and $|1\rangle$ respectively. Although these are less important, the Pauli gates are also worth mentioning, these are exactly the same operators as in the previous chapter however, they are referred to using I, X, Y, Z instead of $\mathbb{1}, \sigma_x, \sigma_y, \sigma_z$, which is simply a convention in quantum computation. The Pauli-X gate is also referred to as the NOT gate since it flips $|0\rangle$ and $|1\rangle$ (which is also why it can be used to represent a bit flip) and in general for a state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, we have

$$X|\psi\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix} = \beta|0\rangle + \alpha|1\rangle. \quad (2.7)$$

The Pauli matrices are also the generators of the *special unitary group of degree 2* or $SU(2)$, which is the group containing unitary 2×2 matrices and of determinant 1. In other words, all unitary matrices 2×2 , i.e., any one-qubit quantum gate can be generated by computing the matrix exponential of a sum of Pauli matrices. It is understood by the fact that the Pauli matrices form a basis of the Hermitian basis vector space and that any unitary matrix can be written as the exponential of a Hermitian matrix times i . This last assertion is easy to understand when we consider a Hermitian matrix M , and U such that

$$U = e^{iM}. \quad (2.8)$$

If we compute the Hermitian conjugate of U , we obtain

$$U^\dagger = (e^{iM})^\dagger = e^{-iM^\dagger} = e^{-iM}, \quad (2.9)$$

which directly yields

$$UU^\dagger = \mathbb{1}, \quad (2.10)$$

which is the definition of a unitary matrix. In particular, each Pauli matrix generates the rotation on the *Bloch sphere* along its corresponding axis, for example, a rotation of angle θ along the z -axis⁸ is obtained by computing

$$R_z(\theta) = e^{-\frac{i}{2}\theta\sigma_z} = \begin{pmatrix} e^{-\frac{i}{2}\theta} & 0 \\ 0 & e^{\frac{i}{2}\theta} \end{pmatrix}. \quad (2.11)$$



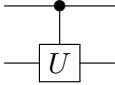
With rotation operators, it becomes possible to describe any unitary operator, for example, any arbitrary one-qubit gate can be decomposed the following way

$$U = e^{i\alpha} R_z(\beta) R_y(\gamma) R_z(\delta), \quad (2.12)$$

with $\alpha, \beta, \gamma, \delta \in [0, 2\pi]$.

We can similarly introduce some of the notorious two-qubit gates, we have the following

⁸It is not direct to see how the rotation comes along on the Bloch sphere, the x, y, z coordinates are obtained by computing the corresponding expected values of $\langle\sigma_x\rangle, \langle\sigma_y\rangle, \langle\sigma_z\rangle$. As such, the rotation is not applied directly on the x, y, z coordinates but on the ket vector from which we compute these coordinates.

Gate	Circuit	Matrix
controlled-NOT		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
SWAP		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
controlled- U		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & U_{11} & U_{12} \\ 0 & 0 & U_{21} & U_{22} \end{bmatrix}$

where the controlled- U gate at the end is there to indicate the representation of an arbitrary controlled-gate and U_{ij} corresponds to the element i, j of the matrix representation of U . The target sign for the controlled-NOT gate is used to indicate a modulo two addition, it is exactly equivalent to the X Pauli matrix and both are used interchangeably. The cNOT gate is remarkable as every arbitrary controlled-gate is built using cNOT gates and as such, it is the true source of entanglement in quantum computing. The SWAP gate is implemented using 3 cNOT gates and as its name suggests, it swaps two qubits, for example

$$\text{SWAP}(|x\rangle \otimes |y\rangle) = |y\rangle \otimes |x\rangle. \quad (2.13)$$

Furthermore, the entanglement that could exist between one of the two qubits with a third qubit is passed on with the SWAP gate. The idea behind every quantum algorithm is to use this set of gates to obtain specific desired results. The quantum advantage is generally achieved through the use of the Hadamard gate to generate superpositions and the cNOT gate to create entanglement.

2.2 Quantum Fourier Transform

In this section, we will introduce our first quantum algorithm, namely the *quantum Fourier transform* (QFT). Although it is notoriously associated to Peter Shor due to its famous application in his algorithm [63], its discovery is associated to Coppersmith [64]. The quantum Fourier transform is a key *subroutine* in many quantum algorithms and in particular in the *quantum phase estimation* algorithm which will be presented in the next section. The QFT possesses a spectacular advantage compared to its classical counterpart, the *fast Fourier transform* FFT . The former performs with an exponential advantage compared to the former. However, this apparent advantage warrants careful consideration, as we will discuss in more detail later.

2.2.1 Definitions and properties

Classically, the discrete Fourier transform is a mathematical operation that takes a vector of complex numbers x_0, \dots, x_{N-1} with N being the length of the vector which remains fixed. It then outputs the vector of complex numbers y_0, \dots, y_{N-1} where

$$y_k \equiv \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2i\pi jk/N}. \quad (2.14)$$

There are different conventions being used. In this case the exponent has a positive pre-factor in its argument but the most common convention has a negative one. Furthermore, some remove the $1/\sqrt{N}$ factor and append a $1/N$ factor in the inverse discrete Fourier transform. The quantum Fourier transform does exactly the same thing as a discrete Fourier transform but with different notations. It acts upon the ket vectors of the computational basis ($|0\dots 00\rangle, |0\dots 01\rangle, \dots |1\dots 11\rangle$) such that

$$\mathcal{QFT} |j\rangle \equiv \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2i\pi jk/N} |k\rangle. \quad (2.15)$$

If we consider any ket vector $|x\rangle$ that can be expressed in the computational basis $|j\rangle$

$$|x\rangle = \sum_{j=0}^{N-1} x_j |j\rangle, \quad (2.16)$$

then the quantum Fourier transform will output the ket vector $|y\rangle$

$$\begin{aligned} |y\rangle &= \mathcal{QFT} \sum_{j=0}^{N-1} x_j |j\rangle \\ &= \sum_{j=0}^{N-1} x_j \mathcal{QFT} |j\rangle \\ &= \sum_{j=0}^{N-1} x_j \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2i\pi jk/N} |k\rangle \\ &= \sum_{k=0}^{N-1} \left(\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2i\pi jk/N} \right) |k\rangle \\ &= \sum_{k=0}^{N-1} y_k |k\rangle, \end{aligned} \quad (2.17)$$

where the y_k are then given by

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2i\pi jk/N}, \quad (2.18)$$

which is exactly the same expression as in the classical discrete case. This transformation is unitary,

it can be seen by proving that the ket vector $|y\rangle$ conserves the norm of $|x\rangle$

$$\begin{aligned}
\langle y|y\rangle &= \sum_{k=0}^{N-1} \left(\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j^* e^{-2i\pi jk/N} \right) \left(\frac{1}{\sqrt{N}} \sum_{j'=0}^{N-1} x_{j'} e^{2i\pi j'k/N} \right) \\
&= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} \sum_{j'=0}^{N-1} x_j^* x_{j'} e^{2i\pi(j'-j)k/N} \\
&= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} |x_j|^2 + \frac{1}{N} \sum_{j=0}^{N-1} \sum_{j' \neq j}^{N-1} x_j^* x_{j'} \frac{1 - e^{2i\pi(j'-j)}}{1 - e^{2i\pi(j'-j)/N}} \\
&= \sum_{j=0}^{N-1} |x_j|^2 \\
&= \langle x|x \rangle,
\end{aligned} \tag{2.19}$$

where we used the formula for a geometric sum to go from the second to the third line and separating the case where $j = j'$ as the formula cannot be applied in this case. The second term in the third line vanishes because $j - j'$ will always give an integer which means that the numerator will always be equal to 0 while the denominator will always be different from 0 because of the $1/N$ factor in the exponential.

The constant N is usually expressed in terms of the number of qubits t of the system as $N = 2^t$. Furthermore, we represent the number j in binary representation $j = j_0 j_1 \dots j_{t-1} \equiv j_0 2^{t-1} + j_1 2^{t-2} + \dots + j_{t-1} 2^0$. We will also use the notation $0.j_1 j_2 \dots j_t$ to represent the binary fraction $j_1 2^{-1} + j_2 2^{-2} + \dots + j_t 2^{-t}$ so that we can rewrite the quantum Fourier transform in its product form

$$QFT |j_1 j_2 \dots j_t\rangle = \frac{(|0\rangle + e^{2i\pi 0.j_t} |1\rangle) (|0\rangle + e^{2i\pi 0.j_{t-1} j_t} |1\rangle) \dots (|0\rangle + e^{2i\pi 0.j_1 j_2 \dots j_t} |1\rangle)}{2^{t/2}}. \tag{2.20}$$

This equality can be proven by starting from (2.15)

$$\begin{aligned}
QFT |j\rangle &= \frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} e^{2i\pi jk/2^t} |k\rangle \\
&= \frac{1}{\sqrt{2^t}} \sum_{k_1=0}^1 e^{2i\pi jk_1 2^{-1}} |k_1\rangle \sum_{k=0}^{2^{t-1}-1} e^{2i\pi jk/2^{t-1}} |k\rangle \\
&= \frac{1}{\sqrt{2^t}} \sum_{k_1=0}^1 e^{2i\pi jk_1 2^{-1}} |k_1\rangle \sum_{k_2=0}^1 e^{2i\pi jk_2 2^{-2}} |k_2\rangle \dots \sum_{k_t=0}^1 e^{2i\pi jk_t 2^{-t}} |k_t\rangle \\
&= \frac{1}{\sqrt{2^t}} \bigotimes_{l=1}^t \left(\sum_{k_l=0}^1 e^{2i\pi jk_l 2^{-l}} |k_l\rangle \right) \\
&= \frac{1}{\sqrt{2^t}} \bigotimes_{l=1}^t \left(|0\rangle + e^{2i\pi j 2^{-l}} |1\rangle \right) \\
&= \frac{(|0\rangle + e^{2i\pi 0 \cdot j_t} |1\rangle) (|0\rangle + e^{2i\pi 0 \cdot j_{t-1} j_t} |1\rangle) \dots (|0\rangle + e^{2i\pi 0 \cdot j_1 j_2 \dots j_t} |1\rangle)}{2^{t/2}},
\end{aligned}$$

where we factorized the first qubit to get from the first to the second line, and then we did the same operation for all the remaining qubits to get from the second to the third line. To go from the fifth to the last line, we used the fact that only the binary fractions of $j2^{-l}$ will have relevant contributions, that is, $0.j_{t-l+1}j_{t-l+2}\dots j_t$.

2.2.2 Circuit implementation

With the quantum Fourier transform being linear, the formulation (2.20) allows us to almost directly guess the associated quantum circuit, which is conveniently represented in Figure 2.1. We must bear in mind that SWAP gates are required after the circuit. The R_k gates are defined as follows

$$R_k \equiv \begin{pmatrix} 1 & 0 \\ 0 & e^{2i\pi/2^k} \end{pmatrix}. \quad (2.21)$$

To prove that the circuit of Fig. 2.1 works, we can compute progressively, starting with the first qubit. We can rewrite the effect of the Hadamard gate as

$$H |x_1\rangle = \frac{1}{\sqrt{2}} (|0\rangle + e^{2i\pi 0 \cdot x_1} |1\rangle), \quad (2.22)$$

owing to the fact that $H |0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)$ and $H |1\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$. Applying the controlled- R_2 gate on the first qubit yields

$$\begin{aligned}
cR_2 \frac{1}{\sqrt{2}} (|0\rangle + e^{2i\pi 0 \cdot x_1} |1\rangle) \otimes |x_2\rangle &= \frac{1}{\sqrt{2}} (|0\rangle + e^{2i\pi 0 \cdot x_1} e^{2i\pi x_2/2^2} |1\rangle) \otimes |x_2\rangle \\
&= \frac{1}{\sqrt{2}} (|0\rangle + e^{2i\pi 0 \cdot x_1 x_2} |1\rangle) \otimes |x_2\rangle,
\end{aligned} \quad (2.23)$$

since $x_1 2^{-2} = 0.0x_2$. Following the same procedure, we apply the remaining gates on the first qubit and we have the state

$$\frac{1}{\sqrt{2}} (|0\rangle + e^{2i\pi 0 \cdot x_1 x_2 \dots x_t} |1\rangle) \otimes |x_1 x_2 \dots x_t\rangle. \quad (2.24)$$

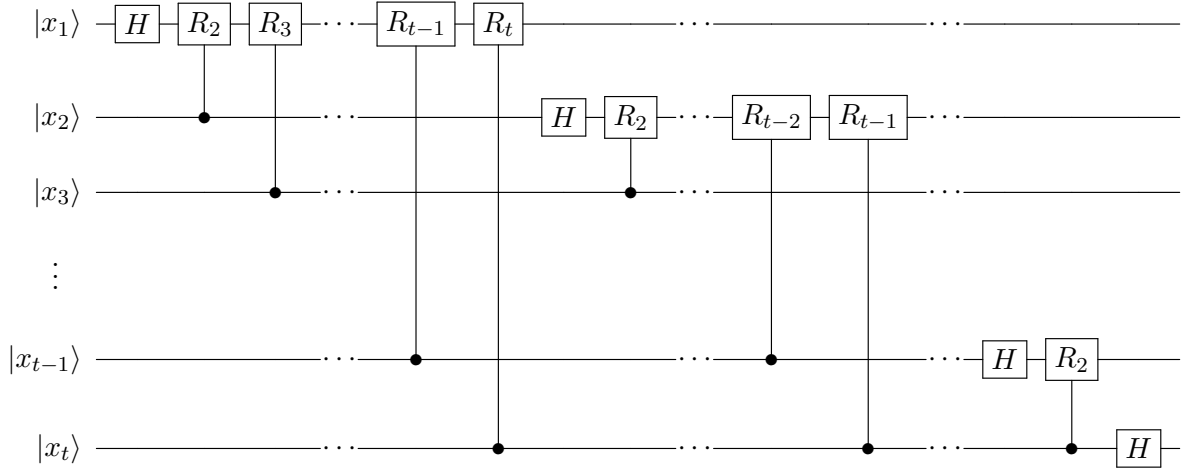


Figure 2.1: Quantum Fourier transform circuit for t qubits. Although it is not explicitly shown, SWAP gates are required at the end of the circuit to set the qubits in the right order.

We can do the same type of computation on the other qubits, and we trivially get the final state

$$\mathcal{QFT} |x_1 x_2 \dots x_t\rangle = \frac{1}{2^{t/2}} (|0\rangle + e^{2i\pi 0.x_1 x_2 \dots x_t} |1\rangle) (|0\rangle + e^{2i\pi 0.x_2 \dots x_t} |1\rangle) \dots (|0\rangle + e^{2i\pi 0.x_t} |1\rangle), \quad (2.25)$$

which is the exact same state as (2.20) but in reverse order, hence the use of SWAP gates at the end. As said previously, here we assumed the initial state to be an eigenstate of the computational basis, however, as the \mathcal{QFT} is linear, any linear combination of states will go through and the circuit will remain valid. It is also necessary to point out that the other commonly used convention discussed in the previous section, which consists in considering that the lowermost qubit on a circuit (which would be $|x_t\rangle$ in our case) corresponds to the leftmost qubit in the ket vector, would lead to the circuit being upside down.

It is useful to introduce an explicit compact notation for the \mathcal{QFT} which encompasses all the gates in the circuit in Fig. 2.1. We easily have

$$\mathcal{QFT} \equiv \prod_{i=1}^t \left(\prod_{j=i+1}^t c^j R_{j-i+1}^i \right) H_i, \quad (2.26)$$

where the notation $c^n R_k^m$ means that we apply the controlled- R_k gate on the m^{th} qubit with the n^{th} qubit being the control.

2.2.3 Complexity analysis and further discussions

We can also easily evaluate the complexity of the quantum Fourier transform, we apply Hadamard gates on all qubits, furthermore, we recognize that the controlled- R_k gates follow an arithmetic sum and finally, we need $t/2$ SWAP gates (Technically it is $t/2$ if t is even and $(t-1)/2 \approx t/2$ if t is

odd) and a SWAP gate requires 3 cNOT gates which means that in total we have

$$\begin{aligned}\#_{\text{gates}} &= t + \frac{1}{2}t(t-1) + \frac{3}{2}t \\ &= \frac{1}{2}t^2 + 2t.\end{aligned}\tag{2.27}$$

Thus, we have that the circuit of the algorithm has a $\mathcal{O}(t^2)$ scaling. By way of comparison, the \mathcal{FFT} (fast Fourier transform) has a scaling of $\mathcal{O}(t^{2t})$ [65]. This represents an exponential speedup over the best-known classical algorithm, however, it is important to stress that the \mathcal{QFT} cannot be directly used in everyday life for digital applications such as signal processing for example. Indeed, the amplitudes of the state (2.25) are not known and require an exponential number of measurements to know them with sufficient precision to be converted in classical bits, making it unsuitable for such purposes. Nonetheless, for applications for which only one information has to be extracted, the \mathcal{QFT} can generally be used to outperform classical algorithms. It is notably the case for the renowned Shor’s algorithm [63] which is a quantum algorithm for prime factorization of integers. Shor’s algorithm uses the quantum Fourier transform to find the period of a function, effectively extracting only one information.

Unfortunately, in the present day, quantum processors have layouts with limited connectivity between qubits. For example, the typical layout of an IBM quantum processor can be seen in Fig. 2.2. For this kind of processor, qubits have at most three neighbors and most only have two, meaning they are on a 1-D array⁹. Knowing this, it is rather evident that the circuit in Fig. 2.1 cannot be directly used or has to be appended with a considerable amount of SWAP gates, and thus cNOT gates, to be effectively implemented. As a consequence, the real number of gates, in the case of a 1-D array (linear nearest neighbor), would scale as $\mathcal{O}(t^3)$. It is easy to understand why; when summing over the R_k gates, we have to multiply by the number of lines that separate two qubits. Instead of summing over integers, we thus sum over squared integers which scale cubically. This problem is well known, and multiple people [67]–[71] have proposed adaptations of the quantum Fourier transform for various qubits layouts and especially for linear nearest neighbor architectures. The most efficient implementation thus far, was recently proposed by Klaver et al.(2024) [72]. They introduced a formalism that leverages the tracking of parity quantum information to enable the implementation of algorithms on devices with restricted connectivity, eliminating the need for additional qubits, SWAP operations, or qubit shuttling (i.e. physically moving the qubits so that they can interact). In their analysis, the exact total number of gates required is $\#_{\text{gates}} = \frac{3}{2}t^2 + \frac{5}{2}t - 3$, furthermore, the total depth (the minimum number of layers of quantum gates required to implement the circuit, a layer being a group of gates that can be applied simultaneously) is $\#_{\text{depth}} = 5t - 3$. Having a lower depth is an advantage in the *NISQ era* (noisy intermediate-scale quantum era) as it reduces the execution time that is needed.

To conclude this section, it is worth noting that the basic algorithm, when not restricted by hardware connectivity limitations, has been significantly improved by various researchers [73], [74]. These improvements often leverage approximation techniques, such as omitting small-angle controlled rotations, to reduce the gate cost. Such approaches have successfully reduced the scaling of the algorithm to $\mathcal{O}(t \log(t))$ while still maintaining an acceptable precision. To the best of our knowledge, Ref. [75] remains the only work to develop an approximate quantum Fourier trans-

⁹The way it works is that the chain that minimizes the total error is selected so in the end, qubits have only two neighbors. The others may be used for quantum error correction techniques.

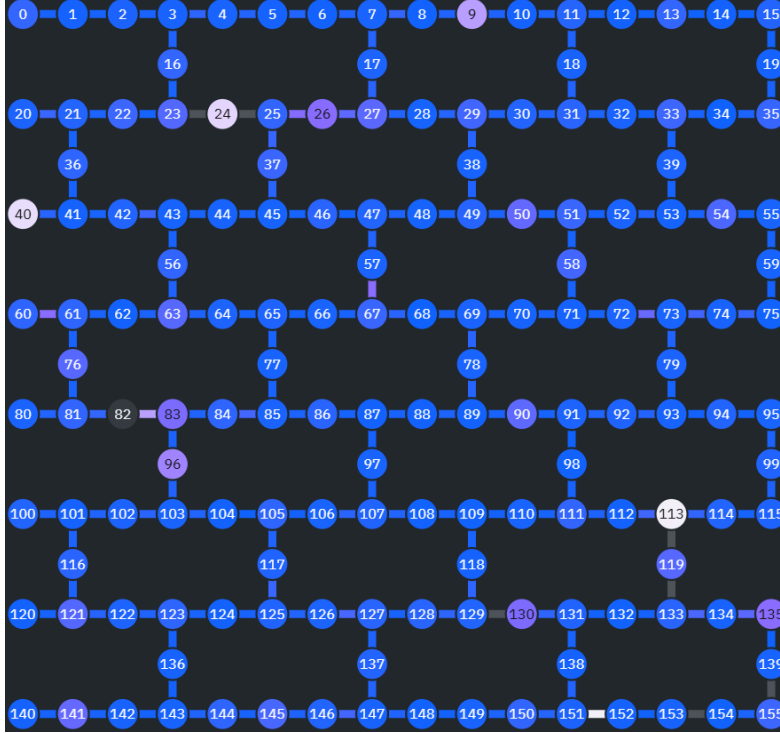


Figure 2.2: Qubits layout of an IBM quantum processor “Heron r2” (Picture taken from [66]). The qubits colors indicate the readout assignment errors and the colors of the segments indicate the connection errors for cNOT gates.

form specifically designed for linear nearest neighbor architectures, achieving a similar scaling of $\mathcal{O}(t \log(t))$ (although it is very likely that the exact number of gates exceeds that of Refs. [73], [74]).

2.3 Quantum Phase Estimation

In this section, we will introduce our second quantum algorithm, one which is central to this thesis, the *quantum phase estimation* (QPE). It was developed by Kitaev [76] and similarly with the quantum Fourier transform, it is used as a subroutine for other, more complex quantum algorithms.

2.3.1 Overview

Let us suppose a unitary operator U possesses an eigenvector $|u_j\rangle$ corresponding to an eigenvalue $e^{2i\pi\varphi_j}$. Any eigenvalue of any unitary operator U can always be written this way since the unitary operator can always be written as $U = e^{iM}$, with M a Hermitian matrix, which has real eigenvalues. The purpose of the quantum phase estimation is to estimate the value of φ_j , which is assumed to be in the interval $[0, 1]$. In order to do so, it is assumed that we have access to *oracles*, i.e., *black boxes* subroutines capable of preparing the eigenstate $|u_j\rangle$ and performing controlled- U^{2^j} operations. The algorithm is composed of two quantum registers, one containing t qubits and corresponding to the eigenvalue register used to store the eigenvalue. The second is composed of n qubits and encodes the eigenvector $|u_j\rangle$ in the computational basis as a *quantum state*. This second part is important, we could naively expect that stocking the eigenvector $|u_j\rangle$ would require $2L$ times more qubits than

the eigenvalue, with L the length of the matrix representation of U ¹⁰. However, this is not the case, the eigenvector register stores the state by specifically using the quantum nature of the qubits. To illustrate this, we will restrain ourselves to a single qubit for simplicity: to classically store the state of the qubit (as a pure state), we would need two real numbers¹¹, while we trivially need only one qubit to represent a qubit (as a quantum state). In general, the number of qubits in this register is fixed by the size of U , such that $n = \log_2(L)$. Conversely the number of qubits in the eigenvalue register is determined by the desired precision, using t qubits will output a t -bit estimation of φ_j . Furthermore, as we will uncover below, a higher number of qubits will also infer on the probability of success.

2.3.2 Circuit and detailed description

The associated circuit of the quantum phase estimation is illustrated in Fig 2.3. We assume that the qubits of the eigenvalue register are all in the state $|0\rangle$ and that the eigenvector register has the state $|u_j\rangle$ already encoded (via the use of the oracle). We then apply Hadamard gates on all the qubits of the eigenvalue register so that we have a superposition of all possible states and the state at the first dashed line is given by

$$\begin{aligned} |\psi\rangle &= \prod_{k=0}^{t-1} H_k |0_k\rangle \otimes |u_j\rangle \\ &= \frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} |k\rangle \otimes |u_j\rangle. \end{aligned} \quad (2.28)$$

We must then apply the controlled-unitary gates onto our state $|\psi\rangle$, however, since $|u_j\rangle$ is an

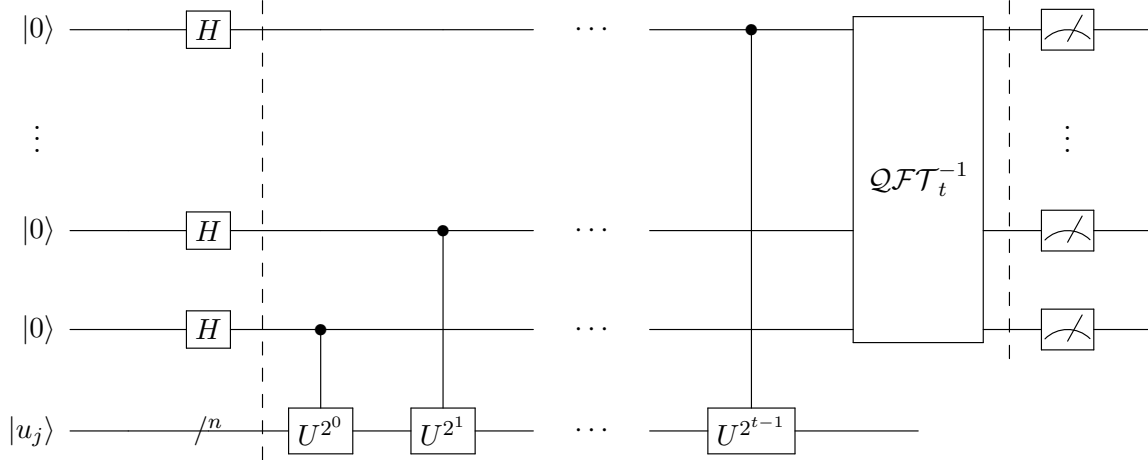


Figure 2.3: Quantum circuit for the quantum phase estimation. The algorithm assumes the eigenvector quantum register is prepared in the state $|u_j\rangle$ and it outputs the best t -bit estimation of φ_j with a probability of success greater than $4/\pi^2$.

eigenstate of U , passing it through a single gate U^{2^k} will output

$$U^{2^k} |u_j\rangle = e^{2i\pi 2^k \varphi_j} |u_j\rangle. \quad (2.29)$$

¹⁰We would use $2L$ and not L owing to the fact that the amplitudes are complex but φ_j is real.

