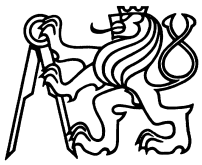
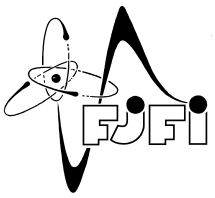


Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering



Quantum Dots and Quantum Optimal Control

Kvantové tečky a kvantové optimální řízení

Master Thesis

Author: **Bc. Ludvík Cigna**

Supervisor: **doc. Ing. Martin Štefaňák, Ph.D.**

Consultant: **Mgr. Jakub Mareček, Ph.D.**

Academic year: 2023/2024

FAKULTA JADERNÁ A FYZIKÁLNĚ INŽENÝRSKÁ

KATEDRA FYZIKY

ZADÁNÍ DIPLOMOVÉ PRÁCE

Akademický rok: 2022/2023

Student:	Bc. Ludvík Cigna
Studijní program:	Kvantové technologie
Název práce: (česky)	Kvantové tečky a kvantové optimální řízení
Název práce: (anglicky)	Quantum Dots and Quantum Optimal Control
Jazyk práce:	Angličtina

Pokyny pro vypracování:

1) Nastudujte základy problematiky kvantových teček, jejich teoretický popis a fyzikální vlastnosti

2) Seznamte se s metodami teorie optimální kontroly, a detailněji prostudujte jejich využití pro řízení kvantových systémů

3) Pokuste se sestavit zjednodušený teoretický model kvantové tečky

4) Prozkoumejte možnosti optimálního řízení kvantové tečky pro dosažení dynamiky využitelné v kvantovém zpracování informace

Doporučená literatura:

[1] S. M. Reimann and M. Manninen, Electronic structure of quantum dots, Rev. Mod. Phys. **74**, 1283 (2002)
[2] R. Hanson et al., Spins in few-electron quantum dots, Rev. Mod. Phys. **78**, 1217 (2007)
[3] J. P. Zwolak and J. M. Taylor, Advances in automation of quantum dot devices control, pre-print arXiv:2112.09362v2 (2022)
[4] A. E. Bryson, Y.-Ch. Ho, Applied Optimal Control, Taylor & Francis, Oxon, 1975
[5] D. D'Alessandro, Introduction to Quantum Control and Dynamics, 2nd edition, Taylor & Francis, Oxon, 2022
[6] G. Dridi, K. Liu, and S. Guérin, Optimal Robust Quantum Control by Inverse Geometric Optimization, Phys. Rev. Lett. **125**, (2020)
[7] L. K. Castelano et al., Optimal control of universal quantum gates in a double quantum dot, Phys. Rev. B **97**, 235301 (2018)

Jméno a pracoviště vedoucího diplomové práce:

doc. Ing. Martin Štefaňák, Ph.D., Katedra fyziky, Fakulta jaderná a fyzikálně inženýrská ČVUT v Praze

Jméno a pracoviště konzultanta diplomové práce:

Mgr. Jakub Mareček, Ph.D., centrum umělé inteligence, Fakulta elektrotechnika ČVUT v Praze

Datum zadání diplomové práce: 01. 03. 2023

Termín odevzdání diplomové práce: 08. 01. 2024

Doba platnosti zadání je dva roky od data zadání.

.....
garant studijního programu

.....
vedoucí katedry

.....
děkan

V Praze dne 01. 03. 2023

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta jaderná a fyzikálně inženýrská

Břehová 7
115 19 Praha 1

PROHLÁŠENÍ

Já, níže podepsaný

Jméno a příjmení studenta:	Ludvík Cigna
Osobní číslo:	478606
Studijní program:	Kvantové technologie
Studijní obor:	
Specializace:	

prohlašuji, že jsem diplomovou práci s názvem:

Kvantové tečky a kvantové optimální řízení

vypracoval samostatně a uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

V Praze dne 8. 1. 2024

Ludvík Cigna

.....

podpis

Acknowledgment:
I would like to thank doc. Ing. Martin Štefaňák, Ph.D. for careful supervision and guidance during the work on my thesis. I would also like to thank Mgr. Jakub Mareček, Ph.D. for consultation and expert advice in the field of polynomial optimization. Finally, I would like to express my gratitude to Assoc. Prof. Tomohiro Otsuka and Motoya Shinozaki, Ph.D. for their support and help not only in the field of quantum dots but also with life in Japan as such.

Název práce:

Kvantové tečky a kvantové optimální řízení

Autor: Ludvík Cigna

Studijní program: Kvantové technologie

Druh práce: Diplomová práce

Vedoucí práce: doc. Ing. Martin Štefaňák, Ph.D., Katedra fyziky, Fakulta jaderná a fyzikálně inženýrská, České vysoké učení technické v Praze

Konzultant: Mgr. Jakub Mareček, Ph.D., Centrum umělé inteligence, Fakulta elektrotechnická, České vysoké učení technické v Praze

Abstrakt: Tato diplomová práce zkoumá optimální kontrolu a interpretaci měření kvantových systémů. Zaměřuje se na kvantové tečky (QDs) a zejména na měření pomocí radiofrekvenční reflektometrie, přičemž představuje nové metodologie pro diskriminaci kvantových stavů. Práce nejprve shrnuje (GaAs) kvantové tečky a následně se zabývá technikami kvantové kontroly jak pro uzavřené, tak otevřené kvantové systémy. Stěžejním bodem této práce je nový přístup k diskriminaci kvantových stavů prostřednictvím identifikace lineárního dynamického systému (LDS) a Kalmanova filtru, využívající jak polynomiální optimalizaci, tak zavedené metody identifikace LDS. Jsou nastíněny dvě hlavní strategie: jedna využívající označené měřené trajektorie a druhá založená na sdruženém shlukování a identifikaci LDS. Práce konstatuje, že ačkoliv navrhované metody dosud nepřekonaly přesnost stávajících technik, mohou slibovat rychlejší rozpoznávání kvantových stavů. Výhledově bude nutné uvážit LDS vyšší dimenze a snížení výpočetní náročnosti pro širší uplatnění.

Klíčová slova: diskriminace kvantového stavu, Kalmanův filtr, kvantové tečky, lineární dynamický systém, optimální kvantová kontrola

Title:

Quantum Dots and Quantum Optimal Control

Author: Ludvík Cigna

Abstract: This master's thesis investigates the precise control and measurement interpretation of quantum systems. It focuses on quantum dots (QDs) and especially the radio-frequency reflectometry measurement, presenting novel methodologies for quantum state discrimination. The thesis first provides an overview of (GaAs) QDs. It then explores quantum control techniques for both closed and open quantum systems. Central to this work is a new approach to quantum state discrimination through linear dynamical system (LDS) identification and Kalman filtering, using both polynomial optimization and established LDS identification methods. Two main strategies are outlined: one using labeled measurement trajectories, and another for joint clustering and LDS identification. The thesis concludes that, while the proposed methods haven't yet surpassed the accuracy of current techniques, there may be a promise for quicker state recognition. Looking ahead, the research suggests exploring higher dimensional LDS and computational improvements for broader applicability.

Key words: Kalman filter, linear dynamical system, optimal quantum control, quantum dots, quantum state discrimination

Contents

1 Introduction **7**

2 Quantum dots overview **9**
2.1 GaAs Quantum Dots . . . 9
2.2 Measuring QDs with radio-frequency reflectometry . . . 11
2.2.1 Basics of radio-frequency reflectometry . . . 11
2.2.2 Application on QDs . . . 15

3 Control of closed quantum systems **18**
3.1 Lie group decomposition . . . 19
3.2 Optimal control of closed quantum systems . . . 20
3.2.1 Pontryagin maximum principle . . . 22
3.3 Globally optimal control of closed quantum systems via polynomial optimization . . . 23

4 Control of open quantum systems **26**
4.1 Open-loop control of open quantum systems . . . 26
4.2 Closed-loop control of open quantum systems . . . 28

5 Quantum state discrimination via LDS identification **31**
5.1 LDS identification and Kalman filtering . . . 31
5.1.1 Standard LDS identification methods . . . 31
5.1.2 LDS identification via polynomial optimization . . . 32
5.1.3 Kalman filter . . . 33
5.2 Quantum state discrimination . . . 34
5.2.1 Discrimination with labeled training trajectories - the algorithm . . . 34
5.2.2 1D data and results . . . 34
5.2.3 Simulated data . . . 36
5.2.4 IQ demodulator setup . . . 37
5.2.5 Results with the artificial IQ data . . . 40
5.3 Discrimination with unlabeled training trajectories . . . 40

6 Conclusion **48**

A Controllability **56**

B Cartan decomposition **58**

C Polynomial optimization **61**

Chapter 1

Introduction

In the realm of current quantum technologies, where noise is a significant factor, the ability to extract maximal information from measurements is essential for ongoing progress. This is especially crucial in the field of quantum computing, where the realization of effective quantum information processing hinges on the precise control of quantum systems [24, 30, 66, 81]. Such precision necessitates highly accurate system identification [8, 14], which, in turn, depends on meticulous measurement and insightful interpretation of measurement data [17, 28, 56].

In this thesis, we review the basics of quantum dots (QDs) which have shown considerable promise as a platform for fault-tolerant quantum computing, owing to their high-fidelity control over coherent states and their compatibility with scalable industrial architectures [91, 79, 25, 33]. We place a special emphasis on radio-frequency (RF) reflectometry [84], a prevalent technique for measuring semiconductor QDs. The review then continues with the summary of quantum control techniques for both closed [24] and open [44] quantum systems.

The main part of this thesis, however, deals with the challenge of accurately determining quantum states based on noisy measurement trajectories. Our attention is particularly drawn to semiconductor quantum dots, although we anticipate that the methodology we propose could be applicable to a broader spectrum of quantum devices. While the standard measurement interpretation techniques are based on finding some optimal threshold value or separating curve [28, 84, 41], we propose a fundamentally different method.

Our approach to quantum state discrimination centers on learning the linear dynamical system (LDS) matrices [4] from measurement trajectories acquired from a specific device. During this learning process, we confront a complex non-convex optimization problem. We deal with it using the recently developed polynomial optimization (POP) framework [2, 94], which has garnered significant interest for its effectiveness in identifying global optima in challenging non-convex scenarios, as well as using more established methods of LDS identification.

We address the problem of state discrimination through two distinct methodologies. Initially, we consider measurement trajectories that are labeled based on whether the QD was initialized in the $|0\rangle$ or $|1\rangle$ state. We then learn the LDS matrices corresponding to each initial state, resulting in two separate sets of matrices. During actual measurements, we process the obtained trajectory with the Kalman filter [40], applying it with each set of system matrices. The final state assignment is based on which set of matrices yields a lower error for the measured trajectory. This approach necessitates precise labeling of the training trajectories, implying the need for already established control over the system.

Alternatively, we explore state discrimination as an unlabeled clustering challenge. This involves the simultaneous task of learning LDS matrices and clustering within the space defined by these matrices.

This method is particularly appealing for experimental devices in their early stages, where adequate control has not yet been achieved, making precise labeling unattainable.

The thesis is organized as follows: in chapter 2 we review the quantum dots and, in particular, QDs based on GaAs. Summary of RF reflectometry then follows. In chapter 3 we give a brief overview of quantum control methods for closed quantum systems, starting with Lie group decomposition and continuing with optimal quantum control based on Pontryagin maximum principle. We conclude this section with an example of globally optimal control based on POP. Very closely related to this chapter are the concepts of controllability and Cartan decomposition with the first being described in Appendix A and the second in Appendix B. In chapter 4 a short introduction to methods of optimal quantum control for open quantum systems is given. This is divided into open-loop and closed-loop techniques. In the later, we focused mainly on the reinforcement learning methods. Finally, in chapter 5 our approach to quantum state discrimination from measurement trajectories is described. This starts with the introduction of all the necessary concepts such as LDS identification and Kalman filter. For these purposes, a brief summary of polynomial optimization is provided in Appendix C. We then introduce the algorithm for the case of labeled measurement trajectories and show the results first for 1D data from a real device and then for the artificial IQ-plane data. The later goes along with the description of the artificial data and the IQ demodulator setup. We conclude the chapter with our proposal for an algorithm for quantum state discrimination in the case of unlabeled trajectories. All codes can be found at: `https://github.com/ludvicheck/QD_LDS_readout`.

Chapter 2

Quantum dots overview

Due to advances in semiconductor physics and technology, it has become possible to isolate a single electron [46]. This isolation is achieved using repelling electric fields that trap the electron, or a few electrons, within a region on the nanometer scale. Such confinement in a potential well leads to a significant outcome: the discretization of energy levels. Therefore, a quantum dot (QD) can be seen as an adjustable artificial atom. This property gives quantum dots a range of interesting physical characteristics and potential uses.

As previously noted, one of the exciting uses of QDs lies in quantum information processing. The capability to control a single spin and employ it as an ideal two-level system for quantum computation – a qubit – is particularly attractive. Loss and DiVincenzo [55] have demonstrated that such a system can meet the five DiVincenzo criteria for universal quantum computation: scalability, initialization, prolonged coherence times, the ability to perform a universal set of gates, and the capability to measure individual qubits.

However, a single electron is never entirely isolated from its surroundings. The most notable interactions affecting the electron include the spin-orbit interaction and the hyperfine interaction between the electron's spin and the nuclear spins in its environment. The intensity of these interactions largely depends on the material used to create the QD. Since our experiment involves (but is not limited to) GaAs quantum dots, we primarily focus on this material. Nevertheless, much of the underlying physics is also relevant for other materials, and we emphasize that our method should be applicable not only to different materials but to other quantum computing platforms as well.

2.1 GaAs Quantum Dots

In Figure (2.1), a schematic of a GaAs quantum dot is presented. A GaAs QD is created at the interface of GaAs and AlGaAs heterostructures, where the AlGaAs layer is doped with Si. This doping process results in free electrons gathering at the interface, forming a two-dimensional electron gas with a relatively low density. On top of the heterostructure, metal gate electrodes are placed. By applying a negative voltage to these electrodes, the resulting electric field depletes certain areas of the electron gas. Then, small regions with lower potential, where the electron gas isn't completely depleted, define the quantum dot.

To measure the properties of the QD, Ohmic contacts are connected to electron gas reservoirs, usually referred to as the source and drain. The most common measurement techniques for QDs are quantum point contact (QPC) or a nearby sensor quantum dot (QDS). Both QPC and QDS are highly sensitive to the charge state of the QD being measured. This sensitivity is shown by the current flowing either

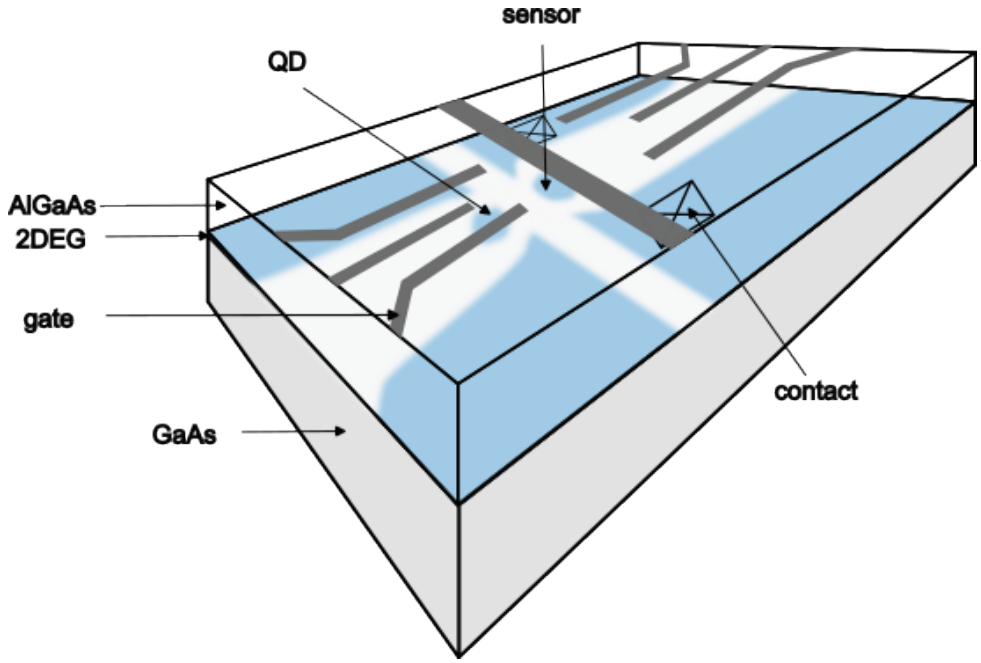


Figure 2.1: Schematic drawing of a GaAs quantum dot.

between the QPC and the QD or through the QDS. We will discuss the measurement process in detail later.

The dominant factor in determining the state of the QD is Coulomb repulsion. Practically, this means that some energy, known as charging energy, is required to add an extra electron to the QD. At low temperatures, where thermal effects are minimal, this leads to the Coulomb blockade phenomenon, which can be understood using the constant interaction model [82]. Let V_s, V_g, and V_d represent the source, gate, and drain voltages, respectively. This model assumes that all Coulomb interactions can be represented by a single constant capacitance $C = C_s + C_g + C_d$ and that the single-electron energy spectrum $\epsilon_i(B)$ (where B is the applied magnetic field) is independent of these interactions. The total energy of the dot with N electrons is then expressed as

$$E(N) = \frac{(-e(N - N_0) + C_s V_s + C_g V_g + C_d V_d)^2}{2C} + \sum_{i=1}^{N} \epsilon_i(B) \tag{2.1}$$

where the sum in the second term covers occupied energy levels and $|e|N_0$ compensates for the positive background charge from the heterostructure's donors.

The electrochemical potential for the transition between an arbitrary N-electron state $|\beta\rangle$ and an $(N-1)$-electron state $|\alpha\rangle$ is defined as

$$\mu_{\alpha\to\beta}(N) = E_\beta(N) - E_\alpha(N-1). \tag{2.2}$$

Defining the charging energy as $E_c = \frac{e^2}{C}$, the formula for a simple transition between the $(N-1)$-electron ground state and the N-electron ground state becomes

$$\mu(N) = (N - N_0 - \frac{1}{2})E_c - \frac{C_s V_s + C_g V_g + C_d V_d}{e} E_c + \epsilon_N. \tag{2.3}$$

Therefore, the energy required to add one more electron to the QD is

$$E_\mathrm{a} = \mu(N+1) - \mu(N) = E_c + \underbrace{\epsilon_{N+1}(B) - \epsilon_N(B)}_{\Delta\epsilon} \tag{2.4}$$

where $\Delta\epsilon$ could be zero if the electron is added to the same spatial orbital.

Transport through the QD can be seen as a process involving a double tunnel barrier. An electron can tunnel from the source to the dot and from the dot to the drain only if the electrochemical potential of the dot falls between the source's and drain's electrochemical potentials, μ_s and μ_d. In other words, applying a bias voltage $V_{sd} = V_s - V_d$ creates an energy difference between the source and drain, allowing electrons to tunnel from the filled states of the source to the empty states of the drain through the QD. This only happens if there's a QD energy state within this energy gap. When no such QD energy level exists within the bias window, it leads to Coulomb blockade, fixing the number of electrons in the QD.

The gradual change of source and drain voltages gives rise to the so-called charge stability diagram. This is shown in figure (2.2), where the yellow spots indicate changes in the number of electrons.

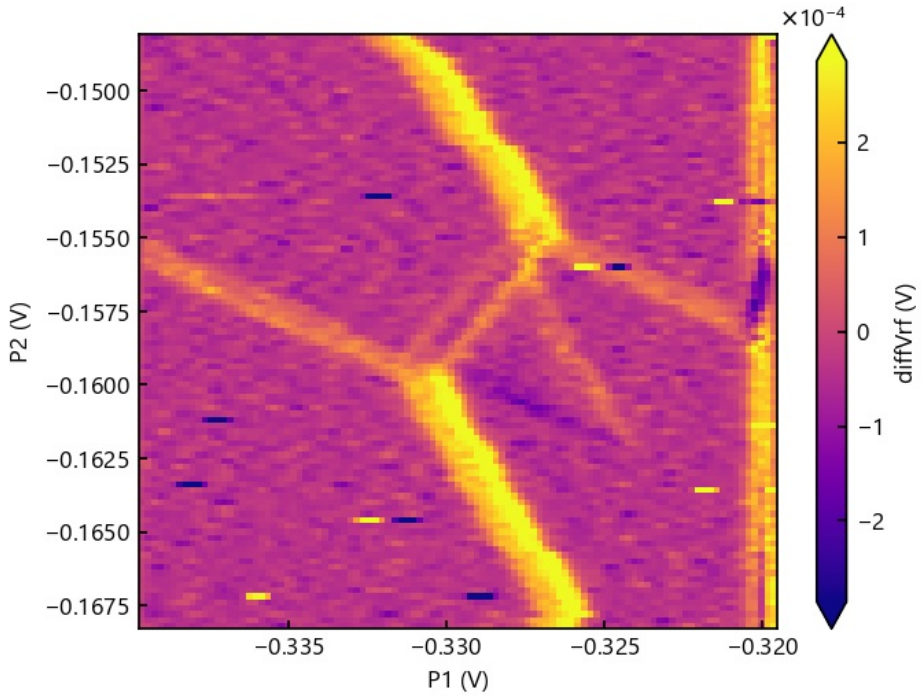


Figure 2.2: Charge stability diagram of the GaAs quantum dot. Data for this diagram can be found in `201125150216.h5` and were measured at the Quantum Device Lab at Tohoku University.

2.2 Measuring QDs with radio-frequency reflectometry

2.2.1 Basics of radio-frequency reflectometry

Radio-frequency reflectometry is a widely used technique for measuring QDs. In this section, we'll cover the basic principles of this method, and later we will delve into its application for single QDs.

A possible setup for the measurement apparatus is illustrated in Figure (2.3). In this figure, the device and its embedding circuit are represented by the impedance Z_{load}. The core idea is to detect changes in the impedance of the device by converting it into a voltage. Z_{load} is measured by sending an input pulse $V_{\text{in}}(t)$ to the device, where it gets reflected. The reflected signal, different from $V_{\text{in}}(t)$, depends on Z_{load}. It is then amplified and demodulated by applying $V_{\text{LO}}(t)$ (often derived from $V_{\text{in}}(t)$) and a low-pass filter.

