Bachelor Thesis in Physics

# Quantum Error Correction with Molecular Spin Qudits

Trabajo de Fin de Grado en Física

# Corrección Cuántica de Errores con Qudits Moleculares de Espín

Alonso Hernández Antón

Directors:

David Zueco Láinez

Alberto Castro Barrigón

Fernando Luis Vitalla

Department of Condensed Matter Physics

Department of Theoretical Physics

University of Zaragoza

Course 2020/2021

"Y con tanto ruido, no se oyó el ruido del mar."
Joaquín Sabina

Este trabajo pone punto y final a una etapa increíble. Estos cuatro años han sido los mejores de mi vida. Gracias de corazón a todos los que me habéis acompañado, en especial a mi familia. Os quiero mucho.

Gracias también a David, Alberto y Fernando, por vuestra inagotable paciencia y vuestra infinita disposición. Sois inspiración y ejemplo para quienes queremos dedicarnos a la investigación. Me lo habéis puesto muy fácil, y me habéis abierto mil puertas.

The work behind this Bachelor Thesis has been carried out under a Collaboration Scholarship granted by the Spanish Ministry of Education.

**Abstract**

Quantum computing is one of the most promising technologies for the upcoming decades, with applications ranging from pharmacy to finance. However, its physical realization is an enormous challenge. The microscopic systems used to process information in these devices are subject to decoherence due to the interaction with their environment, spoiling the results of any intended calculation. Our objective in this thesis work has been to apply optimal control theory to a system described by Lindblad's equation, a strategy that has been proposed to deal with errors. We will address the particular case of spin qubits on molecular nanomagnets. First, we have built a model to describe their dynamics in the presence of noise. Then, making use of it, we have designed and programmed an algorithm capable of simulating these systems and finding the optimal way to implement operations on them, minimizing errors. The results are promising, since we have obtained fidelities over 90%.

# Index

# 1 Introduction: How to Build a Quantum Computer

## 1.1 Quantum Information in a Nutshell

This thesis is oriented towards the physical realization of universal quantum computers. Let us start by summarizing some key concepts about quantum information [1].

Classical computers carry out operations with binary digits (bits), that is, zeros and ones. The proposal for quantum computing is to use quantum bits (qubits), which are a different kind of information units. These are not only capable of using a zero or a one, but any superposition of them, $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, following the rules of quantum mechanics (QM). The reason why this technology has awaken so much interest is that a computer operating with qubits can perform tasks that a classical computer cannot. In fact, certain mathematical problems that show exponential complexity in classical computers can be solved in quantum devices with a non-exponential amount of operations.

In order to carry out complex and useful calculations, quantum computers must be able to cope with more than one qubit simultaneously. As it is stated in QM, a state representing $n$ qubits will live in $\left(\mathbb{C}^2\right)^{\otimes n}$. In order to simplify the notation, we will use the "computational basis", that is, any product state inside an $n$-qubit space, say $|a_{n-1}\rangle \otimes ... \otimes |a_0\rangle \equiv |a_{n-1}...a_0\rangle$, $a_k \in \{0,1\} \forall k$, will be called $|x\rangle$, where $x = a_{n-1} \cdot 2^{n-1} + ... + a_0 \cdot 2^0$. For example, in a 3-qubit system, $|010\rangle = |2\rangle$ and $|111\rangle = |7\rangle$.

The "logic gate" is the basic operation in information theory, and is defined by a transformation of a set of bit states into another set. This is the core of every algorithm in both classical and quantum computers. In fact, most classical logic gates have a quantum counterpart. For example, the classical NOT gate, which acts on one single bit as NOT(0)=1 and NOT(1)=0, becomes the following operator when "quantized":

$$\begin{cases} \text{NOT}|0\rangle = |1\rangle \\ \text{NOT}|1\rangle = |0\rangle \end{cases} \quad , \quad |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad , \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \longrightarrow \quad \text{U}_{\text{NOT}} \equiv \oplus \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

The diagram for this gate as a quantum circuit is the following:

$$|\psi_0\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \;\text{—}\oplus\text{—}\; |\psi_f\rangle = \text{U}_{\text{NOT}}\,|\psi_0\rangle = \alpha\,|1\rangle + \beta\,|0\rangle$$

Every logic gate can be represented as a unitary matrix U in the computational basis. There are also gates involving more qubits, which permit to couple their states. *Controlled* gates operate on a *target* qubit (or set of qubits) depending on the state of another, *control*, one(s). Two important controlled gates are the CNOT and TOFFOLI (TFF) gates:

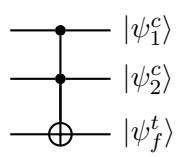
$$\text{U}_{\text{CNOT}} \equiv \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 0 & 1 \\ & & 1 & 0 \end{bmatrix} \begin{matrix} |0\rangle \equiv |00\rangle \\ |1\rangle \equiv |01\rangle \\ |2\rangle \equiv |10\rangle \\ |3\rangle \equiv |11\rangle \end{matrix}$$

$$|\psi^c\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \;\text{—}\bullet\text{—}\; |\psi^c\rangle$$
$$\left|\psi_0^t\right\rangle = \gamma\,|0\rangle + \delta\,|1\rangle \;\text{—}\oplus\text{—}\; |\psi_f^t\rangle$$

$$|\psi_f^t\rangle = [\alpha \cdot \mathbb{I} + \beta \cdot \text{U}_{\text{NOT}}]\,|\psi_0^t\rangle \quad \longrightarrow \quad \left|\psi_f^t\right\rangle \text{ depends on } |\psi^c\rangle \text{ via } \alpha \text{ and } \beta$$

$$
\mathrm{U_{TFF}} \equiv \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & & 0 & 1 \\ & & & 1 & 0 \end{bmatrix}
\begin{matrix} |0\rangle \equiv |000\rangle \\ \vdots \quad\quad \vdots \\ |5\rangle \equiv |101\rangle \\ |6\rangle \equiv |110\rangle \\ |7\rangle \equiv |111\rangle \end{matrix}
$$

$$
\begin{aligned} |\psi_1^c\rangle &= \alpha_1 |0\rangle + \beta_1 |1\rangle \\ |\psi_2^c\rangle &= \alpha_2 |0\rangle + \beta_2 |1\rangle \\ |\psi_0^t\rangle &= \gamma |0\rangle + \delta |1\rangle \end{aligned}
$$




$$
|\psi_t^f\rangle = [(\alpha_1\alpha_2 + \alpha_1\beta_2 + \beta_1\alpha_2) \cdot \mathbb{I} + \beta_1\beta_2 \cdot \mathrm{U_{NOT}}] \, |\psi_t^0\rangle
$$

Any combination and concatenation of quantum logic gates constitutes a quantum circuit. To execute any algorithm on a quantum computer, one must design a specific circuit for it, decomposing its unitary operations into sequences of gates. The main difficulty nowadays lies in buiding a physical implementation of gates and qubits. Quantum systems experience noise, and so will do the results of our operations if we do nothing to mitigate it. In fact, noisy calculations usually return useless results. We have a great challenge ahead.

### 1.2 Physical Qubits: Molecular Nanomagnets

We will focus on a specific physical implementation of qubits: spin qubits. An $S = 1/2$ spin is a two-level system, so we can encode a qubit within it. We can assign, for example, $|0\rangle \equiv |+1/2\rangle$ and $|1\rangle \equiv |-1/2\rangle$. In this work, we consider spins on molecular nanomagnets, as the ones studied by the QMAD group at INMA [2, 3, 4] and several others worldwide. Each molecule consists of a core, made of one or several magnetic ions, surrounded and stabilized by a shell of organic ligand molecules. Chemical design offers nearly unbound possibilites to tune their physical properties via changes in composition and structure. In particular, the use of rare-earch ions is very promising due to their high coherence times.

A good candidate is the $GdW_{30}$ complex [2], whose molecular structure is shown in figure 1a. Its core is a $Gd^{3+}$ ion with a $4f^7$ configuration, whose groundstate has $L = 0$ and $S = 7/2$. This d-level system ($d = 2S + 1 = 8 = 2^3$) can be considered a *qudit*, which can encode three qubits in one single molecule. The ability of embedding non trivial quantum functionalities in a microscopic and fully reproducible system is a big advantage of molecular nanomagnets, since a quantum computer made of these components might be much more scalable [4].

Under the effect of a DC magnetic field $\vec{H}$, the spin Hamiltonian of this molecule can be well approximated by a orthorrombic zero field splitting plus a Zeeman contribution [5, chap. 2]:

$$
\hat{H}_0 = D\left[\hat{S}_z^2 - \frac{1}{3}S(S+1)\right] + E\left(\hat{S}_x^2 - \hat{S}_y^2\right) - g\mu_{\mathrm{B}}\hat{\vec{S}} \cdot \vec{H} \tag{1.1}
$$

where $D = 1281$ MHz and $E = 294$ MHz are experimental values [2]. The splitting of the energy levels is shown in figure 1b.

But this static (bare) Hamiltonian is not enough to operate with qubits. We also need a time-dependent term, able to modify the qubit states and implement the gates. In our case, the molecules will be coupled to a superconducting circuit, capable of making microwave pulses pass through them. The magnetic component of these pulses will interact with the molecular spin,

inducing a certain time evolution on it. This will be our steering wheel to perform operations. To execute any unitary U acting on the mathematical state of the qubits, we must find the microwave pulse that yields the necessary physical evolution on the states of the molecule. Therefore, we must introduce a time-dependent perturbation into the system:

$$\hat{H}(t) = \hat{H}_0 + g(t)\hat{V}, \qquad \hat{V} = -g\mu_B \vec{\hat{S}} \cdot \vec{H}_{\text{mw}} \tag{1.2}$$

The function $g(t)$, which encodes the time dependent amplitude of the magnetic field $\vec{H}_{\text{mw}}$, is what we are going to *control* to perform the target unitaries.

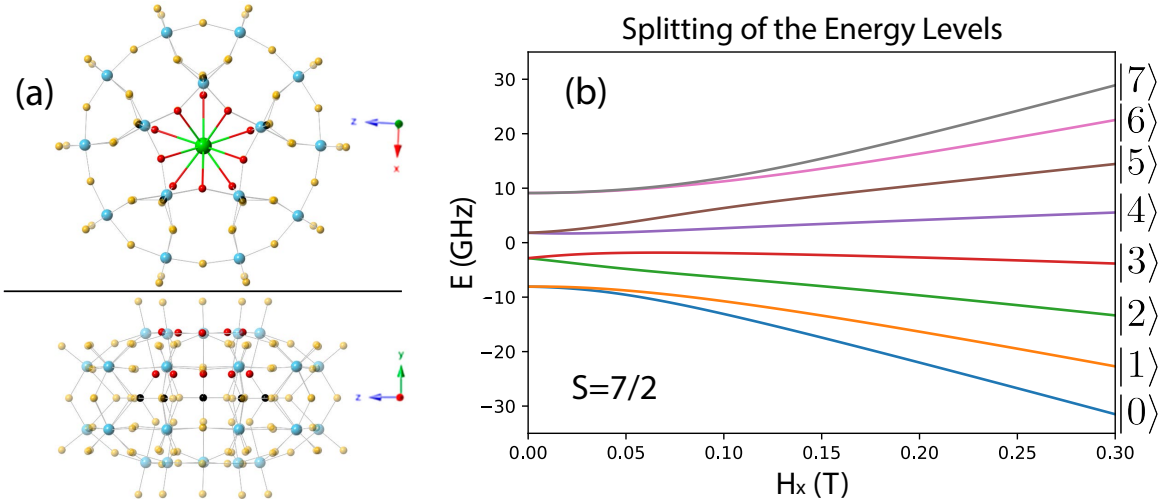



Figure 1: (a) Molecular structure of the $GdW_{30}$ complex. Notice that "$GdW_{30}$" is just an abbreviation: its actual chemical formula is $K_{12}Gd(H_2O)P_5W_{30}O_{110} \cdot 27.5H_2O$. (b) Splitting of the $GdW_{30}$ energy levels when an external static magnetic field is applied. This molecule has been tailored to have a certain anisotropy around the $Gd^{3+}$ ion for its energy levels to show a zero-field splitting. Furthermore, under the effect of a DC magnetic field $\vec{H} \parallel \hat{x}$ they split due to the Zeeman interaction and all the degeneracies are broken. The transition frequencies among all the levels are different and have an affordable magnitude, which makes them all addressable with microwave (GHz) pulses. This enables the realization of quantum operations within the $S = 7/2$ spin manifold.

Our molecules are not isolated, but embedded within a chip inside a dilution refrigerator, with the whole universe around. We do not have a closed system, but an open one, and the interaction of the molecule with its environment will have an important effect on the state of our qubits. This is the aforementioned "noise", and it will make errors arise in our calculations.