¹¹From a ket vector formalism, we have $\alpha|0\rangle + \beta|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$ and since α and β are complex numbers, we would need 3 real numbers to store them. However, states related to a global phase factor are undistinguishable from one another and thus, in practice we need only two real numbers to represent a specific state.

For simplicity, let us suppose that we have an eigenstate of the computational basis in the eigenvalue register $|x\rangle = |x_{t-1}\rangle \otimes |x_{t-2}\rangle \otimes \dots \otimes |x_0\rangle$, with the x_k 's being either 0 or 1. Applying the controlled- U gates would lead to

$$\begin{aligned} \prod_{k=0}^{t-1} c_k U^{2^k} |x\rangle \otimes |u_j\rangle &= e^{2i\pi 2^{t-1} x_{t-1} \varphi_j} |x_{t-1}\rangle \otimes \dots \otimes e^{2i\pi 2^0 x_0 \varphi_j} |x_0\rangle \otimes |u_j\rangle \\ &= \left(\bigotimes_{k=1}^t e^{2i\pi 2^{t-k} x_{t-k} \varphi_j} |x_{t-k}\rangle \right) \otimes |u_j\rangle \\ &= e^{2i\pi x \varphi_j} |x\rangle \otimes |u_j\rangle, \end{aligned} \quad (2.30)$$

where for the first line, we simply have rewritten the effect of a controlled- U^{2^k} gate as adding a phase factor of $e^{2i\pi 2^k \varphi_j}$ conditioned on the value of x_k , and then we simply used that by definition we have

$$x = \sum_{k=0}^{t-1} 2^k x_k = \sum_{k=1}^t 2^{t-k} x_{t-k}, \quad (2.31)$$

$$|x\rangle = |x_{t-1}\rangle \otimes \dots \otimes |x_0\rangle = \bigotimes_{k=1}^t |x_{t-k}\rangle, \quad (2.32)$$

to go from the second to the third line. Applying now the controlled- U gates on the state (2.29) and using the relation (2.30), we find

$$\prod_{l=0}^{t-1} c_l U^{2^l} \frac{1}{\sqrt{2}} \sum_{k=0}^{2^t-1} |k\rangle \otimes |u_j\rangle = \frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} e^{2i\pi k \varphi_j} |k\rangle \otimes |u_j\rangle. \quad (2.33)$$

At this point, what we have done, is that we have created a superposition of all the integer values possible between 0 and $2^t - 1$ and assigned a specific phase factor $e^{2i\pi k \varphi_j}$ to each state $|k\rangle$ in the superposition. To give the reader an intuition of what we will do next, let us assume that φ_j admits a perfect binary fraction decomposition $\varphi_j = 0.x_1 x_2 \dots x_t$ with t bits. The phase factor will be equal to 1 when $k = x_t 2^t + \dots + x_1 2^1$ or an integer multiple of this. The period of the phase factor in this case, is exactly the value of φ_j multiplied by 2^t . The attentive reader will already have understood that we need to apply the quantum Fourier transform (more precisely, the inverse quantum Fourier transform), to map the Fourier basis ¹² onto the computational basis. The inverse quantum Fourier transform is simply defined by the same expression (2.15) but with a negative sign in the exponential. Applying it on the t qubits of the eigenvalue register yields

$$\begin{aligned} \mathcal{QFT}_t^{-1} \frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} e^{2i\pi k \varphi_j} |k\rangle \otimes |u_j\rangle &= \frac{1}{\sqrt{2^t}} \sum_{l=0}^{2^t-1} \left(\frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} e^{2i\pi k \varphi_j} e^{-2i\pi k l / 2^t} \right) |l\rangle \otimes |u_j\rangle \\ &= \sum_{l=0}^{2^t-1} \sum_{k=0}^{2^t-1} \frac{1}{2^t} e^{2i\pi k (\varphi_j - l/2^t)} |l\rangle \otimes |u_j\rangle \\ &= \sum_{l=0}^{2^t-1} \frac{1}{2^t} \frac{1 - e^{2i\pi (\varphi_j 2^t - l)}}{1 - e^{2i\pi (\varphi_j - l/2^t)}} |l\rangle \otimes |u_j\rangle, \end{aligned} \quad (2.34)$$

¹²In the case of quantum computing and the quantum Fourier transform, the Fourier basis can be interpreted as the phase factor basis from an intuitive point of view.

where we used the geometric sum formula to go from the second to the third line. However, the use of this formula is only valid when there exists no l such that $\varphi_j - l/2^t = 0$, which is equivalent to φ_j admitting a perfect binary fraction decomposition with t bits. If the geometric sum formula cannot be used, it is easy to see from the second line in (2.34) that the precise l for which the argument in the exponential is 0, will output a factor equal to 1, since we sum $1/2^t$, 2^t times. All other contributions will be 0 which can be seen either by the fact that the quantum Fourier transform and its inverse are unitary and as such they preserve the norm of the vector. Or it can be seen in the last line in (2.34) that the numerator is 0 because $\varphi_j 2^t$ is equal to an integer and we removed the specific l that is equal to this integer thus the exponential always gives 1.

In the general case where φ_j does not admit an exact binary fraction with t bits, we can rewrite it as

$$\varphi_j = \frac{a}{2^t} + \delta, \quad (2.35)$$

where a is the best t -bit approximation of $\varphi_j 2^t$ and we thus have $-\frac{1}{2} \leq \delta 2^t \leq \frac{1}{2}$. We will prove that in this case, we have a sufficiently high probability of measuring a in the eigenvalue register. We will define the coefficient in (2.34) as

$$\alpha_l = \frac{1}{2^t} \frac{1 - e^{2i\pi(\varphi_j 2^t - l)}}{1 - e^{2i\pi(\varphi_j - l/2^t)}}, \quad (2.36)$$

and we have in particular that

$$\begin{aligned} \alpha_a &= \frac{1}{2^t} \frac{1 - e^{2i\pi(\varphi_j 2^t - a)}}{1 - e^{2i\pi(\varphi_j - a/2^t)}} \\ &= \frac{1}{2^t} \frac{1 - e^{2i\pi 2^t \delta}}{1 - e^{2i\pi \delta}}. \end{aligned} \quad (2.37)$$

The probability of measuring $|a\rangle$ in the eigenvalue register is then given by

$$\begin{aligned} \Pr(a) &= |\alpha_a|^2 \\ &= \frac{1}{2^{2t}} \left| \frac{1 - e^{2i\pi 2^t \delta}}{1 - e^{2i\pi \delta}} \right|^2 \\ &= \frac{1}{2^{2t}} \left(\frac{\sin(\pi 2^t \delta)}{\sin(\pi \delta)} \right)^2 \\ &\geq \frac{1}{2^{2t}} \left(\frac{\sin(\pi 2^t \delta)}{\pi \delta} \right)^2 \\ &\geq \frac{1}{2^{2t}} \left(\frac{2^{t+1} \delta}{\pi \delta} \right)^2 \\ &= \frac{4}{\pi^2} \approx 0.405, \end{aligned} \quad (2.38)$$

where we used the relation $|1 - e^{2ix}|^2 = 4 \sin^2(x)$ to go from the second to the third line, we then used $\sin^2(x) \leq x^2$ to establish a lower bound for $\Pr(a)$. To find the final lower bound, we used the fact that $|2^t \delta| \leq 1/2$ along with the inequality $x \leq \pi \sin(x)$ for $x \leq \pi/2$. The value of the lower bound itself is not very important, what truly matters is that it does not depend on any parameter and is thus $\mathcal{O}(1)$. As a consequence of this, repeating the procedure only a couple of times will assure us that the correct eigenvalue is measured.

2.3.3 Further discussions

In this section, we have established that quantum phase estimation allows us to compute the eigenvalue of a known eigenvector by assuming that we have access to oracles. If the eigenvalue admits an exact binary fraction decomposition using t bits, then we have a 100% chance of success. Furthermore, we have proven that even if the eigenvalue does not admit an exact binary fraction decomposition, we still have at the very least a $4/\pi^2$ probability of measuring the best t -bit estimation of φ_j . While it is not proven in this section¹³, if the prepared state in the eigenvector register is not exactly correct, we still have a good chance of measuring the corresponding eigenvalue.

Although the gate cost of the inverse quantum Fourier transform is known, the scaling of quantum phase estimation cannot be accurately assessed due to the fact that the oracles, especially the controlled- U gates, are the leading factors in the scaling and are unknown. To improve the quantum phase estimation, in scaling, precision and resilience (to noise), it is rather its subroutines that are improved, such as the inverse quantum Fourier transform discussed in the previous section, and the methods used to build the oracles. One such method is the *Suzuki-Trotter expansion* which is presented in the next section. Some researchers [77]–[79] have proposed direct improvements upon the quantum phase estimation algorithm. These methods will not be discussed further as they are not particularly relevant to the use of the QPE in this thesis.

To conclude this section, it is worth noting that the quantum phase estimation in its primary formalism is not very useful and it might not even provide an advantage over classical methods since determining the eigenvalue from a known eigenvector scales with the length of the matrix¹⁴. Its usefulness lies in fact with its ability to create entanglement between an eigenstate and its corresponding eigenvalue. Among the many applications of the quantum phase estimation, we can cite Shor’s algorithm and the Harrow-Hassidim-Lloyd (HHL) algorithm [80], which is a quantum algorithm for solving linear systems of equations. The latter is one of the most important algorithms in quantum computing as it finds applications in many fields in science and engineering. To name only a few, it can be used in machine learning [81]–[83] and it has even been proven to be useful in some cases in the finite element method [84].

2.4 Suzuki-Trotter expansion

In the previous section, we presented the QPE algorithm, which enables us to compute the eigenvalue corresponding to an eigenvector via the use of *oracles*. These oracles are *black-box* pieces of quantum circuits for which the algorithm assumes we have access to. However, in many cases, notably for the quantum algorithms in Chapter 3, these oracles need to be constructed. In particular, the unitary U gates correspond to the matrix exponential of a Hermitian matrix M which can sometimes be associated with the Hamiltonian operator in some schemes [46]. Nonetheless, this raises the question as to why we would need a quantum computer for these problems. Indeed, if we are able to compute the matrix exponential of the Hamiltonian, an operation that requires one to know the eigenvectors and corresponding eigenvalues, then there is no reason to use the QPE in the first place. The

¹³It is indirectly proven in the Ramusat and Savona algorithm in the next chapter 3.1. The α coefficients are multiplied by the corresponding initial amplitudes in the eigenbasis of U . As such, if the prepared state has at least a $1/\sqrt{2}$ overlap with the desired eigenstate, then the probability of measuring the best t -bit estimation of the eigenvalue is $2/\pi^2$ for example.

¹⁴We do not need to compute the entire matrix multiplication of the vector, simply doing it for the first element is enough to compute the eigenvalue.

solution to this problem is to use what we call the *Suzuki-Trotter expansion* (ST expansion), also referred to as *trotterization*. This expansion allows us to find approximations for the unitary U gates without explicitly computing the matrix exponential of a Hermitian matrix.

2.4.1 Theoretical background

Usually, the exponential of the sum of two numbers x and $y \in \mathbb{C}$ is given by

$$e^{x+y} = e^x e^y. \quad (2.39)$$

However, in the general case where we have two matrices $A, B \in \mathbb{C}^N \times \mathbb{C}^N$ that do not commute with each other, the previous formula cannot be used. Instead we have [85]

$$e^{A+B} = \lim_{n \rightarrow \infty} \left(e^{A/n} e^{B/n} \right)^n, \quad (2.40)$$

which is equivalent to

$$e^{(A+B)/n} = e^{A/n} e^{B/n} + \mathcal{O}(1/n^2). \quad (2.41)$$

The second equation can be simply proven using the renowned *Baker-Campbell-Hausdorff* (BCH) formula [86]. This formula gives the expression of $Z \in \mathbb{C}^N \times \mathbb{C}^N$ such that

$$e^X e^Y = e^Z, \quad (2.42)$$

where X and Y are also squared matrices $\in \mathbb{C}^N \times \mathbb{C}^N$ and the terms of Z are given up to third order by

$$Z = X + Y + \frac{1}{2}[X, Y] + \frac{1}{12} \left([X, [X, Y]] + [Y, [Y, X]] \right) + \dots \quad (2.43)$$

If we replace X and Y in Eq. (2.42) by A/n and B/n , Eq. (2.43) gives

$$Z = \frac{1}{n}(A + B) + \frac{1}{2n^2}[A, B] + \frac{1}{12n^3} \left([A, [A, B]] + [B, [B, A]] \right) + \dots \quad (2.44)$$

We thus have

$$e^{A/n} e^{B/n} = e^{(A+B)/n + \mathcal{O}(1/n^2)}, \quad (2.45)$$

which is an equivalent form of Eq. (2.41). How does the formula (2.40) help us build oracles ? The idea is to decompose the Hermitian matrix M as a sum of Pauli strings and then apply the Suzuki-Trotter expansion to approximate the exponential by a succession of Pauli strings exponential. The matrix exponential of a single Pauli operator yields [15]

$$e^{i\theta\sigma_j} = R_j(-2\theta), \quad (2.46)$$

where $R_j(\phi)$ is the operator that rotates the state of a qubit (on the Bloch sphere) by an angle ϕ around the j -axis. Furthermore, for a general Pauli string we have

$$e^{i\theta \bigotimes_{k=1}^{N_{\text{qub}}} \sigma_{j_k}} = \bigotimes_{k=1}^{N_{\text{qub}}} R_{j_k}(-2\theta), \quad (2.47)$$

since the Pauli matrices within the string commute with each other as they are not applied on the same qubit. Rotations around the x , y and z axis by an angle ϕ are easy to engineer as they simply

correspond to specific laser pulses [87]. In practice we set a finite limit to the number n which is called the *total number of Trotter steps* with the *Trotter step* being defined by $e^{A/n}e^{B/n}$ for the first order ST expansion.

The corresponding formula for the second order ST expansion is given by

$$e^{A+B} = \lim_{n \rightarrow \infty} \left(e^{A/2n} e^{B/n} e^{A/2n} \right)^n, \quad (2.48)$$

which is interesting to rewrite for a finite n as

$$\prod_{j=1}^n \left(e^{A/2n} e^{B/n} e^{A/2n} \right) = e^{A/2n} \prod_{j=1}^{n-1} \left(e^{B/n} e^{A/n} \right) e^{B/n} e^{A/2n}, \quad (2.49)$$

which resembles a first order Suzuki-Trotter expansion where we removed the last step and for which we would have multiplied by $e^{A/2n}$ before and $e^{B/n}e^{A/2n}$ after. This means that at almost no additional cost, the error scaling would be $\mathcal{O}(1/n^2)$ instead of $\mathcal{O}(1/n)$. Although it seems surprising that this small correction leads to such an advantage, this feature is obtained for all even orders of the ST expansion [88]. The idea is that symmetrizing the previous odd order expansion cancels the leading error term which is anti-symmetric. It can be seen in Eq. (2.43), the first term is symmetric then the second term, i.e., the first commutator, is anti-symmetric and the third term is also symmetric. Note that the advantage of using a ST expansion instead of a Taylor expansion is that the approximated operator remains unitary, which is not the case for the latter. Since quantum gates must be unitary, we understand from this that a Taylor expansion is unsuitable.

2.4.2 Simple example

Here we will illustrate what we have explained in the previous subsection. We will consider a simple Hermitian matrix defined as

$$M = 2(\sigma_z + \sigma_y) \quad (2.50)$$

$$= \begin{pmatrix} 2 & -2i \\ 2i & -2 \end{pmatrix}, \quad (2.51)$$

so that the corresponding U gate is

$$U = e^{iM} \quad (2.52)$$

$$= \begin{pmatrix} \cos(2\sqrt{2}) + \frac{i \sin(2\sqrt{2})}{\sqrt{2}} & \frac{\sin(2\sqrt{2})}{\sqrt{2}} \\ -\frac{\sin(2\sqrt{2})}{\sqrt{2}} & \cos(2\sqrt{2}) - \frac{i \sin(2\sqrt{2})}{\sqrt{2}} \end{pmatrix} \quad (2.53)$$

$$\approx \begin{pmatrix} -0.951363 + 0.21784i & 0.21784 \\ -0.21784 & -0.951363 - 0.21784i \end{pmatrix}. \quad (2.54)$$

We know from our discussion in the previous subsection that this matrix can be approximated by successive applications of $R_z(-2\phi)$ and $R_y(-2\phi)$ which are given by

$$R_z(-2\phi) = \begin{pmatrix} e^{i\phi} & 0 \\ 0 & e^{-i\phi} \end{pmatrix}, \quad (2.55a)$$

$$R_y(-2\phi) = \begin{pmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{pmatrix}. \quad (2.55b)$$

The successive applications of Trotter steps for multiple total numbers of Trotter steps is shown in Fig. 2.4 on the Bloch sphere ¹⁵. Here we have assumed that our initial state is given by

$$|\psi_0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + e^{i\pi/4} |1\rangle) \quad (2.56)$$

so that it corresponds to the initial point with coordinates

$$x_0 = \frac{1}{\sqrt{2}}, \quad (2.57a)$$

$$y_0 = \frac{1}{\sqrt{2}}, \quad (2.57b)$$

$$z_0 = 0. \quad (2.57c)$$

The first thing to observe is that the second order expansion is already extremely accurate even at only 5 Trotter steps. We also notice that, as expected, the trajectories become more and more accurate as we increase the total number of Trotter steps. Furthermore, we observe that each point remains on the surface of the sphere, i.e., the state of the system remains pure. This is of course a consequence of the fact that the Suzuki-Trotter expansion yields a unitary approximation.

To go further, we will plot the fidelity of the approximated quantum gate as a function of the number of Trotter steps. The fidelity between two quantum states ρ_1 and ρ_2 is defined as [15]

$$\mathcal{F}(\rho_1, \rho_2) = \text{Tr} \left[\sqrt{\sqrt{\rho_1} \rho_2 \sqrt{\rho_1}} \right]^2, \quad (2.58)$$

which corresponds to

$$\mathcal{F}(\psi_1, \psi_2) = |\langle \psi_1 | \psi_2 \rangle|^2, \quad (2.59)$$

for pure states. Intuitively, this quantity corresponds to the probability of measuring the state $|\psi_2\rangle$ if the state before the measurement is $|\psi_1\rangle$. The fidelity is thus contained within the interval $[0, 1]$ with 1 being the maximum fidelity corresponding to both states being equivalent. In order to assess how close the approximation is, we plot the logarithm of $1 - \mathcal{F}$. As we can see, the first order ST expansion follows a power law n^{-2} while the second order follows a power law n^{-4} . This is as expected since their errors scale as n^{-1} and n^{-2} respectively. Computing $1 - \mathcal{F}$ using the expression for the fidelity in Eq. (2.59) would then yield

$$1 - \mathcal{F}(\psi_{\text{ST}}, \psi_{\text{exact}}) = 1 - |\langle \psi_{\text{ST}} | \psi_{\text{exact}} \rangle|^2 \quad (2.60)$$

$$= 1 - \left| \langle \psi_{\text{exact}} + \mathcal{O}(n^{-p}) | \psi_{\text{exact}} \rangle \right|^2 \quad (2.61)$$

$$= 1 - 1 - \left| \mathcal{O}(n^{-p}) \right|^2 \quad (2.62)$$

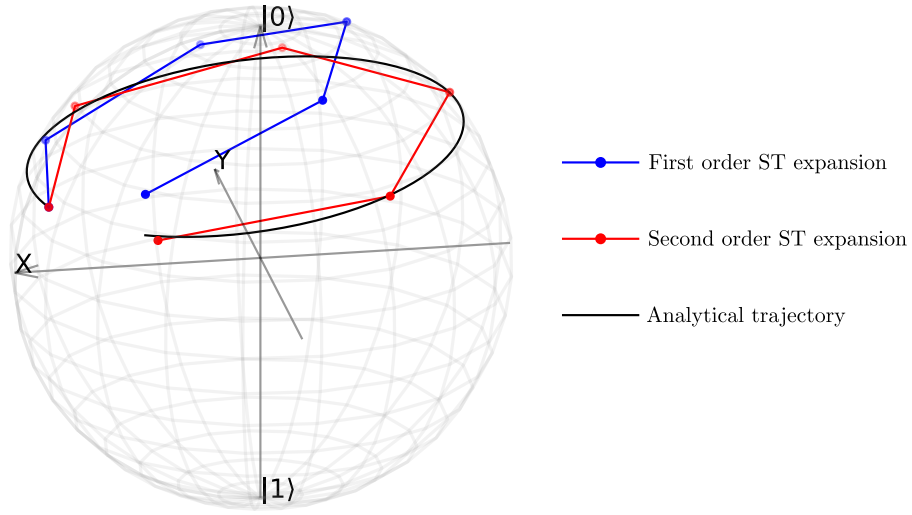
$$= \mathcal{O}(n^{-2p}), \quad (2.63)$$

with p being the order of the Suzuki-Trotter expansion.

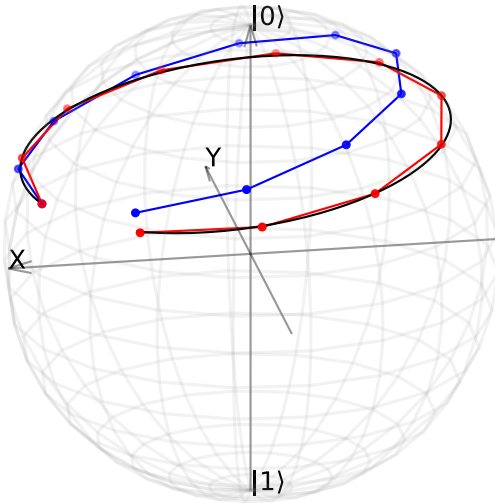
2.5 Summary

This chapter introduced the reader to the basics of quantum computation. In the first section, we explained the concepts of computational basis, quantum circuits, and general quantum gates.

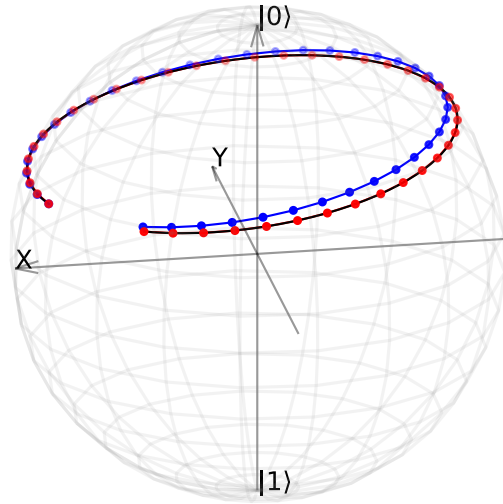
¹⁵The Bloch sphere is a tool to visualize the states of a two-level system. Basically, the x, y, z coordinates are obtained by computing the expected values $\langle \sigma_x \rangle, \langle \sigma_y \rangle, \langle \sigma_z \rangle$ of the state.



(a) 5 Trotter steps



(b) 10 Trotter steps



(c) 40 Trotter steps

Figure 2.4: Plots on the Bloch sphere of the state of the system after each application of a Trotter step with the total number of steps being (a) 5, (b) 10 and (c) 40. The operations are applied on the initial state given in Eq. (2.56). The blue line represents the first order ST expansion, the red line the second order ST expansion while the black line represents the analytical trajectory obtained by explicitly computing the matrix exponential.

The next sections were dedicated to presenting well-known quantum algorithms frequently used as subroutines: the *quantum Fourier transform* (QFT) and *quantum phase estimation* (QPE). These subroutines are precisely what we used in the next chapter regarding quantum algorithms for open quantum systems. The last section was dedicated to the *Suzuki-Trotter expansion*, a method used to approximate specific quantum gates.

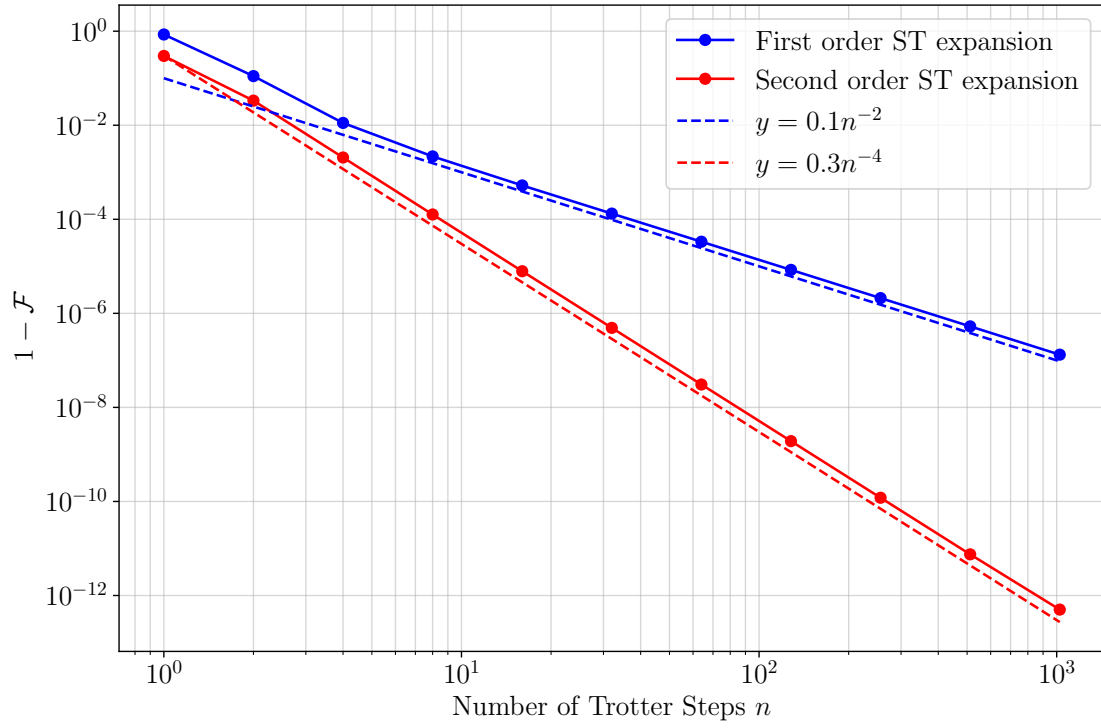


Figure 2.5: Log-Log plot of $1 - \mathcal{F}(\psi_{\text{ST}}, \psi_{\text{exact}})$ in terms of the total number of Trotter steps. The blue line represents the first order ST expansion and the red line the second order ST expansion. The dashed lines are references to show the corresponding power laws.