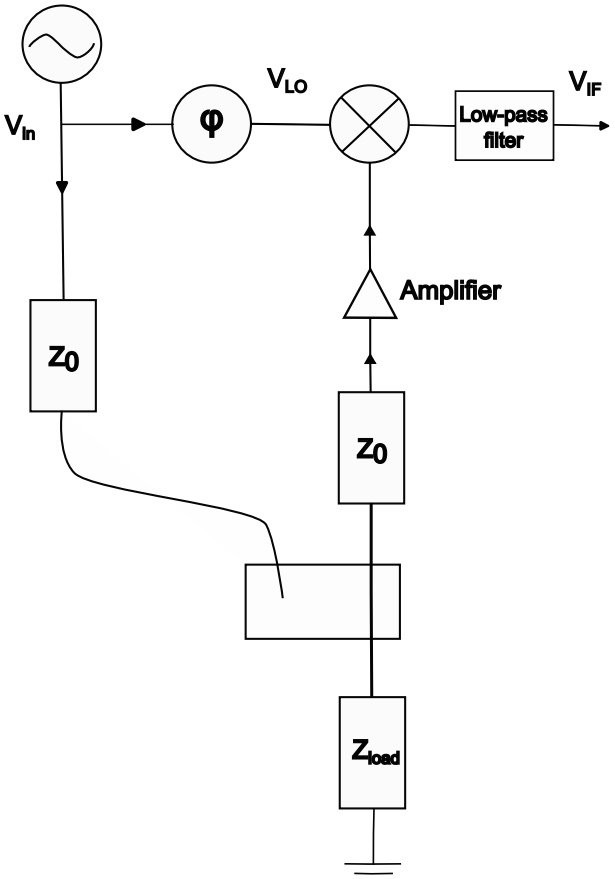


Figure 2.3: Example of RF reflectometry measurement circuit.

From the resulting $V_{\text{IF}}(t)$ the information about Z_{load} is read and interpreted and this is, in fact, the main goal of this thesis.

Following [84], the signal propagation through the entire circuit can be described by voltage $V(x,t)$ and current $I(x,t)$ that satisfy the telegraph equations

$$\frac{\partial V}{\partial x} = -L_u \frac{\partial I}{\partial t} \qquad\qquad \frac{\partial I}{\partial x} = -C_u \frac{\partial V}{\partial t} \tag{2.5}$$

where L_u and C_u are the cable's inductance and capacitance per unit length, respectively. The solution can be expressed as

$$V(x,t) = V_+(t - \sqrt{L_u C_u}x) + V_-(t + \sqrt{L_u C_u}x) \tag{2.6}$$

$$I(x,t) = \frac{1}{Z_0}[V_+(t - \sqrt{L_u C_u}x) - V_-(t + \sqrt{L_u C_u}x)] \tag{2.7}$$

with V_+ and V_- corresponding to solutions propagating forward and backward, respectively, and $Z_0 = \sqrt{\frac{L_u}{C_u}}$ being the characteristic impedance of the transmission line.

Let $V(x,\omega)$ and $I(x,\omega)$ be the Fourier transform of $V(x,t)$ and $I(x,t)$, respectively. When another impedance Z_{load} is connected to the transmission line, the boundary conditions take the form

$$\frac{V(x,\omega)}{I(x,\omega)} = Z_{\text{load}}(\omega). \tag{2.8}$$

It is evident that if $Z_{\text{load}} \neq Z_0$, then V_+ cannot satisfy this boundary condition alone, and therefore, whenever the impedance of the device differs from Z_0, there will also be a reflected wave V_-. Assuming that the boundary is at $x = 0$, the reflection coefficient, measuring the portion of the reflected signal, is given as

$$\Gamma(\omega) \equiv \frac{V_-(0,\omega)}{V_+(0,\omega)} = \frac{Z_{\text{load}}(\omega) - Z_0}{Z_{\text{load}}(\omega) + Z_0}. \tag{2.9}$$

The dependence of $|\Gamma|$ on the ratio $\frac{Z_{\text{load}}}{Z_0}$ is depicted in Figure (2.4). It's clear that the reflection coefficient

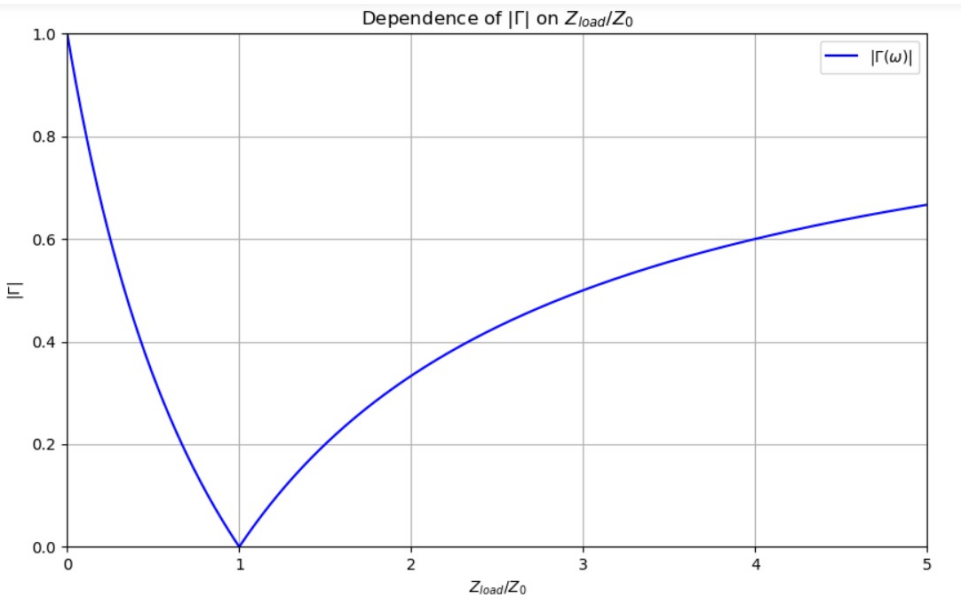


Figure 2.4: Dependence of the reflection coefficient $|\Gamma|$ on $\frac{Z_{\text{load}}}{Z_0}$. The code for this image can be found in `QDs_plots.ipynb`.

Γ is particularly sensitive to changes in Z_{load} when Z_{load} is close to Z_0. However, when the absolute value of Z_{load} is much greater than Z_0, changes in Z_{load} have a minimal impact on Γ. This presents a challenge because the resistance of the quantum device, given by

$$Z_{\text{load}} = \frac{h}{e^2} \approx 25.8\ \text{k}\Omega \tag{2.10}$$

is significantly higher than the typical impedance of the transmission line, which is usually much lower, around

$$Z_0 \approx 50\ \Omega. \tag{2.11}$$

To address this issue, the load is often treated as an electrical resonator approximated by an LCR circuit, consisting of inductance L, capacitance C, and effective resistance R_{eff}, where the details of what is included in R_{eff} can be found in [84]. This approach involves embedding the device in what is known as a tank circuit. The impedance of such a circuit is expressed as

$$Z_{\text{load}}(\omega) = i\omega L - \frac{i}{\omega C} + R_{\text{eff}}. \tag{2.12}$$

The resonance frequency, where the imaginary part of $Z_{\text{load}}(\omega)$ becomes zero, occurs at $\omega = \frac{1}{\sqrt{LC}}$. This significantly influences the reflection coefficient $\Gamma(\omega)$. As demonstrated in Fig. (2.5), there's a

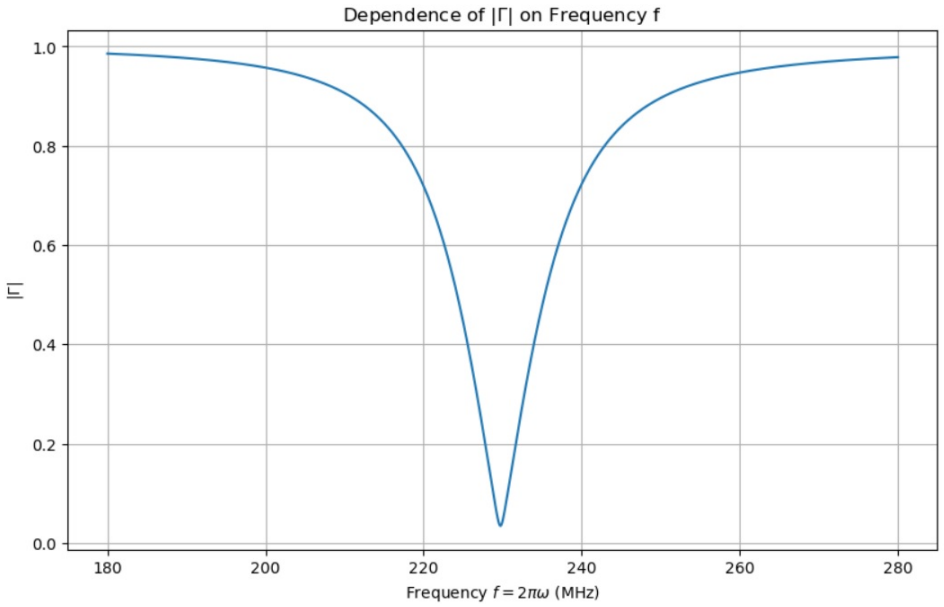


Figure 2.5: Dependence of magnitude of the reflection coefficient $|\Gamma|$ on the frequency. We used values $L = 800$ nH, $C = 0,6$ pF and $R_{\text{eff}} = 46,68\Omega$. The code for this image can be found in `QDs_plots.ipynb`.

notable drop in $|\Gamma|$ at the resonance frequency. At this frequency, any slight variation in the tank circuit parameters results in a significant change in the reflection coefficient, which is the desired outcome. A detailed explanation of how to select the inductance and capacitance of the embedding circuit to match the resistance of the device is found in [84], Section III A.

Next, we describe how the final signal is obtained and extracted using demodulation. For this, we use the IQ-plane representation of the signal. A periodic signal with frequency ω can be represented as

$$V(t) = V_R \cos(\omega t + \phi). \tag{2.13}$$

By applying simple trigonometric identities and defining $V_I = V_R \cos(\phi)$ and $V_Q = V_R \sin(\phi)$, we can express

$$V(t) = V_R \cos(\phi)\cos(\omega t) - V_R \sin(\phi)\sin(\omega t) = V_I \cos(\omega t) - V_Q \sin(\omega t). \tag{2.14}$$

In this way, a periodic signal with a specific amplitude and phase is represented as a point (V_I, V_Q) on the IQ-plane.

Direct measurement of $V(t)$ would require very fast data collection. Therefore, the signal is demodulated to an intermediate-frequency signal $V_{\text{IF}}(t)$ with a much lower frequency but still retaining information about the quadratures V_I and V_Q. This is achieved by mixing $V(t)$ with a signal of the same frequency and phase $V_{\text{LO}} = \cos(\omega t)$. Indeed,

$$V_{\text{IF}}(t) = V(t) \cdot V_{\text{LO}} = \Big(\frac{V_I}{2} + \frac{V_I}{2}\cos(2\omega t)\Big) - \Big(0 + \frac{V_Q}{2}\sin(2\omega t)\Big). \tag{2.15}$$

A low-pass filter is used to remove the rapidly oscillating part, leaving a signal proportional to V_I. Similarly, a signal proportional to V_Q is obtained by mixing $V(t)$ with a phase-shifted demodulating signal $V_{\text{LO}} = -\sin(\omega t)$. Thus, if $V(t)$ and V_{LO} have a phase difference ϕ, then after passing through the low-pass filter, we obtain

$$V(t) \to \frac{V_I}{2}\cos(\phi) - \frac{V_Q}{2}\sin(\phi). \tag{2.16}$$

2.2.2 Application on QDs

As previously mentioned, the most common method for measuring the target QD is to use a quantum point contact (QPC) or another quantum dot, known as a sensor quantum dot (QDS). This section briefly summarizes the process of spin-to-charge conversion and subsequent measurement.

In spin-selective measurement, the two states $|\uparrow\rangle$ and $|\downarrow\rangle$ (sometimes also referred to as $|0\rangle$ and $|1\rangle$) are separated in energy by a strong magnetic field. In a single QD, we can correlate these different energy states with a tunneling event from the QD. Such an event occurs when the higher energy level (in GaAs QDs, it's the $|\downarrow\rangle$ state) is above the Fermi energy of the reservoir, allowing the electron to tunnel into the reservoir. Therefore, to measure the spin state of the QD, we need to shift the potential of the QD so that the $|\downarrow\rangle$ state is above and the $|\uparrow\rangle$ state is below the Fermi level of the reservoir, as shown in Fig. (2.6). This is usually achieved by an illumination signal V_{in}. When the tunneling event occurs, the electron that

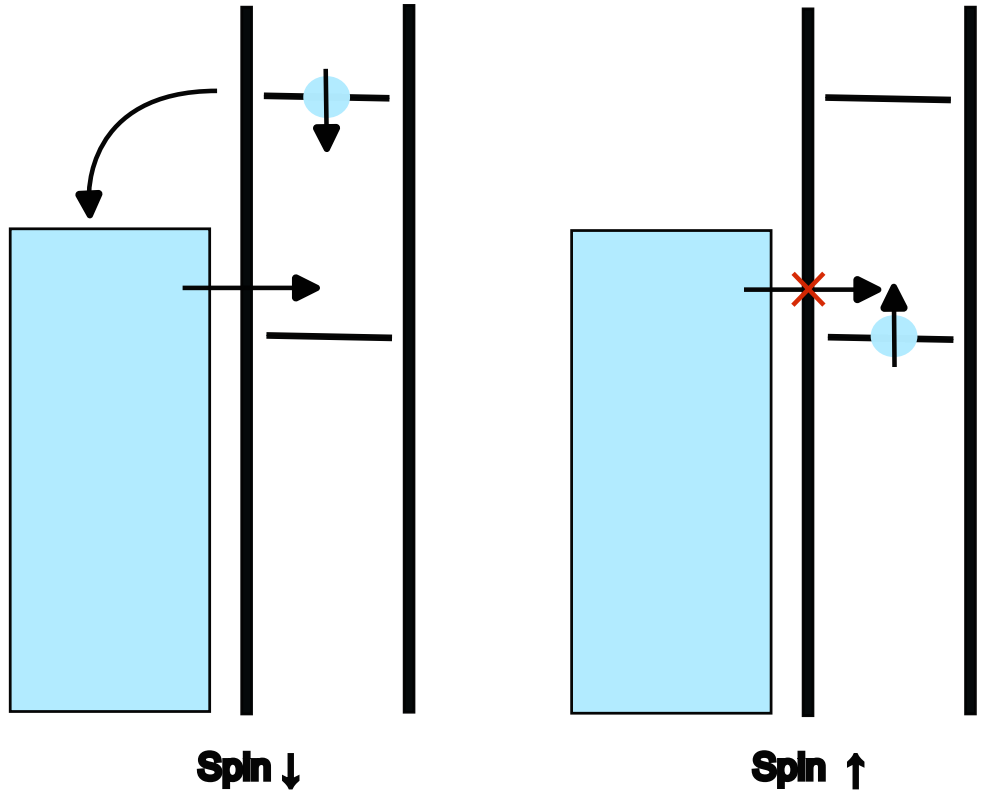


Figure 2.6: Spin to charge conversion.

tunneled out of the dot is quickly replaced by another electron in the $|\uparrow\rangle$ state tunneling into the dot from the reservoir. A fast charge sensor can detect the interval when the QD is temporarily empty. In double quantum dots (DQD), however, the information about the spin state is encoded not in the tunneling effect but in the charge configuration within the DQD.

Now, let's discuss what the QPC or QDS sensors actually measure and how they are tuned. A QPC is created using only one additional depletion gate, positioned so that when a negative voltage V_{QPC} is applied, a narrow channel forms between this gate and the target QD. This channel allows current to flow from one region of the 2D electron gas to another, connected to contact leads for measuring the QPC

conductance G_{QPC}. In the ballistic limit, the conductance is given by

$$G_{QPC} = \frac{e^2}{h} \sum_n f(E_n) g_n \tag{2.17}$$

where f is the Fermi distribution, E_n the energy, and g_n the degeneracy of the nth subband. As the negative voltage is increased, the channel narrows and the conductance decreases. The ideal V_{QPC} is chosen so that the conductance change is most sensitive to V_{QPC}. Hence, any occupation change in the nearby QD (effectively a voltage change) causes a significant conductance change in the QPC.

For a quantum dot sensor (QDS), electron transport is governed by the Coulomb blockade. If the conditions for Coulomb blockade are met, sharp conductance peaks G_{QDS} occur as a function of gate voltage V_{QDS}. These peaks correspond to situations when any energy level of the QDS is within the bias window between two regions of the electron gas. Near the charge degeneracy point V_0, the conductance relation is

$$G_{QDS} = G_{max} \frac{\Delta E}{4k_B T} \cosh^{-2}\Big(\frac{\alpha(V_{QDS} - V_0)}{2k_B T}\Big) \tag{2.18}$$

where G_{max} is the maximum conductance at the degeneracy point, α is the ratio of gate to total capacitance, and ΔE is the energy level spacing. Like the QPC, V_{QDS} is tuned to the point where conductance change is most sensitive, typically on a peak's flank. Any change in the charge configuration of the target QD shifts the peak, causing a significant conductance change. Due to the steeper slope of the QDS curve compared to the QPC, the QDS is usually more sensitive.

We now turn to the most commonly used technique for single-shot measurement and state discrimination. The key goals for single-shot measurement are speed and accuracy. For instance, in quantum error correction protocols, the fidelity of state discrimination should be around 99.9% or higher. Additionally, the measurement must be completed within the qubit coherence time T_1, which is the time it takes for an electron in state $|\downarrow\rangle$ to spontaneously relax to state $|\uparrow\rangle$. However, due to the noise mostly generated by the amplifier, the measurement requires numerous data points (time steps) to accurately determine the state of the QD. These requirements naturally conflict with each other.

A common and straightforward method for state discrimination is to look at the amplitude of the output signal and average it over a specified number of data points. The state is then determined based on whether the average value is above or below a certain threshold value, which differentiates the output voltages for states $|\uparrow\rangle$ and $|\downarrow\rangle$. This approach is clearly demonstrated in Fig. (2.7). Although there are many techniques how to set the optimal measurement time (number of data points to be averaged) and the optimal threshold value [28], the general idea remains the same.

Perhaps the best illustration of the effect of the T_1 process is directly on data from a device from the Quantum Device Lab at Tohoku University. These can be found in the file `201125150755.h5` and can be seen in figure (2.8) , where it is clear how, with a longer averaging time, the T_1 process gradually begins to dominate and even the trajectories, which were correctly assigned to the cluster corresponding to state 1 at a shorter time, are then assigned to state 0. We also add the typical cross section profile for such a histogram at some (optimal) fixed integration time.

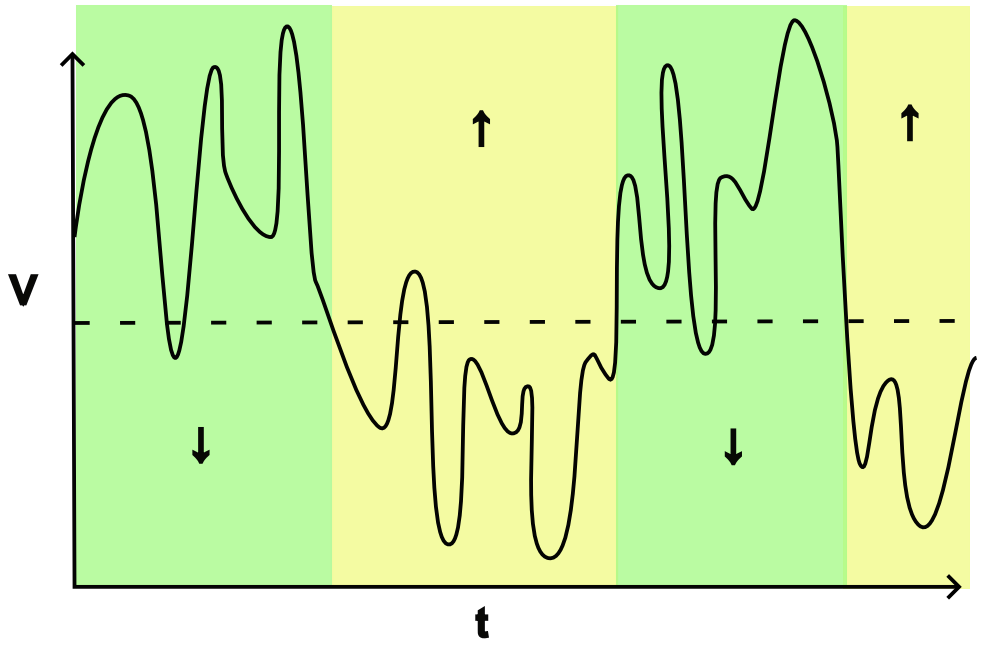
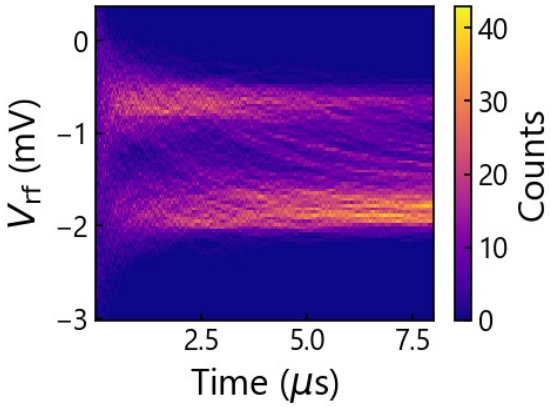
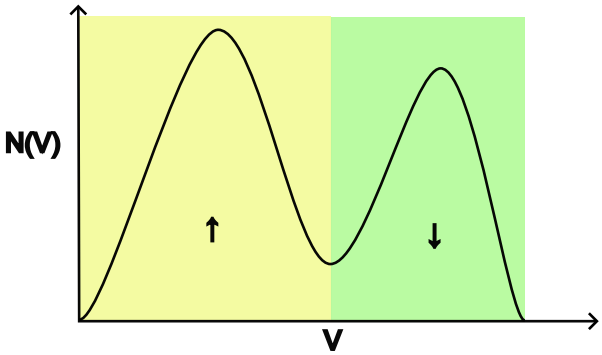


Figure 2.7: A scheme of series of measurements of the QD. The green and yellow background correspond to time when the state is spin down and spin up, respectively.



(a) Time integration histogram for readout signal with gradual change of the integration time. The data can be found in `201125150755.h5`.



(b) Scheme of typical counts for fixed integration time. The data can be found in

Figure 2.8: Time integration and the typical cross section at fixed time.

Chapter 3

Control of closed quantum systems

This chapter is dedicated to provide an overview of the open-loop control techniques for closed quantum systems. Moreover, we consider only finite-dimensional systems, i.e. systems, where the multilevel approximation is relevant. This is almost always the case e.g. for various quantum computing platforms, since the qubits represent an approximation of an ideal two-level quantum system.