The goal of this thesis work is to design optimal hardware-specific operations to mitigate the effect of noise in our qudits. In section 2 we will present a mathematical description of the interaction of a quantum system with its environment. Then, in section 3, we will design an algorithm capable of finding the optimal perturbation $g(t)$ in (1.2) that performs any desired operation on the qubits. Finally, section 4 shows the results we have obtained making use of it.

## 2 Open Quantum Systems: Noise and Errors

In this section, we will present the basics of the theory of open quantum systems: how they behave and evolve in time when they are coupled to their environment and how dissipative effects emerge. Making use of this theory, we will later search for a mathematical description of the $GdW_{30}$ complex, based on previous experimental measurements.

### 2.1 System and Environment: Statistical Description

A quantum system is said to be open when it interacts with its environment. In other words, it is a part of a larger system that contains both the system itself, and its environment. Let us denote the system by S; its states will live inside the Hilbert space $\mathcal{H}_S$, and its behaviour will be determined by the Hamiltonian $\hat{H}_S$, in the absence of interaction with its environment. Likewise, the environment (bath) B will live inside $\mathcal{H}_B$, and evolve according to $\hat{H}_B$. If there is an interaction between them, the total space will be $\mathcal{H}_S \otimes \mathcal{H}_B$, and the Hamiltonian will take the form $\hat{H}_S \otimes \hat{\mathbb{I}} + \hat{\mathbb{I}} \otimes \hat{H}_B + \hat{H}_{int}$, where $\hat{H}_{int}$ is the interaction term.

In principle, one could model the full S+B "universe" as a closed system, but this is impossible, due to the lack of knowledge about the environment, and its size. The environment has a huge number of degrees of freedom, so we will never be able to describe it in detail. This is why we need to use thermodynamics and statistics. The state of our system S will not be well-known anymore, but will obey probability distributions instead. Here, the "ket" description of the system is substituted by the density operator formalism. Due to space constraints, we cannot summarize here the foundations of statistical quantum mechanics, but we have written a quick overview about it in appendix 6.1.

### 2.2 Time Evolution in Open Systems: Master Equation

If $\rho$ is the density matrix that describes the state of the universe S + B, the state of the system itself is obtained by taking the partial trace over the environment degrees of freedom $\rho_S = \mathrm{Tr}_B(\rho)$. The expectation value of an operator $\hat{O}$ defined on the system can then be computed by taking $\langle \hat{O} \rangle = \mathrm{Tr}[\rho_S \hat{O}]$. Now that we know how to describe the states of our system and how to evaluate observables on it, the last ingredient that we need is an equation of motion, that is, a way to describe how the state of the system evolves in time.

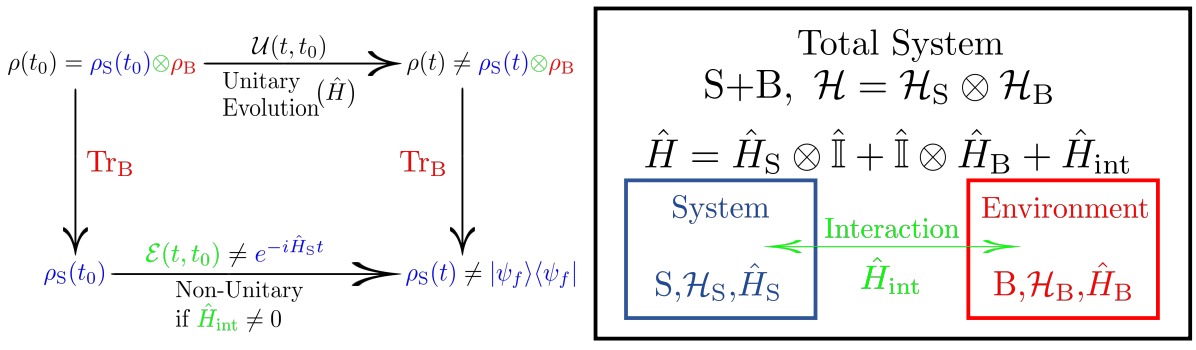



Figure 2: Graphical representation of an open system.

Let us recall the problem that we are facing. Figure 2 sketches the behaviour of an open system. Suppose we start at time $t_0$, with a separable state $\rho(t_0) = \rho_{\mathrm{S}}(t_0) \otimes \rho_{\mathrm{B}}$, where $\rho_{\mathrm{S}}(t_0)$ is, furthermore, pure: $\rho_{\mathrm{S}}(t_0) = |\psi_0\rangle\langle\psi_0|$. The whole system will evolve according to $\hat{H}$, leading to an entangled state. Tracing over the environment B, we could get the initial (pure) and final (mixed) states of S: $\{\rho_{\mathrm{S}}(t_0), \rho_{\mathrm{S}}(t)\}$. Since we are not able to know $\rho_{\mathrm{B}}$, we must find a map $\mathcal{E}$ such that

$$\rho_{\mathrm{S}}(t) = \mathcal{E}(t, t_0)\left[\rho_{\mathrm{S}}(t_0)\right] \tag{2.1}$$

This map will only describe a physically plausible evolution if it is a "universal dynamical map" (UDM), that is, if it meets the following mathematical requirements [6]:

- It is linear: $\mathcal{E}(t, t_0)[\alpha\rho_1 + \beta\rho_2] = \alpha\mathcal{E}(t, t_0)\rho_1 + \beta\mathcal{E}(t, t_0)\rho_2,\ \forall\ \alpha, \beta, \rho_1, \rho_2$.
- It is Markovian: $\mathcal{E}(t_2, t_0) = \mathcal{E}(t_2, t_1)\mathcal{E}(t_1, t_0),\ \forall\ t_0 < t_1 < t_2$.
- It leaves the state unchanged if there is no time evolution: $\mathcal{E}(t_0, t_0) = \mathbb{I},\ \forall\ t_0$.

One key detail is the fact that the inverse of a UDM $\mathcal{E}$ is not, in general, a UDM, unless $\mathcal{E}$ is unitary. The non-unitarity of $\mathcal{E}$ implies the irreversibility of the dissipative evolution suffered by S, and is due to the interaction $\hat{H}_{\mathrm{int}}$. In the language of thermodynamics, if the purity of S, $P_{\mathrm{S}}(t) = Tr\left(\rho_{\mathrm{S}}^2(t)\right)$, decreases with time, its entropy will increase and, therefore, if $\mathcal{E}$ is understood as a thermodynamical process, it will not be reversible, since $\Delta\mathcal{S} > 0$. As one should expect in such a situation, this process is in fact a relaxation towards equilibrium. The final state for $t \longrightarrow \infty$ is the thermal state $\rho_{\mathrm{th}}$ (more about this in appendix 6.1).

Instead of directly trying to build the map $\mathcal{E}$, the usual route consists of using its differential form, i.e.:

$$\frac{d}{dt}\rho(t) = \lim_{h\to 0}\frac{\rho(t+h) - \rho(t)}{h} = \lim_{h\to 0}\frac{\mathcal{E}(t+h, t) - \mathbb{I}}{h}\rho(t) \equiv \mathcal{L}(t)\rho(t) \tag{2.2}$$

$$\mathcal{L}(t) = \lim_{h\to 0}\frac{\mathcal{E}(t+h, t) - \mathbb{I}}{h} \tag{2.3}$$

Notice that, hereafter, we will remove the subscript S from the objects describing the subsystem to ease the notation, i.e. the density $\rho(t)$ in the previous equation in fact refers to subsystem density $\rho_{\mathrm{S}}(t)$.

It can be shown [6] that any differential equation that yields a Markovian evolution can be written as:

$$\frac{d}{dt}\rho(t) = -i\big[\hat{H}(t), \rho(t)\big] + \sum_k \gamma_k(t)\left[\hat{L}_k(t)\rho(t)\hat{L}_k^\dagger(t) - \frac{1}{2}\left\{\hat{L}_k^\dagger(t)\hat{L}_k(t), \rho(t)\right\}\right] = \mathcal{L}(t)\rho(t) \tag{2.4}$$

This is called the "Lindblad equation" [6, 7]. It is a master equation, since it describes the time evolution of a probability distribution, and $\mathcal{L}$ is a linear operator. The particular form of $\{\gamma_k\}$ and $\{\hat{L}_k\}$ will vary from one case to another, this is just the general form of the equation. Taking $\gamma = \sum_k \gamma_k$ and leaving $\tilde{\gamma} = \gamma_k/\gamma$ inside the dissipative term, we can simplify the notation:

$$\dot{\rho} = \mathcal{L}(t)\rho = \underbrace{-i\big[\hat{H}(t), \rho\big]}_{\text{Unitary term, as in (6.1)}} + \underbrace{\gamma\mathcal{D}(t)\rho}_{\text{Non-unitary/dissipative term}} \tag{2.5}$$

The last thing we need is to be able to calculate $\mathcal{E}$ from $\mathcal{L}$, that is, solving the master equation. The solution is [6, 7]:

$$\rho(t) = \mathcal{E}(t, t_0)\rho(t_0) \quad , \quad \mathcal{E}(t, t_0) = \mathcal{T} \exp\left[\int_{t_0}^{t} \mathcal{L}(t')dt'\right] \tag{2.6}$$

where $\mathcal{T}$ is the time-ordering operator [7]. Given the linearity of both Schrödinger's equation for closed systems, and Lindblad's equation for open ones, the form of the solution equations are in fact analagous: see, in appendix 6.1, Eqs. (6.2) and (6.4).

If we knew the exact form of $\hat{H}$, we could deduce the values of the parameters $\{\gamma_k(t)\}$ and the transition operators $\{\hat{L}_k(t)\}$ explicitly, as in [6, 7]. However, this is not always possible. Fourtunately, we will later see that they can be deduced from experimental observations.

One must bear in mind that, as the perturbation $g(t)\hat{V}$ modifies $\hat{H}(t)$, it will also modify the dissipators. Nonetheless, there is an important simplification that can be used in most cases. The microwave pulse is a weak perturbation of the bare Hamiltonian, so $\hat{W}(t) \equiv g(t)\hat{V} \ll \hat{H}_0$ and this term modifies the eigenstates of the molecule to $|\phi_k\rangle = |\phi_k^0\rangle + o(\hat{W})$. Therefore, the dissipative parameters will also become $\hat{L}_k = \hat{L}_k^0 + o(\hat{W})$ and $\gamma_k = \gamma_k^0 + o(\hat{W})$. Moreover, the dissipative term is a weak-coupling approximation, meaning that the transition frequencies $\{\Omega_k\}$ among the levels of the molecule are much bigger than the damping parameters $\{\gamma_k\}$. Combining these two features, we have $\gamma_k \hat{L}_k^2 = \gamma_k^0(\hat{L_k^0}) + o(\gamma\hat{W})$. Using equation (2.4), this yields $\gamma\mathcal{D} \sim \gamma^0\mathcal{D}^0 + o(\gamma\hat{W})$, where the last term is completely neigligible compared to the unitary part of the equation. Consequently, we can consider that the dissipative term is independent of the perturbation $g(t)\hat{V}$. This means that, in our case (actually in most cases), $\{\gamma_k\}$ and $\{\hat{L}_k\}$ are independent of time.

### 2.3 Relaxation and Dephasing Errors

The different dissipators in (2.4) can be associated to different kinds of errors. Let us work in this operational way, i.e. discussing the typical errors and deriving their corresponding dissipators.

One source of errors is the spin-lattice interaction, i.e. the coupling to the vibrational modes both of the molecule itself and of the lattice. The spin can exchange energy with phonons in a charactersitic time, typically called relaxation time, $T_1$. This process makes the populations decay exponentially towards lower energy levels.

The other source of noise is the spin-spin interaction, and is due to the coupling of the molecular spin to other surrounding spins. System and environment do not exchange energy but information. The populations of the states do not change, but their phases do, in a characteristic time $T_2$, in which the non-diagonal terms of the density operator also decay exponentially. This is known as "dephasing".

However, the practical meaning of $T_1$ and $T_2$ goes a bit further. Spin-lattice and spin-spin interactions are not the only sources of noise. Many more phenomena can effectively reduce $T_1$ and $T_2$: Nuclear spins, impurities, fluctuations of the microwave pulses, external contributions to

magnetic fields, fluxons in the superconducting circuits or effective impedances in the electronic devices, among many other, may affect our system too. Therefore, the experimental conditions of our quantum computer and its operation point must be carefully chosen in order to minimize these environmental effects and maximize the coherence times $T_1$ and $T_2$, which must be obtained from measurements. Pararell to the efforts in improving them, in this thesis we will optimize the external driving (1.2) trying to perform useful operations faster than those limiting times.

## 2.4 Experimental Input: Parameters of the Dissipator

The measurements of $T_1$ and $T_2$ will help us build our Linblad equation. Their associated $\hat{L}_k$ operators are $\hat{S}_x$ and $\hat{S}_z$, respectively. In that way, the $T_1$ term in the dissipator is a diagonal matrix, whereas the $T_2$ one only has terms outside the diagonal.