Chapter 3

Quantum computation for open quantum systems

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

Richard P. Feynman 1981

Now that we have introduced the reader to open quantum systems and quantum computation. We will mix the two and explain how we can use quantum computing to solve open quantum system problems and more specifically, the steady state problem. The first section of this chapter will be dedicated to the algorithm of Ramusat and Savona (RS) [19], a quantum algorithm for estimating the steady state of open quantum systems. In the second section, we apply this algorithm in the case of the simple spin model presented in Sec. 1.3 by doing quantum circuit simulations. These simulations are done in Qiskit [89], an open-source Python library for quantum computing which is developed by IBM. We then show that we can bypass these simulations by diagonalizing a matrix defined from the Liouvillian, allowing us to perform advanced convergence analyses. In the second part of this chapter, we show how the RS algorithm can be adapted to non-Markovian systems using the HEOM Liouvillian instead, including a few adjustments. This adaptation represents the original contribution of this thesis. We will then apply this algorithm on the open quantum Rabi model described in Sec. 1.5 and compare it against the original algorithm used in the case of a Markovian embedding approach. Finally, we show how our algorithm gives a non-trivial advantage for multi-mode systems by analyzing the application of both algorithms on a two-mode open quantum Rabi model.

3.1 Ramusat and Savona (RS) algorithm

In this section, the algorithm of the paper of Ramusat and Savona [19] will be extensively presented. They proposed a quantum algorithm to directly estimate steady state expectation values of observables of open quantum systems. The idea of the algorithm is to use a variation of the HHL algorithm [80] and the known spectral properties of the generator of the dynamics to have a high success rate without the use of quantum state amplification. The advantage of the algorithm is that it bypasses the time integration of the master equation, which can imply heavy costs for systems

with high relaxation times, and only computes the steady state. However, the computed steady state cannot be directly accessed and custom measurements must be applied instead, in order to estimate expected values. Alternatively, quantum state tomography could be used to retrieve the exact expression of the steady state. However, this is not ideal and most of the time, not useful as physicists are only interested in the expectation values anyway.

3.1.1 Overview

The algorithm assumes the system to be subjected to the Markovian hypothesis and that it is described by the GKSL master equation (1.28) described in the first chapter of this thesis. To be able to represent the Liouvillian as an operator and to encode the density operator onto a quantum register, the density operator is vectorized using Choi–Jamiołkowski isomorphism (cf. Subsection 1.2.3). Using this procedure, the Liouvillian is downgraded to being merely an operator (instead of a super-operator) which acts linearly upon vectors of the space $\mathcal{H} \otimes \mathcal{H}$. Its explicit matrix representation is reminded below for convenience

$$\mathcal{L} = -i(H \otimes \mathbb{1} - \mathbb{1} \otimes H^T) + \sum_j \gamma_j \left(L_j \otimes L_j^* - \frac{1}{2} L_j^\dagger L_j \otimes \mathbb{1} - \frac{1}{2} \mathbb{1} \otimes L_j^T L_j^* \right).$$

Assuming the Hilbert space \mathcal{H} has dimension 2^N implies the vectorized density operator has a length of 2^{2N} (and the Liouvillian a size $2^{2N} \times 2^{2N}$) which means it can be encoded onto a quantum register which contains $2N$ qubits. If the Hilbert space did not have a dimension of the form 2^N for some N (that is, if the Hilbert space dimension was not a power of 2), the Liouvillian would have to be padded with 1's¹ on the diagonal until it reaches a size which can be written as $2^{2N} \times 2^{2N}$. However, the operator has to be further scaled up as it is in general not Hermitian for dissipative systems. Thus, one usually defines a Hermitian operator M of the form

$$M = \begin{pmatrix} 0 & \mathcal{L} \\ \mathcal{L}^\dagger & 0 \end{pmatrix}, \quad (3.1)$$

which is an operator that acts upon vectors of size 2^{2N+1} , meaning it acts upon $2N + 1$ qubits. The algorithm assumes that the Hamiltonian H and the jump operators L_j are k -local, meaning they act at most on k qubits at the same time, allowing these and the M matrix to be efficiently implemented. Since M is Hermitian and that we need a matrix to be unitary to be implemented as a quantum gate, we set

$$U = e^{2i\pi t_0 M}, \quad (3.2)$$

where t_0 is chosen so that the whole spectrum of $t_0 M$ is in the interval $[0, 1]$. The spectral properties of the Liouvillian imply that the operator M has only two eigenvectors associated with the zero-eigenvalue, which read

$$|\eta_0\rangle_{2N+1} = |0\rangle |\mathbb{1}\rangle_{2N}, \quad (3.3a)$$

$$|\eta_1\rangle_{2N+1} = |1\rangle |\rho_{ss}\rangle_{2N}, \quad (3.3b)$$

where ρ_{ss} is the steady state of the system (satisfying $\mathcal{L}(\rho_{ss}) = 0$ or equivalently, $\mathcal{L}|\rho_{ss}\rangle = 0$) and where $|\mathbb{1}\rangle_{2N}$ is the vectorized identity matrix (the identity matrix being of dimension $2^N \times 2^N$) and thus has dimension 2^{2N} . We chose to explicitly denote the number of qubits for ket vectors with

¹We could pad with any number technically, however, as we will see in the next subsections, we have to be careful that it has no impact on the rest of the algorithm.

an index (most of the time) for clarity. Note that for the vectorized identity, it is implied that it is normalized as a ket vector, furthermore it is not to be mistaken with the notation $|\mathbf{1}\rangle_{2N}$ which would correspond to all qubits being in the state $|1\rangle$. For $|\eta_1\rangle$ it is straightforward to prove it as we have

$$\begin{aligned} M|\eta_1\rangle &= \begin{pmatrix} 0 & \mathcal{L} \\ \mathcal{L}^\dagger & 0 \end{pmatrix} \begin{pmatrix} 0 \\ |\rho_{ss}\rangle \end{pmatrix} \\ &= \begin{pmatrix} \mathcal{L}|\rho_{ss}\rangle \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \end{aligned} \quad (3.4)$$

by definition of the steady state (1.35). The same thing can be done with $|\eta_0\rangle$ and using the fact that $\mathcal{L}^\dagger|\mathbf{1}\rangle = 0$, which is a consequence of the fact that the Liouvillian preserves the trace. Indeed, we have that

$$\text{Tr}[\rho] = \langle \mathbf{1} | \rho \rangle = 1, \quad (3.5)$$

and taking the time derivative yields

$$0 = \langle \mathbf{1} | \mathcal{L} | \rho \rangle. \quad (3.6)$$

This equality being valid for any ρ , we thus find that $\langle \mathbf{1} | \mathcal{L} = 0$ which is of course equivalent to $\mathcal{L}^\dagger|\mathbf{1}\rangle = 0$.

The Liouvillian can always be separated into a Hermitian \mathcal{L}_H and an anti-Hermitian $i\mathcal{L}_A$ contributions such that $\mathcal{L} = \mathcal{L}_H - i\mathcal{L}_A$, with \mathcal{L}_H and \mathcal{L}_A being Hermitian operators. It is trivial to show that taking

$$\mathcal{L}_H = \frac{1}{2}(\mathcal{L} + \mathcal{L}^\dagger), \quad \mathcal{L}_A = \frac{i}{2}(\mathcal{L} - \mathcal{L}^\dagger), \quad (3.7)$$

will always give the correct decomposition. This allows us to rewrite M as

$$M = X \otimes \mathcal{L}_H + Y \otimes \mathcal{L}_A, \quad (3.8)$$

where X and Y are the Pauli operators introduced in the previous chapters. The point of doing this decomposition is that if \mathcal{L}_H and \mathcal{L}_A can be efficiently encoded as quasi-local operators, which is generally easier to do, then it will also be the case for M . We can now plug (1.36) into (3.7) which gives us

$$\mathcal{L}_H = \frac{1}{2} \sum_j \gamma_j \left(L_j \otimes L_j^* + L_j^\dagger \otimes L_j^T - L_j^\dagger L_j \otimes \mathbf{1} - \mathbf{1} \otimes L_j^T L_j^* \right), \quad (3.9a)$$

$$\mathcal{L}_A = (H \otimes \mathbf{1} - \mathbf{1} \otimes H^T) + \frac{i}{2} \sum_j \gamma_j \left(L_j \otimes L_j^* - L_j^\dagger \otimes L_j^T \right). \quad (3.9b)$$

The algorithm can be separated into three parts as shown in Figure 3.1. The first part aims at preparing an initial state with a large overlap with the zero eigenspace of M (i.e., $|\eta_0\rangle$ and $|\eta_1\rangle$) as depicted in Figure 3.2. Then the algorithm uses a quantum phase estimation subroutine to completely project the initial statevector onto the degenerate zero eigenspace. The projection, and thus the algorithm, is successful if the eigenvalue $\mathbf{0}$ is measured in the first qubit register. Having reached this point, custom measurements can be applied onto the second qubit register to attain some information about the steady state.

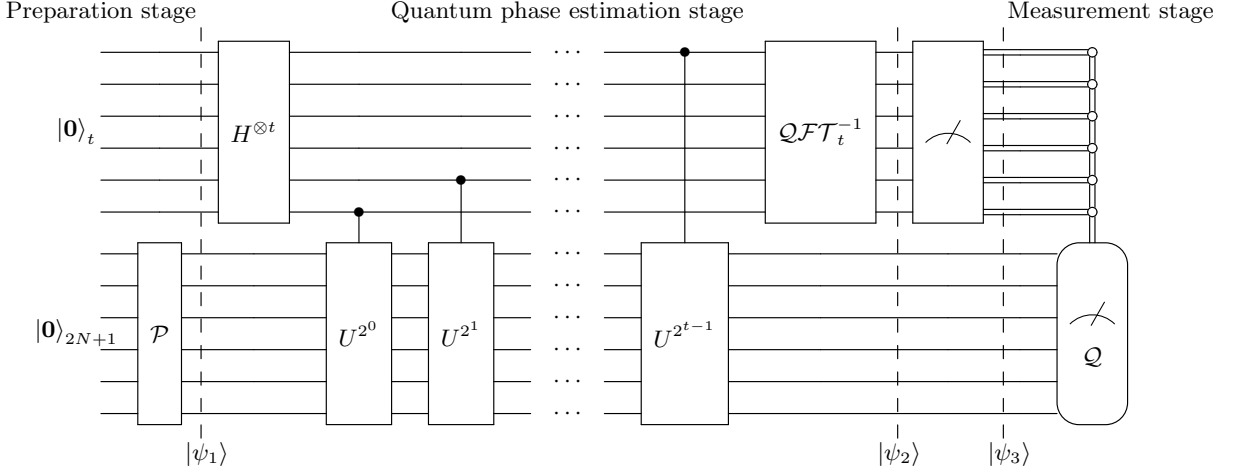


Figure 3.1: Ramusat and Savona algorithm as presented in their paper [19]. The algorithm is composed of an input-state preparation stage, a quantum phase estimation stage and a measurement stage using the operator Q defined in Eq. (3.19). The first qubit register contains t qubits, having more qubits in this register allows for higher precision but at a higher cost in gates and qubits. The second quantum register is composed of $2N + 1$ qubits and represents the statevector of the system. The \mathcal{P} gate is explicitly depicted in Figure 3.2. The white circles represent the conditional measurement applied on the second register is $\mathbf{0}_t$ was measured.

3.1.2 Preparation stage

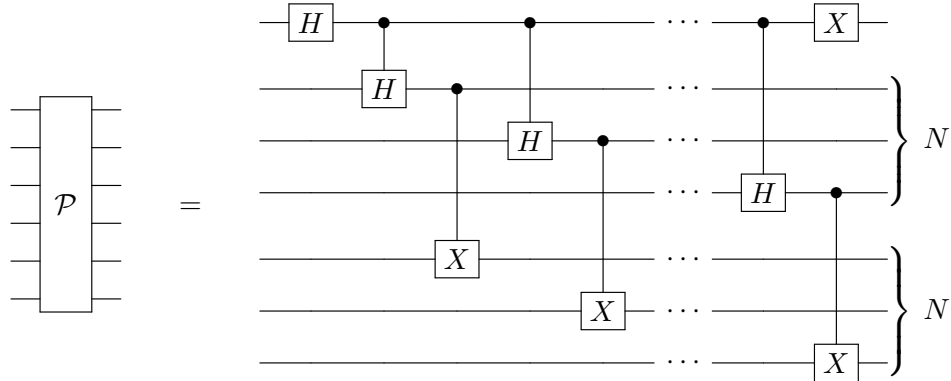


Figure 3.2: \mathcal{P} gate explicitly drawn as a quantum circuit. The circuit takes the $|\mathbf{0}\rangle$ state from $2N + 1$ qubits, as input and outputs the state $|\xi\rangle$. It should be mentioned that this figure is reproduced from Ramusat and Savona's paper but the ordering of the qubits is in reverse order compared to the convention used in Qiskit. It contains $2N$ controlled gates and 2 single-qubit gates. The circuit can be modified to accommodate known symmetries.

As a reminder (cf. Section 2.3) the QPE takes as input the state $|\mathbf{0}\rangle |\eta_j\rangle$, where $|\eta_j\rangle$ is an eigenstate of the unitary operator U where the associated eigenvalue is φ_j . If $2^t \varphi_j$ is a t -bit integer, the QPE will output with 100% certainty the state $|\varphi_j\rangle |\eta_j\rangle$. However, if $2^t \varphi_j$ is not a t -bit integer, then the QPE will output with high probability, the state $|\tilde{\varphi}_j\rangle |\eta_j\rangle$, where $2^t \tilde{\varphi}_j$ is the best t -bit

integer estimate of $2^t \varphi_j$. The algorithm assumes that a circuit exists to efficiently implement the controlled- U^{2^j} gates via the use of a Suzuki-Trotter expansion (cf. Section 2.4).

This algorithm, however, works backwards, starting from a known eigenvalue, it uses the QPE to compute the eigenstate corresponding to the eigenvalue. To do so, the \mathcal{P} gate is used to prepare the $|\xi\rangle$ state where $|\xi\rangle$ has a large overlap with the eigenstates $|\eta_0\rangle$ and $|\eta_1\rangle$. We generally set

$$|\xi\rangle = \frac{|0\rangle|1\rangle_{2N} + |1\rangle|0\rangle_{2N}}{\sqrt{2}} = \frac{|\eta_0\rangle_{2N+1} + |1\rangle|0\rangle_{2N}}{\sqrt{2}}, \quad (3.10)$$

so that we have a good balance between the success rate and the precision of the algorithm. The success rate is determined by the amplitude of the projection of this state onto the zero eigenspace, while some part of the precision is determined by the amplitude of the projection onto the eigenstate $|\eta_1\rangle$. The choice of $|\xi\rangle$ gives at worst, a success rate of 50% as we will prove later on, and the amplitude of the projection is determined by

$$c_1 = \langle \eta_1 | 1 \otimes \mathbf{0} \rangle, \quad (3.11)$$

since $|\eta_0\rangle$ and $|\eta_1\rangle$ are orthogonal. We can now expand $|\xi\rangle$ on the eigenstates of M to rewrite the state of the system after the preparation stage and we have

$$\begin{aligned} |\psi_1\rangle &= |\mathbf{0}\rangle_t |\xi\rangle_{2N+1} \\ &= \frac{1}{\sqrt{2}} |\mathbf{0}\rangle_t |\eta_0\rangle_{2N+1} + \frac{1}{\sqrt{2}} \sum_{j \neq 0} c_j |\mathbf{0}\rangle_t |\eta_j\rangle_{2N+1} \\ &= \frac{|\mathbf{0}\rangle_t |\eta_0\rangle_{2N+1} + c_1 |\mathbf{0}\rangle_t |\eta_1\rangle_{2N+1}}{\sqrt{2}} + \frac{1}{\sqrt{2}} \sum_{j \neq 0,1} c_j |\mathbf{0}\rangle_t |\eta_j\rangle_{2N+1}, \end{aligned} \quad (3.12)$$

where the c_j are defined as $c_j = \langle \eta_j | 1 \otimes \mathbf{0} \rangle$. The advantage of doing this decomposition is that we can easily write the result of any eigenvector of M going through all the controlled- U gates and thus generalize it for any decomposition.

3.1.3 Quantum phase estimation stage

For all eigenstates $|\eta_j\rangle$, we have from (2.34)

$$\begin{aligned} QPE |\mathbf{0}\rangle |\eta_j\rangle &= \frac{1}{2^t} \sum_{k=0}^{2^t-1} \frac{1 - e^{2i\pi(t_0 \varphi_j 2^t - k)}}{1 - e^{2i\pi(t_0 \varphi_j - k/2^t)}} |k\rangle_t |\eta_j\rangle_{2N+1} \\ &= \sum_{k=0}^{2^t-1} \alpha_k^{(j)} |k\rangle_t |\eta_j\rangle_{2N+1}, \end{aligned} \quad (3.13)$$

where QPE designates the action of the quantum phase estimation algorithm before the measurement (i.e., the Hadamard and controlled- U gates) and we defined the $\alpha_k^{(j)}$ coefficients as

$$\alpha_k^{(j)} = \frac{1}{2^t} \frac{1 - e^{2i\pi(t_0 \varphi_j 2^t - k)}}{1 - e^{2i\pi(t_0 \varphi_j - k/2^t)}}. \quad (3.14)$$

However, we have to be careful that this is only valid when $t_0\varphi_j - k/2^t$ is not equal to 0 (or any integer, nonetheless, this is not a problem as we defined t_0 so that all eigenvalues of t_0M will be smaller than 1), otherwise the α coefficient should instead be equal to 1. In our case, we are only interested in the projection onto the zero eigenspace, thus, we only care about $k = 0$ which means that this “problem” only occurs when $\varphi_j = 0$ which merely concerns $|\eta_0\rangle$ and $|\eta_1\rangle$. It is important to notice that when the expression (3.14) is valid, these coefficients decrease exponentially owing to the $1/2^t$ factor in front. We now have everything we need to write the state $|\psi_2\rangle$ at the second dashed line in Fig. 3.1 and we have

$$|\psi_2\rangle = \frac{|\mathbf{0}\rangle_t |\eta_0\rangle_{2N+1} + c_1 |\mathbf{0}\rangle_t |\eta_1\rangle_{2N+1}}{\sqrt{2}} + \frac{1}{\sqrt{2}} \sum_{j \neq 0,1} \sum_{k=0}^{2^t-1} c_j \alpha_k^{(j)} |k\rangle_t |\eta_j\rangle_{2N+1}. \quad (3.15)$$

Once again, we have to be careful that this is technically incorrect as this notation does not consider if $t_0\varphi_j - k/2^t$ can be equal to 0 for $k \neq 0$. It is, however, inconsequential as we will now consider the algorithm was successful and that we measured 0 for all qubits in the first register. Hence, after measuring the first quantum register, the unrenormalized state is given by

$$|\psi_3\rangle = |\mathbf{0}\rangle_t \left(\frac{|\eta_0\rangle_{2N+1} + c_1 |\eta_1\rangle_{2N+1}}{\sqrt{2}} + \frac{1}{\sqrt{2}} \sum_{j \neq 0,1} c_j \alpha_0^{(j)} |\eta_j\rangle_{2N+1} \right). \quad (3.16)$$

The probability of success is given by the squared norm of the state $|\psi_3\rangle$ as it was unrenormalized² and we thus have

$$\begin{aligned} p(\mathbf{0}) &= \langle \psi_3 | \psi_3 \rangle \\ &= \frac{1}{2} \langle \mathbf{0} | \mathbf{0} \rangle \left(\langle \eta_0 | \eta_0 \rangle + |c_1|^2 \langle \eta_1 | \eta_1 \rangle + \sum_{j \neq 0,1} |c_j|^2 |\alpha_0^{(j)}|^2 \langle \eta_j | \eta_j \rangle \right) \\ &= \frac{1}{2} + \frac{1}{2} |c_1|^2 + \sum_{j \neq 0,1} |c_j|^2 |\alpha_0^{(j)}|^2 \\ &\geq \frac{1}{2}, \end{aligned} \quad (3.17)$$

where $p(\mathbf{0})$ designates the probability of measuring $\mathbf{0}$ in the eigenvalue register. This proves that our choice of state $|\xi\rangle$ gives at worst a success rate of 50%, as stated before.

3.1.4 Measurement stage

We can now apply specific measurement operators onto the state $|\psi_3\rangle$ (more specifically, apply measurement on the second quantum register representing the statevector of the system) in order to extract steady state values of system observables. As a reminder, steady state expected values of observables are computed the following way $\langle O \rangle = \text{Tr}[O\rho_{ss}]/\text{Tr}[\rho_{ss}]$, where contrary to what we

²Remember that the probability of measuring an eigenvalue λ of a measurement operator, corresponding to an eigenspace with the associated projector Π , is given by $p(\lambda) = \langle \psi | \Pi | \psi \rangle$. Since a projector is idempotent, i.e., $\Pi^2 = \Pi$ and Hermitian, this probability is equivalently given by $p(\lambda) = \langle \phi | \phi \rangle$ with $|\phi\rangle = \Pi |\psi\rangle$. This means that the probability is given by the squared norm of the projected state (before renormalization).

have seen in the first chapter, we have to normalize the results because even if the expression of (3.16) was normalized, a ket vector and a density operator do not possess the same normalization conditions. To compute the trace directly from the quantum register we will first notice that we have

$$\begin{aligned}
\langle \mathbf{1} | O \otimes \mathbf{1}_{N \times N} | \rho_{ss} \rangle &= \sum_{i,j} \rho_{ij} \sum_{i'} \langle i' | O | i \rangle \langle i' | j \rangle \\
&= \sum_{i,j} \rho_{ij} \langle j | O | i \rangle \\
&= \sum_{i,j} \rho_{ij} O_{ji} \\
&= \sum_i (\rho_{ss} O)_{ii} \\
&= \text{Tr}[\rho_{ss} O],
\end{aligned} \tag{3.18}$$

where we replaced both vectors by their decomposition $|\rho_{ss}\rangle = \sum_{i,j} \rho_{ij} |i\rangle \otimes |j\rangle$, $\langle \mathbf{1} | = \sum_{i'} \langle i' | \otimes \langle i' |$ and rewrote the third line as a matrix product. We can now define the measurement operator \mathcal{Q} the following way

$$\mathcal{Q} = X \otimes O \otimes \mathbf{1}_{N \times N} = \begin{pmatrix} 0 & O \otimes \mathbf{1}_{N \times N} \\ O \otimes \mathbf{1}_{N \times N} & 0 \end{pmatrix}. \tag{3.19}$$

Since $|\eta_0\rangle = |0\rangle |\mathbf{1}\rangle_{2N}$ and $|\eta_1\rangle = |1\rangle |\rho_{ss}\rangle_{2N}$, we have that

$$\langle \eta_0 | \mathcal{Q} | \eta_0 \rangle = \langle \eta_1 | \mathcal{Q} | \eta_1 \rangle = 0. \tag{3.20}$$

This also means that we have

$$\begin{aligned}
\langle \eta_0 | \mathcal{Q} | \eta_1 \rangle &= \langle 0 | 0 \rangle \langle \mathbf{1} | O \otimes \mathbf{1} | \rho_{ss} \rangle \\
&= \text{Tr}[\rho_{ss} O],
\end{aligned} \tag{3.21}$$

where we just took the explicit forms of \mathcal{Q} , $|\eta_0\rangle$, $|\eta_1\rangle$ and used (3.18). Since O is an observable represented by a hermitian matrix, its expected value must be real and we also have $\langle \eta_0 | \mathcal{Q} | \eta_1 \rangle = \langle \eta_1 | \mathcal{Q} | \eta_0 \rangle = \text{Tr}[\rho_{ss} O]$. We can now compute the \mathcal{Q} expected value of the state $|\psi_3\rangle$ using (3.16) and we have

$$\begin{aligned}
\langle \psi_3 | \mathbf{1} \otimes \mathcal{Q} | \psi_3 \rangle &= \frac{1}{2} (c_1 + c_1^*) \text{Tr}[\rho_{ss} O] + \frac{1}{2} \sum_{j \neq 0,1} \left(c_j \alpha_0^{(j)} \langle \eta_0 | \mathcal{Q} | \eta_j \rangle + c_j^* \alpha_0^{(j)*} \langle \eta_j | \mathcal{Q} | \eta_0 \rangle \right. \\
&\quad \left. + c_1^* c_j \alpha_0^{(j)} \langle \eta_1 | \mathcal{Q} | \eta_j \rangle + c_1 c_j^* \alpha_0^{(j)*} \langle \eta_j | \mathcal{Q} | \eta_1 \rangle \right) \\
&\quad + \frac{1}{2} \sum_{i \neq 0,1} \sum_{j \neq 0,1} c_i^* \alpha_0^{(i)*} c_j \alpha_0^{(j)} \langle \eta_i | \mathcal{Q} | \eta_j \rangle
\end{aligned} \tag{3.22}$$

where we used the relations (3.20) and (3.21). We can condense the expression further by rewriting with the real part and finally obtain

$$\begin{aligned}
\langle \psi_3 | \mathbb{1} \otimes \mathcal{Q} | \psi_3 \rangle &= \text{Re}[c_1] \text{Tr}[\rho_{ss} O] + \text{Re} \left[\sum_{j \neq 0,1} \left(c_j \alpha_0^{(j)} \langle \eta_0 | \mathcal{Q} | \eta_j \rangle + c_1^* c_j \alpha_0^{(j)} \langle \eta_1 | \mathcal{Q} | \eta_j \rangle \right) \right] \\
&+ \frac{1}{2} \sum_{i \neq 0,1} \sum_{j \neq 0,1} c_i^* c_j \alpha_0^{(i)*} \alpha_0^{(j)} \langle \eta_i | \mathcal{Q} | \eta_j \rangle.
\end{aligned} \tag{3.23}$$

Note that the third term is also real, it can either be seen by regrouping complex conjugated terms, separating the $i = j$ terms and make it explicitly appear, or we can simply see that since the first two terms are real and \mathcal{Q} is Hermitian, the sum of the three terms has to be real. It is also worth noticing that since the diagonal elements of the density operator are real and positive-valued, as they represent a probability, the c_1 projection constant is strictly real-valued. And as such $\text{Re}[c_1] = c_1$ (remember that the \mathcal{P} gate prepares a state that has a $1/\sqrt{2}$ coefficient exactly where it would be multiplied by the first diagonal element of the steady state density operator when doing the inner product, hence giving a real value). This remark allows us to specify that any adaptation of the \mathcal{P} gate has to be done carefully. Specifically, any modification that would give a c_1 constant strictly imaginary is to be excluded, as it would evidently cancel the term containing the measurement operator expected value. Furthermore, we should be careful that in very particular cases, the steady state may have no overlap with the first diagonal element, giving c_1 equal to 0. If such a case were to happen, the \mathcal{P} gate would have to be modified so that the state prepared corresponds to another diagonal element. The first term is of course what we want and the other two are errors that originate from the quantum phase estimation stage. However, these terms decrease exponentially with the number of qubits t of the quantum register since they all depend on the $\alpha_0^{(j)}$ coefficients which are given by Eq. (3.14). In order to eliminate both the $\text{Re}[c_1]$ coefficient and the norm of ρ_{ss} , we can simply choose O as $\mathbb{1}$ so that \mathcal{Q} would output the value $\langle \psi_3 | \mathcal{Q} | \psi_3 \rangle = \text{Re}[c_1] \text{Tr}[\rho_{ss}]$ which can then be used to normalize the expected value of any operator O .