In general, finite-dimensional control system has the form

$$\dot{x} = f(t, x, u), \tag{3.1}$$

where x represents the state of the system, $u := u(t)$ represents the (one or more) functions of control fields and $f(t, \cdot, u)$ is the family of vector fields parameterized by t and u. In the quantum theory the dynamics of the system is described by the Schrödinger equation. Let $\boldsymbol{\psi}$ denote the finite-dimensional representation of the ket $|\psi\rangle$. Then, setting $\hbar = 1$, the dynamics is given by

$$\frac{\mathrm{d}\boldsymbol{\psi}}{\mathrm{d}t} = -iH(u(t))\boldsymbol{\psi}. \tag{3.2}$$

Here $H(u(t))$ denotes the Hamiltonian of the system which is Hermitian for every u (hence $-iH(u)$ is skew-Hermitian). In most cases the Hamiltonian is of the form

$$H(u) = H_0 + \sum_k H_k u_k. \tag{3.3}$$

For computational purposes it is often needed to express the problem using only real quantities. This is easily done by writing

$$\boldsymbol{\psi} = \boldsymbol{\psi}_R + i\boldsymbol{\psi}_I \tag{3.4}$$

and

$$-iH(u) = R(u) + iI(u), \tag{3.5}$$

where $\boldsymbol{\psi}_R, \boldsymbol{\psi}_I \in \mathbb{R}^n$ and $R(u), I(u) \in \mathbb{R}^{n\times n}$, $R(u)$ is skew-symmetric and $I(u)$ is symmetric for every u. Separating the real and imaginary parts we obtain two real equations, which can be further rewritten to a single equation by introducing

$$\boldsymbol{x} := \begin{pmatrix} \boldsymbol{\psi}_R \\ \boldsymbol{\psi}_I \end{pmatrix}, \qquad \tilde{H}(u) := \begin{pmatrix} R(u) & -I(u) \\ I(u) & R(u) \end{pmatrix}. \tag{3.6}$$

The equation describing the quantum system using only real quantities then has the form

$$\dot{\boldsymbol{x}} = \tilde{H}(u)\boldsymbol{x}. \tag{3.7}$$

When we consider an ensemble instead of a pure state, the state is described by the density matrix $\rho(t)$ and the dynamics is described by the Liouville's equation

$$\dot{\rho}(t) = [-iH(u(t)), \rho]. \tag{3.8}$$

The solution has the form

$$\rho(t) = U(t)\rho(0)U^{\dagger}(t) \tag{3.9}$$

where $U(t)$ is the unitary propagator satisfying

$$\dot{U}(t) = -iH(u)U(t), \qquad U(0) = \mathbb{I}_{n\times n}. \tag{3.10}$$

In fact, relations (3.10) serve as the most common definition of the control system where the propagator $U(t)$, varying on the Lie group of unitary matrices, is the object of control. Obviously, solving the problem for the propagator immediately provides also the solution for the pure state ψ or the ensemble ρ. Posing the problem as the control of the propagator, however, often provides more convenient and natural formulation of the problem, as for example in quantum computing $U(t)$ represents the operation (performed by logic gates) transforming the state ψ in the desired way.

3.1 Lie group decomposition

Lie group decomposition is a tool to analyze dynamics and design quantum control algorithms which is especially useful when the control problem can be decomposed into sub-problems that are easier to solve. Let $\mathcal{L}$ be the Lie algebra and $e^{\mathcal{L}}$ the corresponding Lie group. Let G_j, $j = 1, \dots r$ be the proper subsets of $e^{\mathcal{L}}$ (typically subgroups). Then by decomposition of the Lie group $e^{\mathcal{L}}$ we mean factorization of each element $X \in e^{\mathcal{L}}$ as

$$X = X_r X_{r-1} \dots X_1, \tag{3.11}$$

where $X_j \in G_j$.

This is particularly relevant when we consider $e^{\mathcal{L}}$ being compact (such as $SU(n)$), because then the following lemma holds:

Lemma 1. Let $e^{\mathcal{L}}$ be the compact Lie group and $\{A_1, \dots A_s\}$ the set of generators of the corresponding Lie algebra. Then every element of $e^{\mathcal{L}}$ can be expressed as

$$X = e^{\tilde{A}_r t_r} e^{\tilde{A}_{r-1} t_{r-1}} \dots e^{\tilde{A}_1 t_1} \tag{3.12}$$

for some $t_j \geq 0$ and $\tilde{A}_j \in \{A_1, \dots A_s\}$ for $j = 1, \dots r$.

The proof of this lemma can be found in Appendix D of [24]. Now, consider the definition of the general quantum control problem (3.10) and the common objective to find the control $u(t)$ that steers the unitary X to some desired unitary X_f. Having the Lie group decomposition at hand, we can write

$$X_f = X_{rf} \dots X_{1f}. \tag{3.13}$$

Assuming the existence of u_j driving the X_j from identity to X_{jf}, $j = 1, \dots, r$, it is clear that the resulting control will be piece-wise constant.

In the case of a two-level system the relevant Lie group is *SU(2)* and it leads to the well-known Euler's decomposition

$$X = e^{\tilde{\sigma}_z t_3} e^{\tilde{\sigma}_y t_2} e^{\tilde{\sigma}_z t_1} \tag{3.14}$$

with t_1, t_2, t_3 being the Euler's angles and

$$\tilde{\sigma}_k = \frac{i}{2}\sigma_k, \qquad k = x, y, z \tag{3.15}$$

where σ_k are the Pauli matrices. For this simple example the determination of the angles is straightforward since any matrix from *SU(2)* can be written as

$$X_f = \begin{pmatrix} \cos(\theta)e^{i\phi} & \sin(\theta)e^{i\psi} \\ -\sin(\theta)e^{-i\psi} & \cos(\theta)e^{-i\phi} \end{pmatrix}, \tag{3.16}$$

where $\theta \in [0, \frac{\pi}{2}]$ and $\phi, \psi \in [0, 2\pi)$. As

$$e^{\tilde{\sigma}_z t} = \begin{pmatrix} e^{i\frac{t}{2}} & 0 \\ 0 & e^{-i\frac{t}{2}} \end{pmatrix}, \qquad e^{\tilde{\sigma}_y t} = \begin{pmatrix} \cos\left(\frac{t}{2}\right) & \sin\left(\frac{t}{2}\right) \\ -\sin\left(\frac{t}{2}\right) & \cos\left(\frac{t}{2}\right) \end{pmatrix}, \tag{3.17}$$

choosing

$$t_1 = \phi - \psi, \qquad t_2 = \theta, \qquad t_3 = \phi + \psi \tag{3.18}$$

will give

$$e^{\tilde{\sigma}_z t_3} e^{\tilde{\sigma}_y t_2} e^{\tilde{\sigma}_z t_1} = X_f. \tag{3.19}$$

Going beyond *SU(2)*, quite simple and explicit way to parameterize *SU(n)* is the decomposition into planar rotations described in [60]. Let us denote such a planar rotation as $U_{j,k}(\theta, \beta)$, with $j < k$ denoting the row and column at the intersection of which the elements are

$$\begin{pmatrix} \cos(\theta)e^{i\phi} & -\sin(\theta)e^{-i\beta} \\ \sin(\theta)e^{i\beta} & \cos(\theta) \end{pmatrix},$$

otherwise it is an $n \times n$ identity. The decomposition of any *SU(n)* matrix into planar rotations is given by the following theorem.

Theorem 1. Let

$$D(\alpha_1, \ldots, \alpha_{n-1}) \coloneqq \mathrm{diag}(e^{i\alpha_1}, \ldots, e^{i\alpha_{n-1}}, e^{-i(\sum_{k=1}^{n-1}\alpha_k)}).$$

Then, for any $X \in SU(n)$ there exist parameters $\alpha_1, \ldots \alpha_{n-1}, \theta_1, \ldots, \theta_{\frac{n(n-1)}{2}}, \beta_1, \ldots, \beta_{\frac{n(n-1)}{2}}$, such that

$$X = D(\alpha_1, \ldots, \alpha_{n-1}) U_{1,2}(\theta_1, \beta_1) U_{1,3}(\theta_2, \beta_2) U_{2,3}(\theta_3, \beta_3) \ldots U_{n-1,n}(\theta_{\frac{n(n-1)}{2}}, \beta_{\frac{n(n-1)}{2}}). \tag{3.20}$$

The proof of this theorem is constructive and provides the way of setting the parameters. It proceeds by converting $X \in SU(n)$ to the identity by successively multiplying it by $U_{j,k}$ from the right and finally using the fact that $[U_{j,k}(\theta, \beta)]^{-1} = U_{j,k}(-\theta, \beta)$. The whole proof can be found in [24].

The most general and in quantum mechanics the most ubiquitous Lie group decomposition is the so-called Cartan decomposition. This topic is very broad and cannot be discussed in full here. The most important aspects are, however, discussed in Appendix B.

3.2 Optimal control of closed quantum systems

In this section we introduce some methods from optimal control theory [26, 53] and immediately adjust it to the case of quantum systems. We start with the general formulation of an optimal control problem.

Definition 1. Given a set $\mathcal{X}$ of (state) functions $x : \mathbb{R} \to \mathbb{R}^n$, and a set $\mathcal{U}$ of (control) functions $u : \mathbb{R} \to \mathbb{R}^m$, the objective of the optimal control is to find the functions $x \in \mathcal{X}$ and $u \in \mathcal{U}$ which minimize a cost functional $J : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ and satisfy the dynamical constraint (3.1) almost everywhere.

Such a general definition of a control problem is, however, rarely used in practice. Considering the $\boldsymbol{x}$ and the Hamiltonian as in (3.6) and constrained by the controlled Schrödinger equation (3.2), the three most common equivalent formulations are Mayer's, Lagrange's and Bolza's.

The problem of Mayer emphasizes the final state and/or time and is defined as

$$J(u) \coloneqq \phi(\boldsymbol{x}(T), T) \tag{3.21}$$

with ϕ being a smooth function $\mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$. When the emphasis is solely on the final state, the ϕ is most often defined as

$$\phi(\boldsymbol{x}(T)) = \left\|\boldsymbol{x}(T) - \boldsymbol{x}_f\right\|^2, \tag{3.22}$$

where x_f is some desired final state. If there is an emphasis also on the final time, the time parameter T will appear in the definition of ϕ explicitly.

The problem of Lagrange often gives an emphasis on the minimization of the overall energy used during the control action and the cost function has the form

$$J(u) \coloneqq \int_0^T L(\boldsymbol{x}(t), u(t), t)\mathrm{d}t. \tag{3.23}$$

The combination of the problem Mayer and Lagrange is the problem of Bolza with the cost function defined as

$$J(u) \coloneqq \phi(\boldsymbol{x}(T), T) + \int_0^T L(\boldsymbol{x}(t), u(t), t). \tag{3.24}$$

It is clear, that in the problem of Bolza the cumulative cost increases with time, but the final state still plays an important role.

It can be shown that these three formulations are in fact equivalent as they can be easily transformed from one to another. Take for example the problem of Mayer with the cost function (3.21) and rewrite it as

$$\phi(\boldsymbol{x}(T), T) = \phi(\boldsymbol{x}(0), 0) + \int_0^T \frac{\mathrm{d}}{\mathrm{d}t}\phi(\boldsymbol{x}(t), t)\mathrm{d}t. \tag{3.25}$$

As $\phi(\boldsymbol{x}(0), 0)$ is constant, the cost function to be minimized has the form(3.23) with

$$L(\boldsymbol{x}(t), u(t), t) = \frac{\mathrm{d}}{\mathrm{d}t}\phi(\boldsymbol{x}(t), t), \tag{3.26}$$

which is the problem of Lagrange.

If we consider density matrix instead of the quantum state formulation, the standard cost function may have the form

$$J(u) = \frac{1}{2}\langle O\rangle_\rho + \int_0^T u^2(t)\mathrm{d}t = \frac{1}{2}\operatorname{Tr}\{O\rho\} + \int_0^T u^2(t)\mathrm{d}t \tag{3.27}$$

for some negative definite, Hermitian matrix O.

3.2.1 Pontryagin maximum principle

We will now derive the necessary conditions of optimality under the assumptions that the optimal control exists and that the set of admissible control functions includes at least piecewise continuous functions. Intuitively, if u is the optimal control function with regards to some cost function $J(u)$, then for any other function v from the set of admissible functions it will hold that

$$J(v) - J(u) \geq 0. \tag{3.28}$$

To pose it as a variational problem, we consider strong variations of the optimal control function $u(t)$ on the interval $[0, T]$, i.e. for some fixed $\tau \in (0, T]$ we define $u^\epsilon(t) \coloneqq u(t)$ for $t \in (0, \tau - \epsilon]$ and $t \in (\tau, T]$, and $u^\epsilon(t) \coloneqq v$ for $t \in (\tau - \epsilon, \tau]$ with v being constant. Then the optimality condition has the form

$$J(u^\epsilon) - J(u) \geq 0. \tag{3.29}$$

Such a search over the set of admissible functions is, in principle, infinite-dimensional problem. In the following, we introduce the Pontryagin maximum principle (PMP), which allows us to transform it to finite-dimensional optimization problem. We use the formulation for the problem of Mayer with fixed final time and free final state. Nevertheless, as long as the Lagrangian is not explicitly dependent on t, the resulting conditions can be generalized for the problem of Bolza [26].

Theorem 2 (PMP for the problem of Mayer with fixed final time and free final state)**.** Let u be the optimal control and $\boldsymbol{x}$ the corresponding trajectory solution of (3.1). Then, there exists a nonzero vector $\boldsymbol{\lambda}$ which is a solution of the adjoint equations

$$\dot{\boldsymbol{\lambda}}^T = -\boldsymbol{\lambda}^T \frac{\partial f(\boldsymbol{x}(t), u(t))}{\partial \boldsymbol{x}} \tag{3.30}$$

with terminal condition

$$\boldsymbol{\lambda}^T(T) = -\frac{\partial \phi(\boldsymbol{x}(T))}{\partial \boldsymbol{x}}, \tag{3.31}$$

such that, for almost every $\tau \in (0, T]$, we have

$$\boldsymbol{\lambda}^T(\tau) f(\boldsymbol{x}(\tau), u(\tau)) \geq \boldsymbol{\lambda}^T(\tau) f(\boldsymbol{x}(\tau, v), \tag{3.32}$$

for every v in the set of the admissible values for the control functions. Furthermore, for every $\tau \in [0, T]$,

$$\boldsymbol{\lambda}^T(\tau) f(\boldsymbol{x}(\tau), u(\tau)) = c, \tag{3.33}$$

for a constant c. The vector $\boldsymbol{\lambda}$ is called the costate.

The proof of this theorem can be found in [26]. In practice, the so-called optimal control Hamiltonian is defined as

$$h(\boldsymbol{\lambda}, \boldsymbol{x}, u) \coloneqq \boldsymbol{\lambda}^T f(\boldsymbol{x}, u), \tag{3.34}$$

so that the optimality condition (3.33) can be written as

$$h(\boldsymbol{\lambda}(\tau), \boldsymbol{x}(\tau), u(\tau)) = c. \tag{3.35}$$

With such a definition, the goal is to maximize h over the set of admissible functions, i.e. to choose u such that

$$h(\boldsymbol{\lambda}, \boldsymbol{x}, u) \geq h(\boldsymbol{\lambda}, \boldsymbol{x}, v) \tag{3.36}$$

for every $\boldsymbol{\lambda}, \boldsymbol{x}$ and for every v from the set. This is a finite-dimensional optimization problem over the values of parameters $\boldsymbol{\lambda}, \boldsymbol{x}$ with the resulting control $u = u(\boldsymbol{\lambda}, \boldsymbol{x})$ dependent on these parameters.

This way we obtain the system of equations

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, u(\boldsymbol{\lambda}, \boldsymbol{x})), \tag{3.37}$$

$$\dot{\boldsymbol{\lambda}}^T = -\boldsymbol{\lambda}^T \frac{\partial f(\boldsymbol{x}, u(\boldsymbol{\lambda}, \boldsymbol{x}))}{\partial \boldsymbol{x}} \tag{3.38}$$

with boundary conditions $\boldsymbol{x}(0) = \boldsymbol{x}_0$ and $\boldsymbol{\lambda}^T = -\frac{\partial \phi(\boldsymbol{x}(T))}{\partial \boldsymbol{x}}$, which is a two points boundary value problem. There is usually no way of solving this problem analytically, except for some low-dimensional problems. Hence, various numerical methods have been developed to address this kind of problem, including discretization and iterative methods. Overview of some selected methods can be found in [24].

In case of the quantum optimal control problem defined by the relations (3.7) and (3.21), the optimal control Hamiltonian has the form

$$h(\boldsymbol{\lambda}, \boldsymbol{x}, u) = \boldsymbol{\lambda}^T \tilde{H}(u) \boldsymbol{x}. \tag{3.39}$$

The adjoint equations are

$$\dot{\boldsymbol{\lambda}}^T = -\boldsymbol{\lambda}^T \tilde{H}(u) \quad \overset{\tilde{H} \text{ is skew-symmetric}}{\Longleftrightarrow} \quad \dot{\boldsymbol{\lambda}} = \tilde{H}(u)\boldsymbol{\lambda} \tag{3.40}$$

with the terminal condition $\boldsymbol{\lambda}^T(T) = -\frac{\partial \phi(\boldsymbol{x}(T))}{\partial \boldsymbol{x}}$.

For the density matrix representation, the costate will be represented by matrix Λ satisfying Liouville's equation

$$\dot{\Lambda} = [-iH(u), \Lambda]. \tag{3.41}$$

Since H is skew-Hermitian and $ad_H = [H, \cdot]$ is skew-symmetric, Λ is Hermitian. The optimal control Hamiltonian then can be written as

$$h = \mathrm{Tr}\{\Lambda[-iH(u), \rho]\}. \tag{3.42}$$

For the problem of Mayer with the emphasis on minimizing the expectation value of some observable O at fixed final time T (which for a particular desired final state $\boldsymbol{\psi}_f$ can be defined as $O = \boldsymbol{\psi}_f^T \boldsymbol{\psi}_f$), the cost function has the form $\mathrm{Tr}\{O\rho(T)\}$ and the terminal condition is simply

$$\Lambda(T) = -O. \tag{3.43}$$

The time optimal control of quantum systems is also a very broad topic and won't be included in this thesis. An overview covering both bounded and unbounded control functions can be found in [24].

3.3 Globally optimal control of closed quantum systems via polynomial optimization

In this section we approach general quantum optimal control problem using polynomial optimization (POP) [7]. We decided to include this section for two reasons. First, because POP plays a significant role in the last chapter dealing with the quantum state discrimination. Second, to demonstrate the operator control formalism which hasn't been introduced in detail in the previous sections. Introduction to POP is given in Appendix C.

We consider the setting of operator control (3.10) with the Hamiltonian of the form (3.3). For simplicity, let us consider the Hamiltonian having only one control term, i.e.

$$H(u) = H_0 + H_1 u(t). \tag{3.44}$$

Having the desired target unitary U_f and given the final time T, the common task is to minimize the Frobenius norm of the difference of the final and our control-prepared unitary:

$$\underset{u(t)}{\text{minimize}} \; \|U(T) - U_f\|_F^2 \tag{3.45}$$

$$\text{subject to } \dot{U}(t) = -i(H_0 + H_1 u(t))U(t) \tag{3.46}$$

$$U(0) = \mathbb{I}. \tag{3.47}$$

The resulting unitary generally has the exponential form. Assuming smooth control functions, to formulate the control problem using polynomials, we have to use some kind of expansion. A particularly useful in this case is the so-called Magnus expansion [57]

$$U(T) = \exp\left\{\Omega^{(\infty)}(T)\right\} \qquad \text{with} \qquad \Omega^{(n)} = \sum_{k=1}^{n} \Omega_k \tag{3.48}$$

and $\Omega_k(t)$ being a multiple integral of nested commutators of $-iH(t)$. The first two terms are

$$\Omega_1 = -i \int_0^T \mathrm{d}t_1 H(t_1), \qquad \Omega_2 = \frac{(-i)^2}{2} \int_0^T \int_0^{t_1} \mathrm{d}t_1 \mathrm{d}t_2 [H(t_1), H(t_2)]. \tag{3.49}$$

The main advantage of using the Magnus expansion rather than e.g. the Dyson expansion is that, since all the Ω_k and their sums are in the Lie algebra $u(n)$, the truncation at any order stays in $u(n)$ and, therefore, the approximation of $U(T)$ is unitary.

From the unitary invariance of the Frobenius norm we have

$$\|U(T) - U_f\|_F^2 = \left\| \exp\left\{ \frac{\Omega^{(\infty)}(T)}{2} \right\} - \exp\left\{ -\frac{\Omega^{(\infty)}(T)}{2} \right\} U_f \right\|_F^2. \tag{3.50}$$

Such a matrix exponential can be efficiently represented by Chebyshev polynomials [80]

$$\exp\left\{\alpha \Omega^{(n)}\right\} = \exp\left\{\alpha, \Omega^{(n)}\right\}_\infty, \qquad \exp\left\{\alpha, \Omega^{(n)}\right\}_p = J_0(\alpha)\mathbb{I} + 2 \sum_{k=1}^{p} J_k(\alpha) T_k, \tag{3.51}$$

where $J_k(\alpha)$ is the Bessel function and T_k the matrix valued Chebyshev polynomial defined recurrently as

$$T_0 = \mathbb{I}, \qquad T_1 = \Omega^{(n)}, T_{k+1} = 2\Omega^{(n)} T_k + T_{k-1}. \tag{3.52}$$

What remains is to approximate the smooth control function via polynomial as

$$u(t) = \sum_{k=1}^{m} u_k t^{k-1}, \; \boldsymbol{u} = (u_1, \ldots, u_m). \tag{3.53}$$

It is clear then that $\Omega^{(n)}$ becomes a polynomial in $\boldsymbol{u}$. As also the square of the Frobenius norm of a matrix is a polynomial of its elements, the polynomial formulation of the optimal control problem (3.45) has the form

$$\underset{\boldsymbol{u}}{\text{minimize}} \left\| \exp\left\{ \frac{1}{2}, \Omega^{(n)} \right\}_p - \exp\left\{ \frac{1}{2}, -\Omega^{(n)} \right\}_p U_f \right\|_F^2. \tag{3.54}$$

The constraints given by the Schrödinger equation in (3.45) are now implicit in the Magnus expansion. Hence, this is effectively unconstrained optimization problem.

In the pure-state formalism, the polynomial formulation of the optimal control problem has the form

$$\underset{\boldsymbol{u}}{\text{minimize}} \left\| \exp\left\{ \frac{1}{2}, \Omega^{(n)} \right\}_p |\psi(0)\rangle - \exp\left\{ \frac{1}{2}, -\Omega^{(n)} \right\}_p \left|\psi_f\right\rangle \right\|^2 \tag{3.55}$$

with $\left|\psi_f\right\rangle$ being the desired final state at time T.

Both (3.54) and (3.55) are well-defined problems to be solved e.g. via Sum-of-Squares hierarchy implemented in `TSSOS` Julia library [88, 87].

Chapter 4

Control of open quantum systems

Generally, approaches to control can be divided into two classes: open-loop and closed-loop. Open-loop control involves applying a predetermined sequence of controls to a quantum system without any feedback or modification during the process. The control actions are designed based on a model of the system and executed without real-time adjustment.

On the other hand, closed-loop control involves continuously monitoring the state of the quantum system and dynamically adjusting the control actions based on this feedback. It's a real-time process where the control strategy adapts based on the system's response. This chapter will be divided into two sections, each describing one approach.

4.1 Open-loop control of open quantum systems

Open-loop quantum control is most often model-based and does not involve any kind of monitoring of the state during the process. There has been vast literature published on this topic [44, 39, 23, 74]. We review the most important concepts and methods.