Experiments show that, in general, $T_1 > T_2$, sometimes by a little difference, and sometimes by orders of magnitude. For our $GdW_{30}$ molecule, Ref. [2] shows that $T_1 \approx 2\ \mu s$ and $T_2 \approx 0.5\ \mu s$ at temperatures of about $5\ K$. Note that these coherence times are a spin-1/2 concept and are not characteristic of the whole molecule but of each two-level transition. However, in this case all the transitions have approximately the same decoherence time. As we see, $T_1 > T_2$, and so dephasing is quite faster than relaxation. Even if the difference is not too big, our model will only consider dephasing, for the sake of simplicity. Furthermore, $T_2$ is less dependent on temperature than $T_1$, and relaxation can always be supressed by cooling the spins to very low temperatures to get rid of vibrations.

Hence, we will only have one term in the dissipator, with $\hat{L} = \hat{S}_z$. The only task remaining is to find $\gamma$ and see how it is related to $T_2$. Let us start by writing $\dot{\rho}$ explicitly, using Lindblad equation. We have an S=7/2 spin, so the eigenstates of the molecule are $|s, m\rangle$, where $m = -7/2, ..., 7/2$. The dissipator is:

$$\gamma \mathcal{D}\rho = \gamma \left[ \hat{S}_z \rho \hat{S}_z - \frac{1}{2}\{\hat{S}_z^2, \rho\} \right] \tag{2.7}$$

Since $\rho = \sum_{m,m'} \rho_{mm'} |m\rangle\langle m'|$, we can easily calculate its terms:

$$\begin{aligned}
\hat{S}_z|s,m\rangle &= m|s,m\rangle \\
\hat{S}_z|s,m\rangle\langle s,m'|\hat{S}_z &= mm'|m\rangle\langle m'| \\
\hat{S}_z\rho\hat{S}_z &= \sum_{m,m'} \rho_{mm'} \cdot mm'|m\rangle\langle m'| \\
\hat{S}_z^2\rho &= \sum_{m,m'} \rho_{mm'} \cdot m^2|m\rangle\langle m'| \\
\rho\hat{S}_z^2 &= \sum_{m,m'} \rho_{mm'} \cdot m'^2|m\rangle\langle m'| \\
\hat{S}_z\rho\hat{S}_z - \frac{1}{2}\{\hat{S}_z^2, \rho\} &= -\frac{1}{2} \sum_{m,m'} \rho_{mm'}(m-m')^2|m\rangle\langle m'|
\end{aligned}$$

Writing Lindblad equation term-wise, we find:

$$\dot{\rho}_{mm'} = -i\big[\hat{H},\rho\big] - \frac{\gamma}{2}\Delta m^2 \rho_{mm'}$$
$$\rho_{mm'}(t) = \rho_{mm'}^{\text{unitary}}(t)\cdot e^{-\frac{\gamma}{2}\Delta m^2 t} \tag{2.8}$$

On the other hand, it can be shown that $T_2$ is the decay time of $\langle \hat{S}_x \rangle$, that is $\langle \hat{S}_x \rangle \sim e^{-t/T_2}$. In this case:

$$\langle \hat{S}_x(t)\rangle = \text{Tr}\left[\hat{S}_x \cdot \rho(t)\right] = \frac{1}{2}\,\text{Tr}\left[\sum_{m,m'} \rho_{mm'}\big(\hat{S}_+ + \hat{S}_-\big)|m\rangle\langle m'|\right] =$$
$$= \frac{1}{2}\sum_k \Bigg[\underbrace{\rho_{k-1,k}(t)\sqrt{s(s+1)-k(k-1)}}_{k=m'=m+1} + \underbrace{\rho_{k+1,k}(t)\sqrt{s(s+1)-k(k+1)}}_{k=m'=m-1}\Bigg] \overset{(2.8)}{=}$$
$$\overset{(2.8)}{=} \langle \hat{S}_x(t)\rangle^{\text{unitary}}\cdot e^{-\frac{\gamma}{2}t} \quad \longrightarrow \quad e^{-\frac{\gamma}{2}t} = e^{-t/T_2}$$

finding the relation $\gamma = 2/T_2$, with $T_2 \approx 500$ ns. The master equation is fully characterized.

## 2.5 How to deal with Errors: Mitigation and Correction

Now that we know the behaviour of our molecule as an open system, we need to find the way to avoid its inherent noise and execute fault-tolerant calculations. There are two ways by which we can do this, and we will address both.

The first way consists in designing the microwave pulses that induce the execution of operations in such a way that the effect of the environment is reduced. This task can be carried out numerically, and it has occupied most of our efforts. These techniques (and similar approaches), consisting in designing hardware-specific operations that are as resilient to noise as possible, are known as "error mitigation" techniques. Error mitigation is a key part of what is called the "firmware" [8] of most NISQ[1] devices [9].

There is a second way to deal with noise, which is executing certain sets of gates that are specifically designed to correct errors. These are known as "error correction algorithms" or "error correction codes". They do not avoid decoherence, but are able to detect errors and execute operations to fix them. The strategy consists in codifying a single logical qubit within several physical ones, using the rest of available states to execute these codes. This way, a set of multiple physical qubits can act as one fault-tolerant logical qubit.

Our molecule has S=7/2, which is equivalent to three S=1/2 qubits. We can codify our state of interest in one of them, and use the other two to detect and correct errors. A good candidate to prevent phase errors is the three-qubit code (TQC) proposed in [3], which is shown in figure 3. This algorithm is not designed to execute any operation on the state $|\psi\rangle$, but just to store its information, so that it remains safe until we need to use it. One must keep in mind that this TQC is only capable of correcting single phase errors. To prevent more than one error, one would need more than three physical qubits. However, even with these limitations, the integration of

[1] Noisy Intermediate-Scale Quantum

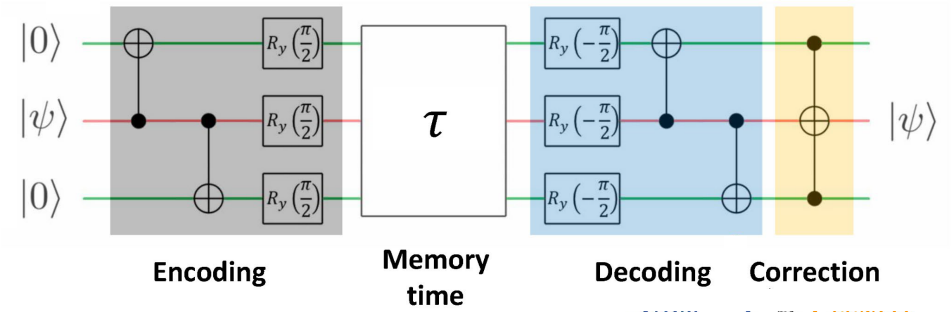



Figure 3: This algorithm has several stages. The first one, called "encoding", leaves the state prepared for the errors to be detected: the CNOT gates entangle the central qubit, which carries the quantum information $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, with the auxiliary (ancilla) ones, that are initialized to $|0\rangle$. This produces an entangled state that is protected from relaxation ($T_1$). The $R_y$ gates rotate the states for them to be robust against pure dephasing ($T_2$) instead. After this stage, there is a "memory time" during which the states remain stored and the errors arise. Finally, the "decoding" and "correction" blocks detect phase errors and correct them.

error correction in each molecular building block constitutes a major advantage of this scheme.

Our objective in this thesis will be to design fault-tolerant unitaries using optimal control theory (OCT), which can be considered a form of error mitigation. Of course, this is not incompatible with the use of error correction codes: one may later build error correction algorithms with the generated unitaries. This will give us an efficient way to both store quantum information and perform operations, minimizing errors.

# 3 Optimal Control Theory for Open Quantum Systems: Error Mitigation

A given quantum gate or circuit can be built with many possible external perturbations or "protocols". Our goal is to design the gates with the help of OCT, in such a way that the external perturbation is fast, and the error induced by the environmental noise is minimized. In order to achieve this objective, we must work with Lindblad's equation.

Let us now sketch the OCT method. We depart from the driven Hamiltonian:

$$\hat{H} = \hat{H}_0 + g(t)\hat{V}\,, \tag{3.1}$$

where $\hat{H}_0$ is the bare Hamiltonian of our molecule, $\hat{V}$ contains the physical means of the perturbation, and $g(t)$ is the temporal shape of the pulse.

Given a unitary $\mathrm{U}_{\mathrm{target}}$, our goal is to find a pulse $g(t)$ which, for any initial pure state $\rho_0 = |\psi_0\rangle\langle\psi_0|$, takes the system to another state as close as possible to $\rho_{\mathrm{target}} = \mathrm{U}_{\mathrm{target}}\rho_0\mathrm{U}_{\mathrm{target}}^{\dagger}$, which is also pure. To do that, we have designed and programmed an algorithm based on OCT, capable of calculating $g(t)$ efficiently. In this section, we will explain the equations the algorithm is based on.

### 3.1 Parametrization

In order to numerically represent the pulse $g(t)$, we will give it a functional dependence on a certain set of parameters $u \equiv \{u_k\}$. The chosen parametrization is a Fourier decomposition, depending on the multiples of $\omega_k = \frac{2\pi}{T}$, where $T$ is the total duration of the pulse:

$$g(u,t) = u_0 + \sum_{k=1}^{M} u_{2k}\cos(\omega_k t) + \sum_{k=1}^{M} u_{2k-1}\sin(\omega_k t) \tag{3.2}$$

Thus, we have $2M+1$ $u_k$ parameters. The series has a maximum of $M$ frecuencies, i.e. there is a frequency cutoff whose choice will be discussed later. The Hamiltonian, and therefore the operators involved in the Lindbladian evolution, can now be considered functions of the parameters $u$:

$$\hat{H} = \hat{H}(u,t) = \hat{H}_0 + g(u,t)\hat{V} \tag{3.3}$$

$$\mathcal{L}\rho = \mathcal{L}(u,t)\rho = -i\left[\hat{H}_0 + g(u,t)\hat{V}, \rho\right] + \gamma\mathcal{D}\rho \tag{3.4}$$

$$\mathcal{E} = \mathcal{E}(u,t,t_0) = \mathcal{T}\exp\left[\int_{t_0}^{t}\mathcal{L}(u,t')dt'\right] \tag{3.5}$$

One must notice that, as we showed before, the dissipator $\mathcal{D}$ does not depend on the pulse $g(t)$ (or at least the dependence is very weak), and therefore it does not depend on the parameters either.

### 3.2 Fidelity and Target Function

Speaking in usual OCT jargon, we need to define a "target function", i.e. a function that depends on the propagated state $\rho(T)$ that measures how well the pulse has achieved its goal. In our case, the goal is to minimize the distance between $\rho(T)$ and the target state $\rho_{\text{target}}$. Therefore, we first need some function to measure the fidelity between two states, that is how similar they are. We first consider the Hilbert-Schmidt product:

$$(\rho_{\text{A}}, \rho_{\text{B}}) = \text{Tr}(\rho_{\text{A}}^{\dagger}\cdot\rho_{\text{B}})\,, \tag{3.6}$$

from which we may define a norm and a distance in the usual way: $||\rho||^2 = (\rho,\rho)$ and $D(\rho_{\text{A}},\rho_{\text{B}}) = ||\rho_{\text{A}} - \rho_{\text{B}}||$. We could define the target functional as $F(\rho(T)) = D(\rho(T),\rho_{\text{target}})$ and this object should then be minimized by the algorithm. However, as it is explained in [10], the Hilbert-Schmidt product is in fact a good-enough measure of the fidelity between two states: noticing that $D^2(\rho_{\text{A}},\rho_{\text{B}}) = ||\rho_{\text{A}}||^2 + ||\rho_{\text{B}}||^2 - 2(\rho_{\text{A}},\rho_{\text{B}})$, they proposed to simplify the target function as:

$$F\left(\rho(T)\right) = \left(\rho(T),\rho_{\text{target}}\right), \tag{3.7}$$

If $\rho_{\text{target}}$ is pure, the maximization of this function is *almost* equivalent to the minimization of the distance, and the difference is related to the purity of the propagated state $||\rho(T)||^2$.

Since a particular choice of the parameter $u$ determines the evolution of the system, we will denote $\rho[u](T)$ to the state propagated with the evolution operator given by a chosen set of

parameters $u$. Our actual target function will be:

$$G(u) = F\left(\rho[u](T)\right) = \mathrm{Tr}\left(\rho_{\mathrm{target}} \cdot \rho[u](T)\right), \tag{3.8}$$

where we recall that $T$ is the execution time of the operation, that is, the total time the pulse will be active. Notice that, due to the properties of the density matrices and their scalar product, $G(u) \leq 1$, and the limit case where $G(u) = 1$ is given only if $\rho[u](T) = \rho_{target}$. Therefore, we can understand $G(u)$ as a measure of the fidelity achieved by the pulse $g(u,t)$.

