國立臺灣大學應用物理研究所

碩士論文

Graduate Institute of Applied Physics

National Taiwan University

Master Thesis

非馬可夫開放系統中含流失態的 CNOT 量子邏輯閘最

佳化控制

Optimal Control of CNOT Gate Operation with Leakage

states and Non-Markovian Environments

楊存毅

Tsun-Yi Yang

指導教授：管希聖 博士

Advisor: Hsi-Sheng Goan, Ph.D.

中華民國 102 年 7 月

July 2013

口試審定書

國立臺灣大學碩士學位論文
口試委員會審定書

非馬可夫開放系統中含流失態的

CNOT 量子邏輯閘最佳化控制

Optimal Control of CNOT Gate Operation with Leakage states and Non-Markovian Environments

本論文係 楊存毅 君（R00245013）在國立臺灣大學物理學系、所完成之碩士學位論文，於民國一〇二年七月十五日承下列考試委員審查通過及口試及格，特此證明

口試委員：

（簽名）

（指導教授）

誌謝

碩士論文能夠如期完成，首先要感謝我的老師管希聖教授這麼詳細的幫我改論文還有指導我，真的比我預想的要詳細很多，老師真的很用心，還麻煩了胡崇德教授和蘇正耀教授當口試委員，很感謝兩位百忙之中還抽空前來，還有黃上瑜學長不厭其煩的指導我，還常常指出我一些錯誤的觀念，沒有學長我大概連程式也很難做出來，另外還要感謝我的良師益友戴榮身常常與我討論問題，感謝好戰友林冠廷在我脆弱的時候還有很多地方幫助我，感謝跟周宜之前合作讓我學到很多，還有感謝簡崇欽學長、陳柏文學長、張晏瑞學長、黃斌學長、萬哥、洪常力、李彥賢，還有許多學長學弟們在501室的交流和合作都讓我感動和感恩。

感謝我的爸爸和媽媽和弟弟，感謝我在西方極樂世界的阿嬤和阿公，還有所有的親戚朋友，雖然爸爸在我讀大學和碩士的這段期間受到了身體上不小的打擊，但和媽媽總是兩人一起很堅強，日後是我面對一切困難的榜樣，這是比其他一切都更重要的課程。 有太多人要感謝，但是一時之間我也無法一一數盡，碩士這兩年我看了很多人、很多事，遇過很多好人，也看過我無法理解的人，在專業素養的養成以外，我覺得做研究的心態才是重要的，如果只是為了學歷或錢去做研究，那肯定是做不好的，我想以後我會盡力的調整自己的心態，去匹配像老師和學長們那樣的傑出研究者，在生活上，但求無愧於心，了然於胸的是自己的心願，如果以後一生我也能堂堂正正的面對自己的一切抉擇，我想我也無遺憾了，而碩士這兩年，確確實實的幫助我了解了更進一步的人生觀，期望自己以後能遇到共度一生的人，分享我此時此刻的心願，大概也就值得了吧。

另外還要感謝我的室友和好朋友們，陳凱評、馬伯倫、施景堯、沙政亞，都是我的好朋友和人生導師，因為遇見你們讓我變得更加成熟，也受過你們的幫助，讓我覺得認識你們真好。

再一次的，感謝所有人。

中文摘要

為了建造一個量子電腦，高精準度的邏輯閘運算是必要的。特別是兩個量子位元(two qubit)所能進行的控制非門(CNOT)的運算，是一個很重要的量子邏輯閘。然而，現實世界中有需多的問題導致實驗上的結果和理論上的預測是不盡相同的。在這篇碩論中，我們探索了在量子最佳化控制理論下，考慮了流失態(leakage)和非馬可夫(non-Markovian)環境下的控制非門的運算。首先，我們給了一個簡短的超導體量子位元介紹，然後決定了單量子位元的漢米爾頓(Hamiltonian)矩陣。然後我們描述了環境的雜訊能譜(noise power spectrum)，接著就用費米黃金定律(Fermi golden rule)去連結馳耗速率(relaxation rate)和關聯函數(correlation function)之間的關係。然後我們描述了一個非馬可夫環境下的控制方程式，接著用我們的模型實行了量子最佳化控制。科羅多夫最佳化控制是我們在這篇碩論中採用的方法，目的是為了最小化邏輯閘運算的誤差，然後找到相對應的控制參數，並且考慮非同時的記憶效應(non-local memory effect)的非馬可夫開放系統。我們也討論了不同控制參數的不同波形，以及控制參數和關聯函數影響最佳化邏輯閘控制的行為。我們發現在環境的參數影響下(此參數參考真實實驗系統結果)，使用超導體電量量子位元是可以完成一個誤差約為$10^{-4} \sim 10^{-5}$的控制非門的高精準度邏輯閘控制的。