The dynamics of an open quantum system can be described by a completely positive, trace preserving (CPTP) map $\mathcal{M}$. The evolution from time 0 to time T is given as

$$\rho_S(t) = \mathcal{M}_{0,T}[\rho_S(0)], \tag{4.1}$$

where ρ_S denotes the density matrix corresponding to the system. There is a very important distinction between Markovian and non-Markovian dynamics. The evolution is called Markovian [11] (or memoryless), if for every $t_1 \in (0, T)$ it holds that

$$\mathcal{M}_{0,T}[\rho_S(0)] = \mathcal{M}_{t_1,T}[\mathcal{M}_{0,t_1}[\rho_S(0)]]. \tag{4.2}$$

From the quantum optimal control point of view, the distinction between Markovian and non-Markovian dynamics is crucial. Markovian evolution can be described using the Lindblad equation. There is, however, no unified framework for non-Markovian dynamics. Instead, a lot of different methods based on different assumptions have been developed for this case [19]. On the other hand, while for most of the quantum information processing purposes the Markovian dynamics is purely detrimental, there have been some proposals to exploit the non-Markovianity [69, 27, 74].

The analogy to the cost function (3.27) for the goal of driving the system to a desired target state in the open quantum system can be given in terms of the Hilbert-Schmidt product

$$J = \mathrm{Tr}\Big\{\mathcal{M}_{0,T}[\rho_S]\rho_{\mathrm{target}}\Big\}. \tag{4.3}$$

There are two fundamental strategies how to deal with open-loop control of open quantum systems. The first consists of exploiting the symmetries in the interaction between the system and the environment. The second is devised from the assumptions on the timescales of this interaction.

The first strategy is generally tied to the conserved quantity in the presence of a symmetry. Consider for example that all the qubits are coupled indistinguishably to the environment. It has been shown in [54], using the semigroup approach, that there exist decoherence-free subspaces in which the universal quantum computation is possible. In this approach, the error generators are identified with the generators of a Lie algebra. The total Hamiltonian of the system and the environment can be written as

$$H_{SE} = H_S \otimes I_E + I_S \otimes H_E + H_I, \tag{4.4}$$

where H_S, H_E and H_I are the system, environment and interaction Hamiltonians, respectively. Under the assumptions of Markovian dynamics, complete positivity and initial decoupling, the evolution of the system can be generally described by the master equation of the form (setting $\hbar = 1$)

$$\frac{\mathrm{d}\rho}{\mathrm{d}t} = \mathcal{M}[\rho] = -i[H_S, \rho] + \sum_{\alpha,\beta=1}^{M} a_{\alpha,\beta}\Big(A_\alpha \rho A_\beta^\dagger - \frac{1}{2}\{A_\beta^\dagger A_\alpha, \rho\}\Big) \tag{4.5}$$

with $\{,\}$ being the anti-commutator. The operators A_α model all the dissipative processes, e.g. $A_\alpha |0\rangle\langle\alpha|$ may stand for the spontaneous decay from the energy level α to the ground state and the coefficients $a_{\alpha,\beta}$ are the rates of these processes. The set $\{A_\alpha\}_{\alpha=0}^{M}$ forms a basis for the vector space of bounded operators acting on the system Hilbert space and it also forms the $N \times N$ matrix representation of the M-dimensional Lie algebra $\mathcal{L}$ (generally, $M \leq N^2 - 1$).

In the case of the collective decoherence, i.e. all qubits undergoing the same decoherence process, there are only three possible errors and hence $\mathcal{L} = su(2)$. Consider now the decoherence-free N_0-dimensional subspace spanned by the states $\{|i\rangle\}_{i=1}^{N_0}$. The density matrix describing the states in this subspace has the form

$$\tilde{\rho} = \sum_{i,j=1}^{N_0} \tilde{\rho}_{ij} |i\rangle\langle j| . \tag{4.6}$$

The "decoherence-free" means that the second, dissipative, term on the right-hand side of (4.5) gives 0 when acting on $\tilde{\rho}$. It is shown in [54] that the necessary and sufficient condition for such a decoherence-free dynamics is that all basis states $|i\rangle$ of the decoherence-free subspace are degenerate eigenstates of all the error generators $\{A_\alpha\}_{\alpha=0}^{M}$ or, if $\mathcal{L}$ is semisimple, that all $|i\rangle$ are annihilated by all $\{A_\alpha\}_{\alpha=0}^{M}$. If we consider having K qubits, it can be shown [92] that, for the boson-spin in case of the collective decoherence, the size of the decoherence-free subspace N_0 is proportional to $K - \frac{3}{2}\log_2 K$. On the other hand, in the case of independent qubit decoherence (here $\mathcal{L} = \bigoplus_{k=1}^{K} su_k(2)$), there does not exist any decoherence-free subspace.

Let us now take a closer look on the second class of strategies - those based on the assumptions on timescales of the system-environment interactions. If the decoherence is slow, the obvious strategy would be to carry out the desired operation as fast as possible, so that decoherence does not have time to disrupt this operation. But this is often not possible. There are, however, very popular methods to mitigate the effects of decoherence, collectively referred to as dynamical decoupling (DD) [85, 77].

DD was originally understood as the selective removal of the undesired contributions to the spin Hamiltonian via the application of suitable pulse sequences [32]. In the framework of quantum control theory, it can be understood as a dynamical control protocol based on repeated application of controls. The essence of the DD can be clearly seen on the simplest example of spin-$\frac{1}{2}$ coupled to a harmonic

environment. The Hamiltonian (4.4) now has the form

$$H_{SE} = \omega_0 \sigma_z \otimes I_E + I_S \otimes \sum_k \omega_k b_k^\dagger b_k + \sigma_z \otimes \sum_k g_k (b_k^\dagger + b_k), \tag{4.7}$$

where ω_0, ω_k, g_k are real parameters and $b_k^\dagger, b_k$ are bosonic creation and annihilation operators, respectively. As $\sigma_z = \mathrm{diag}(1, -1)$, it is easy to see that the last (interaction) term has the opposite signs depending on the spin state. Thus, one may intuitively expect that it can be possible to average these contributions out.

The pioneering Hahn's spin-echo experiment [32] was based on the application of a π-pulse to the spin system after letting it evolve in the magnetic field for time τ. As this reversed the sign of the (Zeeman) interaction, after another time period τ of evolution under the (static) magnetic field, the echo was generated.

Although very simple, current methods of DD are still based on this principle. A variety of methods have been developed to deal with the fluctuating magnetic field, the most prominent being the method of Carr and Purcell [15] based on application not one, but a sequence of pulses. This, however introduces the errors originating from the pulse imperfections. For this purpose, Meiboom and Gill proposed a modification to this sequence to compensate for these errors [58]. This resulted in so-called CPGM sequence.

Finally, let us extend some notions of the optimal quantum control devised in the previous chapter to open quantum systems. The functional to be optimized is given by the equation (4.3), where the dependence on the control fields is given implicitly, i.e. $\mathcal{M}_{0,T} = \mathcal{M}_{0,T}[\{u_j\}]$. One of the prominent methods to calculate extremum of J is the Krotov's method [70, 45]. It is based on the monotonic convergence where the update equation for control is given as [44]

$$\Delta u(t) = u^{i+1}(t) - u^i(t) = \frac{S(t)}{\alpha} \mathrm{Im}\left\{ \mathrm{Tr}\left\{ \lambda^i(t) \frac{\partial \mathcal{M}[\rho]}{\partial u}\Big|_{\rho^{i+1}, u^{i+1}} \right\} \right\}, \tag{4.8}$$

where $S(t)$ is the control shape function, α is the hyper-parameter determining the step size in the change of the control and $\lambda(t)$ denotes the costate. This is an implicit equation (dependence on u^{i+1} on the RHS is through $\rho^{i+1}(t)$) which can be avoided using low order approximation [63]. The state at time t is obtained by solving the Lindblad equation with initial condition $\rho^i(0) = \rho_{\mathrm{init}}$. The costate is, on the other hand, obtained by solving the Lindblad equation with opposite sign for the dissipative terms and with the "initial" condition $\lambda^i(t = T) = \rho_{\mathrm{target}}$, i.e. is solved backward in time.

4.2 Closed-loop control of open quantum systems

The closed-loop control of quantum systems can be roughly divided into three classes of methods [22]:

1. Closed-Loop Learning Control
2. Feedback Control
3. Robust Control.

The first class refers to methods involving a learning process where the control strategy is iteratively refined based on the outcomes of previous control attempts. It is a form of adaptive control that uses a feedback loop not only for immediate control adjustments but also for learning and improving control strategies over time. Such an approaches often involve reinforcement learning (RL) which, nowadays

often powered by deep neural networks, turn out to be very popular as they provide very interesting results [62, 12, 59].

The feedback control [21] is a mechanism where the control action is based on the current state of the system. It involves measuring the state or output of a system and using this information to adjust the inputs to the system in real-time to achieve the desired outcome. The key difference between the feedback and closed-loop learning control is that the latter emphasizes iterative learning and adaptation over time, improving its performance with experience. Feedback control, in contrast, focuses on immediate response to system states without necessarily improving or evolving its control strategy over time.

The methods included in the last class combine the objectives of optimal control (efficiency and effectiveness) with the requirements of robust control. Robust control [93] refers to methods that address the challenge of uncertainties and disturbances which are inherent in any physical system. In quantum systems, these uncertainties could arise from environmental noise, imperfections in the system, or incomplete knowledge of the system's dynamics. Such a strategies should be stable and provide the best possible cost functional under the worst possible uncertainties. In the rest of this section we will focus on the RL-based strategies as they are currently receiving a lot of attention due to the huge boom in neural networks.

RL [78] is a type of machine learning where an agent learns to make decisions by interacting with an environment. In RL, the agent seeks to learn a policy – a strategy of choosing actions based on states – to maximize cumulative reward over time. The environment is typically modeled as a Markov Decision Process (MDP), defined by a set of states S, a set of actions A, a transition function $P(s_{t+1}|s_t, a_t)$ (probability of reaching state s_{t+1} from state s_t by making action a_t), and a reward function $R(s_t, a_t)$. The agent's objective is to learn a policy $\pi(a|s)$ (probability of choosing action a in state s) that maximizes the expected return G_t, which is typically defined as the sum of discounted future rewards, i.e.

$$G_t = \sum_{k=1}^{\infty} \gamma^k R_{t+k+1}, \tag{4.9}$$

where $\gamma \in (0, 1)$ is the discount factor that determines the present value of future rewards. It should be mentioned that sometimes it is more suitable to consider cost function instead of reward and consequently the goal is to minimize the expected overall cost.

Two fundamental approaches in RL are value-based and policy-based methods [61]. Value-based methods, like Q-learning [89], focus on learning the value function, which estimates how good it is to be in a particular state or to perform a certain action. The Q-function in Q-learning, $Q(s, a)$, is updated using the Bellman equation:

$$Q(s, a) \leftarrow Q(s, a) + \alpha[R(s, a) + \gamma \max_{a'} Q(s', a') - Q(s, a)], \tag{4.10}$$

where α is the learning rate. Policy-based methods, on the other hand, directly optimize the policy. An example is the policy gradient method, where the policy $\pi_\theta(a|s)$ is parametrized by θ and updated in the direction of the gradient of expected return, i.e. $\theta \leftarrow \theta + \nabla_\theta G_t$.

Deep Reinforcement Learning (DRL) [3] extends traditional reinforcement learning by integrating deep learning, enabling agents to learn optimal policies in high-dimensional and complex environments. In DRL, deep neural networks are used to approximate the policy (in policy-based methods) or the value function (in value-based methods), which is particularly useful when dealing with large state or action spaces that are impractical for traditional tabular approaches. The most prominent example of DRL is the Deep Q-Network (DQN), which uses a neural network to approximate the Q-function in Q-learning. DQN incorporates techniques like experience replay [73], where transitions are stored and randomly sampled to train the network.

As is obvious from the expressions above, RL requires the measurement of the state at each iteration. This, in principle, requires complete quantum state tomography which is very expensive. There are, however, various ways to bypass this requirement. For example in [62] the goal is to synthesize unitary two-qubit gates with higher fidelity, shorter time and with high robustness. In the paper, the aim is account for the leakage errors to higher energy states than is the computational basis. They focus on the superconducting gmon qubits with Hamiltonian in the rotating wave approximation of the form

$$H_{\mathrm{RWA}} = \frac{\eta}{2}\sum_{j=1}^{2}\hat{n}_j(\hat{n}_j - 1) + g(t)(\hat{a}_2^\dagger\hat{a}_1 - \hat{a}_1^\dagger\hat{a}_2) + \sum_{j=1}^{2}\delta_j(t)\hat{n}_j + \sum_{j=1}^{2} i f_j(t)(\hat{a}_j e^{-i\varphi_j(t)} - \hat{a}_j^\dagger e^{i\varphi_j(t)}), \tag{4.11}$$

where $\hat{a}_j^\dagger, \hat{a}_j$ are the bosonic creation and annihilation operators, $\hat{n}_j$ the number operators and time-independent parameter η represents the Josephson junction anharmonicity. The control parameters are: $f_j(t)$ (amplitude), $\varphi_j(t)$ (phase), $\delta_j(t)$ (qubit detuning) and $g(t)$ (tunable capacitative coupling). In this approximation, only the one-body and the nearest-neighbor two-body terms are included. Although this model is for superconducting qubits, modifications for various models of quantum dots are straightforward. In the following considerations, only the lowest states above computational basis are added, thus truncating the effective space and enabling classically computing the evolution at each iteration.

One of the most important parts of the RL strategy design is the definition of the reward/cost function. This should account for various physical aspects such as the fidelity of the unitary, total time, leakage etc. In [62] the cost was defined as

$$J(\chi,\beta,\mu,\kappa) = \chi[1 - F[U(T)]] + \beta L + \mu \sum_{t\in\{0,T\}} [g^2(t) + f^2(t)] + \kappa T, \tag{4.12}$$

where $F = \frac{1}{16}|\operatorname{Tr}\left\{U^\dagger(T)U_{\mathrm{target}}\right\}|^2$ is the gate fidelity, L accounts for the leakage and its form can be determined via perturbation theory within time-dependent Schrieffer-Wolf transformation formalism [31], the third term deals with the boundary conditions to facilitate convenient gate concatenation and T is the final time to be minimized. (χ,β,μ,κ) are hyper-parameter to be optimized in order to obtain the best possible outcomes.

The RL scheme then uses two neural networks (NN). The first, so-called policy NN, performs the mapping of the state (containing the information about the unitary at time t_i) to the mean and variance of the Gaussian distribution of the proposed control actions at time $t + 1$. The second, so-called value function NN, then takes the simulated proposed unitary as an input and outputs the predicted reward associated with the current unitary. A crucial finding was the significant improvement of the robustness of the control when adding noise in terms of the Gaussian fluctuation of the control signals. Such a scheme provided a significant reduction in both gate time and infidelity.

Another very interesting way to bypass the need for the quantum state tomography at each step has been recently shown in [59]. This method, suited primarily for preparing low-entangled 1D states, leverages the matrix product state (MPS) [75] framework to represent the many-body system and as a trainable architecture for RL agent. MPS provide a compressed representation of many-body wave functions and allow for efficient computation with resources scaling only linearly in the system size for area-law entangled states. This way, the authors have shown that the expensive training step can be efficiently computed classically. Moreover, they provided a mapping of MPS to parameterized quantum circuit on NISQ device, eliminating the need for quantum state tomography and doing the inexpensive inference step directly on the quantum hardware.

There is much room to explore the use of DRL for optimal quantum control. This is an area of very active research and the scalability of different approaches is now a major issue.

Chapter 5

Quantum state discrimination via LDS identification

5.1 LDS identification and Kalman filtering

Linear dynamical system (LDS) identification is a process in control theory and signal processing [76] that aims to determine the parameters of a LDS from the observed input-output data. In a LDS, the relationship between the system's inputs and outputs is described by linear equations. The identification of such systems is essential for understanding their behavior, predicting future responses, and designing effective control strategies. The process typically involves estimating parameters such as the system's state matrices, input matrices, and output matrices, which govern the dynamics of the system.

For formal definition of the LDS (G, F, V, W) we use the standard notation adapted from [90]:

$$\phi_t = G\phi_{t-1} + \omega_t, \tag{5.1}$$

$$Y_t = F'\phi_t + \nu_t. \tag{5.2}$$

Here $\phi_t \in \mathbb{R}^n$ is the hidden state, $Y_t \in \mathbb{R}^m$ is the measured observation, $G \in \mathbb{R}^{n\times n}$ is the process matrix, $F \in \mathbb{R}^{n\times m}$ is the measurement matrix (F' is its transposition) and ω_t, ν_t are the process and measurement noises, respectively. These noises are assumed to be normally distributed and have covariance matrices W and V respectively. The goal is then to identify the LDS (G, F, V, W).

There are some major problems that have to be addressed in the LDS identification. One of them is that the underlying optimization problem is non-convex which is also one of the reasons why polynomial optimization might turn out superior. Another problem is that the dimension of the hidden state is not known in general. In this work, however, we hard-assign the dimension to 2 since we want to preserve the direct relation to the underlying two-dimensional quantum state. Finally, since G, F are matrices, the problem is in principle non-commutative. Although work has been done on the non-commutative POP LDS identification [94], the minimizer extraction (which is vital for this work) remains highly non-trivial and is a part of an active research. For this reason we consider the commutative case and write the constraints in terms of elements of matrices G and F.

5.1.1 Standard LDS identification methods

One common method used for LDS identification is the least squares approach. This technique minimizes the sum of squared errors between the actual output data and the output data predicted by the model based on the estimated parameters. The least squares approach is widely used because it provides

a simple and efficient way to estimate the system's parameters. However, it is essential to ensure that the system being identified is indeed linear, as this method is not suitable for identifying nonlinear systems.

Another powerful and widely used class of methods for state-space model estimation from input-output data are subspace methods. These methods are particularly well-suited for handling noisy and incomplete data, making them valuable in practical applications. The subspace methods utilize the concept of subspaces, which are low-dimensional representations of system data that capture its essential dynamics. Subspace methods can estimate the state matrices of a LDS directly from input-output measurements, without the need to explicitly identify intermediate parameters like poles or transfer functions. This characteristic makes subspace methods computationally efficient and more robust in dealing with time-varying systems.

The leading methods for state-space model identification are implemented in MathWorks™ Matlab™ System Identification Toolbox. In our work we consider three different routines from this toolbox:

1. `ssest` ("def"): routine based on least squares algorithm
2. `N4sid` ("N4sid"): routine based on subspace methods of [83]
3. `N4sid` ("ssa"): routine based on subspace methods with weighting algorithm based on ARX [37]

5.1.2 LDS identification via polynomial optimization

The polynomial optimization approach to LDS identification is based on [94], however, as already mentioned, we consider the commutative case. The loss function is based on the optimal least square estimate. Namely, given the time series of observations $Y_1^i, \dots Y_{t-1}^i$ where i denotes the trajectory and t the time-step, we want to minimize a one-step error function of the form

$$\min_{f_t, t\geq 1} \sum_{i=1}^{N} \sum_{t\geq 1} \left\| Y_t^i - f_t \right\|^2, \tag{5.3}$$

where N is the number of trajectories and f_t are the estimates, i.e. in our case either scalar (if we measure only the "I" component) or two-dimensional vector of decision variables $\begin{pmatrix} f_{t_1} \\ f_{t_2} \end{pmatrix}$.

To put the polynomial optimization problem formally, we consider

$$G = \begin{pmatrix} G_1 & G_2 \\ G_3 & G_4 \end{pmatrix}, \qquad F = \begin{pmatrix} F_1 \\ F_2 \end{pmatrix}, \qquad \omega_t = \begin{pmatrix} \omega_{t_1} \\ \omega_{t_2} \end{pmatrix}, \qquad \nu_t = \nu_t \tag{5.4}$$

for the one dimensional measurement and

$$G = \begin{pmatrix} G_1 & G_2 \\ G_3 & G_4 \end{pmatrix}, \qquad F = \begin{pmatrix} F_1 & F_2 \\ F_3 & F_4 \end{pmatrix}, \qquad \omega_t = \begin{pmatrix} \omega_{t_1} \\ \omega_{t_2} \end{pmatrix}, \qquad \nu_t = \begin{pmatrix} \nu_{t_1} \\ \nu_{t_2} \end{pmatrix}, \tag{5.5}$$

for the IQ measurement. We also include variables $m_t = \begin{pmatrix} m_{t_1} \\ m_{t_2} \end{pmatrix}$ that model the evolution of the hidden state. Finally we add the sum of squares of the process and measurement noise variables as regularisers with sufficient multipliers c_1, c_2 (that should be proportional to $\sqrt{N}$) to the loss function. The polynomial

optimization formulation for identifying LDS (5.1) is then

$$\min_{\substack{f_t, m_t, G \\ F, \omega_t, \nu_t}} \sum_{i=1}^{N} \sum_{t \geq 1} \left\| Y_t^i - f_t \right\|^2 + c_1 \sum_{t \geq 1} \nu_t^2 + c_2 \sum_{t \geq 1} \omega_t^2 \tag{5.6}$$

over feasible set given by constraints

$$m_t = G m_{t-1} + \omega_t, \tag{5.7}$$

$$f_t = F' m_t + \nu_t. \tag{5.8}$$

$F' m_t$ is called noise-free estimate and is regarded as the predicted output. In this approach we in a certain way consider an "averaged trajectory" as in the loss function the noise realizations are common for all the trajectories (this is also the reason behind the proportionality of the multipliers to $\sqrt{N}$). There is yet another approach that consider each trajectory separately and only the system matrices are common to all the trajectories. Such a minimization problem would then be formulated as

$$\min_{\substack{f_t^i, m_t^i, G \\ F, \omega_t^i, \nu_t^i}} \sum_{i=1}^{N} \sum_{t \geq 1} \left\| Y_t^i - f_t^i \right\|^2 + c_1 \sum_{i=1}^{N} \sum_{t \geq 1} (\nu_t^i)^2 + c_2 \sum_{i=1}^{N} \sum_{t \geq 1} (\omega_t^i)^2 \tag{5.9}$$

over feasible set given by constraints

$$m_t^i = G m_{t-1}^i + \omega_t^i, \tag{5.10}$$

$$f_t^i = F' m_t^i + \nu_t^i. \tag{5.11}$$

We tried this approach, however as the number of variables here (unlike the previous approach) scales also with the number of trajectories, this becomes very quickly computationally intractable even on the computing cluster.

5.1.3 Kalman filter

The Kalman filter [40] is a widely used recursive algorithm that provides an optimal estimate of the state of a LDS in the presence of noise. It was developed by Rudolf Kalman in the 1960s and has since found extensive applications in various fields, including control systems [40], navigation [35], and signal processing [5]. The Kalman filter operates based on a two-step prediction-correction process. In the prediction step, the filter estimates the state of the system at the next time step using the system's dynamic model and the previous state estimate. This predicted state is then updated in the correction step by incorporating measurements from sensors or noisy observations. The Kalman filter calculates a weighted average of the predicted state and the measurement to produce an optimal state estimate, taking into account the uncertainties associated with both the system dynamics and the measurements.

The principle underlying the Kalman filter is to optimize the accuracy of state estimation by minimizing the mean square error between the estimated state and the true state of the system. It achieves this optimization by employing a Bayesian probabilistic approach. The Kalman filter maintains a probabilistic representation of the state estimate, incorporating both the state estimate and the covariance matrix that represents the uncertainty associated with that estimate. As new measurements become available, the filter adjusts the state estimate and covariance matrix iteratively, progressively reducing the uncertainty about the true state of the system. This recursive nature allows the Kalman filter to provide real-time estimates with efficient and low computational complexity, making it a fundamental tool for state estimation in various applications where accurate system state estimates are crucial for optimal decision-making and control.