Our objective is to explore the space of the parameters and try to maximize $G(u)$, in order to find the optimal pulse $g(u,t)$ to be executed during the pre-defined time $T$. The procedure will be the following:

1. We start with an initial guess for the set of parameters $u$, and propagate the initial state, obtaining:
$$\rho[u](T) = \mathcal{E}(u, T, t_0) \cdot \rho(t_0) \tag{3.9}$$
2. Then, we must measure "how close" to $\rho_{target}$ this state is, obtaining $G(u)$.
3. Now, we must decide whether $G(u)$ is big enough for our purposes. In case it is, we have finally obtained the pulse $g(u,t)$ we wanted. In case it is not, we must select a new set of parameters $u$ according to some optimization algorithm and go back to step (1).

One can easily notice, however, that a big amount of parameters may give us a better result, but will also steeply increase the cost of this computation as it will increase the size of the search space. The landscape $G(u)$ may be very complex, and not smooth at all. In order to simplify this optimization problem, the algorithms that make use of the gradient of the function typically perform better. We will use the algorithm *LD_SLSQP* [11, 12] from the python *NLopt* library [13]. To do it, we must analitically find a way to compute the gradient. This is not a trivial task, due to how complex the translation of the dependence on $u$ from $g(u,t)$ to $G(u)$ is:

$$g(u,t) \longrightarrow \hat{H}(u,t) \longrightarrow \mathcal{L}(u,t) \longrightarrow \mathcal{E}(u,t,t_0) \longrightarrow \rho[u](T) \longrightarrow G(u) \tag{3.10}$$

### 3.3 Calculation of the Gradient

A more detailed description of the general OCT problem may be found in the appendix 6.2, along with the derivation of the gradient. For our purposes, it is sufficient to consider the case of a linear evolution equation, as Schrödinger and Lindblad equations are. Here, we merely state the results. For an arbitrary linear evolution equation

$$\begin{cases} \dot{y}(t) = \hat{A}(u,t)y(t) \\ y(0) = y_0 \end{cases} \tag{3.11}$$

and for a target function defined as $G(u) = F\left(y[u](t), u\right)$, the gradient takes the form

$$\frac{\partial G}{\partial u} = 2\mathbb{R}\mathrm{e} \int_0^T dt\ \lambda^\dagger(t) \frac{\partial \hat{A}}{\partial u} y(t) \Big|_{\lambda=\lambda[u], y=y[u]} + \frac{\partial F}{\partial u}\left(y(T), u\right) \Big|_{y=y[u]} \tag{3.12}$$

where the Lagrange multiplier $\lambda$ (also called the "costate") is defined by:

$$\begin{cases} \dot{\lambda}(t) = -\hat{A}^\dagger(u,t) \cdot \lambda(t) \\ \lambda(T) = \frac{\partial F}{\partial y^\dagger}\left(y[u](t), u\right) \end{cases} \tag{3.13}$$

This will be our starting point for the following sections, where we will just modify equations (3.12) and (3.13) for certain specific cases.

### 3.3.1 Schrödinger Evolution

A closed quantum system is governed by the following equations:

$$\begin{cases} |\dot{\psi}(t)\rangle = -i\hat{H}(u,t)|\psi(t)\rangle \\ |\psi(0)\rangle = |\psi_0\rangle \end{cases} \tag{3.14}$$

Identifying terms with (3.11), we have $y(t) = |\psi(t)\rangle$ and $\hat{A}(u,t) = -i\hat{H}(u,t) = -i\big[\hat{H}_0 + g(u,t)\hat{V}\big]$.

Suppose that the function we want to maximize is the expectation value of a certain observable $\hat{O}$, that is

$$F\left(|\psi(t)\rangle, u\right) = \langle\psi(T)|\hat{O}|\psi(T)\rangle = \int_0^T \langle\psi(t)|\hat{O}|\psi(t)\rangle\ \delta(t-T)dt \quad , \quad G(u) = F\left(|\psi[u](t)\rangle, u\right)$$

If we wanted to maximize the projection of the final state on another target (pure) state, the observable would be $\hat{O} = \rho_{target} = |\psi_{target}\rangle\langle\psi_{target}|$ and the target function could be simplified to $F\left(|\psi(t)\rangle, u\right) = |\langle\psi_{\text{target}}|\psi(T)\rangle|^2$. In any case, introducing all this into the previous equations (3.13), we find the following expression for the costate (which is a proper ket now):

$$\begin{cases} |\dot{\lambda}(t)\rangle = -i\hat{H}^\dagger(u,t)|\lambda(t)\rangle \\ |\lambda(T)\rangle = \hat{O}|\psi(T)\rangle \end{cases} \tag{3.15}$$

It is typical of OCT that the "initial" conditions for the equations of motion that define the costate are given in terms of its value at the final time $T$ (and therefore, numerically, they must be propagated backwards). Now, substituting for the gradient, we find:

$$\begin{aligned} \frac{\partial G}{\partial u} =& 2\mathbb{R}\mathrm{e}\left[-i\int_0^T \langle\lambda(t)|\frac{\partial\hat{H}}{\partial u}|\psi(t)\rangle\ dt\Big|_{\lambda=\lambda[u],\psi=\psi[u]}\right] + \cancel{\frac{\partial}{\partial u}\Big[\langle\psi(T)|\hat{O}|\psi(T)\rangle\Big]_{\psi=\psi[u]}} = \\ =& 2\mathbb{I}\mathrm{m}\int_0^T \frac{\partial g}{\partial u}(u,t)\langle\lambda(t)|\hat{V}|\psi(t)\rangle\ dt\Big|_{\lambda=\lambda[u],\psi=\psi[u]} \end{aligned} \tag{3.16}$$

We finally have a useful expression. For each guess $u$, we can compute the costate for all $t$ using (3.15), and then use it to calculate the gradient with (3.16). As $g(u,t)$ is linear on the parameters, its derivative with respect to any $u_k$ is exactly its corresponding sine or cosine.

One may wonder why going through all this trouble to get an analytical expression for the gradient, if one could just use a finite-difference formula to compute it numerically. However, if we calculated the gradient by finite differences, we would have to propagate the state at least $2(2M+1)$ times for every step of the optimization. Instead, using the previous formula, we only

need to propagate the state and the costate once per step. The advantage is quite noticeable since, as we will see later, M can be quite large when using long pulse durations $T$.

### 3.3.2 Lindblad Evolution

The case of an open system is, in fact, quite similar, since it also obeys a linear equation:

$$\begin{cases} \dot{\rho}(t) = \mathcal{L}(u,t)\rho(t) \\ \rho(0) = \rho_0 \end{cases} \tag{3.17}$$

This time, $y(t) = \rho(t)$ and $\hat{A}(u,t)\rho = \mathcal{L}(u,t)\rho = -i\big[\hat{H}(u,t),\rho\big] + \gamma\mathcal{D}\rho$. Our target function is:

$$F\left(\rho(t),u\right) = \frac{1}{2}\,\mathrm{Tr}\left[\left(\rho(T) + \rho^{\dagger}(T)\right)\hat{O}\right] = \int_0^T \mathrm{Tr}\left[\left(\rho(t) + \rho^{\dagger}(t)\right)\hat{O}\right]\ \delta(t-T)dt\,,$$

$$G(u) = F\left(\rho[u](t),u\right).$$

Here, the target function uses the Hermitian conjugate of $\rho$, since the hermiticity of the density operator is only guaranteed if it satisfies Lindblad equation, but mathematically, the optimization problem is formulated on a space where not all the possible $\rho's$ are hermitian. In any case, when the matrix is hermitian, the target function is exactly equal to $\mathrm{Tr}[\rho(T)\hat{O}]$.

Once again, to maximize the projection on any state $\rho_{target}$, even if it is mixed, we just have to substitute $\hat{O} = \rho_{target}$. In this case, the differential system for the costate takes the following form, as shown in appendix 6.2:

$$\begin{cases} \dot{\lambda}(t) = -\mathcal{L}^{\dagger}(u,t)\lambda(t) \\ \lambda(T) = \frac{1}{2}\hat{O} \end{cases} \tag{3.18}$$

The gradient can also be rewritten in a more specific way:

$$\frac{\partial G}{\partial u} = 2\mathbb{R}\mathrm{e}\int_0^T \lambda^*(t)\frac{\partial \mathcal{L}}{\partial u}\rho(t)\ dt\Big|_{\lambda=\lambda[u],\rho=\rho[u]} = 2\int_0^T \frac{\partial g}{\partial u}(u,t)\lambda^*(t)\big[\hat{V},\rho(t)\big]\ dt\Big|_{\lambda=\lambda[u],\rho=\rho[u]} \tag{3.19}$$

As we can see, this gradient expression gives us the same numerical advantage with respect to a hypothetical gradient computation through finite differences than in the Schrödinger case. Once again, we first have to compute the costate in order to introduce it into the expression for the gradient. However, there is a subtle detail here one should notice. The "initial" conditions for the costate are once again given for $t = T$. Therefore we have to propagate the costate backwards. In other words, it is defined in terms of $\mathcal{E}^{\dagger}$ (instead of $\mathcal{E}$), which, as it was mentioned above, does not necessarily have to be a UDM. This means that the evolution of $\lambda$ will not represent a physically plausible process. Even so, that is not problematic, since the costate is just a mathematical tool to calculate the gradient.

## 3.4 Multitarget Strategy

So far, we have built the target function such that its optimization leads to a pulse that drives a particular initial state $\rho_0$ to a final state $\rho_{\mathrm{target}}$. However, this is not equivalent to the design of the aforementioned target unitary (or logic gate) $\mathrm{U}_{\mathrm{target}}$, since a unitary must not only drive

$\rho_0$ to $\rho_{\text{target}}$, but every state $\rho_k$ to their respective $\rho_k^{\text{target}} = \mathrm{U}_{\text{target}}\rho_k\mathrm{U}_{\text{target}}^\dagger$. Therefore, as we will se later on the results of our simulations, in order to obtain a unitary, we need a more sophisticated target function.

There is a formulation of OCT based on the equation of motion for the propagator instead of the one for the state. That means that the target function and the gradient could be written in terms of propagators (U for the noiseless aproximation, $\mathcal{E}$ for the noisy one). In the case of the Schrödinger evolution, this can be affordable. For our system, for example, we have an $8 \times 8$ matrix, and so we have to propagate 64 variables. However, if we use Lindblad's equation, the propagator is $\mathcal{E}$, which acts on $8\times8$ density operators, and therefore it is a $64\times64$ super-operator: $d^4$, where $d$ is the dimension of the Hilbert space. This adds up to 4096 variables to propagate. Propagating so many variables and storing their value for all $t \in [0, T]$ for each optimization step is too expensive. We must therefore find an alternative way to face the problem, such that it reduces the computational cost of the optimization, while still providing a reliable execution of the operations.

We need to guarantee that our pulse drives correctly a whole set of $N$ states $\{\rho_k\}$. Our choice is to add terms to the fidelity function that we had for the single-state fidelity, as described in [10]. That is what we will call a "multitarget strategy". Every state $\rho_k$ will have a corresponding $\rho_k^{\text{target}} = U_{\text{target}} \cdot \rho_k \cdot U_{\text{target}}^\dagger$, so we can define the target function as

$$G(u) = \frac{1}{\sum_k^N \mathrm{Tr}\left[\rho_k^2\right]} \sum_k^N \mathrm{Tr}\left[\rho_k(T)\rho_k^{\text{target}}\right] \tag{3.20}$$

Since this function is in fact a linear combination of the fidelities that we were using so far, both the target function and its gradient are still very simple to compute. Our next task is to decide which set $\{\rho_k\}$ to use. If we propagate a whole basis of the space of $8 \times 8$ matrices, we would be back at the $d^4$ situation, and so we must find a smarter choice. Considering that the initial states of our executions will presumably be pure, a first proposal could be the set $\{|k\rangle\langle k|\}_{k=1}^8$, since any unitary can be given in terms of its effect on a basis. However, as projections do not take into account global phases, we could find a fidelity of 1 for a pulse that drove, say, $\alpha|0\rangle + \beta|1\rangle$ to $\alpha U_{\text{target}}|0\rangle + \beta e^{i\phi} U_{\text{target}}|1\rangle$. It would have the maximum fidelity while not acting properly. We need a bigger set of states to be properly propagated for the pulse to guarantee the correct execution of the unitary.

The problem of finding a good set of initial states to define the target functional was addressed by Goerz *et al.* [10]; among all the sets of initial states proposed there, we will choose to add a ninth state $(\rho_9)_{ij} = \frac{1}{8}$ to the set $\{|k\rangle\langle k|\}_{k=1}^8$, which can also be easily shown to be pure. This way, we will only propagate $(d+1)d^2 (= 576)$ variables, in constrast to the $d^4$ variables needed if one uses a full basis of the density matrix space.