This result is important, as Eq. (3.23) predicts that for a number of qubits t in the eigenvalue register large enough, the outcome of the algorithm yields roughly the expected value of the observable. Furthermore it predicts the exact outcome of the algorithm, taking into account the errors contribution. This in turn infers that we can compute $\langle \psi_3 | \mathbb{1} \otimes \mathcal{Q} | \psi_3 \rangle$ in order to directly compute the output of the algorithm for a perfect implementation (i.e. without any hardware limitations and assuming the controlled- U gates are perfectly implemented (cf. Sec 2.4), including the error contribution, all while bypassing the cost of simulating the circuit via Qiskit or any other library. The advantage is that evaluate Eq. (3.23) depends only on the size of the studied system (it requires the same resources regardless of the number of qubits t in the eigenvalue register), whereas the cost of simulating the circuit giving us this result increases exponentially with t . The efficiency of this approach is demonstrated in Sec. 3.2 for a simple open quantum system.

3.1.5 Complexity analysis

The gate cost of the algorithm remains to be determined, and as we shall see, this question is not so trivial. First, the preparation stage consists of 1 Hadamard gate, 1 NOT gate, N controlled-Hadamard gates, and N cNOT gates, leading to a total of $2N + 2$ gates. The measurement stage is already more complicated to evaluate. To estimate expectation values with a precision threshold ϵ_m , the algorithm requires N_m measurements onto the state $|\psi_3\rangle$, meaning N_m successful runs of the algorithm, with $N_m = \mathcal{O}(1/\epsilon_m^2)$ [90], [91], which means that the overall scaling of the algorithm

will be multiplied by $1/\epsilon_m^2$. The quantum phase estimation stage is the most delicate part of this analysis, there are trivially t Hadamard gates and the inverse quantum Fourier transform scales as $\mathcal{O}(t^2)$ for exact precision, while approximate circuits achieve a scaling of $\mathcal{O}(t \log t)$ (cf. Sec. 2.2). The controlled- U gates are not easily implemented. Assuming the use of Suzuki-Trotter expansions and that both the Hamiltonian and jump operators are k -local, the most optimal cost of one gate U^{2^j} is estimated as $\mathcal{O}\left(2\pi t_0 \|M\|_1 (2N+1)^k 2^{j(1+1/p)} / \epsilon_\tau^{1/p}\right)$ [92]. With ϵ_τ the Suzuki-Trotter error, p the order of the Suzuki-Trotter expansion and $\|M\|_1$ the 1-norm of the operator M ³. First of all, as t_0 is chosen so that all eigenvalues are smaller than 1 (i.e. $t_0^{-1} > \max_j(\lambda_j)$), an easy upper bound is conveniently given by the 1-norm of the operator, which allows us to consider that $2\pi t_0 \|M\|_1 = \mathcal{O}(1)$. Furthermore, summing over all U^{2^j} gates, and using the geometric sum formula gives us

$$\sum_{j=0}^{t-1} 2^{j(1+1/p)} = \frac{-1 + 2^{t(1+1/p)}}{-1 + 2^{1+1/p}} \quad (3.24)$$

$$= \mathcal{O}\left(2^{t(1+1/p)}\right). \quad (3.25)$$

One way to estimate the number of qubits required in the eigenvalue register is to suppose the error stemming from the quantum phase estimation scales proportionally with the probability of measuring the wrong eigenvector in the eigenvector basis after having measured the eigenvalue register in the computational basis. This simply corresponds to the error probability in the classical formulation of the quantum phase estimation [15], [93]. Noting this error ϵ_p , we have

$$\epsilon_p = \sum_{j \neq 0,1}^{2^{N+1}-1} |c_j|^2 |\alpha_0^{(j)}|^2, \quad (3.26)$$

where we can develop the factor $|\alpha_0^{(j)}|^2$ as follows

$$\begin{aligned} |\alpha_0^{(j)}|^2 &= \frac{1}{2^{2t}} \left| \frac{1 - e^{2i\pi t_0 \varphi_j 2^t}}{1 - e^{2i\pi t_0 \varphi_j}} \right|^2 \\ &= \frac{1}{2^{2t}} \frac{|\sin(\pi t_0 \varphi_j 2^t)|^2}{|\sin(\pi t_0 \varphi_j)|^2} \\ &\approx |\text{sinc}(\pi t_0 \varphi_j 2^t)|^2 \\ &\leq \frac{1}{\pi^2 t_0^2 \varphi_j^2 2^{2t}}, \end{aligned} \quad (3.27)$$

where we used the relation $|1 - e^{2ix}|^2/4 = \sin^2(x)$ to go from the first to the second line, then we developed the denominator using a Taylor expansion by assuming $\pi t_0 \varphi_j < 1$ ⁴, which allows us to

³As a reminder, the 1-norm of a matrix is defined as the maximum absolute column sum of the matrix, i.e. $\|A\|_1 = \max_j \sum_i |a_{ij}|$.

⁴This step is not very rigorous but it is what is done in the paper of Ramusat and Savona [19]. A more rigorous way to proceed would have been to define t_0 so that the whole spectrum of $t_0 M$ is in the interval $[0, \frac{1}{2}]$. This way, $\pi t_0 \varphi_j$ would be smaller than $\pi/2$. We could then use the same relation as in the QPE section where we have the inequality $x \leq \pi \sin(x)$ for $x \leq \pi/2$. However, we will stick with the original development as redefining t_0 does not change the overall scaling.

rewrite the expression with a squared sine cardinal function which we can bound using $|\text{sinc}(x)|^2 \leq 1/x^2$. We can then insert (3.27) into (3.26) and we have

$$\begin{aligned}\epsilon_p &= \sum_{j \neq 0,1}^{2^{N+1}-1} \frac{|c_j|^2}{\pi^2 t_0^2 \varphi_j^2 2^{2t}} \\ &\leq \frac{1}{\pi^2 t_0^2 g^2 2^{2t+1}},\end{aligned}\tag{3.28}$$

where we found an upper bound by replacing each φ_j by the Liouvillian gap, defined as the smallest nonzero eigenvalue (in norm) of the Liouvillian and then the sum over each $|c_j|^2$ coefficient can also be bounded by $1/2$ which comes from the normalization condition and that $|c_0|^2 = 1/2$. This expression allows us to find an estimation for t and we have

$$t \approx \left\lceil \log_2 \left(\frac{1}{\pi t_0 g \sqrt{2\epsilon_p}} \right) \right\rceil,\tag{3.29}$$

Combining everything gives us the following scaling

$$\#_{\text{gates}} = \mathcal{O} \left(\frac{t + t^2 + 2N}{\epsilon_m^2} + \frac{(2N+1)^k}{t_0 \epsilon_m^2 g^{1+1/p} \epsilon_\tau^{1/p} \epsilon_p^{1/2+1/2p}} \right).\tag{3.30}$$

Note that we did not replace the first t terms by $t = -\log_2(\sqrt{2\epsilon_p}\pi g t_0)$, as could be obtained from Eq. (3.29), simply because these terms will not give any significant contributions compared to the last one. If we neglect the first terms and choose a sufficiently high Suzuki-Trotter expansion order p , the scaling then becomes

$$\#_{\text{gates}} = \mathcal{O} \left(\frac{(2N+1)^k}{t_0 \epsilon_m^2 g \epsilon_p^{1/2}} \right).\tag{3.31}$$

This means that the scaling is polynomial with the size of the system, furthermore, if the Liouvillian gap g behaves as an inverse power law with the size of the system, which is often the case away from critical points [94], then the algorithm performs with an exponential advantage over classical algorithms. Note that choosing a sufficiently high Suzuki-Trotter is not necessarily advantageous. While the number of steps needed decreases, it cannot go below roughly $1/(\pi t_0 g \sqrt{2})$ and will thus saturate at some point. Furthermore, the cost of one Trotter step increases with p (cf. Sec. 2.4) although it is often neglected in complexity analyses. As such, picking a Suzuki-Trotter expansion order moderately high is the preferred approach.

3.2 Direct application of the RS algorithm in the case of a simple open quantum system

In this section, we will perform simulations of the Ramusat and Savona algorithm, presented in Sec. 3.1, in the case of the same simple open quantum system as in Sec. 1.3 of a spin subject to a magnetic field along the x -axis and decay process along the z -axis. We will then compute the expected results from Eq. (3.23) and show that it perfectly predicts the outcomes of the algorithm's simulations.

3.2.1 Circuit simulations

The simulations were done on Qiskit using the Aer simulator backend, specifically, the circuit from Figs. 3.1 and 3.2 are simulated and the final statevector is extracted, which is then used to compute expected values. This kind of simulation does not take into account the measurement stage, the reason behind this is that Qiskit does not yet handle complex dynamical circuits to a satisfying degree and the statevector extraction allows us to bypass all eventual problems and compute the expected value up to a double precision. A dynamical circuit is required in our situation to handle cases where we did not measure $|0\rangle$ in the eigenvalue register, otherwise the circuit will still try to measure the observables on the incorrect final state and this will mess up with the total average. Dynamical circuits can use measurements as controls directly on the circuit, the way it works is that the circuit can use classical bits (which are used to stock the values of the measurements) to verify boolean conditions and apply gates conditionally. This is however, still being developed and the “while” command does not exist yet, thus, the solution is to verify if $|0\rangle$ was measured, do nothing and use the “else” condition to reset the qubits (also by using dynamical circuits) and redo the whole circuit in the “else” condition. We then repeat the initial “if, else” enough times so that at the end, we have a sufficiently high probability of having the correct final state for all shots. Typically, if we have $1/2$ chance of having the correct final state and use 1024 shots to measure an observable, we need the “if, else” conditions to be repeated at the very least 10 times. The reader will already have understood how inefficient this method is. However, the real issue is that there exists a bug when using the Aer simulator on dynamical circuits ⁵ which is still unresolved to this day (January 31, 2025). Thus, it is way easier to only simulate the first and second stages of the algorithm and extract the final state when we know the algorithm was successful so that we can compute the expected values numerically.

Taking the matrix representation (1.54) of the Liouvillian from the example given in Sec. 1.3, i.e.,

$$\mathcal{L} = \begin{pmatrix} -1 & ih & -ih & 0 \\ ih & -\frac{1}{2} & 0 & -ih \\ -ih & 0 & -\frac{1}{2} & ih \\ 1 & -ih & ih & 0 \end{pmatrix}, \quad (3.32)$$

where we directly replaced the components using the dimensionless ratio h for simplicity as it will give the same steady state anyway. Building the circuit, we can simulate the algorithm for various parameters. For simplicity, we set $h = 0.5$ and $t_0 = 1/5$. The system being a two-level system, we have $N = 1$. Vectorizing the density operator and using the Hermitian operator M (3.1) thus yield an eigenvector register with $2N + 1 = 3$ qubits. In the following, we simulate the algorithm using different numbers of qubits in the eigenvalue register, i.e., different t . We also implement the controlled- U gates almost exactly (i.e., without a Suzuki-Trotter expansion) to simulate a perfect implementation of the algorithm as we are only interested in the algorithm in itself for this section. To do so, we use the quantum Shannon decomposition, to synthesize arbitrary gates [95]. The results of our simulations are shown in Fig. 3.3, which displayed the steady state expectation values of σ_y and σ_z as a function of t . As we can see, the simulations indicate that the algorithm converges indeed on the correct value when increasing the number of qubits t in the eigenvalue register.

⁵<https://github.com/Qiskit/qiskit/issues/13162>

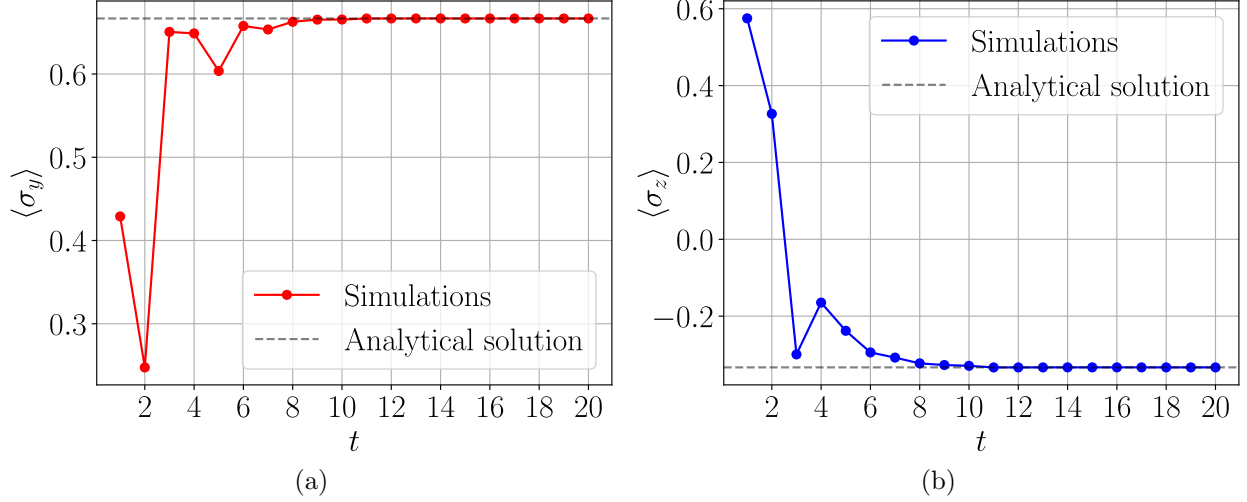


Figure 3.3: plots of $\langle \sigma_y \rangle$ (a) and $\langle \sigma_z \rangle$ (b) in terms of the number of qubits t in the eigenvalue register, with $h = 0.5$ and $t_0 = 1/5$. These plots were obtained using the Aer simulator backend in Qiskit. The dashed lines indicate the values from the analytical solutions (1.60a) and (1.60b).

3.2.2 Numerical predictions for the algorithm's outcome

Now that we have the results of the simulations, we can “manually” compute the state $|\psi_3\rangle$ and use (3.23)⁶. To obtain the expected values, the general procedure is described in the box below.

Numerical prediction procedure

1. Construct the matrix representation of the Liouvillian and then the Hermitian operator M using (3.1).
2. Compute the eigenvalues and eigenvectors of M .
3. Decompose the initial vector $|\xi\rangle$ (3.10) from the preparation stage, into the eigenvector basis of M .
4. To each coefficient c_j , multiply by the $\alpha_0^{(j)}$ coefficient (3.14).
5. Reconstruct the vector with the new coefficients (Optional: renormalize the vector).
6. Compute the expected value of $X \otimes \mathbb{1} \otimes \mathbb{1}$ using Eq. (3.23).
7. Compute the expected value of any measurement operator O using $\mathcal{Q} = X \otimes O \otimes \mathbb{1}$ and Eq. (3.23) and use the expected value of the previous step to normalize the result.

Additionally, since M is Hermitian (and generally sparse, although it is not useful in our case), specific methods can be used to solve the eigenvector problem, such as the Lanczos algorithm [96]–[98] which is simply a specific case of the Arnoldi iteration method [99].

Using the same parameters as in the simulations and plotting both the simulations and the prediction values, we obtain the plots shown in Fig.3.4. As announced earlier, the results from the simulations

⁶We should clarify that we never use (3.23) directly, what we do is we compute the state $|\psi_3\rangle$ from (3.16) and measure \mathcal{Q} which we know from (3.23), converges to the expected value of the measurement operator O , hence the indirect use.

are perfectly predicted using the procedure described above. The predicted results are accurate up to more than 10 decimal places as shown in Fig.3.5, the accuracy drops for higher numbers of qubits t which is simply due to the accumulation of numerical errors.

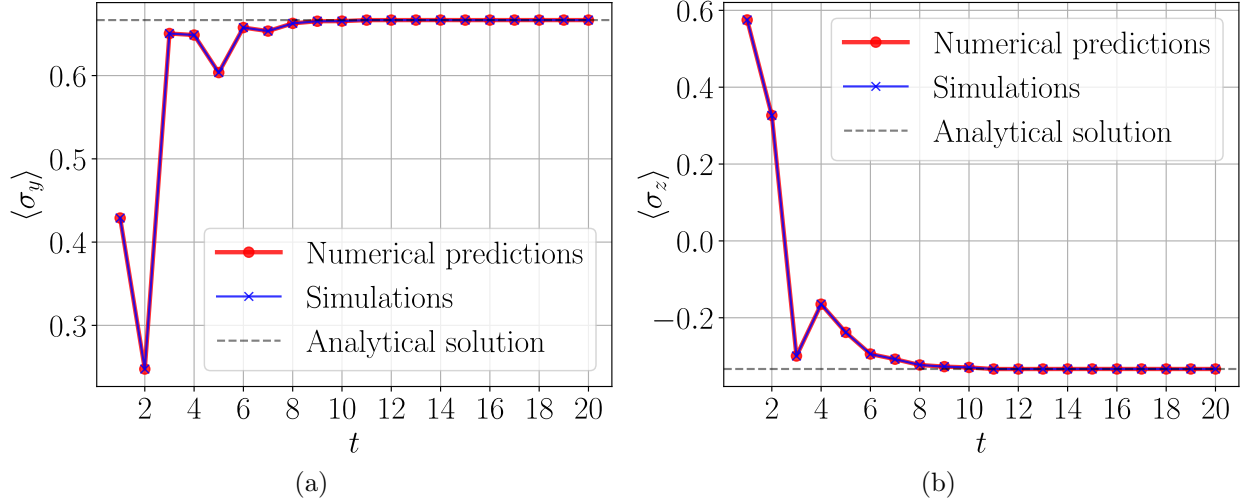


Figure 3.4: plots of $\langle \sigma_y \rangle$ (a) and $\langle \sigma_z \rangle$ (b) in terms of the number of qubits t in the eigenvalue register, with $h = 0.5$ and $t_0 = 1/5$. The lines and points in red indicate the predictions of the outcome of the algorithm using Eq. (3.23), the lines and crosses in blue indicate the results of the simulations using the Aer simulator backend in Qiskit and finally, the grey dashed lines indicate the values from the analytical solutions (1.60a) and (1.60b).

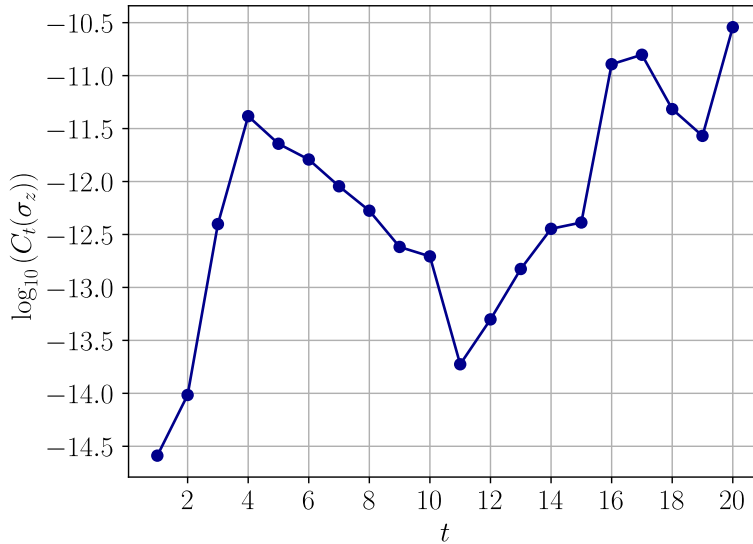


Figure 3.5: Plot of the base-10 logarithm of the precision estimator $C_t(\sigma_z)$ which is defined as $C_t(\sigma_z) = \left| \langle \sigma_z \rangle_{\text{prediction}} - \langle \sigma_z \rangle_{\text{simulation}} \right| (t)$, in terms of t .

The purpose of demonstrating that the simulations can be precisely predicted using Eq. (3.23) is to eliminate the need to actually perform them. This allows us to analyze the algorithm's behavior in more complex systems requiring significantly more qubits, where simulations would be impractical within reasonable timeframes.

3.2.3 Convergence analysis

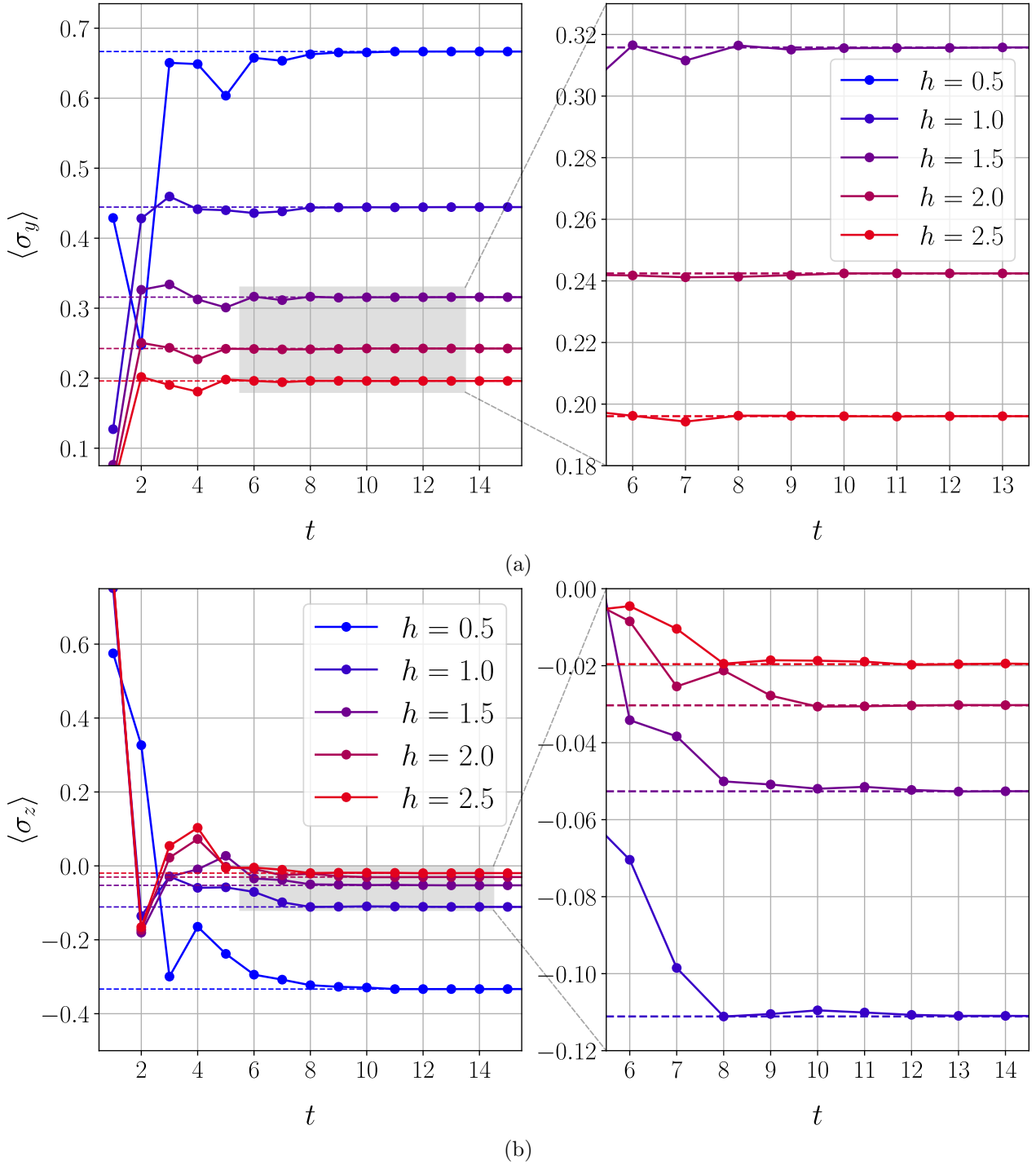


Figure 3.6: plots of $\langle \sigma_y \rangle$ (a) and $\langle \sigma_z \rangle$ (b) in terms of the number of qubits t in the eigenvalue register, for different values of h varying from 0.5 to 2.5. Furthermore t_0 is chosen as $1/(10h)$ and these results are computed from Eq. (3.23). The dashed lines, as per usual, represent the analytical values from eqs. (1.60a) and (1.60b)

To continue this section, we can now compute the expectation values of the algorithm for different

values of h and examine the convergence of the algorithm by comparing it with the analytical solution found in Sec.1.3. The first part is plotted in Fig. 3.6 To evaluate the convergence, we can define an estimator for the algorithm as follows

$$S_t(O) = \left| \langle O \rangle_{\text{Algorithm}}(t) - \langle O \rangle_{\text{Analytical}} \right| \quad (3.33)$$

with t the number of qubits in the eigenvalue register. The dependence on t is only shown for the algorithm expected value because the analytical solution has of course no such dependence. If the analytical solution was not known or simply did not exist, a common approach consists in taking the algorithm expected value for $t + 1$ instead. From now on, we will choose σ_z for the measurement operator, for the simple reason that σ_y will give similar results and thus it is simply a waste of ink and space to plot both. The convergence plots are shown in Fig. 3.7. We unsurprisingly see

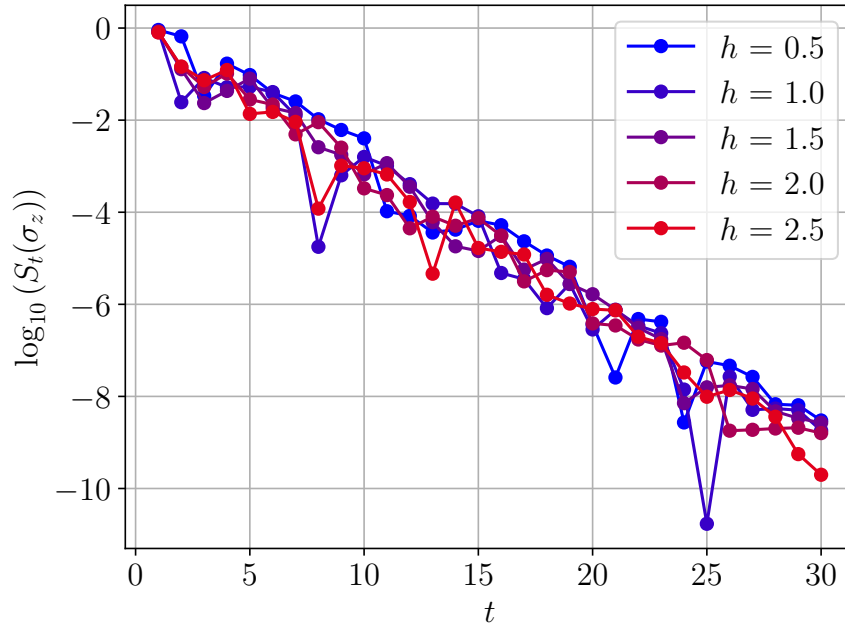


Figure 3.7: Plot of the base 10 logarithm of $S_t(\sigma_z)$ in terms of the number of qubits t in the eigenvalue register, for different values of h varying from 0.5 to 2.5. Furthermore, t_0 is chosen as $1/(10h)$.

that the algorithm converges exponentially with the number of qubits in the eigenvalue register. We also notice that the value of h seems to have as little as almost no impact on the convergence. To conclude this section, it is worth noting that the system studied here is the same as the one analyzed in the paper by Ramusat and Savona [19]. Although their analysis yields results that are very similar to ours, there are slight differences in the errors.