To formally write down the equations that the Kalman filter is based on [47], let the initial state ϕ_0 be a normal random variable with mean m_0 and covariance C_0 and set:

$$\begin{aligned} a_t &= Gm_{t-1} \\ R_t &= GC_{t-1}G' + W \\ Q_t &= F'R_tF + V \\ A_t &= R_tF/Q_t \end{aligned} \tag{5.12}$$

where C_t is the covariance matrix of ϕ_t given $Y_0, \dots Y_t$. Then the Kalman filter update equations are

$$m_t = A_tY_t + (I - F \otimes A_t)a_t \tag{5.13}$$

$$C_t = R_t - A_tQ_tA_t' \tag{5.14}$$

where $\otimes$ denotes an operator for outer product, i.e. the matrix for $a \otimes b$ is ab' for $a, b \in \mathbb{R}^n$. It is important to note that the only element that depends on $Y_0, \dots Y_t$ is m_t.

5.2 Quantum state discrimination

5.2.1 Discrimination with labeled training trajectories - the algorithm

In this section we describe the algorithm and results for quantum state discrimination from labeled trajectories. This procedure assumes that at least some control is developed over the device because for correct label assignment to trajectories we have to know in which state we prepare the QD.

The idea behind the state discrimination method is simple. We learn two sets of LDS matrices from trajectories corresponding to state $|0\rangle$ and $|1\rangle$, respectively. Then, for the trajectories from the validation set, we apply the Kalman filter with both sets of matrices and sum the error between the predicted value and the actual measured value on all the time-steps. Finally, we assign the state according to which set of matrices produces smaller cumulative error. The pseudo-code is summarized in Algorithm (1).

The hyper-parameters to be varied in this approach are the number of training trajectories and the number of time-steps in each trajectory. Different choices will be made for 1D experiment data and 2D simulated data, as the resources (mostly for the POP part) are different.

5.2.2 1D data and results

Although the whole process was first intended for the 2D data (IQ measurement), due to the unavailability of the working IQ demodulator (this will be discussed in detail in the following sections) we adjusted the whole process also for the 1D (only the "I" component) measurement. The measuring device for this kind of measurement was ready and the data can be found in the file `201125150755.h5`.

This data represent one long measurement sequence with 600000 data points. These are separated into 600 subsets with 1000 data points each. This kind of separation, however, wasn't well suited for our purposes and more importantly these data weren't labeled. We therefore used a Python module from Quantum Device Lab at Tohoku University to find the threshold separating the state 0 and 1. The data and the threshold value are depicted in fig. (5.1). Then we reshaped the data so that the 600000 data points were divided into 6000 trajectories, each containing 100 data points. Then each of these trajectories was labeled 0 or 1 according to whether the mean of its data points was below or above the threshold, respectively. Finally, we filtered out those labeled as 0 whose mean of the first 10 data points was above the threshold because those trajectories with high probability corresponded to deexcitation during the long measurement sequence and wouldn't occur during the standard short measurement e.g.

Algoritmus 1: Discrimination with labeled trajectories algorithm

Input : Training set of trajectories labeled 0 or 1, validation set of trajectories labeled 0 or 1
Output: accuracy of given model

Training phase;
Learn LDS (G_0, F_0, V_0, W_0) from trajectories labeled as 0;
; `/* via standard or POP */`
Learn LDS (G_1, F_1, V_1, W_1) from trajectories labeled as 1;
Discrimination phase;
for *trajectory in validation set* **do**
    Apply Kalman filter with (G_0, F_0, V_0, W_0) and compute error $\mathbf{e}_0$
    Apply Kalman filter with (G_1, F_1, V_1, W_1) **and compute error** $\mathbf{e}_1$
    if $\boldsymbol{e_1 \geq e_0}$ **then**
        assign state $|0\rangle$**;**
    else
        assign state $|1\rangle$**;**
    end
end
Compute the accuracy of the assigned labels

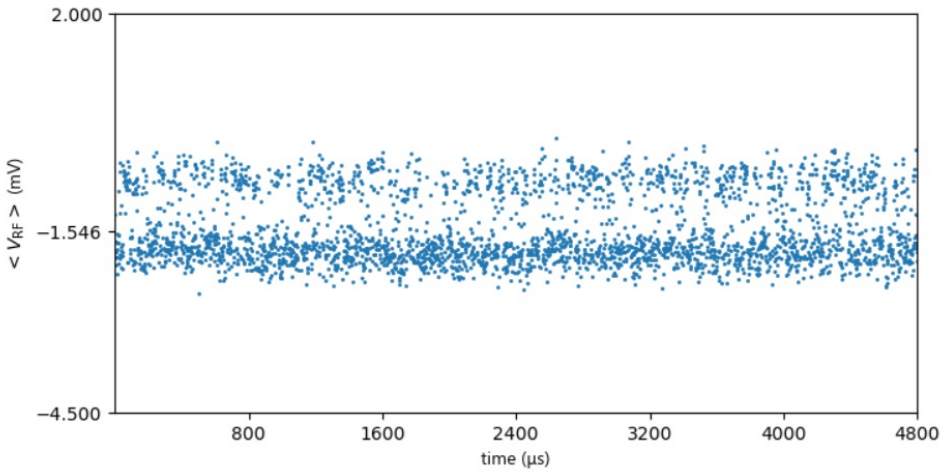


Figure 5.1: Experimental data with threshold value. Each dot represents an average of 200 neighboring data points. The threshold value has been determined to -1.546 mV.

during quantum computation. Similar steps but in the opposite manner has been done for the trajectories labeled as 1. This procedure is implemented in the file `tohoku_1D_data_prep.ipynb`.

These trajectories are then used in the training and inference phase using both POP and standard LDS identification methods. As POP is very resource intensive and these resources scale with the number of variables, which in turn are proportional to the number of time steps, this number of time steps was very limited even when using the RCI cluster located at the Faculty of Electrical Engineering CTU. Therefore, we used the number of time steps from 21 to 30. As the number of trajectories in our approach does not affect the number of variables, we considered 1000 training trajectories. The script used for calculating the LDS matrices can be found in `Julia_1D.jl` and the identified variables for state 0 trajectories in `tohoku_state0_free_init_2023-12-14_11-32-07.txt` and for state 1 in

`tohoku_state1_free_init_2023-12-12_19-33-56.txt`.

The inference phase has been done using 500 trajectories and the accuracy and F1 score was calculated in `POP_1D_accuracy.ipynb`. The results can be found in the table (5.1). From these results

time-steps	accuracy (%)
22	49.9
23	50
24	50
25	46.8
26	53.4
27	50
28	50.8
29	50
30	50
31	50

Table 5.1: Accuracy of POP models trained on 1000 trajectories (both for state 0 and 1) and various number of time-steps.

and the consequent analysis it is obvious that it is almost always the case that one set of matrices works better (produces smaller error) on both state 0 and 1 test trajectories. This is not the desired behavior. Given the huge number of modifications tried, the time spent and the results that will follow, I suspect that the problem lies in the assumption on the system itself. First, there is no guarantee that the physical system will behave linearly. Second, even if there would be any dynamics that could be well approximated by LDS, the signal to noise ratio is rather small, i.e. the noise prevents correctly identifying the LDS and the results show signs of over-fitting. Third, it may be the case that for faithful identification or approximation of the system by LDS the dimension of the LDS should be much higher than 2. Here we, however, again encounter the problem of required resources scaling with the number of variables which scale quadratically with the dimension of the LDS.

For comparison we performed the LDS identification using the earlier mentioned methods implemented in Matlab. We considered learning on 1 and 1000 trajectories and (as these methods are less resource demanding) and 20, 30 and 100 time-steps. The identification algorithm can be found in `sys_idents.m`. The test phase has been done on another 500 trajectories of length 15, 30 and 100 in `accuracy.ipynb`. The accuracy and F1 score has been then again calculated and the results for the accuracy can be found in the table (5.2). Clearly these methods suffer the same problems as in the POP case, although some of the models seem to consistently provide slightly better accuracy.

5.2.3 Simulated data

Since by the start of the project there was no IQ demodulator available in the laboratory, we prepared a program for simulated data generation. The main source of noise in the circuit is often the amplifier, which produces Gaussian noise with different means and different covariance matrices for different quantum states. The educated guess for the mean and the covariance matrix for the ground state were

$$\boldsymbol{x}_0 = \begin{pmatrix} 0.1 \\ 0.2 \end{pmatrix}, \qquad \mathbb{X}_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{5.15}$$

	time-steps: 15	time-steps: 30	time-steps: 100
def_1_20	64.77	69.86	78.94
def_1_30	54.9	51.5	50.4
def_1_100	50.0	50.0	50.0
def_1000_20	50.0	50.0	50.0
def_1000_30	50.0	50.0	50.0
def_1000_100	50.0	50.0	50.0
N4sid_1_20	50.0	50.0	50.0
N4sid_1_30	53.99	53.79	53.79
N4sid_1_100	50.0	50.0	50.0
N4sid_1000_20	50.0	48.0	50.0
N4sid_1000_30	50.0	51.0	50.0
N4sid_1000_100	50.0	52.5	50.0
ssa_1_20	71.06	71.06	56.59
ssa_1_30	65.27	64.77	59.88
ssa_1_100	63.57	63.17	58.88
ssa_1000_20	63.77	63.47	58.58
ssa_1000_30	63.77	63.37	58.68
ssa_1000_100	63.57	62.97	58.38

Table 5.2: Rows: different models with naming convention (model)_(number of training trajectories)_(number of time-steps in each trajectory). Columns: lengths of the validation trajectories. Results are in %.

while for the excited state the choice was

$$\boldsymbol{x}_1 = \begin{pmatrix} 0.8 \\ 0.1 \end{pmatrix}, \qquad \mathbb{X}_1 = \begin{pmatrix} 0.6 & 0 \\ 0 & 0.6 \end{pmatrix}. \tag{5.16}$$

For the trajectories corresponding to the excited state we also included the strongest of the processes affecting the physical system and thus also the measurement results - spontaneous relaxation from the excited state to the ground state with an exponential dependence on time. This dependence is determined by the time T_1, which was set here to 0.1 ms.

To illustrate how this process affects the standard averaging measurement procedure we prepared the diagrams of measured trajectories averaged over different times. This is depicted in Fig. (5.2) (which is two-dimensional equivalent to fig. (2.8)) and it is clear that as one extends the averaging time, the variance of the trajectories decreases and the respective underlying states become more distinguishable. This distinguishability peaks somewhere around 50 time-steps. If we continue averaging over longer trajectories, however, the T_1 process starts to play an important role and the distinguishability again decreases as the mean of the trajectories corresponding to state $|1\rangle$ shifts towards the mean of the of the trajectories corresponding to state $|0\rangle$. The algorithm for the data generation and averaging can be found in `Simulation_with_T1.ipynb`.

5.2.4 IQ demodulator setup

The IQ demodulator came to the lab by the very end of the COLABS program and so the installation had to be done very fast. The IQ demodulator is depicted in Fig. (5.3). Unfortunately, the installation of this device to the circuit was not successful. From Fig. (5.4) it is clear that for the previous mixer

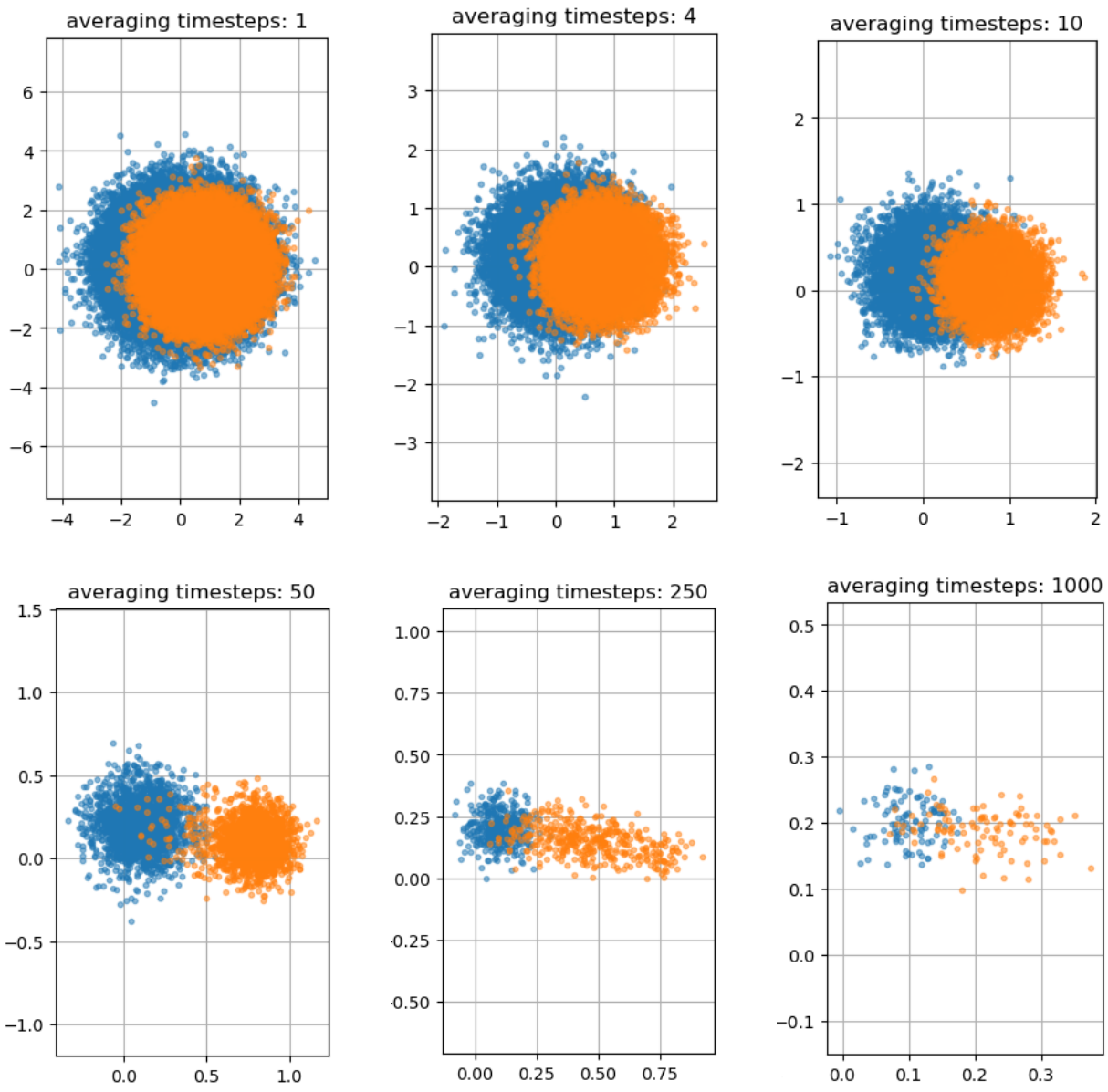


Figure 5.2: Simulation trajectories averaged over 1, 4, 10, 50, 250 and 1000 time-steps, respectively.

(that is only built for "I" component measurement) there is an apparent dependence of the output signal on the frequency and also the shift at $-27V$ gate voltage is manifest. For the I component of the newly installed IQ demodulator, on the other hand, there was no dependence on the frequency and gate voltage. This state remained even after trying different source power, attenuators etc. So the measurement with IQ demodulator was concluded as unsuccessful and the device has to be investigated. This is the reason why all consequent methods were used only on simulated data. As soon as the demodulator will be ready to use, the data will be sent and the methods will be verified on the real data.

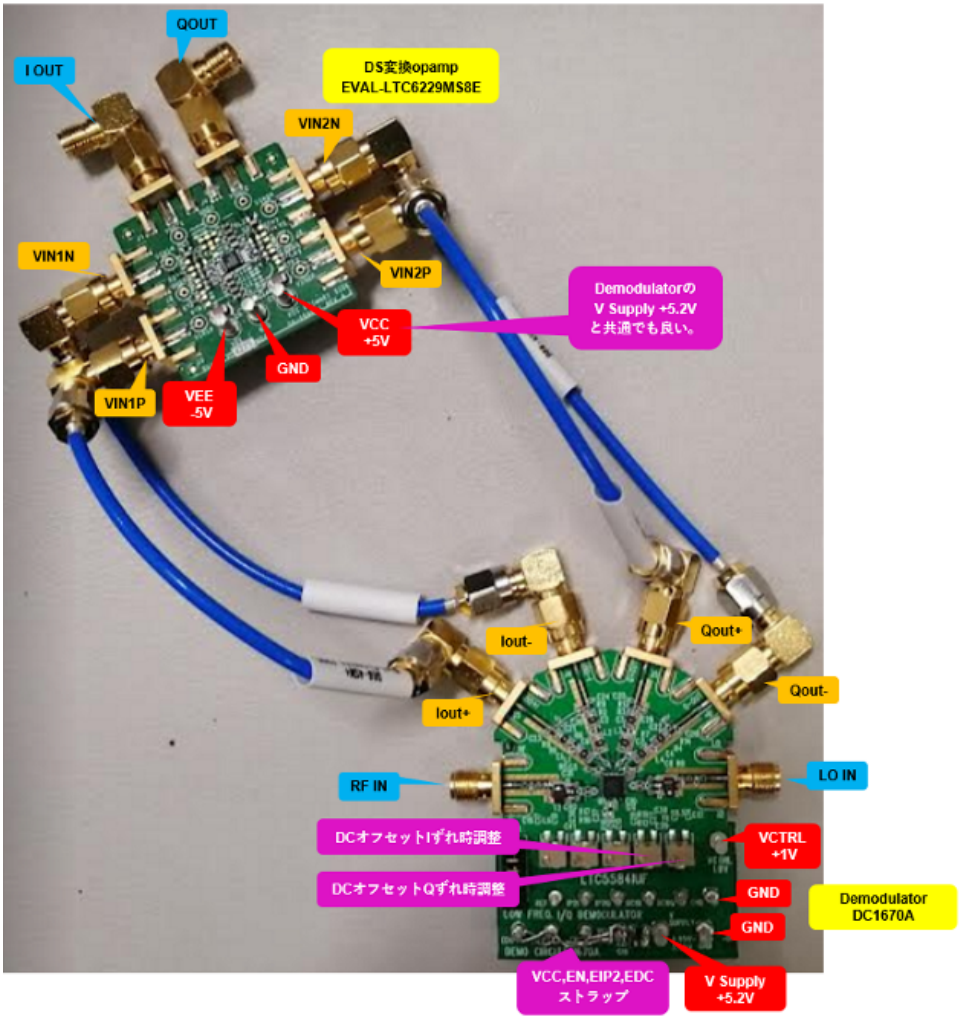
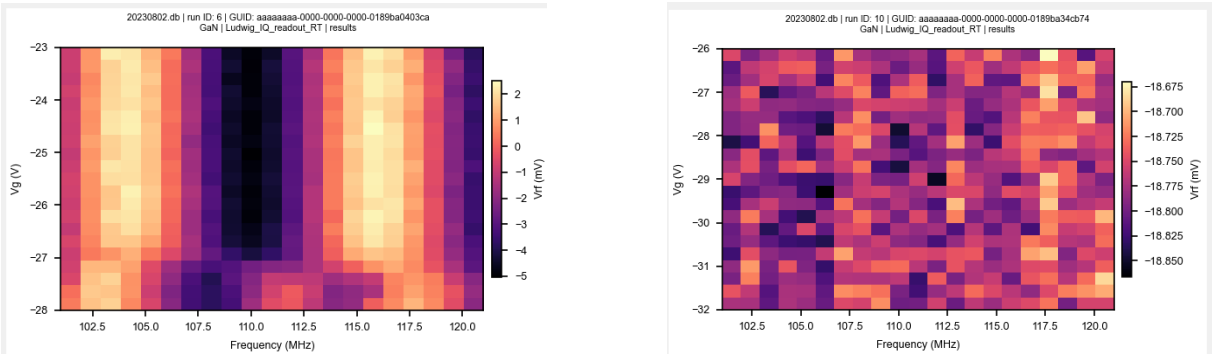


Figure 5.3: IQ demodulator



(a) 1D mixer

(b) I component of IQ demodulator

Figure 5.4: Dependence of the response signal on frequency and gate voltage.

5.2.5 Results with the artificial IQ data

In this case of 2D IQ measurement data we again tried to employ both POP and standard LDS identification methods. However, POP failed to produce any remotely reasonable results for several reasons. On the one hand, there are reasons preventing better results already on the previously discussed 1D data. Furthermore, higher dimensionality implies more variables, and therefore the number of time steps had to be considerably reduced due to computing power. This, in the presence of significant noise, naturally prevents better identification of LDS, as the matrices show strong signs of over-fitting. Ultimately, this data is generated as two Gaussians with different means, with trajectories corresponding to state $|1\rangle$ having an exponential probability of deexcitation to state $|0\rangle$. Such data will not inherently behave as LDS. Although attempts at identification using POP were made, they were not successful and therefore there is no point in publishing the results. Further attempts will take place after the delivery of data from IQ measurements on a real device.

For standard methods of LDS identification we again varied the number of trajectories from which was the LDS identified and the number of time-steps in each trajectory. The number of trajectories was now varied within the set $\{1, 10, 50\}$ and the number of time-steps within $\{15, 30, 60, 200\}$. For each combination we estimated all three standard models - `ssest` (def), `N4sid`, `ssarx` (ssa) - first on trajectories corresponding to the state $|0\rangle$ a then on trajectories corresponding to the state $|1\rangle$.

In Figs (5.5, 5.6, 5.7, 5.8) we plot the comparison of the output of the identified models with selected parameters and the actual output. It can be seen that especially on the longer trajectories corresponding to the state $|1\rangle$, at least one of the models was able to identify that the mean for these trajectories may change and hence the outputs were estimated as decreasing in the case of the amplitude.

The results in terms of accuracy of label assignment for all the models after applying Kalman filters on trajectories with different lengths can be found in table (5.3).

It can be again seen that for most systems one set of matrices is simply better on both types of trajectories and this is the reason why exactly 50% is so common result. Even though most of the results are not very optimistic and are in general worse than would be the results of simple averaging technique, there are also reasons for mild optimism.

If we take a look at the shortest validation trajectories (12 or 30 time-steps), models like def_50_200, N4sid_10_30 and N4sid_10_200 reach accuracy over 60%. Especially for the 12 time-steps trajectories this is comparable to performance of the averaging method. Furthermore, and this is very important, these models were identified on simulated data which are almost perfect Gaussian noise with just different means and variances without any underlying dynamics. We can expect that for real measurements where underlying physical processes take place, the models would be more distinguishable and this would result in better accuracy. This will be explored as soon as the measurement data will be available.

There is one more thing that works in favor of LDS identification method. On the experimental devices even the number of electrons in the QD can be unstable and during the measurement it can produce trajectories that would be very uncertain in the averaging method. Here, however, we can simply consider LDS of higher dimension that may be able to identify also the measurements where some unexpected trajectory occurred and we may be able to exclude this measurement.