### 3.5 Experimental Restrictions

Once the algorithm is completely designed and the route is charted, we must consider several further details regarding the optimization process, for the results to be useful for later experimental use. In the end, we want our pulses to be executed on a real molecule with real instrumentation,

and it will surely have amplitude, frequency and time restrictions. Therefore, we cannot run a fully unconstrained search for any external pulse. We must add constraints to the optimization algorithm that enforce experimentally realizable pulses.

First of all, our arbitrary signal generator will have a maximum output frequency $\omega_{\max}$. In our case, $\omega_{\max} = 5$ GHz. It is important that our generator can handle at least the characteristic frequencies of the system (the seven transition frequencies for adjacent states). In a Fourier expansion parametrization such as the given in Eq. (3.2), the cutoff is naturally implemented by setting the maximum frequency component $M$. We will choose the largest $M$ such that $\omega_M = \frac{2\pi}{T} M < \omega_{\max}$. The number of parameters will be $2M + 1$, and note that it will increase with $T$.

Second, our parameters are the fourier components of the pulse, that is, the amplitudes of the magnetic field. Our instrumentation will not be able to generate arbitrarily large fields, and therefore we must impose a constraint on the module of our parameters. The algorithm we have chosen allows us to introduce inequality constraints (that is, setting boundaries to the parameters) and other linear or nonlinear equality constraints that we will mention later. Therefore, we set $u_k \leq H_{\max}$ boundary constraints. The choice of $H_{\max}$ for the total amplitude to be affordable for the instrumentation is key for our calculations to be useful. If the maximum field for our generator is $H_{\text{gen}}$, the safest call would be $H_{\max} = \frac{H_{\text{gen}}}{2M+1}$, although it might be too restrictive. In any case, we will pick arbitrary values for $H_{\max}$ and not address this discussion here.

The other two restrictions that we will impose on the optimization are equality constraints. First, we need the pulse to be continuous in time, that is, $g(0) = g(T) = 0$. Since $\sin(\omega_k \cdot 0) = \sin(\omega_k T) = 0$ and $\cos(\omega_k \cdot 0) = \cos(\omega_k T) = 1$, the condition translates to $u_0 + \sum_k u_{2k} = 0$. Second, we will require the total integral of the pulse to be zero, which is satisfied if $u_0 = 0$. Putting these together, the conditions are:

$$u_0 = 0 \qquad \sum_k u_{2k} = 0 \tag{3.21}$$

### 3.6 Implementation and Performance of the Optimization Process

The algorithm we have just sketched has been implemented in a numerical code. We have used python. The software development has been done using the *qocttools* package [14], a plugin for *qutip* developed by Dr. Alberto Castro. We have carried out two tasks: First, we extended the *qocttools* package to allow for multitarget optimizations and then we used it to simulate the $GdW_{30}$ molecule, obtain the pulses, and try and find the optimal parameters for them to achieve the best fidelities, as will be shown later.

Before moving on to describing our results in section 4, a word about the protocol followed to perform the optimizations. Our first approach consisted in running optimizations using Lindblad equation, initializing the parameters at random, and trying different seeds if the results were not good enough. However, the starting point was always too far from the optimal pulse, so we always obtained low fidelities. For that reason, we decided to split the optimization into two

steps. We are going to explain the protocol in depth for the notation to be clear from now on.

First, starting from a random guess $u$, we execute the optimization considering no dissipation (i.e. using Schrödinger's equation). We will refer to this as "Schrödinger optimization", and its resulting pulses and fidelities will be colored in blue in the following graphs. The fidelity of the optimal pulse thus obtained is not realistic (since it implies assuming $T_1, T_2 \to \infty$) as we will check by calculating its real fidelity using Lindblad equation. This check will be called "Lindblad propagation" and its results will be drawn in orange. As we will see, this real fidelity is lower than the one we had first obtained. Nevertheless, in spite of being quite unrealistic, the Schrödinger optimization provides us with a good inital guess for the next step: a second optimization, this time with Lindblad's equation. We will refer to this as "Lindblad optimization", and will draw its resulting pulses and fidelities in green.

We display an example of the convergence of the optimization in figure 4. This is a case in which the objective is a state-to-state transition ($|6\rangle \to |7\rangle$). One can see the convergence history of the first optimization. It achieves a high fidelity, but it is not realistic, as it ignores the decoherence. The convergence history of the second optimization departs from the pulse obtained in the previous case. It converges more slowly towards the optimal pulse, whose fidelity is not as high.

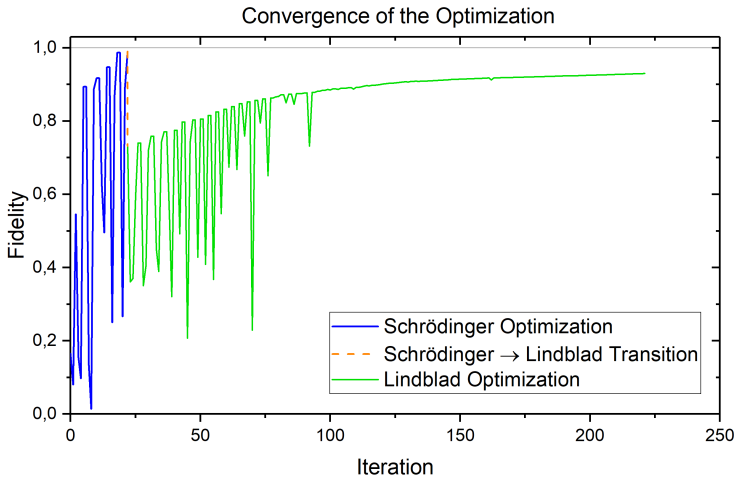



Figure 4: Convergence of the optimization for the transition $|\psi_0\rangle = |6\rangle$ and $|\psi_{\text{target}}\rangle = |7\rangle$. The duration has been set to T=44 ns. The first optimization (−) is performed using Schrödinger equation. The second one (−) starts from the result of the first one, but takes into account decoherence.

There is another important issue about the optimization problem. As the simulations have confirmed, the landscape of fidelities as a function of the parameters is not smooth at all, but complex and spiky. For that reason, it is very likely to fall into local maxima, that is, pulses whose fidelities are bigger than the ones surrounding them in the space of parameters, but not the biggest. That is why our actual strategy has included the use of several ($\sim 50$) Schrödinger optimizations, each one started with a different random seed for the initialization of the parameters. The best of them has been used as initial guess for the second step.

# 4 Results

It is time to apply the OCT theory to a real task. In this work, we focus on a three-qubit gate that is both a simple and a useful example: the Toffoli gate. Furthermore, it is key for the "correction" stage of the algorithm in figure 3. This three-qubit gate ($8 \times 8$) consists in a NOT gate controlled by two ancillas. In case the state of both ancillas is $|1\rangle$, the NOT operation will be executed on the target qubit. Let us recall its corresponding unitary matrix:

$$\mathrm{U_{TFF}} \equiv \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & 0 & 1 & \\ & & & 1 & 0 & \end{bmatrix} \begin{matrix} |0\rangle \\ \vdots \\ |5\rangle \\ |6\rangle \\ |7\rangle \end{matrix} \tag{4.1}$$

As we can see, this unitary causes a transition between the states $|6\rangle \leftrightarrow |7\rangle$ and leaves every other state of the basis $(|0\rangle, ..., |5\rangle)$ unchanged.

## 4.1 State-to-State Transitions

Before moving to the our actual objective, the optimization of the Toffoli gate, let us discuss a simpler problem, which is the optimization of the $(|6\rangle \leftrightarrow |7\rangle)$ transition, instead of the whole unitary $\mathrm{U_{target}} = \mathrm{U_{TFF}}$. Obviously, optimizing this transition is not the full story for the gate, since in the latter the rest of the states must remain unchanged, whereas, if we optimize only this transition, we do not care about them. However, this simpler task will allow us to understand some important aspects of our algorithm and its underlying physics. In this case, the target function is:

$$G(u) = F(\rho[u](T)) = \mathrm{Tr}\left[\rho_{\mathrm{target}} \cdot \rho[u](T)\right] = \langle 7|\rho(T)|7\rangle\,, \tag{4.2}$$

Notice that we use no multitarget strategy here. This target function only guarantees that the optimal pulse achieves both $|6\rangle \to |7\rangle$ and $|7\rangle \to |6\rangle$ transitions.

Our first calculations were carried out considering a realistic coherence time, that is $T_2 = 500$ ns, with the durations $T$ of the pulse ranging from 2 to 60 ns. An example of the obtained results is shown in figure 5. As one should expect, the fidelity of the Lindblad propagation (orange curve in the right panel) decreases with time with respect to the Schrödinger optimization (blue), due to the external noise that the latter does not account for. This is a clear picture of the effects of the dephasing encoded within $T_2$. Furthermore, since $T \ll T_2$, these two curves have very similar shapes. The second (Lindblad) optimization is able to find a pulse that performs better than the previous (Schrödinger) one in presence of decoherence (the green curve in the right panel performs better than the orange one). Notice that, in contrast to the slowly-growing fidelity of the Schrödinger pulse, the one associated to the Lindblad optimization is almost a step function, and there is a certain part of the pulse responsible for that (squared in red).

Combining the results we have obtained for several durations $T$, we observe a very particular behaviour of the fidelities, as shown in figure 6. For short durations, the three curves behave in similar ways, because the effects of noise are not important yet. As $T$ increases, the noiseless

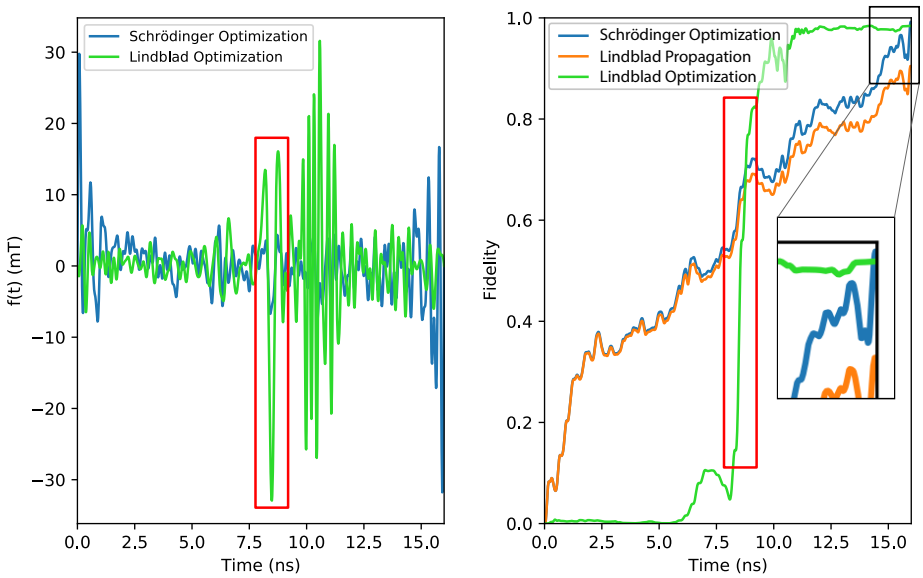



Figure 5: *Left*: Optimal pulses obtained for the $|6\rangle \rightarrow |7\rangle$ transition, for $T = 16$ ns and $H_{\max} = 1$ mT. In blue and green, the optimal pulses obtained with Schrödinger and Lindblad equations, respectively. *Right*: Their effect on the fidelity, as a function of time.

fidelity of the Schrödinger optimization (•) saturates to 1, while its real fidelity (•) starts decreasing and falls down to much lower values. This is the difference between unitary and dissipative evolutions. However, we can see how the second optimization (•) clearly outplays the first one when we take into account dissipation (•). One could think that the optimal pulses found with the Schrödinger propagation could not have the expected fidelity (as the orange curve clearly shows) but would still be the best in any case. If that were true, the implementation of Lindblad equation in our program would only have been useful for giving more realistic expectations on the performance of the pulses. However, we can see how the second optimization finds a pulse with a substantially better fidelity than the one we had before. This result justifies all the work, since we have found the way to perform high fidelity operations in the presence of noise.

The valleys that appear in some regions of the in the blue and green curves are due to local maxima in the optimization landscape. We have tried to dodge them, but it is quite difficult[2]. The fluctuations in the orange curve are not due to local maxima, since these points are not obtained optimizing, but propagating a previously optimized pulse. One should not have expected a smooth curve. For some durations, it is possible that some levels are more populated, just because $T$ could be a multiple $2\pi/\omega_j$, where $\omega_j$ is the frequency of some of the transitions of the system.

We also want to study the performance of the algorithm and the results obtained for durations $T$ that represent a big fraction of $T_2$. As our program cannot simulate very long durations, we repeat the previous executions for a shorter $T_2$ (40 ns), that is, a much stronger dephasing.