3.3 Adapting the RS algorithm to non-Markovian systems using the HEOM method: the RS HEOM algorithm

In this section, we will prove that the algorithm presented in Section 3.1 can be seamlessly adapted to non-Markovian systems using the HEOM method with only a few adjustments.

3.3.1 Overview

We use the HEOM Liouvillian in matrix form using Choi–Jamiołkowski isomorphism, in place of the standard Liouvillian to define the Hermitian operator

$$M_{K_{max}} = \begin{pmatrix} 0 & \mathcal{L}_{\text{HEOM}}(K_{max}) \\ \mathcal{L}_{\text{HEOM}}^\dagger(K_{max}) & 0 \end{pmatrix}, \quad (3.34)$$

where we have explicitly written the dependency of $\mathcal{L}_{\text{HEOM}}$ on K_{max} to emphasize that both the quantum circuit and the size of the quantum register now depend on the chosen cutoff. This means that the preparation gate \mathcal{P} and the measurement operators will have to be adapted in consequence. Furthermore, we will distinguish, in the whole density operator, the part representing the physical system $\rho^{(0,0)}$ and we will similarly assume that its Hilbert space has a size 2^{N_0} , where the 0 index is used to highlight the fact that this does not represent the entirety of the system. Another important detail to take into account is that the entire size of the system cannot be considered as a pure power of 2 even though the physical system is ⁷. The total size varies with K_{max} and thus, has to be systematically increased to the next power of 2 by appending either 1's on the diagonal of the HEOM Liouvillian or 0's but in this case, one has to be careful to prepare a state with the \mathcal{P} gate that has no overlap with the inevitably new eigenvectors composing the nullspace. Assuming the entire system is now of size 2^N ⁸, we can define $N_1 = N - 2N_0$, to account for the number of qubits representing the ancillary states in the total number of qubits for the system, where the factor 2 in front of N_0 comes from the vectorization of the density operator. The exact value for N and N_1 can be found by using the fact that the HEOM method uses in total $\frac{(2M+K_{max})!}{(2M)!K_{max}!}$ ancillary states with M the total number of pseudo-modes coupled to the system. This yields the following expressions

$$N_1 = \left\lceil \log_2 \left(\frac{(2M + K_{max})!}{(2M)!K_{max}!} \right) \right\rceil, \quad (3.35a)$$

$$N = 2N_0 + \left\lceil \log_2 \left(\frac{(2M + K_{max})!}{(2M)!K_{max}!} \right) \right\rceil. \quad (3.35b)$$

The full quantum circuit is illustrated in Fig 3.8. Similarly as the RS algorithm, it is composed of a preparation stage, a quantum phase estimation stage and a measurement stage.

⁷Indeed, the factorial implies of course that successive numbers will be multiplied, the $2M$ also implies that ($M = 1$), there will always be at the very least, two successive number, the only two numbers that can give a strict power of 2 in this scenario are 1 and 2 ($K_{max} = 0$). Furthermore, it is strictly impossible that for $M > 1$, a strict power of 2 exists since three successive numbers multiplied will always give a number that can be divided by 3.

⁸Note that we do not need to increase the size to 2^{2N} with N being an integer (Thus a strict power of 4), to match the scenario from RS algorithm. We still need to increase the size to a power of 2 for the simple reason that this is required to be implemented as a quantum gate (It has to act on a number of qubits N which implies that any operator of this Hilbert space has a size $2^N \times 2^N$)

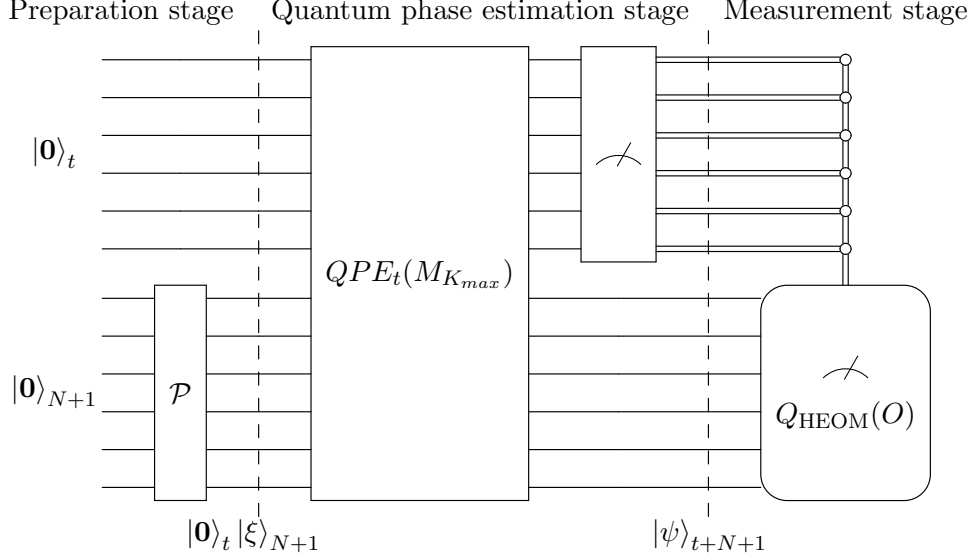


Figure 3.8: Quantum circuit diagram for the full algorithm. The algorithm is composed of an input-state preparation stage, a quantum phase estimation stage and, if successful, a measurement stage. The first qubit register contains t qubits, having more qubits in this register allows for higher precision but at a higher cost in gates and qubits. The second quantum register is composed of $N+1$ qubits and represents the statevector of the system. The quantum phase estimation algorithm uses t controlled- U^{2^j} gates defined in Eq. (3.39), applied on the second quantum register and an inverse quantum Fourier transform applied on the first quantum register. The classical wires (double lines) indicate the conditional measurement applied on the second register if the algorithm was successful, i.e., that $\mathbf{0}_t$ was measured in the eigenvalue register. The \mathcal{P} gate is explicitly depicted in Figure 3.9. And the measurement operator $Q_{\text{HEOM}}(O)$ is defined in Eq. (3.49).

3.3.2 Preparation stage

The idea of Ramusat and Savona's paper was to prepare a state having a large overlap with the degenerate nullspace by taking advantage of the fact that the left eigenvector of \mathcal{L} , $|\mathbf{1}\rangle$ was known. This idea still works for HEOM, however, we must take into account that the left eigenvector of $\mathcal{L}_{\text{HEOM}}$ is $|\mathbf{1}^{(0,0)}\rangle$ [56]. This notation designates that we have the vectorized identity on the first $2N_0$ qubits and then $\mathbf{0}$ for the remaining N_1 qubits. This is a consequence of the fact that the normalization condition for ρ is that $\text{Tr}[\rho^{(0,0)}] = 1$. We thus have that the two eigenvectors of $M_{K_{\max}}$ associated with the eigenvalue 0 are given by

$$|\eta_0\rangle = |\mathbf{0}\rangle |\mathbf{0}\rangle_{N_1} |\mathbf{1}\rangle_{2N_0}, \quad (3.36a)$$

$$|\eta_1\rangle = |\mathbf{1}\rangle |\rho_{ss}\rangle_{N_1+2N_0}, \quad (3.36b)$$

where we added subscripts to clarify the dimensions of each part. This means that the state we need to prepare is

$$|\xi\rangle = \frac{|\mathbf{0}\rangle |\mathbf{0}\rangle_{N_1} |\mathbf{1}\rangle_{2N_0} + |\mathbf{1}\rangle |\mathbf{0}\rangle_{N_1} |\mathbf{0}\rangle_{2N_0}}{\sqrt{2}} = \frac{|\eta_0\rangle_{N+1} + |\mathbf{1}\rangle |\mathbf{0}\rangle_{N_1} |\mathbf{0}\rangle_{2N_0}}{\sqrt{2}}, \quad (3.37)$$

where the indices are used to indicate the number of qubits required for each sub-state. Note that this state does not depend on the cutoff K_{\max} apart from its size and, as such, neither does the \mathcal{P} gate. This is a major difference and a convenient advantage, compared to the original algorithm

used in the context of a Markovian embedding approach, where the vectorized identity has to be prepared by taking into account the Fock space cutoff N_c and thus the gate has to be modified accordingly. The state (3.37) can be prepared through the \mathcal{P} gate, as shown in Fig. 3.9.

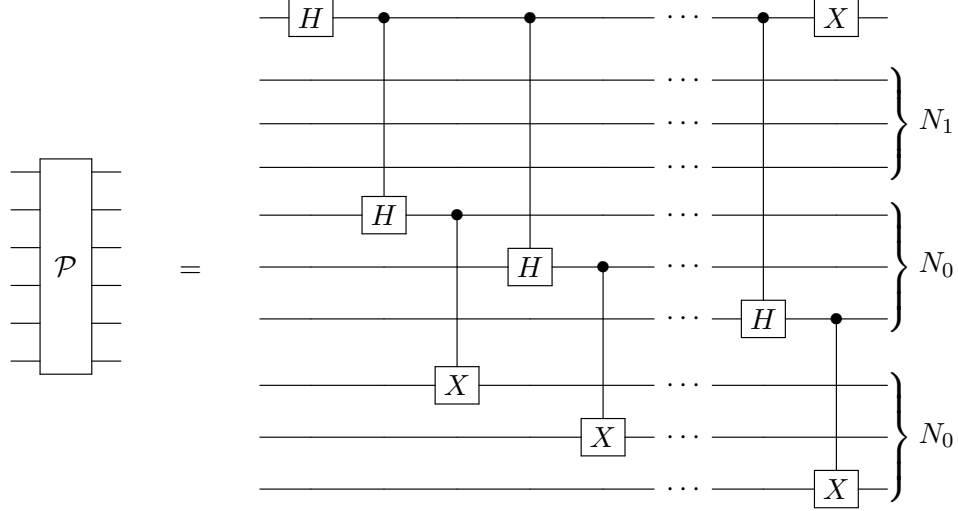


Figure 3.9: \mathcal{P} gate explicitly drawn as a quantum circuit for the HEOM adaptation.

The state of the circuit after the \mathcal{P} gate is then given by

$$|0\rangle_t |\xi\rangle_{N+1} = \frac{1}{\sqrt{2}} |0\rangle_t (|0\rangle_{N_1} |0\rangle_{N_1} |1\rangle_{2N_0} + |1\rangle_{N_1} |0\rangle_{N_1} |0\rangle_{2N_0}) \quad (3.38)$$

3.3.3 Quantum phase estimation stage

The quantum phase estimation stage follows the same logic as in Sec. 3.1. Evidently, we need to use the Hermitian operator $M_{K_{max}}$ for the controlled- U^{2^j} gates which are thus defined as

$$U^{2^j} = e^{2\pi i t_0 M_{K_{max}} 2^j}, \quad (3.39)$$

where t_0 is chosen so that the whole spectrum of $M_{K_{max}}$ is in the interval $[0, 1]$. A convenient estimation of t_0 can be obtained by determining an upper bound of the 1-norm of $\mathcal{L}_{\text{HEOM}}$, which can be straightforwardly determined by considering the 1-norms of the sub-matrices of $\mathcal{L}_{\text{HEOM}}$. However, the attentive reader may have already noticed that the $\mathcal{L}_{\text{HEOM}}$ matrix has its sub-matrices of different physical dimensions. For example, using the expression (1.78) for HEOM at one mode, the A and B matrices have s^{-2} units, the D matrices have s^{-1} units while the C matrices are dimensionless. We have empirically found that replacing the $|g_j|^2$ by $|g_j|$ and multiplying by $|g_j|$ when the term is dimensionless so that the whole expression is dimensionally coherent yields a better upper bound for $\max_k |\lambda_k|$. The expression found is given below.

$$\begin{aligned}
\max_k |\lambda_k| &\leq 2\|H_S\|_1 \\
&+ K_{max} \left(\max_j \left(|\kappa_j + i\omega_j| + |g_j| \|L\|_1 \right) \right) \\
&+ 2 \sum_{j=1}^M |g_j| \|L\|_1,
\end{aligned} \tag{3.40}$$

where the 1-norm of H_S and L depend on the studied system and can generally be obtained with relative ease. Note that this inequality is almost equal for $|g_j| \ll |\kappa_j + i\omega_j|$ as shown in Fig. 3.10. Note that the choice of t_0 directly impacts the number of qubits required in the eigenvalue register. The classical formulation of the quantum phase estimation can be summarized (cf. Sec. 2.3) for an eigenvector $|\eta_j\rangle$ of $M_{K_{max}}$, associated with the eigenvalue φ_j , through the relation

$$QPE_t(M_{K_{max}}) |\mathbf{0}\rangle_t |\eta_j\rangle_{N+1} = \left(\sum_{l=0}^{2^t-1} \alpha_l^{(j)} |l\rangle_t \right) |\eta_j\rangle_{N+1}, \tag{3.41}$$

where

$$\alpha_l^{(j)} = \frac{1}{2^t} \frac{1 - e^{2i\pi(t_0\varphi_j 2^t - l)}}{1 - e^{2i\pi(t_0\varphi_j - l/2^t)}},$$

which is the same definition as (3.14). This relation is only valid if $t_0\varphi_j \neq l/2^t$, $\forall l = 0, 1, \dots, 2^t - 1$, otherwise, if we assume an integer k such that $t_0\varphi_j = k/2^t$, the sum in (3.41) vanishes, and we are simply left with $|k\rangle_t |\eta_j\rangle_{N+1}$. The next step is then to measure the t qubits composing the eigenvalue register, in the computational basis. In the general case, there is a probability of at least $4/\pi^2$ to measure the best t bits estimate of φ_j and 1 if k exists. In our situation, we have a state having a large overlap with the nullspace of $M_{K_{max}}$ and which is not an eigenstate. The purpose is to create entanglement between the eigenstates and eigenvalues using the QPE , and hope we measure $\mathbf{0}_t$ in the eigenvalue register so that the final state will be the projection of the initial state onto the nullspace. Expanding $|\xi\rangle$ on the basis of eigenvectors $|\eta_j\rangle$ of $M_{K_{max}}$ yields

$$|\xi\rangle = \frac{1}{\sqrt{2}} |\eta_0\rangle + \sum_{j \neq 0}^{2^{N+1}-1} c_j |\eta_j\rangle, \tag{3.42}$$

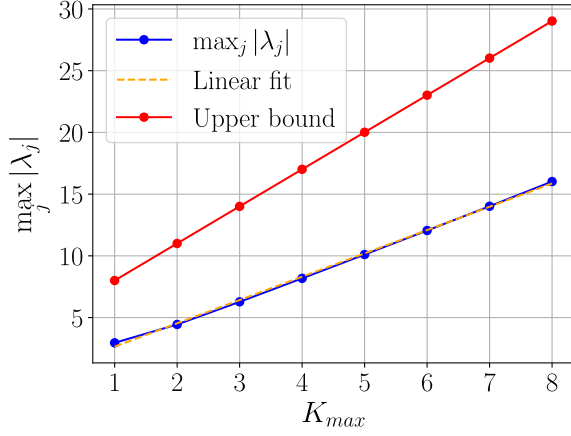
with c_j being the projection of $|\xi\rangle$ onto the eigenstate $|\eta_j\rangle$, i.e.,

$$c_j = \langle \eta_j | \xi \rangle \tag{3.43a}$$

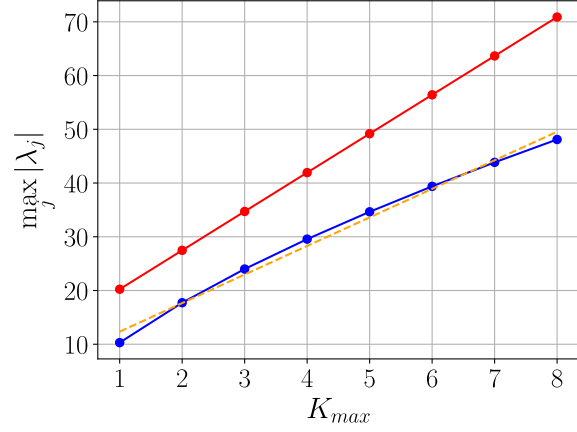
$$= \langle \eta_j | 1 \otimes \mathbf{0}_{N1} \otimes \mathbf{0}_{2N_0} \rangle, \tag{3.43b}$$

since $\langle \eta_0 | \eta_j \rangle = 0$ for $j \neq 0$. We can then apply the quantum phase estimation on our state $|\xi\rangle$ which gives

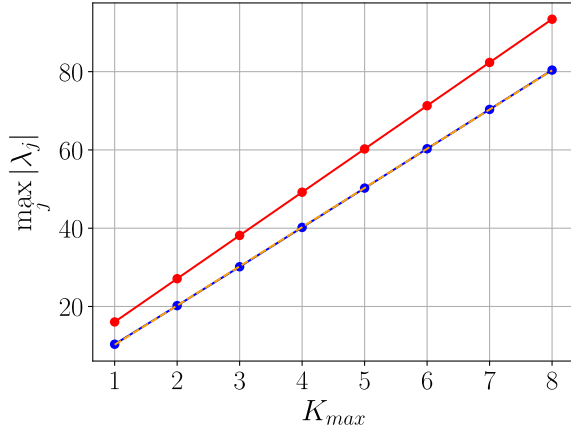
$$\begin{aligned}
QPE_t(M_{K_{max}}) |\mathbf{0}\rangle_t |\xi\rangle_{N+1} &= |\mathbf{0}\rangle_t \left(\frac{1}{\sqrt{2}} |\eta_0\rangle_{N+1} + c_1 |\eta_1\rangle_{N+1} \right) \\
&+ \sum_{j \neq 0,1}^{2^{N+1}-1} c_j \left(\sum_{l=0}^{2^t-1} \alpha_l^{(j)} |l\rangle_t \right) |\eta_j\rangle_{N+1},
\end{aligned} \tag{3.44}$$



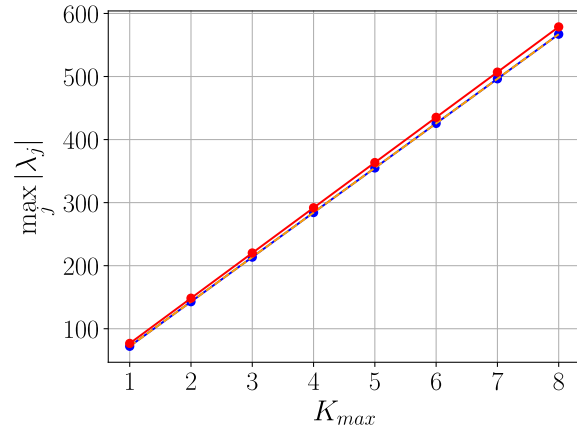
(a) $\omega_0 = 1.0, \omega_1 = 0.1, g_1 = 1.0, \kappa_1 = 2.0$



(b) $\omega_0 = 1.0, \omega_1 = 1.0, g_1 = 5.0, \kappa_1 = 2.0$



(c) $\omega_0 = 1.0, \omega_1 = 1.0, g_1 = 1.0, \kappa_1 = 10.0$



(d) $\omega_0 = 1.0, \omega_1 = 50.0, g_1 = 1.0, \kappa_1 = 50.0$

Figure 3.10: Plots of the largest eigenvalue (in blue) of $\mathcal{L}_{\text{HEOM}}$ in terms of K_{max} in the case of the two-mode open quantum Rabi model presented in Sec. 3.4.3, for multiple parameter values. Here we set $\omega_2 = 0.5, g_2 = 0.5$ and κ_2 for all 4 plots as only the largest parameters among the multiple modes contribute. The orange dashed line is the linear fit obtained with the blue points and the red line represents the upper bound (3.40).

where we assume for simplicity, that the non-zero eigenvalues of $M_{K_{max}}$ do not admit exact binary decompositions with t bits, so that the $\alpha_l^{(j)}$ coefficients are well-defined. Assuming we measured $\mathbf{0}_t$ in the eigenvalue register, we are left with the following state

$$|\psi\rangle_{N+t+1} = |\mathbf{0}\rangle_t |\Phi\rangle_{N+1}. \quad (3.45)$$

where

$$|\Phi\rangle_{N+1} = \frac{|0\rangle |\mathbf{0}\rangle_{N_1} |\mathbf{1}\rangle_{2N_0}}{\sqrt{2}} + c_1 |1\rangle |\rho_{ss}\rangle_N + \sum_{j \neq 0,1}^{2^{N+1}-1} c_j \alpha_0^{(j)} |\eta_j\rangle_{N+1}, \quad (3.46)$$

where we have written explicitly the expression for $|\eta_0\rangle$ and $|\eta_1\rangle$. The probability to measure $\mathbf{0}_t$ will always be greater than $1/2$ owing to the $1/\sqrt{2}$ factor in front of $|\eta_0\rangle$ in Eq. (3.42)(cf. Sec 3.1).

3.3.4 Measurement stage

The underlying principle of the measurement stage in Ramusat and Savona's paper was to use the identity

$$\begin{aligned} \langle O \rangle &= \mathcal{N} \text{Tr}[O\rho] \\ &= \mathcal{N} \langle \mathbf{1} | O \otimes \mathbf{1} | \rho \rangle \end{aligned} \quad (3.47a)$$

$$= \mathcal{N} \langle \rho | O \otimes \mathbf{1} | \mathbf{1} \rangle, \quad (3.47b)$$

with \mathcal{N} a normalization constant coming from the fact that the state $|\rho\rangle$ is unnormalized from a density operator standpoint, O is a measurement operator and $\mathbf{1}$ the identity operator, both acting on the Hilbert space of the system. The purpose was then to use a quantum measurement Q defined as

$$Q = X \otimes O \otimes \mathbf{1},$$

so that we can take advantage of the fact that the states primarily surviving the quantum phase estimation stage are the eigenstates associated with the eigenvalue 0 which are precisely $|\mathbf{1}\rangle$ and ρ_{ss} .

In our non-Markovian case, the eigenstates that survive are $|\mathbf{1}^{(0,0)}\rangle$ and $|\rho_{ss}\rangle$. Furthermore, the former is what we need (from a projector perspective) to measure the part $\rho^{(0,0)}$ from the total density operator through the relation

$$\text{Tr}[O\rho^{(0,0)}] = \left\langle \mathbf{1}^{(0,0)} \left| \mathbf{1}_{N_1} \otimes O \otimes \mathbf{1}_{N_0} \right| \rho \right\rangle \quad (3.48a)$$

$$= \left\langle \mathbf{1}^{(0,0)} \left| \mathbf{1}_{N_1} \otimes O \otimes \mathbf{1}_{N_0} \right| \rho \right\rangle, \quad (3.48b)$$

where O is a measurement operator acting on the Hilbert space of the system and thus on N_0 qubits, and $\mathbf{1}_{N_j}$ is an operator acting trivially on N_j qubits, i.e. $\underbrace{\mathbf{1}_{2 \times 2} \otimes \cdots \otimes \mathbf{1}_{2 \times 2}}_{N_j}$. We can similarly define

the operator

$$Q_{\text{HEOM}}(O) = X \otimes \mathbf{1}_{N_1} \otimes O \otimes \mathbf{1}_{N_0}, \quad (3.49)$$

where we added a O argument to specify which measurement operator is used. This operator acts

on $N + 1$ qubits and has the following properties

$$\langle \eta_0 | Q_{\text{HEOM}}(O) | \eta_0 \rangle = 0, \quad (3.50a)$$

$$\langle \eta_1 | Q_{\text{HEOM}}(O) | \eta_1 \rangle = 0, \quad (3.50b)$$

$$\langle \eta_0 | Q_{\text{HEOM}}(O) | \eta_1 \rangle = \text{Tr}[O\rho], \quad (3.50c)$$

$$\langle \eta_1 | Q_{\text{HEOM}}(O) | \eta_0 \rangle = \text{Tr}[O\rho], \quad (3.50d)$$

which stems from the fact that the X gate flips the corresponding qubit and the expressions (3.36a) and (3.36b) for $|\eta_0\rangle$ and $|\eta_1\rangle$ and that Q is Hermitian. Measuring the state (3.46) with the operator (3.49) yields the following expression

$$\begin{aligned} \langle \Phi | Q_{\text{HEOM}}(O) | \Phi \rangle &= c_1 \sqrt{2} \text{Tr} \left[O \rho_{ss}^{(0,0)} \right] \\ &+ 2 \text{Re} \left[\sum_{j \neq 0,1}^{2^{N+1}-1} c_j \alpha_0^{(j)} \left(\frac{1}{\sqrt{2}} \langle \eta_0 | Q_{\text{HEOM}}(O) | \eta_j \rangle + c_1^* \langle \eta_1 | Q_{\text{HEOM}}(O) | \eta_j \rangle \right) \right] \\ &+ \sum_{j \neq 0,1}^{2^{N+1}-1} \sum_{k \neq 0,1}^{2^{N+1}-1} c_j^* c_k \alpha_0^{*(j)} \alpha_0^{(k)} \langle \eta_j | Q_{\text{HEOM}}(O) | \eta_k \rangle, \end{aligned} \quad (3.51)$$

where the second and third terms are errors originating from the quantum phase estimation stage and decrease exponentially with the number of qubits t in the eigenvalue register. In order to eliminate both the norm of the steady state and the $c_1 \sqrt{2}$ pre-factor, we can replace the measurement operator O by the identity operator $\mathbb{1}_{N_0}$ so that the expression (3.51) goes to $c_1 \sqrt{2} \text{Tr} \left[\rho_{ss}^{(0,0)} \right]$ for $t \rightarrow \infty$. We can then compute any expectation value using

$$\begin{aligned} \langle O \rangle &= \frac{\text{Tr}[O \rho_{ss}]}{\text{Tr}[\rho_{ss}]} \\ &\approx \frac{\langle \Phi | Q_{\text{HEOM}}(O) | \Phi \rangle}{\langle \Phi | Q_{\text{HEOM}}(\mathbb{1}_{N_0}) | \Phi \rangle}. \end{aligned} \quad (3.52)$$

This relation, along with the expression (3.46) for $|\Phi\rangle$, is what we use to predict the outcome of the algorithm for benchmarking purposes, thus bypassing the need to emulate the whole circuit. This allows us to be relatively unrestricted in the number of qubits in the eigenvalue register, the downside being that in order to compute the state $|\Phi\rangle$, we need to know every eigenvector and eigenvalue of the matrix $M_{K_{max}}$ which can become computationally challenging for large systems.