5.3 Discrimination with unlabeled training trajectories

Many experimental devices provide very poor control over the state which implies that the labels cannot be reliably assigned to the trajectories since there is a huge uncertainty about the prepared state. For this purpose, we created an algorithm that is built up on the previous case but which does not rely on the labels.

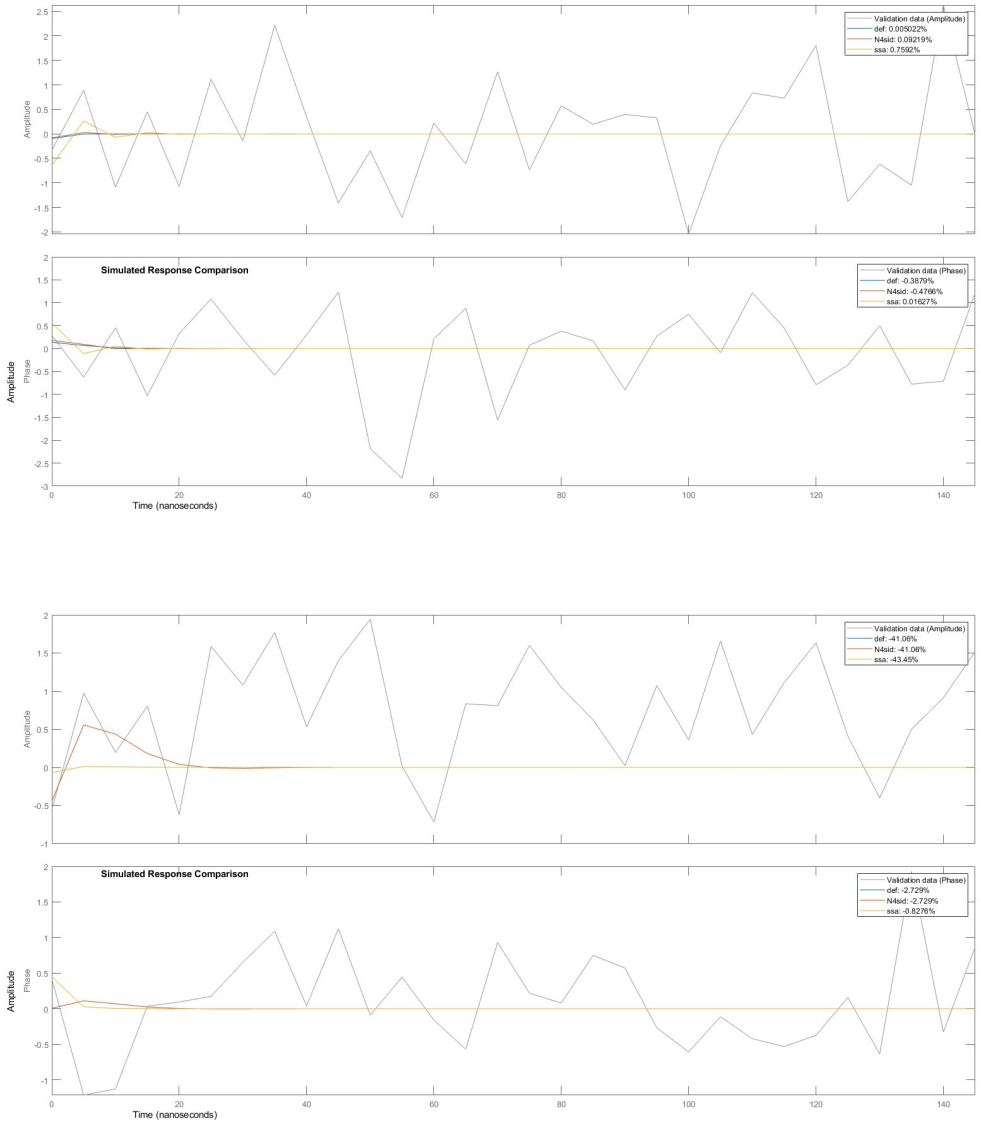


Figure 5.5: Comparison of models trained on 1 trajectory with 30 time-steps. Upper: trained on trajectory corresponding to $|0\rangle$, lower: trained on trajectory corresponding to $|1\rangle$.

Instead of working with pre-assigned labels, this approach creates its own clusters in joint clustering and LDS identification algorithm. Although there are many ways how to formulate this joint algorithm, as the first step we chose probably the most straightforward way. This is based on agglomerative clustering in the space of system matrices.

First, we properly formulate the problem. The general formulation of joint clustering and LDS identification problem can be written as

$$\arg\min_{\boldsymbol{S},\mathcal{L}} L(\boldsymbol{S},\mathcal{L}|\boldsymbol{Y}) \tag{5.17}$$

where L is a properly chosen loss function such that the empirical risk will be minimized, $\boldsymbol{S} = \{S_1, S_2, \dots\}$

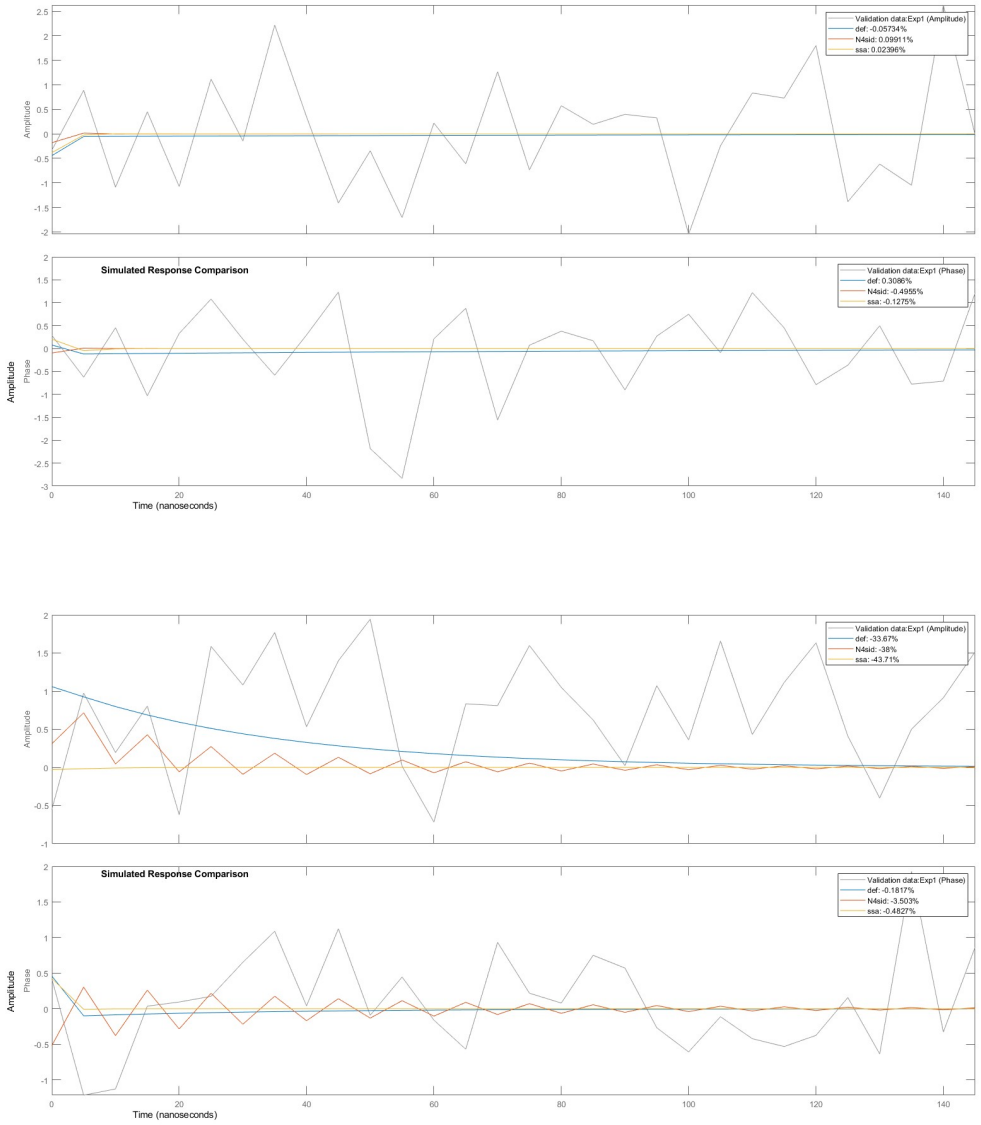


Figure 5.6: Comparison of models trained on 10 trajectories with 30 time-steps. Upper: trained on trajectories corresponding to $|0\rangle$, lower: trained on trajectories corresponding to $|1\rangle$.

is the set of clusters, $\mathcal{L} = (G, F, W, V)$ is the set of system matrices for all the trajectories and $\boldsymbol{Y} = \{Y^1, Y^2, \dots\}$ is the set of all the trajectories. By "arg" we mean that the result of the problem are the optimally determined LDSs and their optimal clustering.

Holding one of $\{\boldsymbol{S}, \mathcal{L}\}$ fixed, we can decompose the optimization as

$$\arg\min_{\boldsymbol{S}} L(\boldsymbol{S}|\boldsymbol{Y}, \mathcal{L})$$
$$\arg\min_{\mathcal{L}} L(\mathcal{L}|\boldsymbol{Y}, \boldsymbol{S})$$

The first line describes standard clustering in the space of system matrices. The second line corresponds

	time-steps: 12	time-steps: 30	time-steps: 60	time-steps: 250
def_1_15	50.5	50.0	50.0	50.0
def_1_30	52.0	50.0	50.0	50.0
def_1_60	50.0	50.0	50.0	50.0
def_1_200	31.5	33.5	34.0	44.0
def_10_15	50.0	50.0	50.0	50.0
def_10_30	50.0	50.0	50.0	50.0
def_10_60	50.0	50.0	50.0	50.0
def_10_200	50.0	50.0	50.0	50.0
def_50_15	50.0	50.0	50.0	50.0
def_50_30	50.0	50.0	50.0	50.0
def_50_60	50.0	50.0	50.0	50.0
def_50_200	58.5	61.0	59.5	51.5
N4sid_1_15	50.0	50.0	50.0	50.0
N4sid_1_30	50.0	50.0	50.0	50.0
N4sid_1_60	50.0	50.0	50.0	50.0
N4sid_1_200	49.5	48.0	47.0	50.0
N4sid_10_15	49.0	51.0	50.5	49.5
N4sid_10_30	60.5	52.5	50.0	50.0
N4sid_10_60	50.0	50.5	50.0	50.0
N4sid_10_200	60.0	56.0	50.0	50.0
N4sid_50_15	42.5	40.5	47.0	49.5
N4sid_50_30	49.0	49.5	50.0	50.0
N4sid_50_60	54.0	50.0	50.0	50.0
N4sid_50_200	50.0	49.0	50.0	50.0
ssa_1_15	50.0	50.0	50.0	50.0
ssa_1_30	43.0	50.0	50.0	50.0
ssa_1_60	50.5	50.0	50.0	50.0
ssa_1_200	33.5	44.0	48.5	50.0
ssa_10_15	27.0	32.5	36.5	48.5
ssa_10_30	37.0	49.5	50.0	50.0
ssa_10_60	46.5	50.0	50.0	50.0
ssa_10_200	29.5	38.5	42.5	49.5
ssa_50_15	35.0	49.5	50.0	50.0
ssa_50_30	33.5	45.0	49.0	50.0
ssa_50_60	47.0	50.0	50.0	50.0
ssa_50_200	28.0	34.5	36.5	46.5

Table 5.3: Rows: different models with naming convention (model)_(number of training trajectories)_(number of time-steps in each trajectory). Columns: lengths of the validation trajectories. Results are in %.

to "supervised" learning of LDS matrices (or better to say, learning LDS through trajectories associated with LDSs in given cluster).

Agglomerative clustering can be described as at each step trying to merge two clusters S_a, S_b such

that:

$$\{S_a, S_b\} = \arg \min_{S_i, S_j \in \boldsymbol{S}, i \neq j} \mathcal{A}(S_i, S_j)$$

where $\mathcal{A}(S_i, S_j)$ is a function to measure affinity between two clusters.

The high-level view on the algorithm is as follows: first, for each trajectory learn the LDS and consider this LDS a cluster on its own in the space system matrices. Then, merge the two most affine systems and on their corresponding trajectories compute new LDS. Consider this LDS as a new cluster in the space of system matrices. Repeat this procedure until only two LDS computed on the corresponding trajectories remain. The discrimination phase is then the same as in the previous case, i.e. for new trajectory apply the Kalman filter with both sets of the system matrices and assign the state that produces smaller error.

Since the formulation for standard LDS identification methods is trivial and apparent from the high-level overview, we formalize more precisely only the POP method. In the k-th step of the algorithm there will be $N - k$ clusters, each corresponding to one LDS (i.e. here by "cluster" we mean that the single LDS was computed from cluster of trajectories that were merged), where N denotes the total number trajectories. Let $\boldsymbol{S}^{k-1} = \{S_1^{k-1}, S_2^{k-1}, \dots S_{N-k+1}^{k-1}\}$ be the set of clusters after $k-1$ steps. Let S_i^{k-1}, S_j^{k-1} be the most affine LDSs according to affinity measure $\mathcal{A}$. Let S_l^k be the newly produced set of trajectories. The LDS computed from trajectories in this cluster will be based on objective function of the form

$$\min_{\substack{f_t, m_t, G \\ F, \omega_t, \nu_t}} \sum_{Y^i \in S_l^k} \sum_{t \geq 1} \left\| Y_t^i - f_t \right\|^2 + c_1 \sum_{t \geq 1} \nu_t^2 + c_2 \sum_{t \geq 1} \omega_t^2 \tag{5.18}$$

with constraints

$$m_t = G m_{t-1} + \omega_t \tag{5.19}$$

$$f_t = F' m_t + \nu_t \tag{5.20}$$

The pseudo-code for this algorithm is summarized in Algorithm (2).

This algorithm obviously relies very much on the performance of the previous labeled approach. For this reason we left for now the formulation quite general (e.g. the affinity measure $\mathcal{A}$ can be later specified as the Frobenius norm) and we will proceed with specific implementation after we solve all the problems in the previous approach with the real measurement data.

There are, of course, other possible clustering algorithms like e.g. k-means (for which we are also working on formulation within the framework of joint clustering and LDS identification), but these are out of scope of this work.

Algoritmus 2: Discrimination with unlabeled trajectories algorithm

Input : Training set of trajectories unlabeled, validation set of trajectories labeled 0 or 1 (if available)

Output: accuracy of given model

Training phase;
Learn LDS fro each trajectory;
while *number of clusters* $\neq 2$ **do**
    Join two clusters of trajectories that are most affine in the space of system matrices;
    Learn LDS from trajectories in the newly produced cluster
end
Denote the two resulting LDSs as (G_0, F_0, V_0, W_0) and (G_1, F_1, V_1, W_1);
Discrimination phase;
for *trajectory in validation set* **do**
    Apply Kalman filter with (G_0, F_0, V_0, W_0) and compute error $\mathbf{e}_0$
    Apply Kalman filter with (G_1, F_1, V_1, W_1) **and compute error** $\mathbf{e}_1$
    if $\boldsymbol{e_1 \geq e_0}$ **then**
        assign state $|0\rangle$**;**
    else
        assign state $|1\rangle$**;**
    end
end
Compute the accuracy of the assigned labels

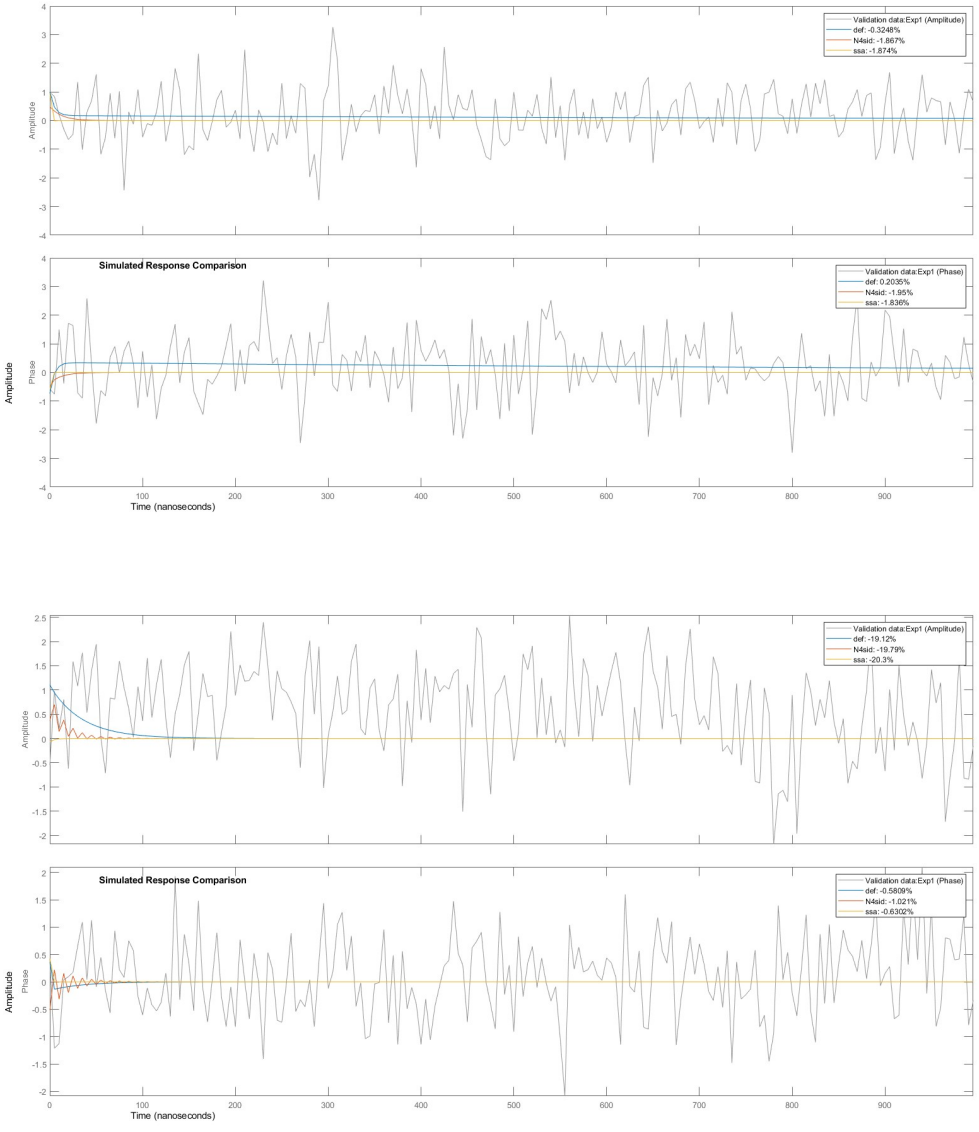


Figure 5.7: Comparison of models trained on 10 trajectories with 200 time-steps. Upper: trained on trajectories corresponding to $|0\rangle$, lower: trained on trajectories corresponding to $|1\rangle$.

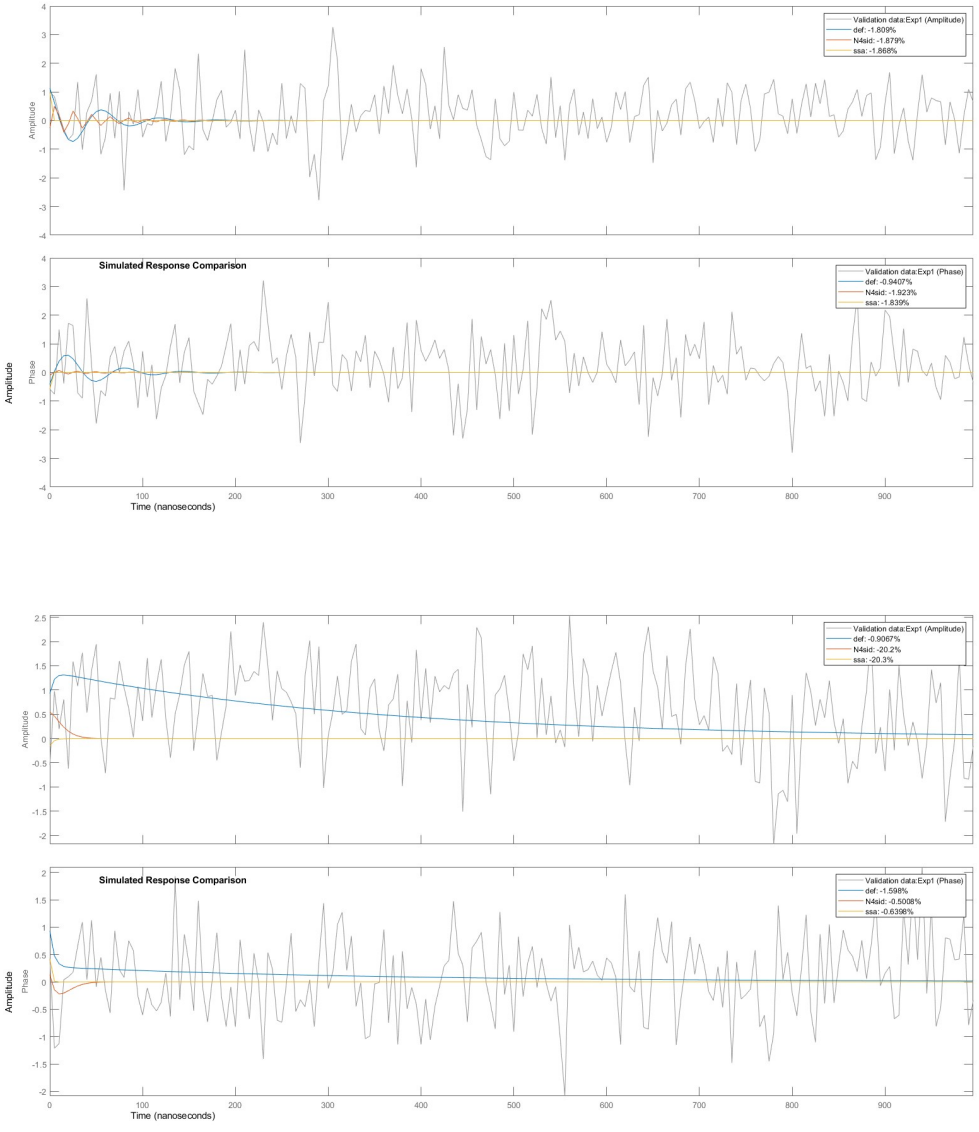


Figure 5.8: Comparison of models trained on 50 trajectories with 200 time-steps. Upper: trained on trajectories corresponding to $|0\rangle$, lower: trained on trajectories corresponding to $|1\rangle$.

Chapter 6

Conclusion

In this master's thesis, we have explored several key areas in the field of quantum technology, particularly focusing on quantum dots, their measurement techniques, and the challenges in controlling both closed and open quantum systems. Our journey through these topics has led to the development of new methods for determining the states of quantum systems. This conclusion brings together the main findings of our research, acknowledges its limitations, and suggests directions for future study.