[2]For the blue curve, we have run 80 seeds and found no progress. For the green one, we haven't had enough time to run a better search, since every Lindblad optimization for those durations may take up to 22 hours, starting from pulses that must be previously calculated too.

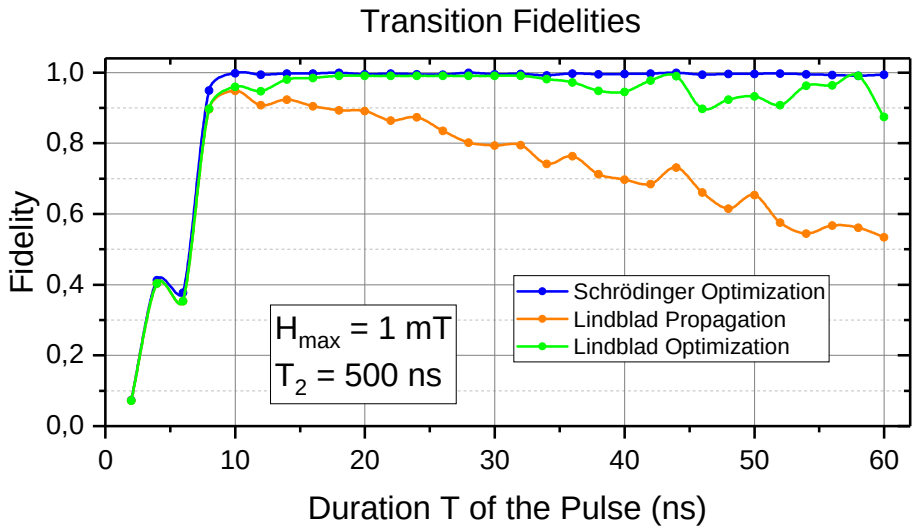



Figure 6: Transition fidelities up to $T = 60$ ns. The only calculated values are the dotted ones, the solid lines are spline interpolations for the reader to better understand the graph. (●) The fidelity of the Schrödinger optimization saturates at its maximum value at about $T = 10$ ns, whereas the noisy propagation of those pulses (●) starts decreasing around that same duration. (●) The result of the second optimization gives much better results than the first one, reaching fidelities almost as high as the ones achieved by the first pulse in the absence of noise.

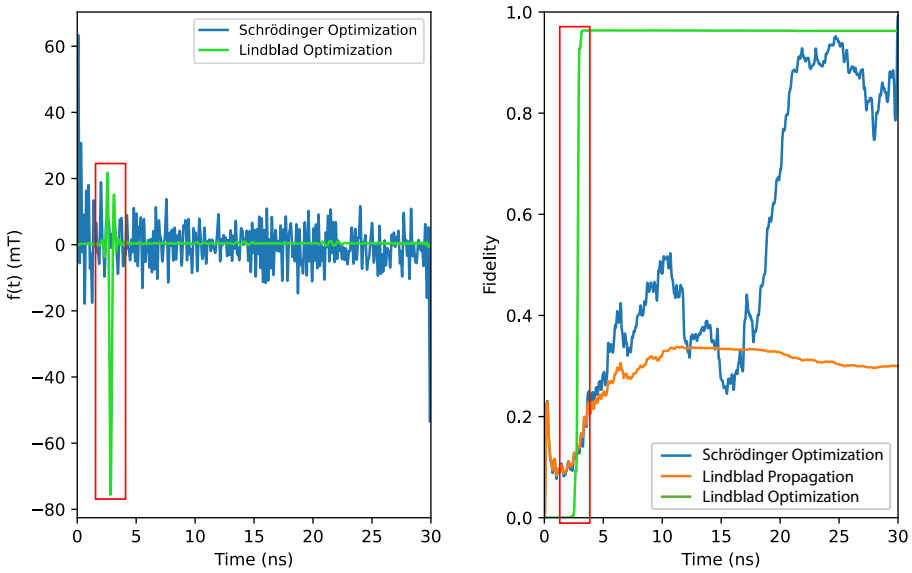



Figure 7: Optimal pulses $T = 30$ ns, $H_{\max} = 1$ mT and $T_2 = 40$ ns, together with their effect on the real-time fidelity.

The results for $T = 30$ ns are shown in figure 7. In this case, the fidelity of the noisy propagation of the pulse optimized with Schrödinger equation does not preserve the shape of the noiseless curve at all, because the decoherence is too strong. Moreover, the result of Lindblad optimization is now even steeper than it was for $T_2 = 500$ ns. We can see how the optimal pulse only takes relevant values in a small time interval around $t \approx 3$ ns, which once again causes a step-function behaviour. An important question arises here, concerning why the second optimization tends to return pulses that are concentrated in time and whose effect is a step function. The explanation is tightly related to the nature of $T_2$. As dephasing only affects the non-diagonal terms of $\rho$, the only states that are affected by $T_2$ are the superpositions of eigenstates of the Hamiltonian. If the state of the system is not superposed, it does not suffer dephasing. That is why the optimal pulses are those that yield a step-function behaviour. The quickest the transition, the least noisy the evolution. After the strong part of the pulse, $\rho \approx |7\rangle\langle 7|$, so the effect of dephasing is neigligible, and that is why the fidelity *plateau* for $t > 5$ ns does not decay over time. This also explains why the orange curve on the right shows that much dephasing compared to the green one. As the Schrödinger optimization does not consider $T_2$, its optimal pulse is extended over time, and the evolution of its fidelity is not a step function at all. Therefore, when we simulate its effect with Lindblad's equation, it suffers very much from dephasing.

Combining results for $T \in [0, 60]$ ns, we obtain figure 8. One can clearly see how the real fidelity of the pulses obtained from the Schrödinger optimization (●) is a competition between the effect of the pulse (●) and the dissipation due to $T_2$ (–). For short times, when dephasing is not important yet, the orange curve is close to the blue one. However, as time runs, decoherence increases and the separation becomes bigger and bigger.

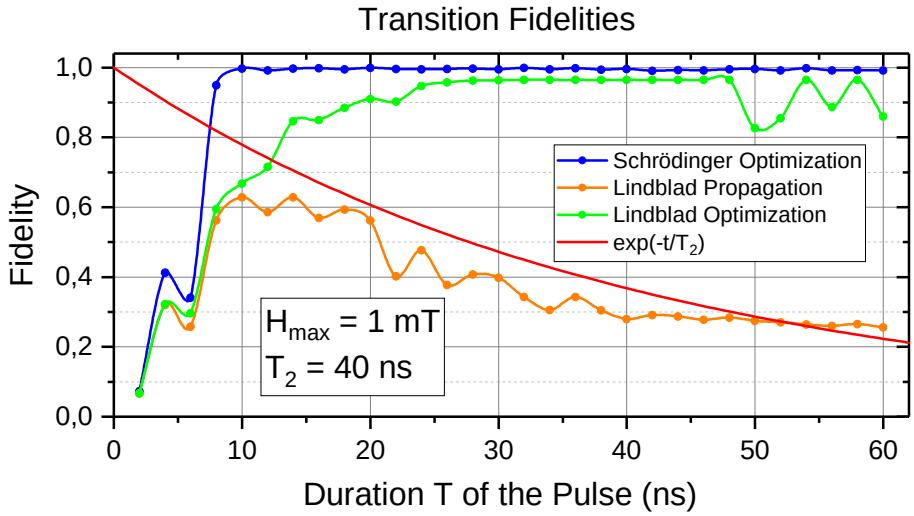



Figure 8: Transition fidelities for durations up to $T = 60$ ns, this time for $T_2 = 40$ ns. In this case, we are able to simulate values of $T$ that are not small compared with $T_2$. This will give us a better understanding of what is happening.

After obtaining the result of this competition in the orange curve, the second step of the optimization returns the green one which is, once again, significantly better than the orange one. The Schrödinger pulses are extended over time, while the Lindblad ones are concentrated

around small time intervals, and that is why the red curve acts as an order-of-magnitude barrier only for the first one. In any case, the reader must keep in mind that these points are the result of the propagation of different pulses each. They are not the steps of the time evolution of the same system, because the hamiltonian varies from one to another. That is why some phenomena such as the fact that the orange curve surpasses the red one for $T > 50$ ns are not especially problematic.

### 4.2 Unitaries: Toffoli Gate

As we have seen in figure 6, a simple $|6\rangle \leftrightarrow |7\rangle$ transition is not equivalent to a Toffoli gate. The second is a unitary, and therefore our optimization must be more restrictive. We have to use the whole multitarget fidelity from equation (3.20), with the $d+1 = 9$ states from [10] and $\mathrm{U}_{\mathrm{target}} = \mathrm{U}_{\mathrm{TFF}}$. Two examples of the results of this optimization with the realistic $T_2 = 500$ ns are shown in figures 9 and 10, and most of the comments we made in the previous section remain valid.

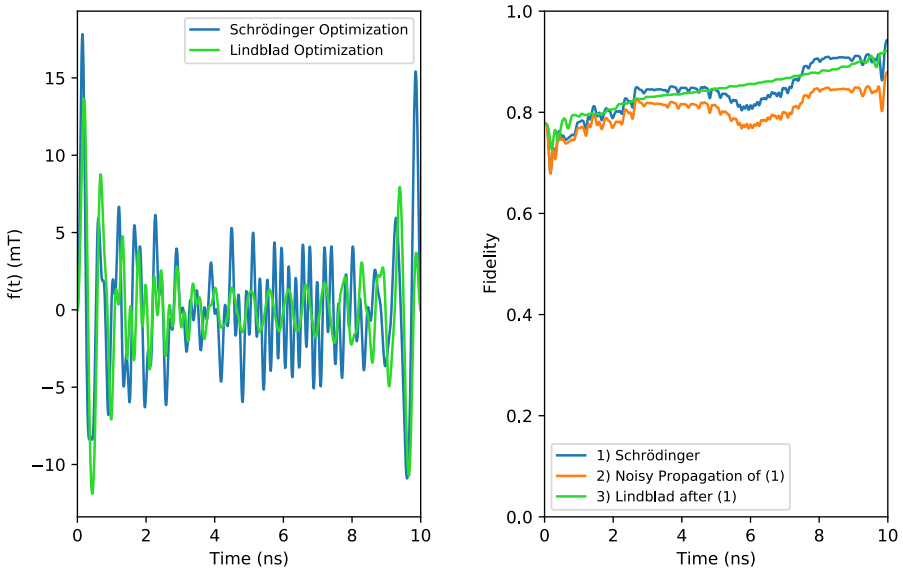



Figure 9: Multitarget optimization of a Toffoli gate for $T = 10$ ns and $H_{\max} = 1$ mT.

Considering the shapes of these pulses and real-time fidelities, and many others that are not shown here, we have observed that for shorter values of $T$ the pulses tend to force a smooth evolution in the fidelity, as in figure 9, whereas for longer ones, as the dissipation becomes stronger, a step-by-step behaviour arises, as in figure 10. In the former case, the pulse is extended over time. Meanwhile, although the latter pulse is not as clean as the one shown in figure 7, it also has a shape where there seems to be a wave packet responsible for each step. For durations $T$ that are not extremely short compared to $T_2$, the optimization tends to find pulses that drive the initial states to their respective targets one by one and in sudden jumps, in order to minimize dephasing.

Putting together the fidelities we have calculated for $T \in [0, 20]$ ns, $T_2 = 500$ ns and two different values of $H_{\max}$, we obtain figure 11. The need to propagate nine states instead of one made the

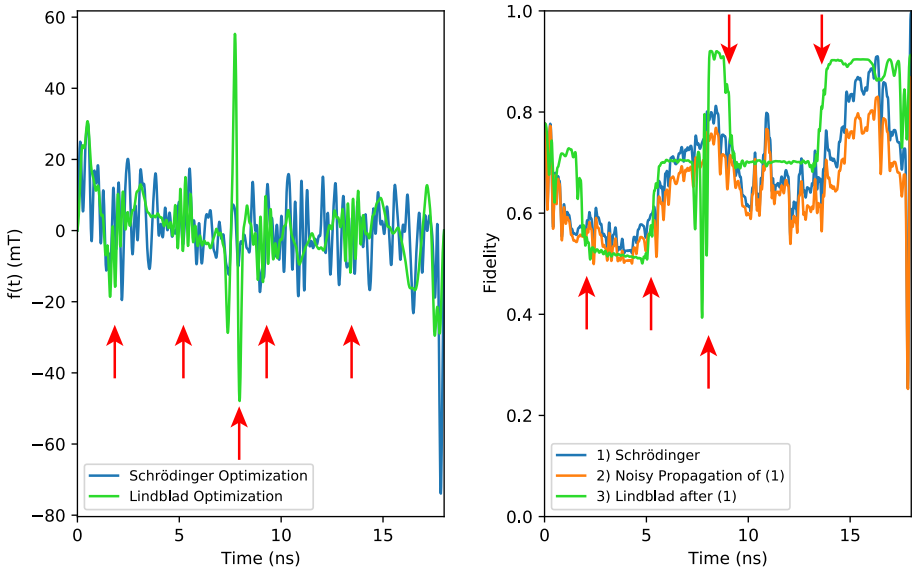



Figure 10: Multitarget optimization of a Toffoli gate for $T = 18$ ns and $H_{\max} = 2$ mT. The red arrows on the left point at five wave packets on the Lindblad pulse that seem to be responsible for the five steps pointed in the green curve on the right.

executions nine times longer, so we only could simulate durations up to $T = 20$ ns in affordable times.