3.3.5 Complexity analysis

We still have to characterize the scaling of the algorithm, luckily for us, the discussion at the end of Section 3.1 remains valid and we simply need to account for the increased size of the system. Furthermore, we will only consider the QPE scaling (combined with the measurement stage scaling ϵ_m) as it dominates the overall scaling of the algorithm. We have in this setup that the number of gates scales as

$$\#_{\text{gates}} = \mathcal{O}\left(\frac{(N+1)^k}{(t_0 g)^{1+1/p} \epsilon_p^{1/2+1/2p} \epsilon_\tau^{1/p}}\right), \quad (3.53)$$

where the expression for N is defined in (3.35b). We see that the HEOM method primarily introduced a linear dependence of K_{\max} which is hidden in the expression of t_0 with (3.40). Since N_0 was logarithmically defined with regards to the size of the system, $(N+1)^k$ scales polylogarithmically. Furthermore, the scaling depends on the ratio between the largest eigenvalue (through t_0) and the HEOM Liouvillian gap g . Assuming this ratio scales as a power law of the system, this algorithm is thus expected to perform with an exponential advantage compared to exact diagonalization. For notational convenience, we will refer to this adaptation as “RS HEOM algorithm” for the remainder of this work.

3.4 Applications of the RS HEOM algorithm

In this section, we will apply the algorithm explained in the previous section to the same model as in Sec. 1.5, i.e., the open Rabi model. We will then compare it against the original RS algorithm in the case of a Markovian embedding approach. As a second stage, we will apply both algorithms to a two-mode open quantum Rabi model.

3.4.1 Open quantum Rabi model

The point of this subsection is the same as Sec. 3.2. The first step consists in doing simulations of the circuit and then proving that we can bypass the simulations using Eq. (3.52) in order to study large system sizes and the algorithm’s convergence.

For self-evident reasons, the simulations were only done for $K_{\max} = 1, 2$. Note that the tools and procedures used here are exactly the same as in Sec. 3.2 and as such, they will not be detailed again. The comparison plots with the numerical predictions using Eq. (3.52) are directly shown in Fig. 3.11. Evidently, these results were to be expected, the simulations being done as a verification for the algorithm validness.

We can now evaluate the convergence of the algorithm. However, a subtlety is that we have two parameters that control the convergence in this algorithm: the order K_{\max} and the number of qubits t in the eigenvalue register. As such we define the following two convergence estimators $S_{K_{\max}}(O)$ and $S_t(O)$

$$S_{K_{\max}}(O) = |\langle O(K_{\max} + 1, t) \rangle - \langle O(K_{\max}, t) \rangle| \quad (3.54)$$

$$S_t(O) = |\langle O(K_{\max}, t + 1) \rangle - \langle O(K_{\max}, t) \rangle|. \quad (3.55)$$

Note that these estimators, even if they are defined to estimate the convergence for only one parameter, depend in fact on both parameters. As a second remark, we defined the second estimator by comparing successive increments. Nevertheless, one could alternatively compare it directly to the classical nullspace computation. However, this approach was not adopted because, from a purely quantum computing perspective, the classical results are assumed to be unknown. Therefore, we opted for this point of view in defining the estimator. Finally, from these two estimators, we can

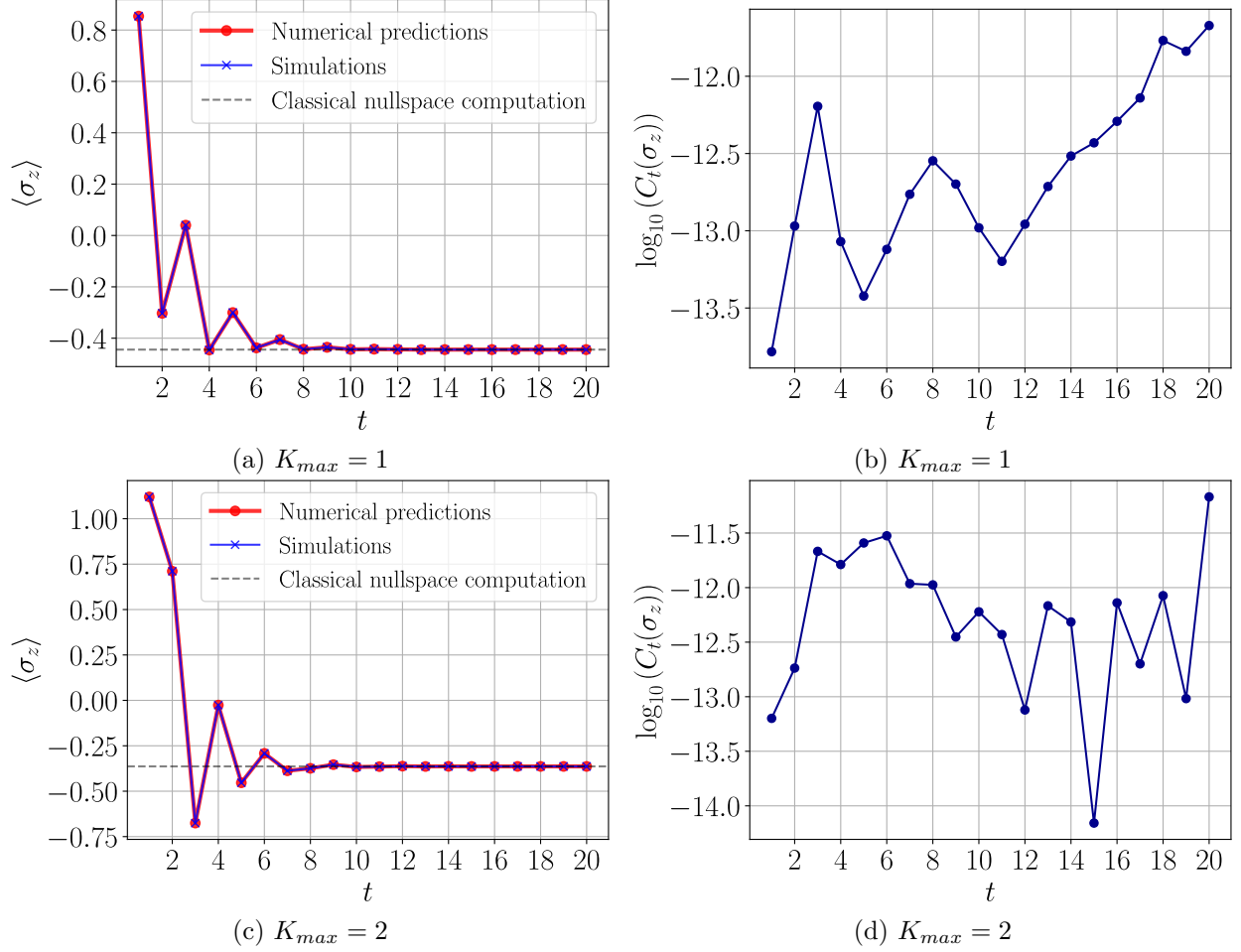


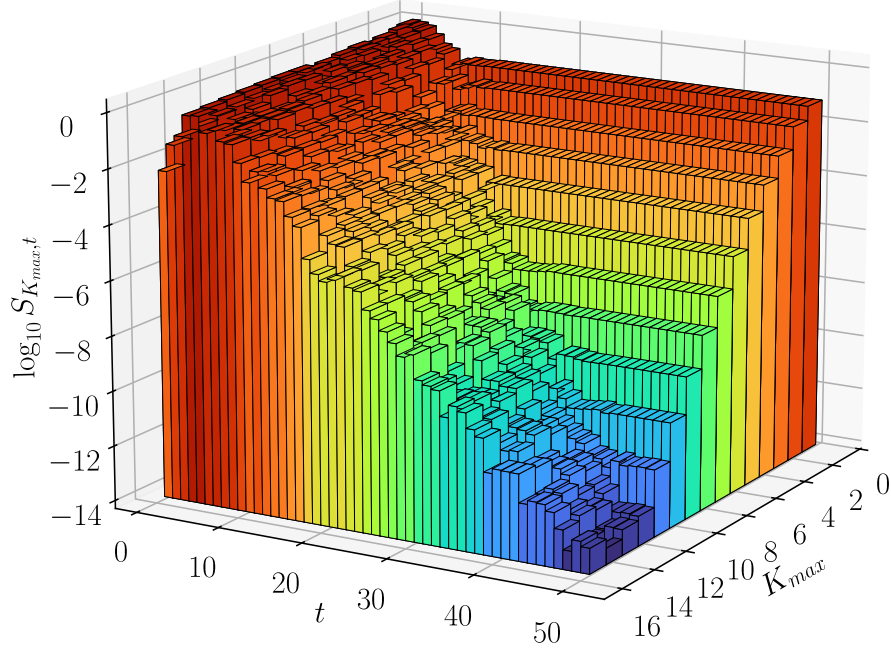
Figure 3.11: Left figures: $\langle \sigma_z \rangle$ in terms of the number of qubits t in the eigenvalue register for $K_{max} = 1$ (a) and $K_{max} = 2$ (c). For all computations and simulations, we have $\omega_0 = g = \omega = \kappa/2 = 1.0$ and $t_0 = 1/10$. The line and dots in red represent the numerical predictions obtained via Eq. (3.52) while the line and dots in blue represent the simulations done on the Aer simulator in Qiskit. The dashed grey line represents the classical nullspace computation obtained in the previous Section, using Eq. (1.98)(a) and Eq. (1.99)(c). Right figures: Precision estimator $|\langle \sigma_z \rangle_{\text{prediction}} - \langle \sigma_z \rangle_{\text{simulation}}|$ plotted in base 10 logarithm in terms of the number of qubits t in the eigenvalue register for $K_{max} = 1$ (b) and $K_{max} = 2$ (d). The numerical predictions and simulations are in accordance to more than 11 decimals.

define a third one that aims to estimate the complete convergence

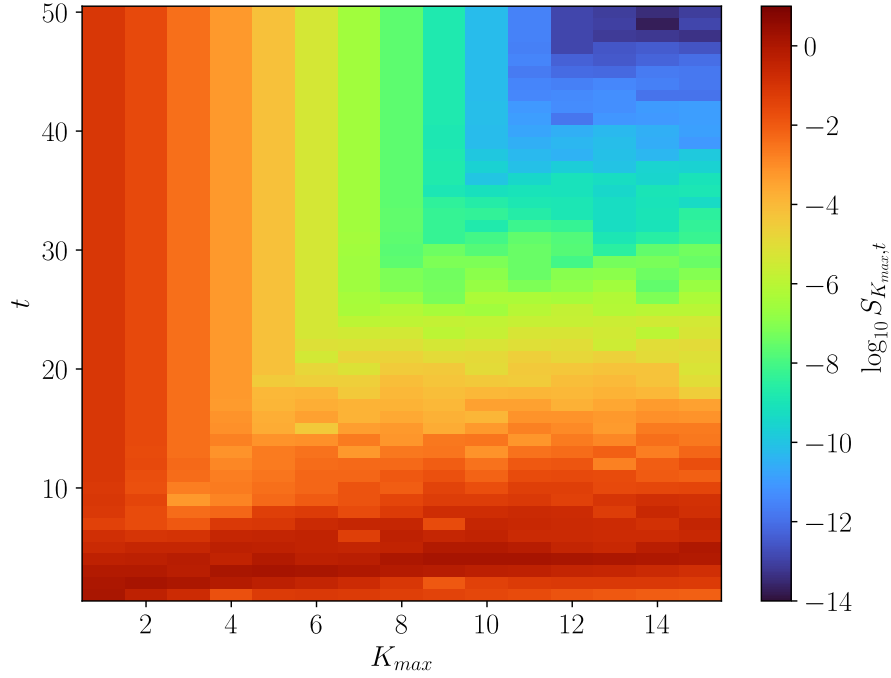
$$S_{K_{max},t}(O) = \sqrt{(S_{K_{max}}(O))^2 + (S_t(O))^2}. \quad (3.56)$$

This estimator simply corresponds to the Euclidian norm of the first two and this definition is purely arbitrary, as we could have chosen the Manhattan norm for example. The results, however, would not differ much.

The estimator (3.56) is plotted for different parameters in Fig 3.12. We can see on both plots the two different regimes, either i) t is not large enough compared to K_{max} and the “steps” seem erratic or ii) K_{max} is not large enough compared to t and the “steps” seem remarkably regular. Furthermore, while the exponential convergence was already established in Section 3.2, we notice



(a)



(b)

Figure 3.12: 3D bar plot (a) and density plot (b) of the base 10 logarithm of the convergence estimator $S_{K_{max}, t}(\sigma_z)$ in the case of the open quantum Rabi model for $g = \omega = \omega_0 = \kappa/2 = 1.0$, with the number of qubits t in the eigenvalue register varying from 1 to 50 and the HEOM cutoff parameter K_{max} varying from 1 to 15. The color scale is the same for both plots. Note that t_0 was not chosen using the determined upper bound (3.40) but with a linear fit. The reason behind this choice was so that these results could be compared later, with the Markovian embedding approach.

that the results here suggest a faster than exponential convergence for K_{max} ⁹ since the difference between each “step” increases. As previously said, any number of qubits t in the eigenvalue register takes the same amount of time when computing, the reason we did not go further than 50 qubits is because the precision of the machine is reached around there and as such, the errors stagnate.

3.4.2 RS VS RS HEOM algorithm: open quantum Rabi model

As mentioned earlier in this work (cf. Chapter 1 and more precisely Subsection 1.4.1), the simplest way conceptually, to simulate non-Markovian systems is to encompass the part of the environment containing the interaction with the studied system (i.e., the pseudo-modes), into a larger Markovian system. This approach, commonly referred to as Markovian embedding, can be used directly in the case of the RS algorithm without any major adaptation. The purpose of this section is to compare both non-Markovian methods to determine if the HEOM adaptation has any meaningful advantage.

In the case of the Markovian embedding approach, the enlarged Hilbert space is $\mathcal{H} = \mathcal{H}_{sys} \otimes \mathcal{H}_{env}$ ¹⁰, where \mathcal{H}_{env} represents the pseudo-modes Hilbert space, and we use the notion of tensor product extension to adapt operators acting on only one of the Hilbert space to the enlarged Hilbert space. For example, any operator O_{sys} acting on \mathcal{H}_{sys} can be adapted to \mathcal{H} using

$$O = O_{sys} \otimes \mathbb{1}_{env}. \quad (3.57)$$

In the case of the open quantum Rabi model, the system is the two-level system representing the atom while the environment is the cavity mode surrounding the atom. These operator extensions allow the Hamiltonian of the system (and thus the Liouvillian) to be expressed as a single equation given by (1.88) where the absence of indices indicates that the operators are extended to the enlarged Hilbert space and as such, we can implement it using RS algorithm. The size of the Hilbert space of the cavity mode is infinite in theory as there is no limit to the number of photons that can be present within the cavity. However, for numerical computations, we introduce a cutoff N_c for the size of the Fock space describing the cavity, as in Sec. 1.4.1. The cutoff N_c is analogous to the cutoff parameter K_{max} used for the HEOM method, nonetheless, there are important differences between the two. Notably, N_c is a square cutoff, meaning if we write explicitly the density operator for the cavity, we have

$$\rho_{env} \approx \sum_{j=1}^{N_c} p_j |\psi_j\rangle \langle \psi_j|, \quad (3.58)$$

if we vectorize this density operator, we obtain

$$|\rho_{env}\rangle \approx \left(\rho^{(1,1)}, \rho^{(1,2)}, \dots, \rho^{(1,N_c)}, \rho^{(2,1)}, \dots, \rho^{(N_c,N_c)} \right)^T. \quad (3.59)$$

This means that any $\rho^{(i,j)}$ where $\max(i, j) \leq N_c$, will be considered in the computations. Nevertheless, as mentioned in Section 3.3, the pre-factor has to be increased to a pure power of 2 to be implemented on quantum computers as gates.

In order to compare both methods without biases, we computed the largest eigenvalues numerically to determine the best possible value for t_0 ¹¹. Something similar was already done for the

⁹At least faster than a simple exponential.

¹⁰Note that the “env” index is used somewhat loosely, as it represents only a part of the environment, i.e., the pseudo-modes.

¹¹Estimating the upper bound for the Markovian embedding method and using the one determined in Sec. 3.3 could lead to an advantage for one or the other. We want to compare first without taking this into account.

HEOM adaptation in Fig 3.10, for the Markovian embedding approach, it is displayed in Fig 3.13. Another detail to keep in mind is that the minimal value for N_c is 2, whereas K_{max} can be equal to 1. Indeed, the operators a and a^\dagger link two different states of the cavity, meaning we need at least two different states for them to contribute to the dynamics.

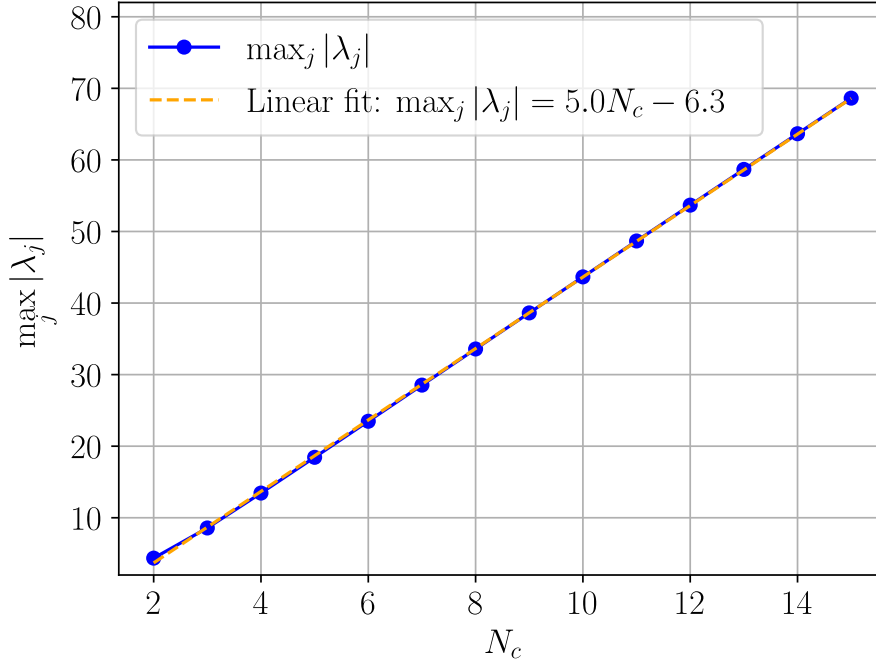


Figure 3.13: Plot of the largest eigenvalue of \mathcal{L}_M in terms of N_c in the case of the Rabi model for $g = \omega = \omega_0 = \kappa/2 = 1.0$. The blue dotted line is the plot of the largest eigenvalue computed numerically with the orange dashed line being its linear fit.

The outcome of the algorithm can be predicted using the exact same approach as in Section 3.2. Furthermore, we can define a convergence estimator $S_{N_c,t}(O)$ using the same principles as in Section 3.4.1. This estimator leads to plots that are extremely similar to those in Fig. 3.12 which are displayed in Fig. 3.14. It is rather complicated using Figs. 3.12 3.14 to compare which method is better. One way to compare the two methods is to compute the required resources to reach an arbitrary precision threshold and compare them. For example, we can take the difference with the number of qubits required in the eigenvalue register and compute the ratio of the dimensions of the systems. This approach is illustrated in Fig. 3.15, where we have considered the dimensions of the Liouvillian for which their size are not increased to the next pure power of 2. As we can see, there is no real advantage in terms of HEOM or Markovian embedding Liouvillian dimensions, but there is a clear advantage in terms of required numbers of qubits in the eigenvalue register, t . At first glance, this difference may seem insignificant, especially for higher precision thresholds where the relative advantage is less than 10%¹². However, recalling the discussion in Section 3.1 regarding the scaling of the algorithm, the number of gates required for implementing one controlled- U^{2^j} scales with 2^j (and we have t such controlled gates as $j = 0, t-1$). As such, even one qubit of difference means that we would require twice as many gates using the Markovian embedding approach, assuming one Trotter step takes the same resources for both methods. On average, the RS HEOM adaptation requires three qubits less than its counterpart, which translates to a gate cost approximately 8 times cheaper.

¹²Typically, for the last point, we have $t_M = 48$ and $t_{HEOM} = 45$.

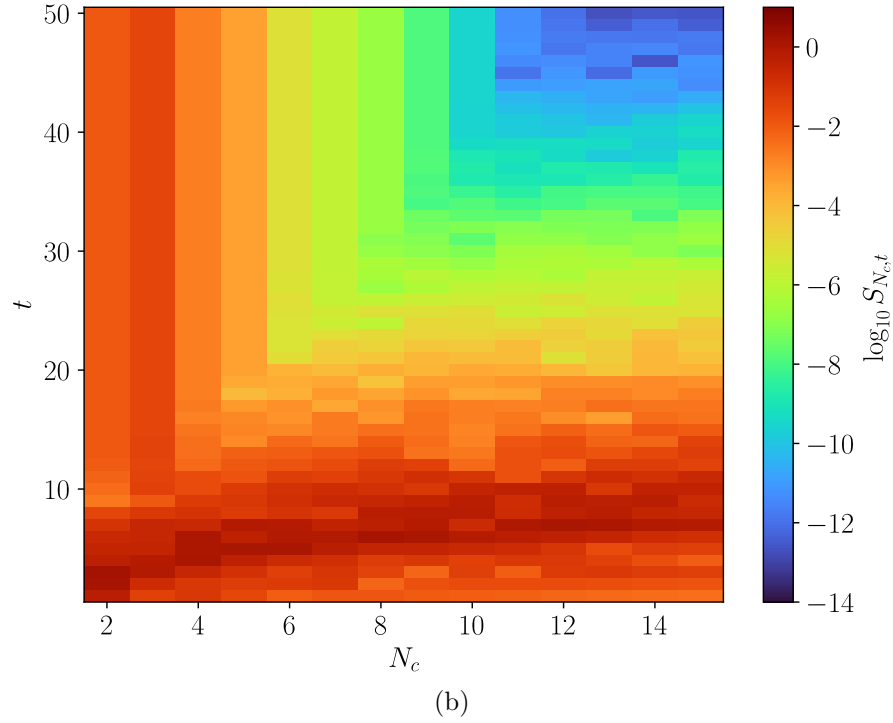
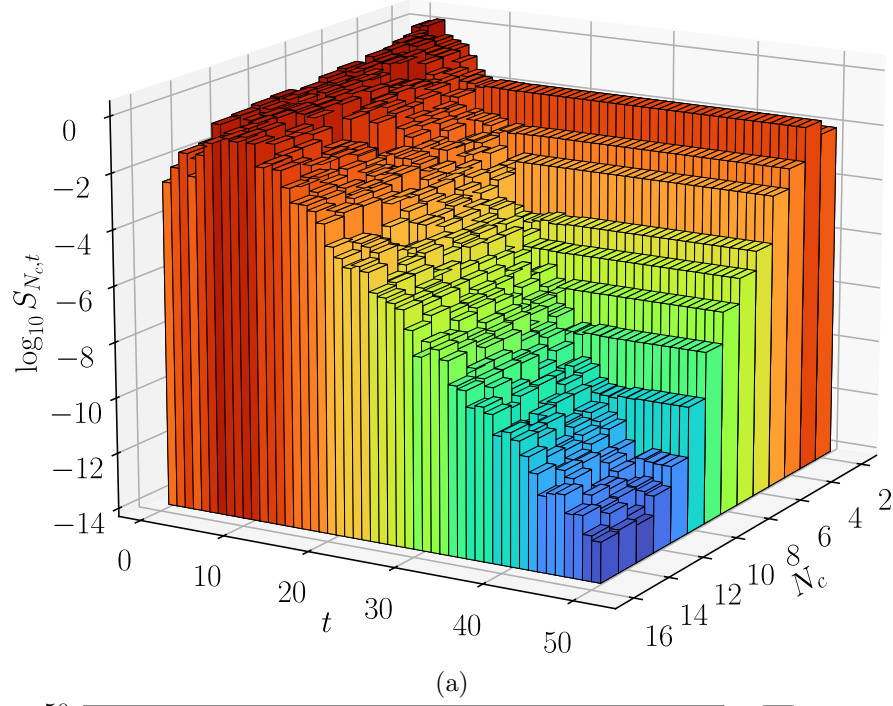


Figure 3.14: 3D bar plot (a) and density plot (b) of the base 10 logarithm of the convergence estimator $S_{N_c, t}(\sigma_z)$ in the case of the open quantum Rabi model for $g = \omega = \omega_0 = \kappa/2 = 1.0$, with the number of qubits t in the eigenvalue register varying from 1 to 50 and the Markovian embedding cutoff parameter N_c varying from 2 to 15. The color scale is the same for both plots. t_0 is chosen according to the plot in Fig. 3.13.

However, a few points should be emphasized: firstly, the results of this comparison are inherent

to our choice of parameters, as we will see later on. Secondly, the difference is slightly erratic, and this irregularity can sometimes lead to a negative difference, i.e. a disadvantage, for some precision thresholds and other choices of parameters. And thirdly, we have no means to predict for which precision thresholds this can happen.