We started the thesis with a brief overview of quantum dots. As GaAs QDs were used in our following research, these were of our main interest. Since the last chapter of the thesis was dedicated to improving the readout interpretation, we reviewed in greater detail the radio-frequency reflectometry technique and its application on QDs. We described the IQ-plane measurement and discussed the role of T_1 process in quantum state discrimination.

We then moved on to the topic of quantum control. We started with the case of a finite dimensional closed quantum system. For this case, there is a comprehensive theoretical framework based on the decomposition of the Lie group. In the main text, we provided an example of $SU(2)$ decomposition and a theorem for $SU(n)$ decomposition into planar rotations. A discussion of the more advanced concept of Cartan decomposition, as well as the concept of controllability, was done in Appendix B and A, respectively. We also presented a framework for optimal quantum control based on Pontryagin maximum principle, and we concluded the case of closed quantum systems with an example of globally optimal control using polynomial optimization.

Subsequently, we focused on the control of open quantum systems. Here, in contrast to closed systems, there is still no comprehensive theoretical framework and the question of the Markovianity of the system is crucial. We divided the discussion into open-loop and closed-loop methods. Within open-loop methods, there are two basic approaches. The first is based on decoherence-free subspaces given by the symmetries of the system. The second class of methods uses assumptions on the time scales of system-environment interactions. Here, we discussed dynamic decoupling as a leading method to reduce decoherence. From the closed-loop techniques, we focused primarily on methods based on reinforcement learning, as these are now experiencing great popularity with the boom in deep neural networks. We presented the basic concepts of reinforcement learning, possible implementations of the loss function and options for bypassing the seemingly necessary tomography of the quantum state.

In the last chapter, we focused on quantum state discrimination. We proposed a method based on the identification of a linear dynamic system and the subsequent application of a Kalman filter. For these purposes, we first summarized the fundamental ideas of the linear dynamic system and its identification, as well as the principle of the Kalman filter. A fundamental innovation was the use of polynomial optimization for LDS identification, the basics of which we summarized in Appendix C. We proposed two possible formulations, of which only one is currently computationally feasible for practical pur-

poses. Subsequently, we presented in detail the algorithm for quantum state discrimination for the case of marked measured trajectories. The idea was based on identifying two sets of system matrices, each identified on trajectories corresponding to a different state. The result of the new measurement was then interpreted according to which of the sets of matrices generated a smaller cumulative error using the Kalman filter on the new trajectory.

We have divided the discussion of data preparation and results into the case of 1D measurement and the case of measurement in the IQ-plane. For the first case, we used data from GaAs quantum dots, which first needed to be properly processed. We then applied the aforementioned algorithm both using POP and using traditional LDS identification methods. In doing so, we considered models learned on a different number of trajectories of different lengths. We also tried different lengths of test trajectories. We chose a similar procedure for the IQ data collection. Here, however, due to the absence and subsequent unsuccessful installation of the IQ-demodulator, we had to implement a data generator based on an educated guess. Due to the nature of the data and the computational complexity, in this case we only considered standard LDS identification methods and omitted the results for POP.

From these first results, it is clear that the level of accuracy has not yet been reached as for the currently standard methods based on averaging data points. In the case of POP, the issue of regularization and over-fitting must first be solved, but even then we see the problem with traditional identification methods that one set of LDS matrices provides a smaller error on any trajectory than the other. However, there are reasons for mild optimism. If these problems can be overcome, preliminary results suggest that it may be possible to recognize trajectory type from fewer data points, and thus more quickly than is necessary with current methods. Such a feature would be very useful, for example, for error correction, where the measurement speed is a crucial parameter.

A first step to overcome the current problems in further research will be to consider higher dimensional LDS. The approximation of real dynamics can then be much more accurate and, in principle, can also capture phenomena that could not be accurately projected into a two-dimensional system. Due to the computational complexity, it is probably not practical to consider this approach for POP at the present time, however, active research is underway to reduce the computational demands using matrix sparsity and other methods that can change this situation. If increasing the dimension turns out to be the right way, it will also be possible to continue working on the algorithm for the case of unmarked trajectories presented at the end of the last chapter. The latter could subsequently provide an advantage for early experimental devices.

As such, this work represents not only a comprehensive overview of quantum control methods and their possible use for quantum dots, but also an attempt at an original synthesis of quantum technologies and computer science. Although the current results do not bring significant improvement in the accuracy of measurement interpretation, the ideas presented here pave the way for further research in the field of optimal readout using methods fundamentally different from those currently used.

Bibliography

[1] Altafini, C. "Controllability properties for finite dimensional quantum Markovian master equations." In: *Journal of Mathematical Physics* 44.6 (2003), pp. 2357–2372.

[2] Anjos, M. F. and Lasserre, J. B. *Handbook on semidefinite, conic and polynomial optimization*. Vol. 166. Springer Science & Business Media, 2011.

[3] Arulkumaran, K., Deisenroth, M. P., Brundage, M., and Bharath, A. A. "Deep reinforcement learning: A brief survey." In: *IEEE Signal Processing Magazine* 34.6 (2017), pp. 26–38.

[4] Åström, K. J. and Eykhoff, P. "System identification—a survey." In: *Automatica* 7.2 (1971), pp. 123–162.

[5] Auger, F., Hilairet, M., Guerrero, J. M., Monmasson, E., Orlowska-Kowalska, T., and Katsura, S. "Industrial applications of the Kalman filter: A review." In: *IEEE Transactions on Industrial Electronics* 60.12 (2013), pp. 5458–5471.

[6] Blanco, V. and Puerto, J. "Some algebraic methods for solving multiobjective polynomial integer programs." In: *Journal of Symbolic Computation* 46.5 (2011), pp. 511–533.

[7] Bondar, D. I., Jacobs, K., Korpas, G., Marecek, J., et al. "Quantum optimal control via polynomial optimization: A globally convergent approach." In: *arXiv preprint arXiv:2209.05790* (2022).

[8] Bondar, D. I., Popovych, Z., Jacobs, K., Korpas, G., and Marecek, J. "Recovering models of open quantum systems from data via polynomial optimization: Towards globally convergent quantum system identification." In: *arXiv preprint arXiv:2203.17164* (2022).

[9] Boussaid, N., Caponigro, M., and Chambrion, T. "Weakly coupled systems in quantum control." In: *IEEE transactions on automatic control* 58.9 (2013), pp. 2205–2216.

[10] Breiding, P. and Timme, S. "HomotopyContinuation. jl: A package for homotopy continuation in Julia." In: *Mathematical Software–ICMS 2018: 6th International Conference, South Bend, IN, USA, July 24-27, 2018, Proceedings 6*. Springer. 2018, pp. 458–465.

[11] Breuer, H.-P. and Petruccione, F. *The theory of open quantum systems*. Oxford University Press, USA, 2002.

[12] Bukov, M., Day, A. G., Sels, D., Weinberg, P., Polkovnikov, A., and Mehta, P. "Reinforcement learning in different phases of quantum control." In: *Physical Review X* 8.3 (2018), p. 031086.

[13] Bunse-Gerstner, A., Byers, R., and Mehrmann, V. "A chart of numerical methods for structured eigenvalue problems." In: *SIAM Journal on Matrix Analysis and Applications* 13.2 (1992), pp. 419–453.

[14] Burgarth, D. and Yuasa, K. "Quantum system identification." In: *Physical Review Letters* 108.8 (2012), p. 080502.

[15] Carr, H. Y. and Purcell, E. M. "Effects of diffusion on free precession in nuclear magnetic resonance experiments." In: *Physical review* 94.3 (1954), p. 630.

[16] Chambrion, T., Mason, P., Sigalotti, M., and Boscain, U. "Controllability of the discrete-spectrum Schrödinger equation driven by an external field." In: *Annales de l'Institut Henri Poincaré C* 26.1 (2009), pp. 329–349.

[17] Chefles, A. "Quantum state discrimination." In: *Contemporary Physics* 41.6 (2000), pp. 401–424.

[18] D'Alessandro, D. "Constructive decomposition of the controllability Lie algebra for quantum systems." In: *IEEE transactions on automatic control* 55.6 (2010), pp. 1416–1421.

[19] De Vega, I. and Alonso, D. "Dynamics of non-Markovian open quantum systems." In: *Reviews of Modern Physics* 89.1 (2017), p. 015001.

[20] Dirr, G, Helmke, U, Kurniawan, I, and Schulte-Herbrüggen, T. "Lie-semigroup structures for reachability and control of open quantum systems: Kossakowski-Lindblad generators form Lie wedge to Markovian channels." In: *Reports on Mathematical Physics* 64.1-2 (2009), pp. 93–121.

[21] Doherty, A. C., Habib, S., Jacobs, K., Mabuchi, H., and Tan, S. M. "Quantum feedback control and classical control theory." In: *Physical Review A* 62.1 (2000), p. 012105.

[22] Dong, D., Chen, C., Jiang, M., Wang, L.-C., et al. *Quantum control and quantum information technology*. 2013.

[23] Dridi, G., Liu, K., and Guérin, S. "Optimal robust quantum control by inverse geometric optimization." In: *Physical Review Letters* 125.25 (2020), p. 250403.

[24] d'Alessandro, D. *Introduction to quantum control and dynamics*. CRC press, 2021.

[25] Fabian, J., Matos-Abiague, A., Ertler, C., Stano, P., and Zutic, I. "Semiconductor spintronics." In: *arXiv preprint arXiv:0711.1461* (2007).

[26] Fleming, W. H. and Rishel, R. W. *Deterministic and stochastic optimal control*. Vol. 1. Springer Science & Business Media, 2012.

[27] Floether, F. F., De Fouquieres, P., and Schirmer, S. G. "Robust quantum gates for open systems via optimal control: Markovian versus non-Markovian dynamics." In: *New Journal of Physics* 14.7 (2012), p. 073023.

[28] Gambetta, J., Braff, W., Wallraff, A, Girvin, S., and Schoelkopf, R. "Protocols for optimal readout of qubits using a continuous quantum nondemolition measurement." In: *Physical Review A* 76.1 (2007), p. 012325.

[29] Ghaddar, B., Marecek, J., and Mevissen, M. "Optimal power flow as a polynomial optimization problem." In: *IEEE Transactions on Power Systems* 31.1 (2015), pp. 539–546.

[30] Glaser, S. J., Boscain, U., Calarco, T., Koch, C. P., Köckenberger, W., Kosloff, R., Kuprov, I., Luy, B., Schirmer, S., Schulte-Herbrüggen, T., et al. "Training Schrödinger's cat: Quantum optimal control: Strategic report on current status, visions and goals for research in Europe." In: *The European Physical Journal D* 69 (2015), pp. 1–24.

[31] Goldin, Y. and Avishai, Y. "Nonlinear response of a Kondo system: Perturbation approach to the time-dependent Anderson impurity model." In: *Physical Review B* 61.24 (2000), p. 16750.

[32] Hahn, E. L. "Spin echoes." In: *Physical review* 80.4 (1950), p. 580.

[33] Hanson, R., Kouwenhoven, L. P., Petta, J. R., Tarucha, S., and Vandersypen, L. M. "Spins in few-electron quantum dots." In: *Reviews of modern physics* 79.4 (2007), p. 1217.

[34] Helgason, S. *Differential geometry, Lie groups, and symmetric spaces*. Academic press, 1979.

[35] Hu, C., Chen, W., Chen, Y., Liu, D., et al. "Adaptive Kalman filtering for vehicle navigation." In: *Journal of Global Positioning Systems* 2.1 (2003), pp. 42–47.

[36] Huang, G. M., Tarn, T. J., and Clark, J. W. "On the controllability of quantum-mechanical systems." In: *Journal of Mathematical Physics* 24.11 (1983), pp. 2608–2618.

[37] Jansson, M. "Subspace identification and ARX modeling." In: *IFAC Proceedings Volumes* 36.16 (2003), pp. 1585–1590.

[38] Jurdjevic, V. and Sussmann, H. J. "Control systems on Lie groups." In: *Journal of Differential equations* 12.2 (1972), pp. 313–329.

[39] Kallush, S., Dann, R., and Kosloff, R. "Controlling the uncontrollable: Quantum control of open-system dynamics." In: *Science Advances* 8.44 (2022), eadd0828.

[40] Kalman, R. E. "A new approach to linear filtering and prediction problems." In: (1960).

[41] Keith, D, Gorman, S., Kranz, L, He, Y, Keizer, J., Broome, M., and Simmons, M. "Benchmarking high fidelity single-shot readout of semiconductor qubits." In: *New Journal of Physics* 21.6 (2019), p. 063011.

[42] Knapp, A. W. and Knapp, A. W. *Lie groups beyond an introduction*. Vol. 140. Springer, 1996.

[43] Kobayashi, K. and Hiraishi, K. "Optimal control of probabilistic Boolean networks using polynomial optimization." In: *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences* 95.9 (2012), pp. 1512–1517.

[44] Koch, C. P. "Controlling open quantum systems: tools, achievements, and limitations." In: *Journal of Physics: Condensed Matter* 28.21 (2016), p. 213001.

[45] Konnov, A. and Krotov, V. F. "DETERMINISTIC SYSTEMS-On Global Methods of Successive Improvement of Controlled Processes." In: *Automation and Remote Control* 60.10 (1999), pp. 1427–1436.

[46] Koppens, F. H., Buizert, C., Tielrooij, K.-J., Vink, I. T., Nowack, K. C., Meunier, T., Kouwenhoven, L., and Vandersypen, L. "Driven coherent oscillations of a single electron spin in a quantum dot." In: *Nature* 442.7104 (2006), pp. 766–771.

[47] Kozdoba, M., Marecek, J., Tchrakian, T., and Mannor, S. "On-line learning of linear dynamical systems: Exponential forgetting in kalman filters." In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 4098–4105.

[48] Lasserre, J. B. "Convergent SDP-relaxations in polynomial optimization with sparsity." In: *SIAM Journal on optimization* 17.3 (2006), pp. 822–843.

[49] Lasserre, J. B. "Global optimization with polynomials and the problem of moments." In: *SIAM Journal on optimization* 11.3 (2001), pp. 796–817.

[50] Lasserre, J. B. *Moments, positive polynomials and their applications*. Vol. 1. World Scientific, 2009.

[51] Latorre, F., Rolland, P., and Cevher, V. "Lipschitz constant estimation of neural networks via sparse polynomial optimization." In: *arXiv preprint arXiv:2004.08688* (2020).

[52] Laurent, M. "Sums of squares, moment matrices and optimization over polynomials." In: *Emerging applications of algebraic geometry* (2009), pp. 157–270.

[53] Lee, E. B. and Markus, L. *Foundations of optimal control theory*. Vol. 87. Wiley New York, 1967.

[54] Lidar, D. A., Chuang, I. L., and Whaley, K. B. "Decoherence-free subspaces for quantum computation." In: *Physical Review Letters* 81.12 (1998), p. 2594.

[55] Loss, D. and DiVincenzo, D. P. "Quantum computation with quantum dots." In: *Physical Review A* 57.1 (1998), p. 120.

[56] Magesan, E., Gambetta, J. M., Córcoles, A. D., and Chow, J. M. "Machine learning for discriminating quantum measurement trajectories and improving readout." In: *Physical review letters* 114.20 (2015), p. 200501.

[57] Magnus, W. "On the exponential solution of differential equations for a linear operator." In: *Communications on pure and applied mathematics* 7.4 (1954), pp. 649–673.

[58] Meiboom, S. and Gill, D. "Modified spin-echo method for measuring nuclear relaxation times." In: *Review of scientific instruments* 29.8 (1958), pp. 688–691.

[59] Metz, F. and Bukov, M. "Self-correcting quantum many-body control using reinforcement learning with tensor networks." In: *Nature Machine Intelligence* 5.7 (2023), pp. 780–791.

[60] Murnaghan, F. D. "The orthogonal and symplectic groups." In: *Commun. Dublin Inst. Adv. Stud., A* 13 (1958), p. 156.

[61] Nachum, O., Norouzi, M., Xu, K., and Schuurmans, D. "Bridging the gap between value and policy based reinforcement learning." In: *Advances in neural information processing systems* 30 (2017).

[62] Niu, M. Y., Boixo, S., Smelyanskiy, V. N., and Neven, H. "Universal quantum control through deep reinforcement learning." In: *npj Quantum Information* 5.1 (2019), p. 33.

[63] Palao, J. P. and Kosloff, R. "Optimal control theory for unitary transformations." In: *Physical Review A* 68.6 (2003), p. 062308.

[64] Parrilo, P. A. "Semidefinite programming relaxations for semialgebraic problems." In: *Mathematical programming* 96 (2003), pp. 293–320.

[65] Pauwels, E., Henrion, D., and Lasserre, J.-B. "Inverse optimal control with polynomial optimization." In: *53rd IEEE Conference on Decision and Control*. IEEE. 2014, pp. 5581–5586.

[66] Press, D., Ladd, T. D., Zhang, B., and Yamamoto, Y. "Complete quantum control of a single quantum dot spin using ultrafast optical pulses." In: *Nature* 456.7219 (2008), pp. 218–221.

[67] Price, J. F. *Lie groups and compact groups*. Vol. 25. Cambridge University Press, 1977.

[68] Putinar, M. "Positive polynomials on compact semi-algebraic sets." In: *Indiana University Mathematics Journal* 42.3 (1993), pp. 969–984.

[69] Rebentrost, P, Serban, I, Schulte-Herbrüggen, T., and Wilhelm, F. "Optimal control of a qubit coupled to a non-Markovian environment." In: *Physical review letters* 102.9 (2009), p. 090401.

[70] Reich, D. M., Ndong, M., and Koch, C. P. "Monotonically convergent optimization in quantum control using Krotov's method." In: *The Journal of chemical physics* 136.10 (2012).

[71] Renner, P. and Schmedders, K. "A polynomial optimization approach to principal–agent problems." In: *Econometrica* 83.2 (2015), pp. 729–769.

[72] Rossmann, W. *Lie groups: an introduction through linear groups*. Vol. 5. Oxford University Press, USA, 2006.

[73] Schaul, T., Quan, J., Antonoglou, I., and Silver, D. "Prioritized experience replay." In: *arXiv preprint arXiv:1511.05952* (2015).

[74] Schmidt, R., Negretti, A., Ankerhold, J., Calarco, T., and Stockburger, J. T. "Optimal control of open quantum systems: Cooperative effects of driving and dissipation." In: *Physical review letters* 107.13 (2011), p. 130404.

[75] Schollwöck, U. "The density-matrix renormalization group in the age of matrix product states." In: *Annals of physics* 326.1 (2011), pp. 96–192.

[76] Simpkins, A. “System identification: Theory for the user, (ljung, l.; 1999)[on the shelf].” In: *IEEE Robotics & Automation Magazine* 19.2 (2012), pp. 95–96.

[77] Souza, A. M., Álvarez, G. A., and Suter, D. “Robust dynamical decoupling.” In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 370.1976 (2012), pp. 4748–4769.

[78] Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

[79] Takeda, K., Kamioka, J., Otsuka, T., Yoneda, J., Nakajima, T., Delbecq, M. R., Amaha, S., Allison, G., Kodera, T., Oda, S., et al. “A fault-tolerant addressable spin qubit in a natural silicon quantum dot.” In: *Science advances* 2.8 (2016), e1600694.

[80] Tal-Ezer, H. and Kosloff, R. “An accurate and efficient scheme for propagating the time dependent Schrödinger equation.” In: *The Journal of chemical physics* 81.9 (1984), pp. 3967–3971.

[81] Taylor, J., Petta, J., Johnson, A., Yacoby, A., Marcus, C., and Lukin, M. “Relaxation, dephasing, and quantum control of electron spins in double quantum dots.” In: *Physical Review B* 76.3 (2007), p. 035315.

[82] Van Houten, H, Beenakker, C., and Staring, A. “Coulomb-blockade oscillations in semiconductor nanostructures.” In: *Single charge tunneling: Coulomb blockade phenomena in nanostructures* (1992), pp. 167–216.

[83] Van Overschee, P. and De Moor, B. *Subspace identification for linear systems: Theory—Implementation—Appli* Springer Science & Business Media, 2012.

[84] Vigneau, F., Fedele, F., Chatterjee, A., Reilly, D., Kuemmeth, F., Gonzalez-Zalba, M. F., Laird, E., and Ares, N. “Probing quantum devices with radio-frequency reflectometry.” In: *Applied Physics Reviews* 10.2 (2023).

[85] Viola, L. “Advances in decoherence control.” In: *Journal of Modern Optics* 51.16-18 (2004), pp. 2357–2367.

[86] Waki, H., Kim, S., Kojima, M., and Muramatsu, M. “Sums of squares and semidefinite program relaxations for polynomial optimization problems with structured sparsity.” In: *SIAM Journal on Optimization* 17.1 (2006), pp. 218–242.

[87] Wang, J., Magron, V., and Lasserre, J.-B. “Chordal-TSSOS: a moment-SOS hierarchy that exploits term sparsity with chordal extension.” In: *arXiv preprint arXiv:2003.03210* (2020).

[88] Wang, J., Magron, V., and Lasserre, J.-B. “TSSOS: A moment-SOS hierarchy that exploits term sparsity.” In: *SIAM Journal on optimization* 31.1 (2021), pp. 30–58.

[89] Watkins, C. J. and Dayan, P. “Q-learning.” In: *Machine learning* 8 (1992), pp. 279–292.

[90] West, M. and Harrison, J. *Bayesian forecasting and dynamic models*. Springer Science & Business Media, 2006.

[91] Yoneda, J., Takeda, K., Otsuka, T., Nakajima, T., Delbecq, M. R., Allison, G., Honda, T., Kodera, T., Oda, S., Hoshi, Y., et al. “A quantum-dot spin qubit with coherence limited by charge noise and fidelity higher than 99.9%.” In: *Nature nanotechnology* 13.2 (2018), pp. 102–106.

[92] Zanardi, P. and Rasetti, M. “Noiseless quantum codes.” In: *Physical Review Letters* 79.17 (1997), p. 3306.

[93] Zhou, K. and Doyle, J. C. *Essentials of robust control*. Vol. 104. Prentice hall Upper Saddle River, NJ, 1998.

[94] Zhou, Q. and Marecek, J. “Proper learning of linear dynamical systems as a non-commutative polynomial optimisation problem.” In: *arXiv preprint arXiv:2002.01444* (2020).

Appendix A

Controllability

Having a control system described by the Schrödinger equation (3.10), controllability theory focuses on the study of so-called reachable set - set of states that can be obtained with a given experimental set-up. In the language of quantum operations, the question may be as follows: what unitary operations can we perform on a given device? Here we, of course, consider closed quantum systems, which will be of the main interest in this section.

For the closed finite-dimensional systems, there is a complete framework to determine the controllability of the system. In short, it can be said that, having the Hamiltonian (3.3), the system is controllable provided the Lie algebra spanned by the nested commutators of H_0 and H_k is full rank. In another words, the system is controllable if the set of matrices that can be obtained through (3.10) is the set of all unitary matrices $U(n)$ [24, 36].