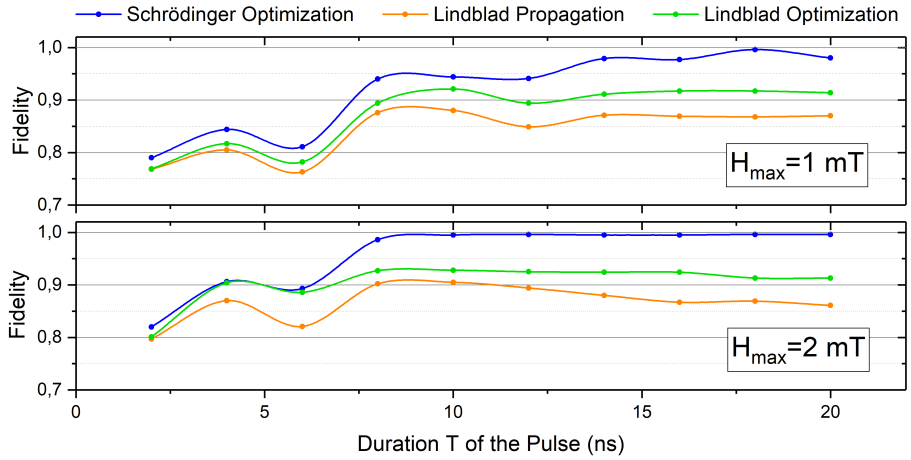



Figure 11: Fidelities for the Toffoli gate as a function of the duration of the pulse, for two different values of $H_{\max}$. There is the same local-maxima problem than before, and the results could be even better if we could deal with it, but we will not it because it seems to be quite complex.

Once again, we can see how the second optimization finds better pulses than the first one, for every duration $T$ of the pulse, and for both values of $H_{\max}$. This demonstrates that the implementation of high-fidelity unitaries on molecular spin qudits is achievable, although we will make several necessary clarifications later.

Another observation we can make is about the time $T$ where the blue curve saturates for

each $H_{\max}$. While for the smaller one it approaches 1 erratically, the bigger one clearly finds its optimal performance around $T \approx 10$ ns. From that point on, the noiseless fidelity does not increase, while the orange curve starts decreasing. If we could not execute the second optimization, the optimal duration of the pulse would be that. Nevertheless, the green curves show a better performance, and they do not have such a clear decreasing tendency. It would be interesting to simulate longer durations to check for the global behaviour, but the efficiency of our algorithm is not good enough to do that in affordable times yet.

# 5 Conclusions and Further Work

The algorithm that we have developed in this thesis work has found the way to implement high-fidelity transitions and unitaries. The results of the optimizations executed on top of the Lindblad equation clearly outperform the calculations carried out considering only Schrödinger's equation. Furthermore, our programs have not only generated pulses capable of dodging dephasing. In some cases, the generated pulses taught us something we had not predicted: sequences of sudden pulses and step-by-step evolutions of the fidelity are especially effective to avoid the effects of $T_2$.

Moreover, the algorithm that we have developed is versatile and flexible, for two reasons:

First, the implementation of high-fidelity operations is not restricted to simple logic gates. The input $\mathrm{U}_{\mathrm{target}}$ of our program can be any unitary, that is, a logic gate or the product of several. The decomposition of algorithms into sequences of logic gates has been clasically done to simplify the understanding of logic operations, and because some kinds of quantum hardware are specifically designed for the implementation of certain logic gates that constitute a universal set[3] to be especially simple. However, when it comes to our molecules, it may be faster to work directly with unitaries instead of gates. Further research might shed light on this. As our calculations admit the product of an arbitrary number of gates as a target unitary, each stage of an algorithm can be run with a single pulse. For instance, the error correction code in figure 3 could be executed with just two pulses: one for the encoding, and another for decoding and correction. In principle, this could improve the velocity of any quantum circuit, reducing decoherence. It would be interesting to test this methodology by trying to replicate figure 4 from [3], to see if one can obtain better results.

The second reason that makes our program flexible is that it is not tied to specific models: $\hat{H}_0$ and $\hat{V}$ can be easily modified, and therefore its applications range beyond magnetic molecules. Any system that can be described as a static Hamiltonian plus a perturbation can be optimized as the physical support for a quantum computer with this algorithm.

After these results, there are several doors that remain open for further research.

[3] A logic gate or set of logic gates is called "universal" if any logic operation can be decomposed into a sequence of the operations of this set. An example of classical universal gates are the NAND and the NOR gates, which are universal each. As for quantum gates, a set containing the CNOT and unitary one-qubit operations is universal.

First, to be even more realistic, we need to introduce relaxation, that is, $T_1$. Although it is bigger than $T_2$ and its effects become noticeable after longer times, the consequences of relaxation are different than the ones of dephasing. The plateaus in the step functions that we have obtained are flat because dephasing does not affect the eigenstates of the bare Hamiltonian, but relaxation does. If we take it into account, the fidelities would decay after the step, so the optimal pulses that we have found so far might not be that effective when considering both sources of noise. We may find other shapes of pulses in that case.

Second, we must search for a deeper physical understanding of the curves of the time evolution of the fidelity. It could help us simplify the pulses, design better target functions or reduce the execution times. If the best way to implement gates is as a discrete set of sudden transitions, maybe it would be possible to "cut and slice" the pulses, and put the ones that are responsible for each step (red arrows) together, supressing the parts of the pulse that are apparently negligible. An example of this idea can be found in figure 7: If the pulse had stopped at $t \approx 5\ ns$, the fidelity would have been the same than the obtained at the final time, but the execution would have been six times shorter.

As for the experimental implementation of our results, there is an important problem yet to be addressed. We have obtained high fidelities, but the amplitude of our pulses is usually about several tens of mT, and the instrumentation is limited. As we explained in section 3.5, the reduction of the amplitude constraints for the magnetic field is a delicate issue, since it must combine feasability with efficience. This reduction would demand longer execution times for the pulses to achieve high fidelities, so we would have to deal with much more noise. The good news is that the green curves in figures 6, 8 and 11 do not show a clear decreasing tendency. Maybe one can design optimal pulses that achieve high fidelities efficiently for those long execution times. In any case, one obstacle that we have so far is the fact that our algorithm is not able to simulate very long times yet.

To sum up, this work is a first analysis of the feasability of the implementation of fault tolerant calculations on molecular spin qudits, and the results so far are optimistic. Optimal control seems to be able to deal with noise, although there are several important issues that still need to be studied.

# References


[1] Michael A. Nielsen and Isaac L. Chuang. *"Quantum Computation and Quantum Information"*. Cambridge University Press, 2010.

[2] M. D. Jenkins, Yan Duan, B. Diosdado, J. J. García-Ripoll, Alejandro Gaita-Ariño, C. Giménez-Saiz, P. J. Alonso, Eugenio Coronado, and Fernando Luis. "Coherent manipulation of three-qubit states in a molecular single-ion magnet". *Physical Review B*, 95(6):S36.002, feb 2017.

[3] Emilio Macaluso, Marcos Rubín, David Aguilà, Alessandro Chiesa, Leoní A. Barrios, Jesús I. Martínez, Pablo J. Alonso, Olivier Roubeau, Fernando Luis, Guillem Aromí, and Stefano Carretta. "A heterometallic [LnLn'Ln] lanthanide complex as a qubit with embedded quantum error correction". *Chemical Science*, 11(38):10337–10343, aug 2020.

[4] S. Carretta, D. Zueco, A. Chiesa, Á. Gómez-León, and F. Luis. "A perspective on scaling up quantum computation with molecular spins". *Appl. Phys. Lett.*, 118(24):240501, jun 2021.

[5] Robert M. White. *"Quantum Theory of Magnetism : Magnetic Properties of Materials"*. Springer, 2007.

[6] Ángel Rivas and Susana F. Huelga. "Open Quantum Systems. An Introduction". 2011.

[7] Heinz Peter Breuer and Francesco Petruccione. *"The Theory of Open Quantum Systems"*. Oxford University Press, 2007.

[8] Harrison Ball, Michael J. Biercuk, and Michael R. Hush. "Quantum firmware and the quantum computing stack". *Physics Today*, 74(3):28–34, 2021.

[9] John Preskill. "Quantum computing in the NISQ era and beyond". *Quantum*, 2:79, aug 2018.

[10] Michael H. Goerz, Daniel M. Reich, and Christiane P. Koch. "Optimal control theory for a unitary operation under dissipative evolution". *New Journal of Physics*, 16(5):55012, may 2014.

[11] Dieter Kraft. "A software package for sequential quadratic programming". Technical report, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen, 1988.

[12] Dieter Kraft. "Algorithm 733: TOMP–Fortran Modules for Optimal Control Calculations". *ACM Trans. Math. Softw.*, 20(3):262–281, 1994.

[13] Steven G. Johnson. The NLopt nonlinear-optimization package, http://github.com/stevengj/nlopt.

[14] Alberto Castro. qocttools, https://gitlab.com/acbarrigon/qocttools.

[15] W. Wirtinger. "Zur formalen Theorie der Funktionen von mehr komplexen Veränderlichen". *Mathematische Annalen*, 97:357–376, 1927.

# 6 Appendix

## 6.1 Density Operator

There is an alternative formulation of quantum mechanics we have not mentioned before. It is based on the density operator $\rho$. If the state $|\psi\rangle$ of a system is perfectly known, the density operator associated to it is $\rho = |\psi\rangle\langle\psi|$. Using it, one can find equivalent expressions for the evolution and physical properties of the states of the system.

$$i\frac{d}{dt}|\psi\rangle = \hat{H}|\psi\rangle \quad \rightarrow \quad \dot{\rho} = -i\big[\hat{H}, \rho\big] \tag{6.1}$$

$$|\psi(t)\rangle = \mathcal{U}(t, t_0)|\psi(t_0)\rangle \quad \rightarrow \quad \rho(t) = \mathcal{U}(t, t_0) \cdot \rho(t_0) \cdot \mathcal{U}^{\dagger}(t, t_0) \tag{6.2}$$

$$\langle\hat{O}\rangle = \langle\psi|\hat{O}|\psi\rangle \quad \rightarrow \quad \langle\hat{O}\rangle = \mathrm{Tr}\left[\hat{O} \cdot \rho\right] \tag{6.3}$$

where the evolution operator $\mathcal{U}$ takes the form [6, 7]:

$$\mathcal{U}(t, t_0) = \mathcal{T}\exp\left[-i\int_{t_0}^{t}\hat{H}(t')dt'\right] \overset{\frac{\partial\hat{H}}{\partial t}=\hat{0}}{\longrightarrow} \mathcal{U}(t, t_0) = e^{-i\hat{H}(t-t_0)} \tag{6.4}$$

However, if, instead of having a well-known state, we have a set of states $\{|\psi_k\rangle\}$ obeying a probability distribution $\{p_k\}$, the bra/ket representation is not valid anymore, and we will need to use the density operator to describe the state of the system. In that case,

$$\rho = \sum_k p_k|\psi_k\rangle\langle\psi_k| \tag{6.5}$$

and equation (6.3) still holds. A state whose density operator can be written as $\rho = |\psi\rangle\langle\psi|$ is called a "pure" state. Otherwise, it is said to be "mixed". Density operators meet certain mathematical properties:

1. $\rho^{\dagger} = \rho$
2. All density operators are positive semi-definite.
3. $\mathrm{Tr}(\rho) = \sum_k p_k = 1$, since $\{p_k\}$ must be a normalized probability distribution.
4. $\rho^2 = \rho \iff \rho$ is pure. Therefore, we can define the purity of a state as $P(\rho) = Tr(\rho^2) \leq 1$. It will be smaller than 1 unless $\rho$ is pure, in which case it is exactly 1.
5. If the evolution of a state is given by (6.1), its purity is preserved over time:

$$\frac{d}{dt}P(\rho) = \frac{d}{dt}\mathrm{Tr}(\rho^2) = 2\,\mathrm{Tr}(\rho \cdot \dot{\rho}) = -2i\,\mathrm{Tr}\left(\rho\big[\hat{H}, \rho\big]\right) = -2i\left[Tr\big(\rho\hat{H}\rho\big) - Tr\big(\rho\rho\hat{H}\big)\right] = 0$$

6. The Von-Neumann Entropy of a state is $S(\rho) = -\mathrm{Tr}\left[\rho\log(\rho)\right]$. It also gives information about the purity of $\rho$, since $S(\rho_{pure}) = 0$, while $S(\rho_{mixed}) > 0$.