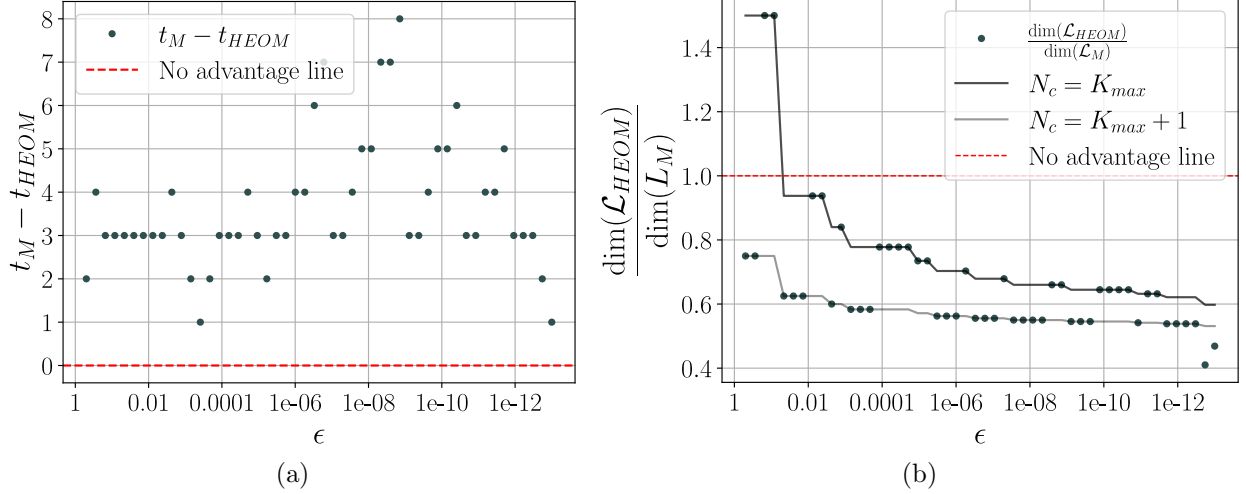


Figure 3.15: (a) Difference between the number of qubits in the eigenvalue register between the RS HEOM adaptation and the Markovian embedding approach and (b) Ratio between the dimensions of the HEOM Liouvillian and the Markovian embedding Liouvillian, to reach a precision threshold ϵ in logarithmic scale in the case of the open quantum Rabi model for $g = \omega = \omega_0 = \kappa/2 = 1.0$. The dimensions of the Liouvillians are not increased to pure powers of 2 in order to show the black/gray lines which are respectively the projected ratios where we replaced K_{max} by N_c and $N_c - 1$ in the dimension computations.

In order to evaluate the difference in the number of qubits in the eigenvalue register for multiple choices of parameters, we will take the average for different precision thresholds so that we only take into account the global tendency. This approach allows us to have an objective criterion for comparing both methods with multiple combinations of parameters. The full plot for the convergence parameter space analysis is shown in Fig. 3.16. There are multiple instances where neither method managed to converge and parameter sets for which only the Markovian embedding approach converged. These instabilities seem to occur around critical coupling values and when the system is under strong non-Markovianity as Fig. 3.17 indicates. The dimensionless ratio $g^2/\kappa g_c$ thus seems to predict rather roughly where these instabilities occur. The critical coupling g_c is defined as

$$g_c = \frac{1}{2} \sqrt{\omega\omega_0 + \frac{\kappa^2\omega_0}{4\omega}} \quad (3.60)$$

for the one-mode open quantum Rabi model [100]. Overall, this suggests that the Markovian embedding algorithm is more robust than its RS HEOM counterpart¹³. Such differences stem from the fact that it has been proven that HEOM suffers from instability issues under specific regimes [57]. Although these results are slightly unfortunate, this is not necessarily problematic as the RS algorithm is initially expected to perform poorly near critical points [19]. Away from these

¹³The white cells where only the RS HEOM adaptation converged are a bit misleading as the HEOM values barely fit the convergence criterion while the Markovian embedding approach fell short by a very small margin ($\approx 2.3 \times 10^{-6}$). However, if we still aim to compare the number of qubits required for lower precision thresholds, the Markovian embedding algorithm still holds an advantage.

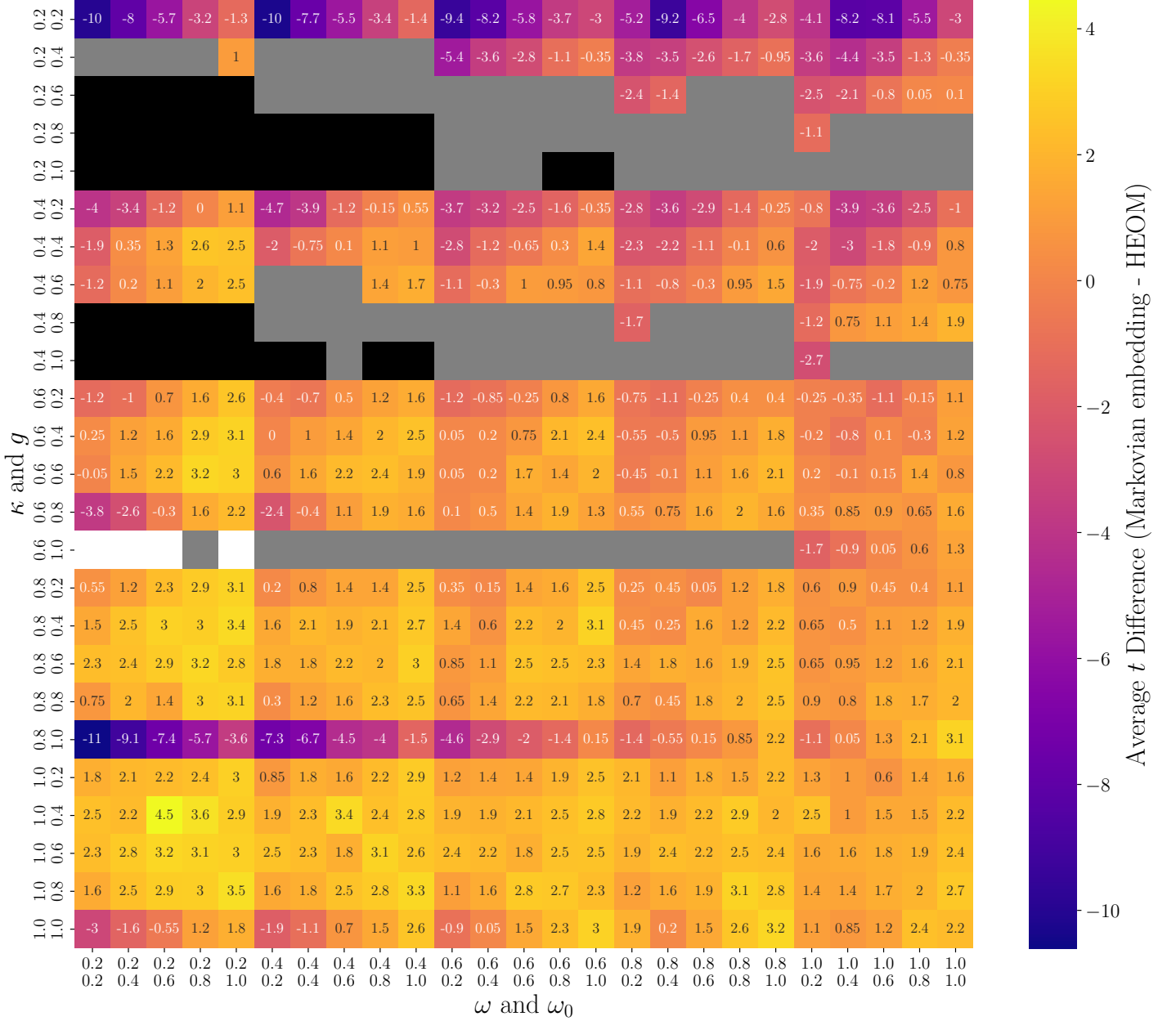
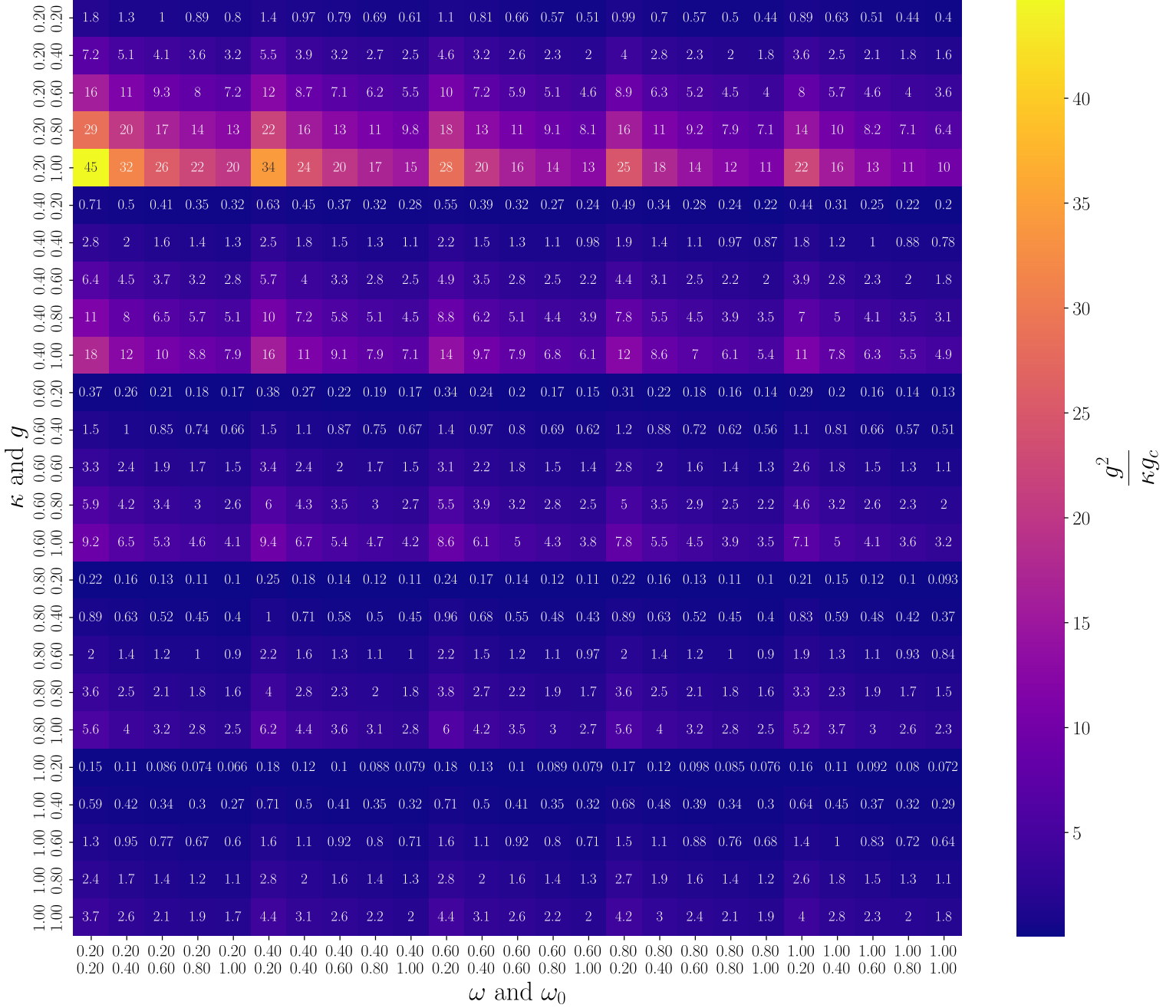


Figure 3.16: Full convergence analysis across the parameter space for the RS HEOM and Markovian embedding methods, shown as a heatmap of the average difference in the required number of qubits in the eigenvalue register across 20 logarithmically spaced precision thresholds from 10^{-1} to 10^{-5} . Black cells indicate parameter sets where neither method converged to the minimal threshold (10^{-6}); grey cells denote cases where only the Markovian embedding approach converged while white cells show situations where only the RS HEOM algorithm converged to a satisfying degree. The convergence criterion is based on the highest precision obtained using the maximal value of K_{max} and N_c , which is 15 in these computations. The plasma colormap quantifies $\langle t_{\text{Markovian}} - t_{\text{HEOM}} \rangle$ where both methods converged.



points, the RS HEOM adaptation performs better than its counterpart, holding a steady advantage of roughly two (and up to 4.5) qubits, which also corresponds to a gate cost 4 times cheaper as mentioned earlier. Note also that a possible outlook of this work could be to overcome the problem of convergence of HEOM in critical regimes by finding a modified version of HEOM through alternative definition of auxiliary states, as done in [101] for non-Markovian quantum trajectory methods.

For classical computations, the advantage of the HEOM method stems from its system size scaling, a feature not observed in our analysis. This is not necessarily surprising as quantum supremacy is partly rooted in quantum computers' superior scaling efficiency for encoding quantum states. In this analysis, we studied the open quantum Rabi model composed of only a single mode. The dimension of the HEOM Liouvillian, at equal cutoffs, goes to half the dimension of the Markovian embedding Liouvillian (cf. Subsection 1.4.2). One additional qubit allows us to encode quantum states twice as large; this, in turn, infers that the advantage of the HEOM approach for one mode is almost nullified. As we increase the size of the systems to pure powers of 2, either both methods use the same number of qubits or HEOM uses one qubit less, as previously shown in Fig 3.15b. However, the advantage of the HEOM approach in terms of size scaling becomes more significant as we increase the number of modes. An example at two modes is the subject of the next section.

3.4.3 RS VS RS HEOM algorithm: two-mode open quantum Rabi model

As we have mentioned earlier, the HEOM method is expected to perform better with multi-mode systems, and the advantage for a single-mode system is hardly noticeable in the case of quantum computing. For two modes, the leading order term in the dimension ratio of the Liouvillian is $1/4! = 1/24$, which means that for large values of K_{max} and N_c , we could expect an advantage of around 4 to 5 qubits for the system register. In order to verify this hypothesis, we will do the same comparison than in the previous sections using the two-mode Rabi model instead.

The Hamiltonian of the two-mode Rabi model is given by

$$H_{\text{Rabi}}^{\text{two-mode}} = \omega_0 \sigma_z + \sum_{j=1,2} \omega_j a_j^\dagger a_j + \sum_{j=1,2} g_j (a_j + a_j^\dagger) \sigma_x, \quad (3.61)$$

where the ω_k is the frequency of the mode k , a_k is the corresponding annihilation operator for the mode ω_k and g_k is the associated coupling parameter between the mode and the two-level system. The associated BCF is given by

$$\alpha(\tau) = \sum_{j=1,2} g_j^2 e^{-\kappa_j |\tau| - i\omega_j \tau}. \quad (3.62)$$

We will follow the same procedure as in the previous sections however, we will only focus on the convergence comparisons between the two methods. Due to the numerical cost of the computations, we have restrained ourselves to a single set of parameters, and a figure similar to 3.16 was not produced for the two-mode Rabi model ¹⁴. In order to have a convergence rate (relative to the

¹⁴Note that the plot at one mode took 57 hours to complete, unparallelized. However the computation time is not the only limiting factor at two modes, we need a lot of RAM memory in order to store the Liouvillian matrix and its eigenvectors. Since we need every eigenvector and every eigenvalue of the matrix, we cannot use sparse methods. Typically, picking $N_c = 8$ for the Markovian embedding cutoff for the two-mode Rabi model, gives a M matrix of size 32768×32768 , assuming 128 bits per element (double precision for a complex number) means that we need 16GB to store it. Furthermore, diagonalizing such a matrix takes a considerable amount of time and parallelizing is not feasible due to the ram consumption unless we have access to a cluster which was not our case in this work.

cutoff parameters) neither too slow nor too fast, we picked the following set of parameters

$$1.0 = \omega_0 = \kappa_1 = \kappa_2/0.5 = \omega_1/0.5 = \omega_2/0.35 = g_1/0.25 = g_2/0.2.$$

The two-mode HEOM Liouvillian was implemented by using the library available in QuTiP [50]. Contrary to the one-mode convergence analysis, we chose to estimate the convergence, not by comparing with the next increment of the parameters but by comparing the computations, to a classically obtained estimation of $\langle \sigma_z \rangle_{ss}$. The reason being that we have more points to compare using this approach¹⁵. The classical method employed consisted in propagating the dynamics for an extended period of time, typically $t = 5000\omega_0^{-1}$, to reach the steady state and using high cutoff parameters, $K_{max} = 20$ and $N_c = 12$. This yielded the two values

$$\langle \sigma_z \rangle_M = -0.35683403225378906853, \quad (3.63a)$$

$$\langle \sigma_z \rangle_{HEOM} = -0.35683403225375598389, \quad (3.63b)$$

which are in accordance to 13 decimal places. Alternatively, we could have used sparse matrix methods to obtain the eigenvector corresponding to the eigenvalue 0. However, due to the low relaxation time of the system¹⁶, it is faster to propagate the dynamics. Computing the dynamics up to $t = 5000\omega_0^{-1}$ yields results accurate to 17 decimal places compared with the diagonalization at lower cutoff values. The estimator is thus defined as

$$S_{X,t}(\sigma_z) = |\langle \sigma_z(X, t) \rangle - \langle \sigma_z \rangle_{CC}|, \quad (3.64)$$

where X designates both K_{max} or N_c depending on the context and CC refers to Classically Computed. The convergence plots for both methods are illustrated in Fig. 3.18

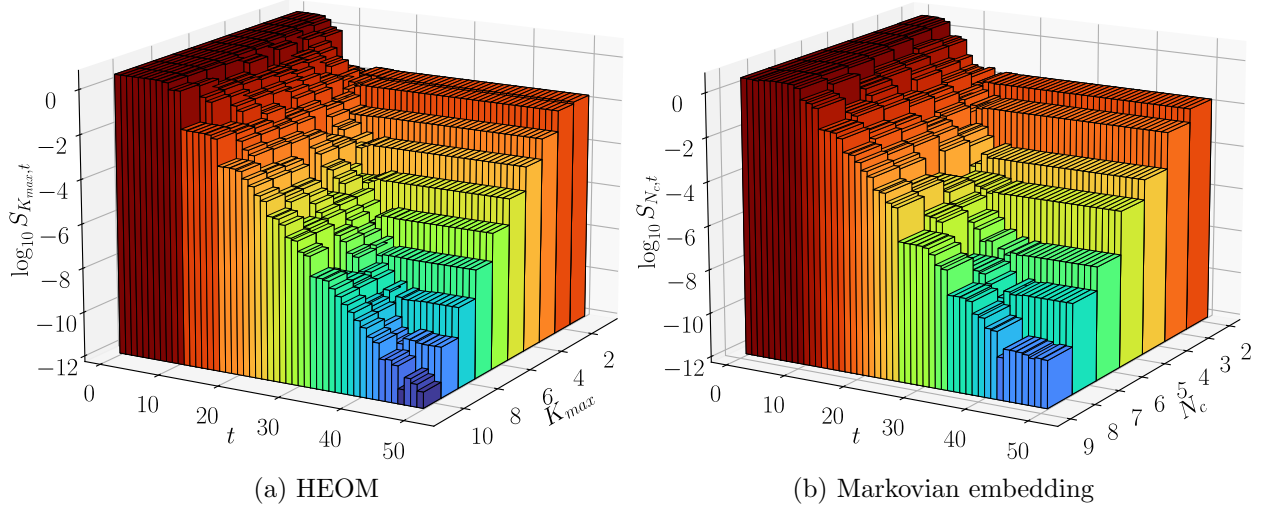


Figure 3.18: 3D bar plot for HEOM (a) and the Markovian embedding approach (b) of the base 10 logarithm of the convergence estimator (3.64), in the case of the two-mode open quantum Rabi model, with the number of qubits t in the eigenvalue register varying from 1 to 50, the HEOM cutoff parameter varying from 1 to 10 and the Markovian embedding cutoff parameter ranging from 2 to 8. The color and height scale is the same for both plots. t_0 is chosen by computing the largest eigenvalue of the matrix, multiplying by 1.05 and taking the inverse.

¹⁵The maximum value for N_c we could obtain was 8. Using the previous approach, this value would have served to estimate the convergence relative to $N_c = 7$. Comparing to another value obtained classically, allows us to use $N_c = 8$ in the convergence analysis.

¹⁶The relaxation time is defined as the typical timescale over which a system returns to equilibrium after a perturbation.

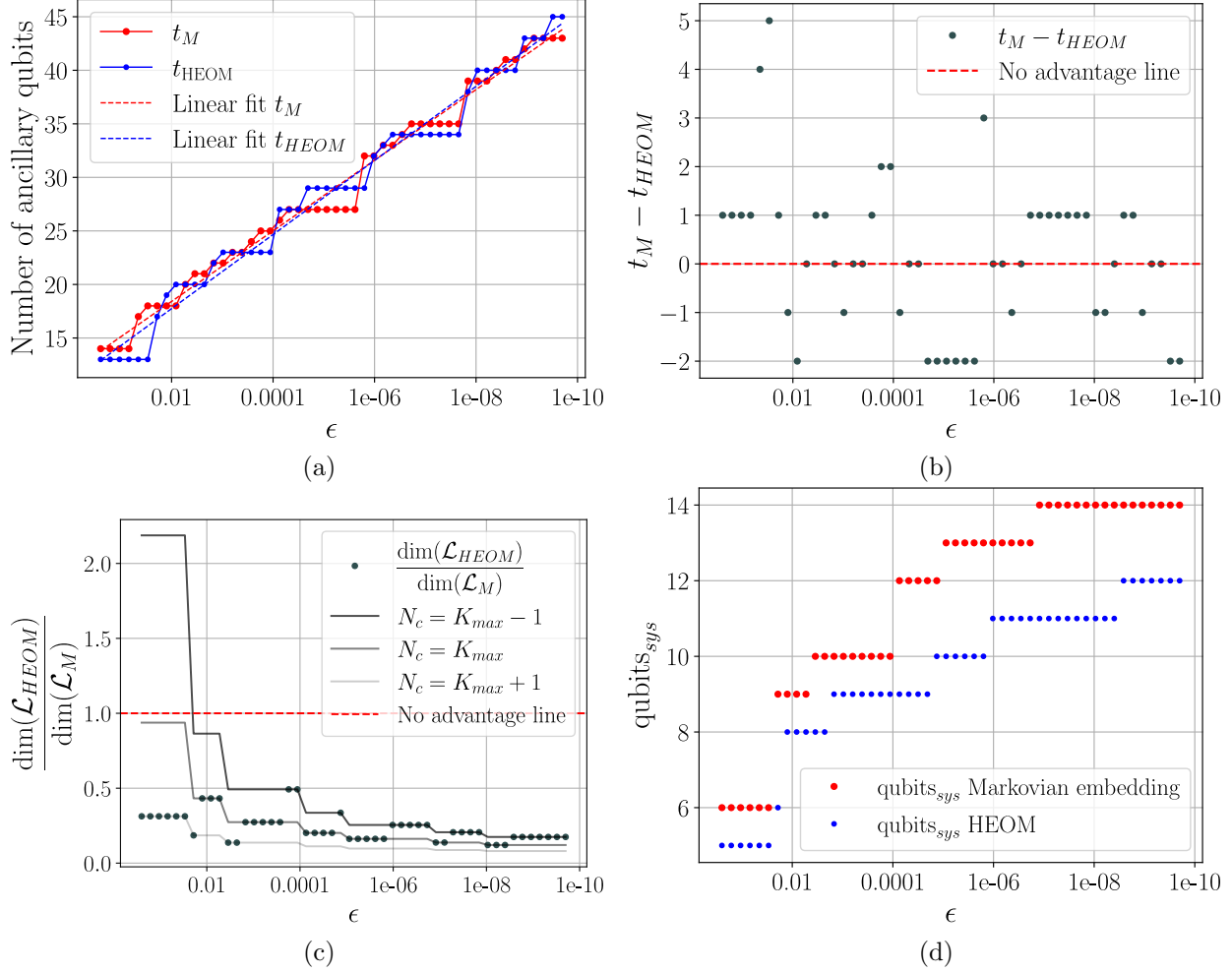


Figure 3.19: (a) Plot of the number of ancillary qubits required to reach a precision threshold ϵ ranging from 0.25 to 2×10^{-10} for both methods with (b) being the plot of the difference between the Markovian embedding and HEOM numbers of ancillary qubits. The linear fits are $t_M = -3.297 \log_{10}(\epsilon) + 11.817$ and $t_{HEOM} = -3.459 \log_{10}(\epsilon) + 10.823$. (c) Ratio between $\dim(\mathcal{L}_{HEOM})$ and $\dim(\mathcal{L}_M)$ as a function of the precision threshold ϵ . The dimensions of the Liouvillians are not increased to pure powers of 2 in order to show the black/gray lines which are respectively the projected ratios where we replaced K_{max} by $N_c + 1$, N_c and $N_c - 1$ in the dimension computations. The Liouvillian dimensions increased to pure powers of 2 are implicitly shown in (d) which is the plot of the number of qubits representing the complete system for both methods as a function of the precision threshold ϵ . For all computations, we used $1.0 = \omega_0 = \kappa_1 = \kappa_2/0.5 = \omega_1/0.5 = \omega_2/0.35 = g_1/0.25 = g_2/0.2$, which is justified by the fact that this set of parameters allows a rate of convergence that is neither too slow nor too fast for our numerical resources.

The results seem more irregular due to the comparison with $\langle \sigma_z \rangle_{CC}$ instead of the next parameter iteration. We can then proceed as in the previous sections and reverse the plots in Fig. 3.18 to obtain the minimal parameter values as a function of the precision threshold. All the plots of the convergence analysis are illustrated in Fig. 3.19. The first feature to notice is that the advantage regarding the number of qubits in the eigenvalue register has disappeared. Such a result is not necessarily surprising owing to the fact that the choice of parameters has a considerable impact on

whether a method holds an advantage over the other in this regard. As such, it seems likely that this specific choice of parameters translates to no advantage in terms of qubits in the eigenvalue register and that parameters where HEOM holds an advantage over the Markovian embedding approach exist. Nonetheless, such a conclusion is overly stretched in the case of the two-mode open quantum Rabi model and deserves further research since a plot similar to Fig. 3.16 for two modes was not conducted.