Let $U(t, u)$ be the solution of (3.10) at time t with control function u. We define the reachable set at time $T > 0$, $\mathcal{R}(T)$, as the set of all the unitary matrices $\tilde{U}$ such that there exists an admissible control function u with $U(T, u) = \tilde{U}$. The reachable set $\mathcal{R}$ is then defined as $\mathcal{R} = \bigcup_{T\geq 0} \mathcal{R}(T)$. In the following we will consider u from the set of piecewise constant functions. We also note that this is sometimes called operator controllability or complete controllability to distinguish it from the state controllability.

The following theorem is the basis for testing the controllability of the system [24]:

Theorem 3 (Lie algebra rank condition)**.** The set of reachable states for system (3.10) is dense (in the subset topology induced by $U(n)$) in the connected Lie group $e^{\mathcal{L}}$ associated with the Lie algebra $\mathcal{L}$ generated by $\{-iH(u)|u \in \mathcal{U}\}$, $\mathcal{U}$ being the set of admissible functions. Furthermore, if $e^{\mathcal{L}}$ is a Lie subgroup of $U(n)$ then $\mathcal{R} = e^{\mathcal{L}}$.

The proof of this theorem can be found in Appendix D of [24]. When the dynamical Lie algebra $\mathcal{L}$ is equal to $u(n)$ then $e^{\mathcal{L}} = U(n)$ and $\mathcal{R} = U(n)$. The statement of the theorem is that all the information about $\mathcal{R}$ is contained in the dynamical Lie algebra $\mathcal{L}$. The crucial discussion is then whether $\mathcal{R} = e^{\mathcal{L}}$ and it can be shown that the assumption on $e^{\mathcal{L}}$ being the subgroup of $U(n)$ is essential. Such a discussion can be reduced to a question whether a Lie group is a Lie subgroup of another Lie group, which is a typical question a Lie theory. The answer is provided by so-called closed subgroup theorem [72] which roughly states that $e^{\mathcal{L}}$ is a Lie subgroup if and only if $e^{\mathcal{L}}$ is closed in the topology of $U(n)$.

This property can be checked using so-called Killing form defined as

$$\langle x, y\rangle_K \coloneqq \mathrm{Tr}\Big\{\mathrm{ad}_x \mathrm{ad}_y\Big\}, \tag{A.1}$$

where $\mathrm{ad}_x : L \to [x, L]$ is a matrix representation of linear operator on $\mathcal{L}$. The Proposition 4.27 in [42] then states that if the Killing form of $\mathcal{L}$ is negative definite, then $e^{\mathcal{L}}$ is a compact (and therefore closed) Lie group. This, however, provides only sufficient condition. Another approach utilizing the structure of

Lie algebra $\mathcal{L}$ is based on the fact that any Lie subalgebra of $u(n)$ is reductive [18], i.e. can be written as $\mathcal{L} = \mathcal{S} \oplus \mathcal{Z}$, where $\mathcal{S}$ is semisimple and $\mathcal{Z}$ is Abelian center - subspace of elements commuting with all of $\mathcal{L}$. The question whether $e^{\mathcal{S}}$ is a Lie subgroup of $U(n)$ is answered by the following theorem [67]:

Theorem 4 ($e^{\mathcal{S}}$ is subgroup of $U(n)$)**.** Let $\mathcal{S}$ be the a semisimple Lie subalgebra of a Lie algebra $\mathcal{G}$ and $e^{\mathcal{S}}$ a compact Lie group (in our case $U(n)$). Then $e^{\mathcal{S}}$ is a compact Lie subgroup of $e^{\mathcal{G}}$.

This reduces the analysis only to whether $e^{\mathcal{Z}}$ is a subgroup of $U(n)$. This is often very easy as $\mathcal{Z}$ tends to be low-dimensional and commutes with the whole group. The standard procedure is then to first find the basis of $\mathcal{L}$, identify $\mathcal{Z}$ and check whether $e^{\mathcal{Z}}$ is closed. Since $e^{\mathcal{L}}$ is then a subgroup of $U(n)$, theorem (3) implies that $\mathcal{R} = e^{\mathcal{L}}$. Simple recursive procedure to generate a basis of $\mathcal{L}$ is given in [24].

An interesting question is also the dependence of the reachable set $\mathcal{R}(t)$ on time t. This is answered by the following theorem [38]:

Theorem 5 ($e^{\mathcal{S}}$ is subgroup of $U(n)$)**.** Assume that the dynamical Lie algebra $\mathcal{L}$ is semisimple ($\mathcal{L} = su(n)$ for controllable systems). Then there exists a critical time T_c such that for every $T \geq T_c$, $\mathcal{R}(T) = e^{\mathcal{L}}$.

Accurate estimation of T_c is an active area of research. Detailed discussion of pure state and density matrix controllability can be found in [24].

Regarding the open quantum systems, any complete framework for controllability analysis does not exist and the extension for these purposes remains an open problem to date [44]. As the evolution is no longer unitary, controllability can not be deduced purely from the Hamiltonian. The dissipative processes can prevent or even enable reaching some states. The simplest example is the impossibility of transition between pure states under fast decoherence.

There are two ways to approach the controllability analysis of the open quantum systems. First, dealing with the complete system-environment description. In such a case some of the methods for the unitary evolution can be employed. The advantage of this approach is that there are no a priori assumptions on the dynamics of the reduced system. Although the description and the controllability analyses of the complete system-environment dynamics is extremely challenging, some advances in controllability analysis of infinite-dimensional system may be utilized [9, 16].

Second, considering controllability analyses solely for the reduced dynamics. Rigorous analyses of the reachable sets has been done only for the Markovian case [1], where the Lindblad operators were shown to form a Lie wedge [20]. For non-Markovian systems no rigorous analyses has been done to date [44].

Appendix B

Cartan decomposition

Cartan decomposition of Lie groups represents a very general framework for achieving a desired unitary evolution given a model of a closed quantum system. Here we provide a short overview of this concept.

Definition 2 (Cartan decomposition)**.** Let $\mathcal{L}$ be a semisimple Lie algebra. A decomposition of the form

$$\mathcal{L} = \mathcal{K} \oplus \mathcal{P},\ \mathcal{P} \coloneqq \mathcal{K}^{\perp} \tag{B.1}$$

is called Cartan decomposition if:

1.
$$[\mathcal{K}, \mathcal{K}] \subseteq \mathcal{K} \tag{B.2}$$
(i.e. $\mathcal{K}$ is subalgebra of $\mathcal{L}$)

2.
$$[\mathcal{K}, \mathcal{P}] \subseteq \mathcal{P} \tag{B.3}$$

3.
$$[\mathcal{P}, \mathcal{P}] \subseteq \mathcal{K}. \tag{B.4}$$

The following theorem is the basis for utilizing the Cartan decomposition for control purposes.

Theorem 6 (PK decomposition)**.** For a semisimple Lie algebra $\mathcal{L}$ (such as $su(n)$) with Cartan decomposition (B.1), every $X \in e^{\mathcal{L}}$ can be written in the form

$$X = PK, \tag{B.5}$$

where $P \in e^{\mathcal{P}}$ and $K \in e^{\mathcal{K}}$.

The proof of this theorem can be found e.g. in [34]. This paves the way for yet another decomposition. Consider $\mathcal{A}$ being a subalgebra of $\mathcal{L}$ and simultaneously a subspace of $\mathcal{L}$. Because $[\mathcal{A}, \mathcal{A}] \subseteq \mathcal{A} \subseteq \mathcal{P}$ and $[\mathcal{A}, \mathcal{A}] \subseteq [\mathcal{P}, \mathcal{P}] \subseteq \mathcal{K}$, then clearly $[\mathcal{A}, \mathcal{A}] = \{0\}$, i.e. $\mathcal{A}$ is Abelian. For a given Cartan decomposition of $\mathcal{L}$, maximal Abelian subalgebra of $\mathcal{P}$ is called Cartan subalgebra of $\mathcal{L}$. The following two theorems are the necessary ingredients for so-called KAK decomposition [24].

Theorem 7 (Every element of $\mathcal{P}$ belongs to a Cartan subalgebra)**.** Let $\mathcal{A}$ be a Cartan subalgebra associated with the Cartan decomposition (B.1). Then

$$\bigcup_{K \in e^{\mathcal{K}}} K \mathcal{A} K^{-1} = \mathcal{P}. \tag{B.6}$$

Theorem 8 (All the Cartan subalgebras are conjugate via K)**.** For any two subalgebras $\mathcal{A}, \tilde{\mathcal{A}}$ in $\mathcal{P}$, there exists $K \in e^{\mathcal{K}}$ such that

$$K\mathcal{A}K^{-1} = \tilde{\mathcal{A}}. \tag{B.7}$$

With these two theorems at hand it is straightforward to prove the KAK decomposition:

Theorem 9 (KAK decomposition)**.** For a semisimple Lie algebra $\mathcal{L}$ with Cartan decomposition (B.1), every $X \in e^{\mathcal{L}}$ can be written in the form

$$X = K_1 A K_2, \tag{B.8}$$

with $K_1, K_2 \in e^{\mathcal{K}}$ and $A \in e^{\mathcal{A}}$.

In particular, it can be shown [24] that for every $X \in SU(n)$ the matrices K_1, K_2 are orthogonal and

$$A \coloneqq \mathrm{diag}(e^{i\alpha_1}, \ldots, e^{i\alpha_{n-1}}, e^{-i\sum_{k=1}^{n-1}\alpha_k}). \tag{B.9}$$

This important result, nevertheless, doesn't tell us anything about the way of finding the factors in the KAK decomposition. For $su(n)$ (which is relevant for quantum computing), however, Cartan classification of symmetric spaces revealed that there are, up to conjugation, only three types of decompositions, often denoted AI, AII, AIII. These are:

- AI:
$$\mathcal{K} = so(n) \tag{B.10}$$
- AII:
$$\mathcal{K} = sp(\frac{n}{2}) \tag{B.11}$$
for n even with $sp(.)$ denoting the symplectic Lie algebra
- AIII:
$$\mathcal{K} = \mathrm{span}\Big\{\begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix}\Big| A \in u(p), B \in u(q), p+q = n, \mathrm{Tr}\{A\} + \mathrm{Tr}\{B\} = 0\Big\}. \tag{B.12}$$

For each type of decomposition, corresponding subalgebra $\mathcal{A}$ can be found [24]. E.g. for the type AI, $\mathcal{A}$ is the subalgebra of diagonal matrices.

To calculate the factors in the KAK decomposition, the first step is often to put the subspaces $\mathcal{K}, \mathcal{P}$ in the standard form via some similarity transform T. Finding such a transform can be significantly simplified using the concept of Cartan involution [24]. Having the standard for, the goal is then to reduce the calculation KAK factors for each type of Cartan decomposition to so-called structured eigenvalue problem. E.g. for type AI we have $U = O_1 D O_2$ with O_1, O_2 orthogonal and D diagonal matrix. By simple manipulation we get

$$UU^T O_1 = O_1 D^2, \tag{B.13}$$

i.e. finding the eigenvalues and eigenvectors of UU^T. Calculating O_1, D and finally O_2 is then straightforward. There exist several algorithms for solving such problems [13].

Let us demonstrate the Cartan decomposition on an example. Very simple and yet for quantum dots very relevant model is two weakly coupled spins with Ising interaction. The Schrödinger operator equation, after proper scaling and considering the z component of the magnetic field constant, reads

$$\dot{U} = -i\Big[\sigma_z \otimes \sigma_z + \Big(\sum_{k=x,y} u_k(t)\sigma_k \otimes \mathbb{I}\Big) + \Big(r\sum_{k=x,y} u_k(t)\mathbb{I} \otimes \sigma_k\Big)\Big]U, \tag{B.14}$$

where $r = \gamma_2/\gamma_1$ is the ration of gyromagnetic ratios of the two spins.

Here (for $r \neq \pm 1$) the algebra generated by the control matrices is

$$\mathcal{K} = \text{span}\{i\sigma_k \otimes \mathbb{I}, i\mathbb{I} \otimes \sigma_k | k = x, y, z\}. \tag{B.15}$$

This imply the corresponding Lie group of the form

$$e^{\mathcal{K}} = \{X_1 \otimes X_2 | X_1, X_2 \in SU(2)\}, \tag{B.16}$$

i.e. no entangling dynamics without the interaction. It is not hard to see that the orthogonal complement to $su(4)$ is

$$\mathcal{P} = \{\sigma_k \otimes \sigma_l | k, l = x, y, z\} \tag{B.17}$$

and the decomposition is of the type AI. The maximal subalgebra in $\mathcal{P}$ is

$$\mathcal{A} = \text{span}\{i\sigma_k \otimes \sigma_k | k = x, y, z\}. \tag{B.18}$$

Hence, according to KAK decomposition, every $U \in SU(4)$ can be written as

$$U = X_1 \otimes X_2 e^{it_1 \sigma_x \otimes \sigma_x} e^{it_2 \sigma_y \otimes \sigma_y} e^{it_3 \sigma_z \otimes \sigma_z} X_3 \otimes X_4 \tag{B.19}$$

for some real parameters $t_{1,2,3}$. As Pauli matrices are unitarily equivalent, we can write $\sigma_z = U_x \sigma_x U_x^\dagger = U_y \sigma_y U_y^\dagger$ and the previous relation can be further rewritten as

$$U = X_1 \otimes X_2 e^{it_1 \sigma_z \otimes \sigma_z} X_3 \otimes X_4 e^{it_2 \sigma_z \otimes \sigma_z} X_5 \otimes X_6 e^{it_3 \sigma_z \otimes \sigma_z} X_7 \otimes X_8 \tag{B.20}$$

with $X_j \in SU(2)$, $j = 1, \ldots, 8$. This means that we can perform the desired unitary operation by alternating very fast single spin operations (during which we can neglect the coupling) and turning off all the control fields and letting the the system evolve under $e^{i\sigma_z \otimes \sigma_z}$ for times $t_{1,2,3}$.

Appendix C

Polynomial optimization

Polynomial optimization is a branch of mathematics and optimization theory that deals with finding the minimum or maximum of a polynomial function over a given domain. In contrast to general global optimization problem

$$f^* := \inf_{\boldsymbol{x}}\{\boldsymbol{x}|\boldsymbol{x} \in \boldsymbol{K}\} \text{ for some feasible set}$$
$$\boldsymbol{K} = \{\boldsymbol{x} \in \mathbb{R}^n | g_j(\boldsymbol{x}) \geq 0, j = 1, \dots m\} \tag{C.1}$$

where $f, g_j : \mathbb{R}^n \to \mathbb{R}$ are arbitrary continuous functions, which is in general intractable, in polynomial optimization we restrict ourselves to the case where f, g_j are polynomials. It turns out, that with such a restriction it is in principle possible to obtain the global optimum f^* or its approximation that is as close as needed. It is important to mention that this remain true also for highly non-convex problems for which standard methods often struggle to find the global optima. This was enabled by representation theorems from real algebraic geometry produced in 1990s [68] and is the reason why polynomial optimization attracted a lot of attention recently.

At first glance, it may seem that the conditions of polynomial optimization are quite restrictive, but on closer inspection it is clear that many problems can be expressed as POPs. Examples can be:

- Linear and quadratic optimization problems, i.e. problems with objective function in the form

$$\inf_{\boldsymbol{x}}\{\boldsymbol{x}^T \mathbb{A}_0 \boldsymbol{x} + \boldsymbol{b}_0^T \boldsymbol{x} | \boldsymbol{x}^T \mathbb{A}_j \boldsymbol{x} + \boldsymbol{b}_j^T \boldsymbol{x} - c_j \geq 0, j = 1, \dots, m\} \tag{C.2}$$

- Boolean optimization by enforcing the constraints $x_i^2 - x_i = 0$. This obviously produces a Boolean variable $x_i \in \{0, 1\}$.

- Mixed Integer programming by enforcing the constraints

$$(x_i + N) \cdot (x_i + N - 1) \dots x_i \cdot (x_i - 1) \dots (x_i - N) = 0. \tag{C.3}$$

  Clearly now $x \in \{-M, \dots M\}$.

Solving polynomial optimization problems can be challenging due to the complexity of the polynomial functions and the constraints involved. Various approaches have been developed to tackle these problems, and some of the prominent methods include:

1. Algebraic Methods: For certain types of polynomials, especially those with low degrees, algebraic methods can be employed to find the optimal points analytically. These methods involve manipulating the polynomial algebraically to identify critical points and check for their optimality using calculus techniques [6].

2. Convex Optimization: Convex optimization deals with finding optimal solutions for convex polynomial functions subject to convex constraints. Convex polynomials are well-behaved, and efficient algorithms exist to find the global optimum for such problems. Semidefinite programming (SDP) is a popular technique used in convex optimization for certain classes of polynomial optimization problems [2].

3. Sum-of-Squares (SOS) Optimization: SOS optimization is a powerful technique used to handle non-convex polynomial optimization problems. It transforms the original problem into a hierarchy of convex optimization problems by expressing the polynomial as a sum of squares of other polynomials. The higher the degree of the SOS relaxation, the better the approximation to the true optimal solution [86].

4. Polynomial Homotopy Continuation: This method converts the polynomial optimization problem into a series of simpler problems, and by gradually adjusting the parameters, it finds the solution to the original problem. It is particularly useful for finding all the solutions of a polynomial system and is commonly used in algebraic geometry and engineering [10].

One of the key aspects of polynomial optimization is that it has a wide range of applications across various fields, including engineering [29, 65], computer science [43], economics [71], and physics [8, 7]. For instance, in engineering, it is used in control systems to find optimal parameter values for system performance, while in computer science, it plays a vital role in machine learning algorithms [51], where finding optimal parameters can lead to better model accuracy. Additionally, in economics, polynomial optimization is utilized to maximize utility functions subject to budget constraints, helping economists make better decisions. These examples demonstrate the practical significance of polynomial optimization in solving real-world problems.

Let us formalize the moment/SOS hierarchy, first introduced in the fundamental papers of POP [49, 64]. Let us refer to POP problem (C.1) (i.e. with f, g_j being polynomials) as P. We then define the family of problems

$$P_\lambda : \ f^* = \sup\{\lambda | f(\boldsymbol{x}) - \lambda, \boldsymbol{x} \in \boldsymbol{K}\}. \tag{C.4}$$

It was in the last-mentioned articles that Lasserre and Parrilo presented a systematic algorithmic construction of certificates of positivity on $\boldsymbol{K}$ for $f(\boldsymbol{x}) - \lambda$. These then serve to obtain solution of (C.4).

The basis of the construction is Putinar's Positivstellensatz.

Theorem 10 (Putinar's Positivstellensatz)**.** Let $\boldsymbol{K}$ be as in (C.1), with g_j, f polynomial functions, and compact. Let us denote the quadratic module generated by the family (g_j) as

$$Q(g) \coloneqq \Big\{\sigma_0 + \sum_{j=1}^{m} \sigma_j g_j : \sigma_j \in \Sigma[\boldsymbol{x}], j = 1, \ldots, m\Big\}, \tag{C.5}$$

where $\Sigma[\boldsymbol{x}]$ denotes the space of SOS polynomials, i.e. polynomials that can be written as $p(\boldsymbol{x}) = \sum_j p_j(\boldsymbol{x})^2$ for some finite family of polynomials (p_j). Assume that there exists $u \in Q(g)$ such that level set $\{\boldsymbol{x} \in \mathbb{R}^n : u(\boldsymbol{x}) \geq 0\}$ is compact. Then, if f is strictly positive on $\boldsymbol{K}$, then $f \in Q(g)$, that is,

$$f = \sigma_0 + \sum_{j=1}^{m} \sigma_j g_j, \tag{C.6}$$

for some SOS polynomials $\sigma_j \in \Sigma[\boldsymbol{x}]$, $j = 1, \ldots, m$.

Testing the possibility of writing f as in (C.6) means solving the SDP.

The dual of this SDP deals with the problem that, given a real sequence $\boldsymbol{y} = (y_{\boldsymbol{\alpha}})$ with multi-index $\boldsymbol{\alpha} \in \mathbb{N}^n$, whether there exists a Borel measure μ on $\boldsymbol{K}$ such that

$$y_\alpha = \int_{\boldsymbol{K}} \boldsymbol{x}^\alpha \mathrm{d}\mu \tag{C.7}$$

for any α. μ is then said to be a representing measure of the sequence $\boldsymbol{y}$. We introduce Riesz linear functional L_y from real polynomials as

$$f = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^n} f_{\boldsymbol{\alpha}} \boldsymbol{x}^{\boldsymbol{\alpha}} \rightarrow L_{\boldsymbol{y}}(f) = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^n} f_\alpha y_{\boldsymbol{\alpha}}. \tag{C.8}$$

The Riesz-Haviland theorem roughly states, that the sequence $\boldsymbol{y}$ above has a representing measure on $\boldsymbol{K}$ if and only if $L_{\boldsymbol{y}}(f) \geq 0$ for all polynomials nonnegative on $\boldsymbol{K}$. Then, if relations

$$L_{\boldsymbol{y}}(h^2) \geq 0, \qquad L_{\boldsymbol{y}}(h^2 g_j) \geq 0, \, j = 1, \ldots, m \tag{C.9}$$

hold for all polynomials h with maximum degree d, certain $m + 1$ moment and localizing matrices, with entries linear in $(y_{\boldsymbol{\alpha}})$, are positive semidefinite, i.e.

$$M_d(\boldsymbol{y}) \geq 0, \qquad M_d(g_j \boldsymbol{y}) \geq 0, \, j = 1, \ldots, m. \tag{C.10}$$

This can be modeled as a so-called generalized moment problem (GMP), which has the form

$$\inf_{\mu_i} \Big\{ \sum_{i=1}^{s} \int_{\boldsymbol{K}_i} f_i \mathrm{d}\mu_i : \sum_{i=1}^{s} \int_{\boldsymbol{K}_i} h_{ik} \mathrm{d}\mu_i \geq b_k \Big\}, \tag{C.11}$$

where $\boldsymbol{K}_i \subset \mathbb{R}^{n_i}$ is a compact semi-algebraic set and $f_i, g_{ik} : \mathbb{R}^{n_i} \rightarrow \mathbb{R}$ are polynomials. In this work, we will not go into the details of GMP, but a comprehensive description can be found in [50].

Nevertheless, from both primal and dual point of view, solving the moment/SOS amounts to solving a hierarchy of (convex) semidefinite programs of increasing size:

$$f_d^* = \sup_{\lambda, \sigma_j} \Big\{ \lambda : \sigma_0 + \sum_{j=1}^{m} \sigma_j g_j, \text{ with } \deg(\sigma_j g_j) \leq 2d \Big\} \tag{C.12}$$

The sequence $\{f_d^*, d \in \mathbb{N}\}$, is monotone and non-decreasing. For $\boldsymbol{K}$ compact, the global optimum of (C.4) is obtained as

$$f^* = \lim_{d \rightarrow \infty} f_d^*. \tag{C.13}$$

We note that finite convergence of (C.13) is a generic property.

As POP over a basic semialgebraic set is NP-hard problem [52], since the size of the matrices corresponding to the d-th step of the moment/SOS relaxation with n variables is of the order $\binom{n+d}{n}$, several proposals have been put forward to lower the required computational resources. One way is to exploit the sparsity patterns appearing in the POP problems proposed by Lasserre [48] and implemented e.g. in `TSSOS` Julia library [88].