But, even with this formulation, we cannot describe the state of the whole system $S + B$, since the complexity of $B$ is still there. We must find a way to extract some information about our system of interest $S$, in spite of entanglement and statistical mixtures. The way to do this is

using a Partial Trace:

$$\rho_S = \mathrm{Tr}_B(\rho)$$

This is a sum over the degrees of freedom that correspond to the environment, in order to extract statistical information about the system $S$, making some sort of average over the influence of the environment $B$ on $S$. However, this carries a loss of information, since

$$\mathrm{Tr}_B(\rho) \otimes \mathrm{Tr}_S(\rho) \neq \rho$$

unless $\rho$ is a separable state ($\rho = \rho_S \otimes \rho_B$). In that situation, there would be no loss of information after the partial trace because there was no previous entanglement. In any case, we will be interested in $\rho_S$ and, although the purity of $\rho$ will remain constant in time, the purity of $\rho_S$ will not, since the evolution of $\rho_S$ will not be given by (6.1). A discussion about how it actually is may be found in section (2.2).

But even without saying anything about evolution, we can already notice one thing. Due to equation (6.5), the state of our system, $\rho_S$, is a sum of pure states weighted by their probabilities, that is a mean value. Thus, it will evolve towards a thermal equilibrium state, given by statistical mechanics:

$$\lim_{t\to\infty} \rho(t) = \rho_{th}(T \equiv \beta) = \frac{1}{Z} e^{-\beta \hat{H}} \quad , \quad Z = \mathrm{Tr}\left[e^{-\beta \hat{H}}\right] \tag{6.6}$$

This has two main consequences:

- Since the final state will be a function of T for any initial $\rho(t_0)$, the whole evolution will also depend on the temperature of the system.
- As in any every other relaxation towards equilibrium, the entropy of the system will increase, so the evolution will be dissipative.

## 6.2 Optimal Control Theory: A Standard Formulation of the Problem, and Derivation of the Gradient Expression

In this appendix, we present the "typical" formulation of the problem addressed by OCT, and sketch a derivation of the equation for the gradient that we have used. Given a differential system,

$$\begin{cases} \dot{y}(t) = f\left(y(t), u, t\right) \\ y(0) = y_0 \end{cases} \tag{6.7}$$

for the evolution of a function y(t), depending on a set of "control" parameters $u \equiv (u_1, \ldots, u_P)$, and a functional

$$F\left(y(T), u\right) = \int_0^T dt\ \delta(t-T) F\left(y(t), u\right)\ ,$$

we want to optimize $G(u) = F\left(y[u](T), u\right)$, where $y[u](T)$ is the solution to (6.7).The problem reduces, therefore, to finding the maximum of a multivariable function $G(u)$, a problem for which hundreds of methods have been proposed. Many of them require of a means to compute the gradient; in the following, we sketch a proof for the gradient expression that we have used above. Notice that, for our purposes, (6.7) may represent the evolution of any quantum system, being $y$ either $|\psi\rangle$ or $\rho$, and $f$ the dynamical function given by Schrödinger's or Lindblad's equation,

respectively.

The problem can be formulated as the maximization of $F$ subject to the conditions given by equations (6.7). We will have to use the method of Lagrange multipliers. Let

$$J(\lambda, y, u) = F(\lambda, y, u) + L(\lambda, y, u)$$

where $y(t) \in \mathbb{C}^n$ is our system, $\lambda(t) \in \mathbb{C}^n$ is its associated Lagrange multiplier, and

$$L(\lambda, y, u) = -2\mathbb{R}\mathrm{e} \int_0^T dt \cdot \cancel{\lambda^*(t)}\lambda(t) \cdot [\dot{y}(t) - f(\lambda(t), u, t)]$$

Notice that $a \cdot b = \sum_i a_i^* b_i$ is a dot product in $\mathbb{C}^n$. Now we want to take derivatives, and we must bear in mind that $F$, $L$ and $J$ are real functions whose variables are complex functions. Therefore, we should use the so-called Wirtinger derivatives (i.e. "write the functions in terms of both the variable and its complex conjugate, and take the derivative with respect with one of those, treating the other one as if it were independent") [15]. Thus, for example:

$$\frac{\delta L}{\delta \lambda^*(t)} = \frac{\delta}{\delta \lambda^*(t)} \sum_i \left[ -\int_0^T dt\ \lambda_i^*(t) \left(\dot{y}_i - f_i(u, u, t)\right) - \int_0^T dt \left(\dot{y}_i^* - f_i^*(y, u, t)\right) \lambda_i(t) \right] \tag{6.8}$$

$$= -\left[\dot{y} - f(y(t), u, t)\right] \tag{6.9}$$

Now we want to take the derivative of $L$ with respect to $y^*(t)$. We rewrite it first as:

$$L \doteqdot \sum_i \int_0^T dt \left[\dot{\lambda}_i^*(t) + \delta(t)\lambda_i^*(t) - \delta(t-T)\lambda_i^*(t)\right] y_i(t) + \sum_i \int_0^T dt\ \lambda_i^*(t) f_i(y(t), u, t) +$$
$$+ \sum_i \int_0^T dt y_i^*(t) \left[\dot{\lambda}_i(t) + \delta(t)\lambda(t) - \delta(t-T)\lambda_i(t)\right] + \sum_i \int_0^T dt\ f_i^*(y_i(t), u, t)\, \lambda_i(t) 0$$

and we obtain:

$$\frac{\delta L}{\delta y^*(t)} = \frac{\partial f}{\partial y^*}(y(t), u, t)\, \lambda(t) + \dot{\lambda}(t) - \delta(t-T)\lambda(t) + \frac{\partial f^*}{\partial y^*}(y(t), u, t)\, \lambda(t)\, .0$$

Notice that the $\delta(t)\lambda(t)$ term disappears: the reason is that, when taking the functional derivative, the variations of $y$ at $t = 0$ are not allowed, due the to the $y(0) = y_0$ constraint. The definition of the Lagrange multiplier term $J$ is given in such a way that setting its functional derivative with respect to $\lambda$ equal to zero implies the equations of motion for the system:

$$0 = \frac{\delta J}{\delta \lambda^*} = \frac{\delta L}{\delta \lambda^*} = -\left[\dot{y} - f(y(t), u, t)\right] \quad \longrightarrow \quad \dot{y} = f(y(t), u, t)$$

The differential equation that defines $\lambda$ is analogously given by taking the functional derivative

with respect to $y$ and equating it to zero:

$$\begin{aligned}
&\frac{\delta J}{\delta y^*} = \frac{\partial F}{\partial y^*}\left(y(T),u\right)\delta(t-T) + \frac{\delta L}{\delta y^*} \quad \Rightarrow \\
\Rightarrow \quad &\frac{\delta J}{\delta y_i^*} = \dot{\lambda}_i(t)\delta(t-T)\lambda_i(t) + \sum_j \lambda_j^*(t)\cdot\frac{\partial f_j}{\partial y_i^*}\left(y(t),u,t\right) \\
&\qquad + \sum_j \frac{\partial f_j^*}{\partial y_i^*}\left(y(t),u,t\right)\cdot\lambda_j(t) + \frac{\partial F}{\partial y_i^*}\left(y(t),u\right)\delta(t-T)
\end{aligned}$$

Therefore, if the functional derivative of $J$ is equal to 0:

$$\begin{cases} \dot{\lambda}_i(t) = -\sum_j \lambda^*(t)\cdot\frac{\partial f_j}{y_i^*}\left(y[u](t),u,t\right) - \sum_j \frac{\partial f_j^*}{y_i^*}\left(y[u](t),u,t\right)\cdot\lambda(t) \\ \lambda_i(T) = \frac{\partial F}{\partial y_i^*}\left(y[u](t),u\right) \end{cases} \tag{6.10}$$

where we have explicitly written $y[u](t)$ to emphasize that it is the system trajectory that meets the constraint given by (6.7) for parameters $u$. Likewise, we can use the notation $\lambda[u](t)$ to refer to the costate trajectory () for parameters $u$.
Considering that $G(u) = J(\lambda[u], y[u], u)$, now we can finally compute the gradient of G by taking the chain rule:

$$\begin{aligned}
\frac{\partial G}{\partial u} = &\sum_j \int_0^T dt \cancel{\frac{\delta J}{\delta \lambda_j^*[u](t)}} \frac{\partial \lambda_j^*[u](t)}{\partial u} + \sum_j \int_0^T dt \cancel{\frac{\delta J}{\delta \lambda_j[u](t)}} \frac{\partial \lambda_j[u](t)}{\partial u} + \\
&+ \sum_j \int_0^T dt \cancel{\frac{\delta J}{\delta y_j^*[u](t)}} \frac{\partial y_j^*[u](t)}{\partial u} + \sum_j \int_0^T dt \cancel{\frac{\delta J}{\delta y_j[u](t)}} \frac{\partial y_j[u](t)}{\partial u} + \\
&+ \frac{\partial}{\partial u} J(\lambda, y, u)\Big|_{\lambda=\lambda[u], y=y[u]} = \\
&= 2\mathbb{R}\mathrm{e}\int_0^T dt\ \lambda^*(t)\frac{\partial f}{\partial u}\left(y(t),u\right)\Big|_{\lambda=\lambda[u],y=y[u]} + \frac{\partial}{\partial u}F\left(y(T),u\right)\Big|_{y=y[u]}
\end{aligned} \tag{6.11}$$

The previous equation is rather general. Now we will simplify it considering that the differential equation encoding the evolution of a physical system, with or without dissipation, is linear. Our system will take the form:

$$\begin{cases} \dot{y}(t) = \hat{A}(u,t)\cdot y(t) \\ y(0) = y_0 \end{cases} \tag{6.12}$$

We can identify terms with the previous section as follows:

$$f(y,u,t) = \hat{A}(u,t)\cdot y(t) \quad \longrightarrow \quad f^\dagger(y,u,t) = y^\dagger(t)\hat{A}^\dagger(u,t) \equiv f^\dagger(y^\dagger,u,t)$$

$$(6.2) \Rightarrow \begin{cases} \dot{\lambda}(t) = -\hat{A}^\dagger(u,t)\cdot\lambda(t) \\ \lambda(T) = \frac{\partial F}{\partial y^\dagger}\left(y[u](t),u\right) \end{cases} \tag{6.13}$$

$$(6.12) \Rightarrow \frac{\partial G}{\partial u} = 2\mathbb{R}\mathrm{e}\int_0^T dt\ \lambda^\dagger(t)\frac{\partial \hat{A}}{\partial u}y(t)\Big|_{\lambda=\lambda[u],y=y[u]} + \frac{\partial F}{\partial u}\left(y(T),u\right)\Big|_{y=y[u]} \tag{6.14}$$

To further particularize OCT problems for quantum systems, we will just have to specify the meaning of $y(t)$ and $\hat{A}$, and introduce specific expressions of $F$ into the equations.

In case our system follows a Lindbladian evolution, the deduction of the expression for the costate boundary condition is not straightforward. We will use (6.2), with $F\left(y(t),u\right)=\mathrm{Tr}\left[\rho_{\mathrm{target}}\cdot\rho[u](T)\right]$. First, we must write the content of the trace term-wise:

$$\left[(\rho+\rho^{\dagger})\hat{O}\right]_{ij}=\sum_{m}\left(\rho_{im}+\rho_{mi}^{*}\right)O_{mj}.$$

Now we write the explicit form of the target function in terms of the previous expression:

$$F=\frac{1}{2}\,\mathrm{Tr}\left[(\rho+\rho^{\dagger})\hat{O}\right]=\frac{1}{2}\sum_{k}\left[(\rho+\rho^{\dagger})\hat{O}\right]_{kk}=\frac{1}{2}\sum_{k,m}\left(\rho_{km}+\rho_{mk}^{*}\right)\hat{O}_{mk}.$$

Finally, we differentiate it and find a compact expression

$$\frac{\delta F}{\delta\rho_{ij}^{*}}=\frac{1}{2}\sum_{k,m}\left(\frac{\partial\rho_{km}}{\partial\rho_{ij}^{*}}+\frac{\partial\rho_{mk}^{*}}{\partial\rho_{ij}^{*}}\right)O_{mk}=\frac{1}{2}\sum_{k,m}\left(0+\delta_{im}\delta_{jk}\right)O_{mk}=\frac{1}{2}O_{ij}$$

that is,

$$\lambda(T)=\frac{\delta F}{\delta\rho^{\dagger}}\left(y[u](T),u\right)=\frac{1}{2}\hat{O}. \tag{6.15}$$