The particular element to notice is that the advantage regarding the number of qubits allocated to the system, i.e. the eigenvector register, has grown compared to the one-mode open quantum Rabi model. Furthermore, the gray lines in Fig. 3.19c seem to indicate that K_{max} and N_c scale indeed rather similarly. Although there seems to have been a shift of 1 between the respective values of K_{max} and N_c compared to the one-mode Rabi model, such a difference is inconsequential: we have shown in the first chapter of this thesis (cf. Sec 1.4.2) that for neither method to be able to provide an advantage over the other, we would need that $K_{max} \approx (2M/e)N_c$ which is evidently not the case here. This in turn suggests that our initial intuition that the advantage for the HEOM method would get larger for multi-mode systems seems correct. Moreover, although the difference for the number of qubits for the system is around 2 to 3, it is within expectations since we have only gone up to $N_c = 8$ which only leads to a ratio of around 1/8. As such, increasing the Liouvillians to pure powers of two would either yield 1/4, 1/8 or 1/16. However, the latter does not occur as increasing to pure powers of two tends to favor the Markovian embedding approach as discussed in the previous sections ¹⁷.

3.5 Summary

The first section of this chapter was dedicated to the algorithm of Ramusat and Savona (RS) [19], a quantum algorithm for estimating the steady states of Markovian open quantum systems. In the second section, we applied this algorithm to the simple spin model from Sec. 1.3 through quantum circuit simulations. We then demonstrated how to bypass simulations by diagonalizing a Liouvillian-derived matrix, that enabled us to perform advanced convergence analyses.

In the latter part of the chapter, we showed how the RS algorithm could be adapted to non-Markovian systems using the HEOM Liouvillian with a few key adjustments, which is the original contribution of this master's thesis. We applied this modified algorithm to the open quantum Rabi model (Sec. 1.5) and compared it against the Markovian embedding approach. Finally, we demonstrated our algorithm's non-trivial advantage for multi-mode systems through the analysis of a two-mode open quantum Rabi model.

¹⁷To have an advantage of 4 qubits it is expected to be at least at $N_c = 10$.

Conclusion

The main objective of this master’s thesis was to develop a quantum algorithm for finding the steady state of non-Markovian open quantum systems. In the first chapter, we introduced the reader to the field of open quantum systems. We presented the density operator formalism, showed why it is necessary, and proved some of its key properties. We then explained how to compute the dynamics of open quantum systems using the GKSL master equation that relies on the Born and the Markov approximations. Afterwards, we described how the Choi–Jamiołkowski isomorphism allows us to map density operators to ket vectors and CPTP maps to operators in order to implement them on quantum computers. We also outlined how to compute the dynamics of non-Markovian systems, first by using Markovian embedding techniques and second by using the HEOM method, and clarified why the latter is generally advantageous. Furthermore, we provided two examples of open quantum systems, one Markovian and the other non-Markovian, to equip the reader with a stronger intuition on the subject.

The second chapter aimed at explaining the general formalism of quantum information and computation by introducing the concepts of computational basis, quantum circuits, and quantum gates. We then described the quantum Fourier transform and quantum phase estimation algorithms as they are part of the key subroutines of this thesis algorithm. Moreover, we showed how the oracles for the controlled- U gates can be implemented using the Suzuki-Trotter expansion.

In the third chapter, we extensively presented the algorithm of Ramusat and Savona to find the steady states of Markovian open quantum systems. As an original contribution, we demonstrated how this algorithm could be generalized to non-Markovian systems using HEOM and a few adaptations; notably to the \mathcal{P} gate, the choice of t_0 , and the measurement operator. Furthermore, we used the same open quantum systems introduced in the first chapter to apply both algorithms. We first implemented the algorithm by employing circuit simulations in Qiskit using the Aer Simulator backend, and then showed that the outcome of the algorithm could be accurately classically predicted at the cost of needing to entirely diagonalize the M matrix of the corresponding Liouvillian. In the case of the one-mode open quantum Rabi model, we have established that away from critical points and where the systems do not exhibit strong non-Markovianity, our adaptation performs with a steady advantage regarding the number of qubits in the eigenvalue register compared to the original algorithm used in the case of a Markovian embedding approach. Additionally, we confirmed in the case of the two-mode open quantum Rabi model that our algorithm exhibits a clear advantage regarding the number of qubits for the system, a feature not observed at one mode. These observations seem to substantiate the initial assumption that our HEOM adaptation should perform with a non-trivial advantage for the number of qubits in the eigenvector register. A manuscript gathering all these results is currently under preparation.

Moving forward, one could use the HEOM method to adapt other algorithms to non-Markovian

systems. In particular, algorithms that use quantum annealing instead of digital quantum computers may provide interesting challenges. An additional and promising prospect could be to develop a quantum algorithm to compute the dynamics of non-Markovian open quantum systems that takes full advantage of the noise present in the hardware to implement the decoherence effects. While this idea was already proposed by Seth Lloyd in 1996 [102], only few people have actively tried to do so [23], [103]. Furthermore, one could also use measurement operations to simulate decoherence [104]. Combining these two approaches also offers considerable advantages; for example, removing the non-unitary parts in the Liouvillian eliminates the need to embed it into a larger Hermitian matrix. The measurements used to implement the decoherence would inevitably yield information about the system, and lastly, such an algorithm would be utterly noise-resilient.

Bibliography

- [1] D. J. Griffiths, *Introduction to Quantum Mechanics*, 2nd ed. Cambridge, England: Cambridge University Press, Aug. 2016.
- [2] H.-P. Breuer and F. Petruccione, *The theory of open quantum systems*. London, England: Oxford University Press, Jan. 2007.
- [3] J. Von Neumann, *Mathematical foundations of quantum mechanics* (Princeton Landmarks in Mathematics and Physics), en. Princeton, NJ: Princeton University Press, Oct. 1996.
- [4] J.-M. Raimond and V. Rivasseau, Eds., *Quantum decoherence* (Progress in Mathematical Physics), en, 2007th ed. Basel, Switzerland: Springer, Dec. 2006.
- [5] V. Gorini, A. Kossakowski, and E. C. G. Sudarshan, “Completely positive dynamical semigroups of N -level systems,” en, *J. Math. Phys.*, vol. 17, no. 5, pp. 821–825, May 1976.
- [6] G. Lindblad, “On the generators of quantum dynamical semigroups,” en, *Commun. Math. Phys.*, vol. 48, no. 2, pp. 119–130, Jun. 1976.
- [7] G. A. L. White, C. D. Hill, F. A. Pollock, L. C. L. Hollenberg, and K. Modi, “Demonstration of non-markovian process characterisation and control on a quantum processor,” en, *Nat. Commun.*, vol. 11, no. 1, p. 6301, Dec. 2020.
- [8] C. Ayaz, *Non-Markovian modeling of protein folding*, 2021.
- [9] X. Li, “Markovian embedding procedures for Non-Markovian stochastic schrödinger equations,” Apr. 2020. arXiv: 2005.00103 [physics.comp-ph].
- [10] H. I. Nurdin, “Markovian embeddings of non-markovian quantum systems: Coupled stochastic and quantum master equations for non-markovian quantum systems,” Nov. 2023. arXiv: 2312.00134 [quant-ph].
- [11] Y. Tanimura and R. Kubo, “Time evolution of a quantum system in contact with a nearly gaussian-markoffian noise bath,” en, *J. Phys. Soc. Jpn.*, vol. 58, no. 1, pp. 101–114, Jan. 1989.
- [12] P. Benioff, “The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines,” en, *J. Stat. Phys.*, vol. 22, no. 5, pp. 563–591, May 1980.
- [13] R. P. Feynman, “Simulating physics with computers,” en, *Int. J. Theor. Phys.*, vol. 21, no. 6-7, pp. 467–488, Jun. 1982.
- [14] Y. I. Manin, *Mathematics as metaphor* (Collected Works). Providence, RI: American Mathematical Society, Nov. 2007.
- [15] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge, England: Cambridge University Press, Dec. 2010.

- [16] Z. Hu, R. Xia, and S. Kais, “A quantum algorithm for evolving open quantum dynamics on quantum computing devices,” en, *Sci. Rep.*, vol. 10, no. 1, p. 3301, Feb. 2020.
- [17] N. Yoshioka, Y. O. Nakagawa, K. Mitarai, and K. Fujii, “Variational quantum algorithm for nonequilibrium steady states,” en, *Phys. Rev. Res.*, vol. 2, no. 4, Nov. 2020.
- [18] H.-Y. Liu, T.-P. Sun, Y.-C. Wu, and G.-P. Guo, “Variational quantum algorithms for steady states of open quantum systems,” Jan. 2020. arXiv: 2001.02552 [quant-ph].
- [19] N. Ramusat and V. Savona, “A quantum algorithm for the direct estimation of the steady state of open quantum systems,” en, *Quantum*, vol. 5, no. 399, p. 399, Feb. 2021.
- [20] A. W. Schlimgen, K. Head-Marsden, L. M. Sager, P. Narang, and D. A. Mazziotti, “Quantum simulation of open quantum systems using a unitary decomposition of operators,” en, *Phys. Rev. Lett.*, vol. 127, no. 27, p. 270 503, Dec. 2021.
- [21] H. Kamakari, S.-N. Sun, M. Motta, and A. J. Minnich, “Digital quantum simulation of open quantum systems using quantum imaginary time evolution,” Apr. 2021. arXiv: 2104.07823 [quant-ph].
- [22] Z. Hu, K. Head-Marsden, D. A. Mazziotti, P. Narang, and S. Kais, “A general quantum algorithm for open quantum dynamics demonstrated with the Fenna-Matthews-Olson complex,” Jan. 2021. arXiv: 2101.05287 [quant-ph].
- [23] J. Leppäkangas, N. Vogt, K. R. Fratus, *et al.*, “A quantum algorithm for solving open system dynamics on quantum computers using noise,” Oct. 2022. arXiv: 2210.12138 [quant-ph].
- [24] J. W. Z. Lau, K. H. Lim, K. Bharti, L.-C. Kwek, and S. Vinjanampathy, “Convex optimization for nonequilibrium steady states on a hybrid quantum processor,” en, *Phys. Rev. Lett.*, vol. 130, no. 24, p. 240 601, Jun. 2023.
- [25] X. Li and C. Wang, “Succinct description and efficient simulation of Non-Markovian open quantum systems,” Nov. 2021. arXiv: 2111.03240 [quant-ph].
- [26] M. Cygorek and E. M. Gauger, “ACE: A general-purpose non-markovian open quantum systems simulation toolkit based on process tensors,” en, *J. Chem. Phys.*, vol. 161, no. 7, Aug. 2024.
- [27] M. Cygorek, J. Keeling, B. W. Lovett, and E. M. Gauger, “Sublinear scaling in non-markovian open quantum systems simulations,” en, *Phys. Rev. X*, vol. 14, no. 1, Feb. 2024.
- [28] X. Li, S.-X. Lyu, Y. Wang, R.-X. Xu, X. Zheng, and Y. Yan, “Toward quantum simulation of non-markovian open quantum dynamics: A universal and compact theory,” en, *Phys. Rev. A (Coll. Park.)*, vol. 110, no. 3, Sep. 2024.
- [29] A. Seneviratne, P. L. Walters, and F. Wang, “Exact non-markovian quantum dynamics on the NISQ device using kraus operators,” en, *ACS Omega*, vol. 9, no. 8, pp. 9666–9675, Feb. 2024.
- [30] A. Seneviratne, P. L. Walters, and F. Wang, “Quantum algorithm for the simulation of non-markovian quantum dynamics using Feynman-Vernon influence functional,” en, *J. Chem. Phys.*, vol. 162, no. 19, May 2025.
- [31] P. L. Walters and F. Wang, “Path integral quantum algorithm for simulating non-markovian quantum dynamics in open quantum systems,” en, *Phys. Rev. Res.*, vol. 6, no. 1, Feb. 2024.
- [32] X. Dan, E. Geva, and V. S. Batista, “Simulating non-markovian quantum dynamics on NISQ computers using the hierarchical equations of motion,” en, *J. Chem. Theory Comput.*, vol. 21, no. 4, pp. 1530–1546, Feb. 2025.

- [33] E. Bairey, C. Guo, D. Poletti, N. H. Lindner, and I. Arad, “Learning the dynamics of open quantum systems from their steady states,” *New J. Phys.*, vol. 22, no. 3, p. 032001, Mar. 2020.
- [34] J. Guo, O. Hart, C.-F. Chen, A. J. Friedman, and A. Lucas, “Designing open quantum systems with known steady states: Davies generators and beyond,” en, *Quantum*, vol. 9, no. 1612, p. 1612, Jan. 2025.
- [35] M. Orszag, *Quantum optics*, en, 3rd ed. Cham, Switzerland: Springer International Publishing, Apr. 2016.
- [36] M. O. Scully and M. S. Zubairy, *Quantum Optics*. Cambridge, England: Cambridge University Press, Jun. 2012.
- [37] C. Cohen-Tannoudji, B. Diu, and F. Laloë, *Quantum mechanics, volume 1*, en, 2nd ed. Berlin, Germany: Blackwell Verlag, Oct. 2019.
- [38] M. Le Bellac, *Quantum Physics*, en. Cambridge, England: Cambridge University Press, Mar. 2006.
- [39] A. Nordsieck, W. E. Lamb Jr, and G. E. Uhlenbeck, “On the theory of cosmic-ray showers I the furry model and the fluctuation problem,” en, *Physica*, vol. 7, no. 4, pp. 344–360, Apr. 1940.
- [40] D. Manzano, “A short introduction to the lindblad master equation,” Jun. 2019. arXiv: 1906.04478 [quant-ph].
- [41] M. E. Peskin and D. V. Schroeder, *An introduction to quantum field theory*, en. London, England: CRC Press, Sep. 2019.
- [42] M.-D. Choi, “Completely positive linear maps on complex matrices,” en, *Linear Algebra Appl.*, vol. 10, no. 3, pp. 285–290, Jun. 1975.
- [43] A. Jamiolkowski, “Linear transformations which preserve trace and positive semidefiniteness of operators,” en, *Rep. Math. Phys.*, vol. 3, no. 4, pp. 275–278, Dec. 1972.
- [44] B. Debecker, J. Martin, and F. Damanet, “Controlling matter phases beyond markov,” en, *Phys. Rev. Lett.*, vol. 133, no. 14, p. 140403, Oct. 2024.
- [45] A. Lemmer, C. Cormick, D. Tamascelli, T. Schaetz, S. F. Huelga, and M. B. Plenio, “Simulating spin-boson models with trapped ions,” Apr. 2017. arXiv: 1704.00629 [quant-ph].
- [46] P. M. Q. Cruz, G. Catarina, R. Gautier, and J. Fernández-Rossier, “Optimizing quantum phase estimation for the simulation of hamiltonian eigenstates,” *Quantum Sci. Technol.*, vol. 5, no. 4, p. 044005, Aug. 2020.
- [47] C. Meier and D. J. Tannor, “Non-Markovian evolution of the density operator in the presence of strong laser fields,” en, *J. Chem. Phys.*, vol. 111, no. 8, pp. 3365–3376, Aug. 1999.
- [48] G. Ritschel and A. Eisfeld, “Analytic representations of bath correlation functions for ohmic and superohmic spectral densities using simple poles,” en, *J. Chem. Phys.*, vol. 141, no. 9, p. 094101, Sep. 2014.
- [49] R. Hartmann, M. Werther, F. Grossmann, and W. T. Strunz, “Exact open quantum system dynamics: Optimal frequency vs time representation of bath correlations,” en, *J. Chem. Phys.*, vol. 150, no. 23, p. 234105, Jun. 2019.
- [50] N. Lambert, T. Raheja, S. Cross, *et al.*, “QuTiP-BoFiN: A bosonic and fermionic numerical hierarchical-equations-of-motion library with applications in light-harvesting, quantum control, and single-molecule electronics,” en, *Phys. Rev. Res.*, vol. 5, no. 1, Mar. 2023.

- [51] Z. Tang, X. Ouyang, Z. Gong, H. Wang, and J. Wu, “Extended hierarchy equation of motion for the spin-boson model,” en, *J. Chem. Phys.*, vol. 143, no. 22, p. 224 112, Dec. 2015.
- [52] Y. Tanimura, “Nonperturbative expansion method for a quantum system coupled to a harmonic-oscillator bath,” en, *Phys. Rev. A*, vol. 41, no. 12, pp. 6676–6687, Jun. 1990.
- [53] Y. Tanimura and P. G. Wolynes, “Quantum and classical Fokker-Planck equations for a Gaussian-Markovian noise bath,” en, *Phys. Rev. A*, vol. 43, no. 8, pp. 4131–4142, Apr. 1991.
- [54] Y. Tanimura, “Stochastic liouville, langevin, Fokker-Planck, and master equation approaches to quantum dissipative systems,” en, *J. Phys. Soc. Jpn.*, vol. 75, no. 8, p. 082 001, Aug. 2006.
- [55] Y. Tanimura, “Reduced hierarchical equations of motion in real and imaginary time: Correlated initial states and thermodynamic quantities,” en, *J. Chem. Phys.*, vol. 141, no. 4, p. 044 114, Jul. 2014.
- [56] B. Debecker, J. Martin, and F. Damanet, “Spectral theory of non-markovian dissipative phase transitions,” en, *Phys. Rev. A (Coll. Park.)*, vol. 110, no. 4, Oct. 2024.
- [57] M. Krug and J. Stockburger, “On stability issues of the HEOM method,” en, *Eur. Phys. J. Spec. Top.*, vol. 232, no. 20-22, pp. 3219–3226, Dec. 2023.
- [58] V. Link, K. Müller, R. G. Lena, *et al.*, “Non-markovian quantum dynamics in strongly coupled multimode cavities conditioned on continuous measurement,” en, *PRX quantum*, vol. 3, no. 2, Jun. 2022.
- [59] M. Malekakhlagh and A. W. Rodriguez, “Quantum rabi model with two-photon relaxation,” en, *Phys. Rev. Lett.*, vol. 122, no. 4, Jan. 2019.
- [60] O. Koska, M. Baboulin, and A. Gazda, “A tree-approach pauli decomposition algorithm with application to quantum computing,” Mar. 2024. arXiv: 2403.11644 [quant-ph].
- [61] M. Dion, T. Belabbas, and N. Bastien, “Efficiently manipulating pauli strings with PauliArray,” May 2024. arXiv: 2405.19287 [quant-ph].
- [62] B. Reggio, N. Butt, A. Lytle, and P. Draper, “Fast partitioning of pauli strings into commuting families for optimal expectation value measurements of dense operators,” May 2023. arXiv: 2305.11847 [quant-ph].
- [63] P. W. Shor, “Algorithms for quantum computation: Discrete logarithms and factoring,” in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, Santa Fe, NM, USA: IEEE Comput. Soc. Press, 2002.
- [64] D. Coppersmith, “An approximate fourier transform useful in quantum factoring,” Jan. 2002. arXiv: quant-ph/0201067 [quant-ph].
- [65] J. W. Cooley and J. W. Tukey, “An algorithm for the machine calculation of complex fourier series,” en, *Math. Comput.*, vol. 19, no. 90, pp. 297–301, 1965.
- [66] *Ibm quantum processors*, Accessed: January 21, 2025. [Online]. Available: <https://quantum-computing.ibm.com/>.
- [67] A. G. Fowler, S. J. Devitt, and L. C. L. Hollenberg, “Implementation of shor’s algorithm on a linear nearest neighbour qubit array,” Feb. 2004. arXiv: quant-ph/0402196 [quant-ph].
- [68] D. Maslov, “Linear depth stabilizer and quantum fourier transformation circuits with no auxiliary qubits in finite-neighbor quantum architectures,” en, *Phys. Rev. A*, vol. 76, no. 5, Nov. 2007.

- [69] A. Bhattacharjee, C. Bandyopadhyay, R. Wille, R. Drechsler, and H. Rahaman, “Improved look-ahead approaches for nearest neighbor synthesis of 1D quantum circuits,” in *2019 32nd International Conference on VLSI Design and 2019 18th International Conference on Embedded Systems (VLSID)*, Delhi, NCR, India: IEEE, Jan. 2019.
- [70] A. Holmes, S. Johri, G. G. Guerreschi, J. S. Clarke, and A. Y. Matsuura, “Impact of qubit connectivity on quantum algorithm performance,” *Quantum Sci. Technol.*, vol. 5, no. 2, p. 025 009, Mar. 2020.
- [71] B. Park and D. Ahn, “Reducing CNOT count in quantum fourier transform for the linear nearest-neighbor architecture,” in *Sci. Rep.*, vol. 13, no. 1, p. 8638, May 2023.
- [72] B. Klaver, S. Rombouts, M. Fellner, *et al.*, “SWAP-less implementation of quantum algorithms,” Aug. 2024. arXiv: 2408.10907 [quant-ph].
- [73] R. Cleve and J. Watrous, “Fast parallel circuits for the quantum fourier transform,” Jun. 2000. arXiv: quant-ph/0006004 [quant-ph].
- [74] Y. Nam, Y. Su, and D. Maslov, “Approximate quantum fourier transform with $O(n \log(n))$ T gates,” Mar. 2018. arXiv: 1803.04933 [quant-ph].
- [75] Y. Takahashi, N. Kunihiro, and K. Ohta, “The quantum fourier transform on a linear nearest neighbor architecture,” *Quantum Inf. Comput.*, vol. 7, no. 4, pp. 383–391, May 2007.
- [76] A. Y. Kitaev, “Quantum measurements and the abelian stabilizer problem,” Nov. 1995. arXiv: quant-ph/9511026 [quant-ph].
- [77] H. Ni, H. Li, and L. Ying, “On low-depth algorithms for quantum phase estimation,” Feb. 2023. arXiv: 2302.02454 [quant-ph].
- [78] H. Mohammadbagherpoor, Y.-H. Oh, P. Dreher, A. Singh, X. Yu, and A. J. Rindos, “An improved implementation approach for quantum phase estimation on quantum computers,” Oct. 2019. arXiv: 1910.11696 [quant-ph].
- [79] J. Li, “An iterative method to improve the precision of quantum phase estimation algorithm,” Feb. 2024. arXiv: 2402.14191 [quant-ph].
- [80] A. W. Harrow, A. Hassidim, and S. Lloyd, “Quantum algorithm for solving linear systems of equations,” Nov. 2008. arXiv: 0811.3171 [quant-ph].
- [81] S. Lloyd, M. Mohseni, and P. Rebentrost, “Quantum algorithms for supervised and unsupervised machine learning,” Jul. 2013. arXiv: 1307.0411 [quant-ph].
- [82] P. Rebentrost, M. Mohseni, and S. Lloyd, “Quantum support vector machine for big data classification,” Jul. 2013. arXiv: 1307.0471 [quant-ph].
- [83] Z. Zhao, A. Pozas-Kerstjens, P. Rebentrost, and P. Wittek, “Bayesian deep learning on a quantum computer,” Jun. 2018. arXiv: 1806.11463 [quant-ph].
- [84] A. Montanaro and S. Pallister, “Quantum algorithms and the finite element method,” Dec. 2015. arXiv: 1512.05903 [quant-ph].
- [85] N. Hatano and M. Suzuki, “Finding exponential product formulas of higher orders,” in *Quantum Annealing and Other Optimization Methods*, ser. Lecture notes in physics, Berlin, Heidelberg: Springer Berlin Heidelberg, Nov. 2005, pp. 37–68.
- [86] A. Bonfiglioli, *Topics in Noncommutative Algebra* (Lecture notes in mathematics), en, 2012th ed. New York, NY: Springer, Jan. 2012.

- [87] A. Z. Goldberg, A. M. Steinberg, and K. Heshami, “Beyond transcoherent states: Field states for effecting optimal coherent rotations on single or multiple qubits,” en, *Quantum*, vol. 7, no. 963, p. 963, Mar. 2023.
- [88] F. Casas, A. Escorihuela-Tomàs, and P. A. Moreno Casares, “Approximating exponentials of commutators by optimized product formulas,” en, *Quantum Inf. Process.*, vol. 24, no. 2, Jan. 2025.
- [89] G. Aleksandrowicz, T. Alexander, P. Barkoutsos, *et al.*, *Qiskit: An open-source framework for quantum computing*, 2019.
- [90] E. Knill, G. Ortiz, and R. D. Somma, “Optimal quantum measurements of expectation values of observables,” en, *Phys. Rev. A*, vol. 75, no. 1, Jan. 2007.
- [91] A. Roggero and A. Baroni, “Short-depth circuits for efficient expectation value estimation,” May 2019. arXiv: 1905.08383 [quant-ph].
- [92] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, “A theory of trotter error,” Dec. 2019. arXiv: 1912.08854 [quant-ph].
- [93] X. Li, “Some error analysis for the quantum phase estimation algorithms,” Nov. 2021. arXiv: 2111.10430 [quant-ph].
- [94] M. Žnidarič, “Relaxation times of dissipative many-body quantum systems,” en, *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, vol. 92, no. 4, p. 042143, Oct. 2015.
- [95] V. V. Shende, S. S. Bullock, and I. L. Markov, “Synthesis of quantum logic circuits,” Jun. 2004. arXiv: quant-ph/0406176 [quant-ph].
- [96] C. Lanczos, “An iteration method for the solution of the eigenvalue problem of linear differential and integral operators,” *J. Res. Natl. Bur. Stand. (1934)*, vol. 45, no. 4, p. 255, Oct. 1950.
- [97] I. U. Ojalvo and M. Newman, “Vibration modes of large structures by an automatic matrix-reduction method,” en, *AIAA J.*, vol. 8, no. 7, pp. 1234–1239, Jul. 1970.
- [98] C. C. Paige, “The computation of eigenvalues and eigenvectors of very large sparse matrices,” Ph.D. dissertation, London University Institute of Computer Science, 1971.
- [99] W. E. Arnoldi, “The principle of minimized iterations in the solution of the matrix eigenvalue problem,” en, *Quart. Appl. Math.*, vol. 9, no. 1, pp. 17–29, 1951.
- [100] M.-J. Hwang, P. Rabl, and M. B. Plenio, “Dissipative phase transition in the open quantum rabi model,” *Phys. Rev. A (Coll. Park.)*, vol. 97, no. 1, Jan. 2018.
- [101] K. Müller and W. T. Strunz, *Nuhops: A quantum trajectory method for highly excited environments in non-markovian open quantum dynamics*, 2025. arXiv: 2503.03368 [quant-ph]. [Online]. Available: <https://arxiv.org/abs/2503.03368>.
- [102] S. Lloyd, “Universal quantum simulators,” en, *Science*, vol. 273, no. 5278, pp. 1073–1078, Aug. 1996.
- [103] C. Cao and X. Wang, “Noise-assisted quantum autoencoder,” en, *Phys. Rev. Appl.*, vol. 15, no. 5, May 2021.
- [104] J. P. T. Stenger, G. Bazargan, N. T. Bronn, and D. Gunlycke, “Method for simulating open-system dynamics using mid-circuit measurements on a quantum computer,” 2025. eprint: 2504.15187 (quant-ph).