中文關鍵字：最佳化、位元、邏輯閘、科羅多夫、約瑟芬

Abstract

When building a quantum computer, high precision gate operations are needed. In particular, the controlled-not (CNOT) operation regarded as a crucial universal two-qubit gate is a very important quantum gate to implement. However, real world contains a lot of problems and causes the difference between experimental results and theoretical simulations. In this thesis, we investigate CNOT gate operation using quantum optimal control theory for superconducting charge qubit system taking into account the effects of leakage states and a non-Markovian environment. First, we give a brief introduction to superconducting qubits and decide the Hamiltonian of a simple one-qubit model. Then we describe the noise power spectrum of environments, which gives the relation between the relaxation rate and the bath correlation function through the Fermi golden rule . After that, we describe the non-Markovian master equation approach and apply it to our model together with the quantum optimal control theory. The Krotov optimal control method that we used in this thesis can minimize the error of gate operations and find the corresponding optimal control pulses to realize the gate operations. Considering the non-local memory effect in non-Markovian open quantum systems. We also discuss the effect of different shapes and behaviors in the bath correlation function on the optimal control gate fidelity. We find that it is possible to implement high-fidelity CNOT gates with error about $10^{-4} \sim 10^{-5}$ in superconducting charge qubit system with environment parameters extracted from the realistic noise power spectrum of experiments.
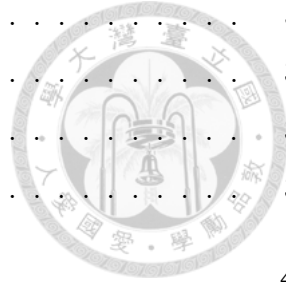
Keywords: optimal, qubit, gate, Krotov, Josephson

Contents

Chapter 1

Introduction and Background

1.1 Introduction

The field of solid-state quantum computation grows very fast. Since some quantum algorithms (e.g. Shor's algorithm) are shown to outperform significantly their best known classical counterparts, the idea and implementation of quantum information processing becomes important. The research of quantum bits, or qubits, is rapidly growing. There are many types of physical objects that have potential to be implemented as qubits. However, solid-state circuits, and superconducting circuit are of great interest as they offer great scalability. That is, the possibility of making circuits with a larger number of interacting qubits. The Josephson charge qubit circuit is a good candidate because it can implement qubit, i.e., quantum two-level systems. Such systems possess coherence, can be initialized and read out. Also, the logic gates operations can be constructed and gate operation can be done. First, In chapter 1, we introduce the background such as Josephson charge qubit Hamiltonian, power spectrum, and transition rate. In chapter 2, we describe our main purpose, i.e., two-qubit gate operations and present some necessary material that are useful to achieve our goal. We will give an overview of the performing two qubit Hamiltonian. We will discuss the noise spectral density and the bath correlation functions. We will try to use a more realistic bath spectral density parameters extracted from relevant experiments. In chapter 3, we introduce the Krotov optimal control method and the derivation to

obtain the iteration equations. In chapter 4, we present the optimal control results of four different kinds of situations. They are the combination of open/closed systems and with/without leakage levels. Finally, we conclude this whole thesis in Chapter 5. Some details about how to perform the calculation are put in appendices.
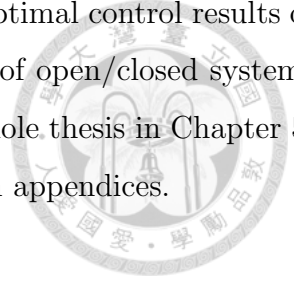
1.2 Review of Josephson Charge Qubit

The Josephson charge qubit consists basically of a Cooper pair box as a superconducting island which is connected via a Josephson junction to a large electrode, called a superconducting reservoir. Figure 1.1 shows a schematic illustration of a Josephson charge qubit. There are some paper reviewing the property of superconducting qubits, [1, 16]. Typically there are around $10^7 - 10^8$ conduction electrons in the island and the dimensions are about 1000 nm $\times$ 50 nm $\times$ 20 nm. The Cooper pairs can tunnel into or out of the box with a tunneling amplitude described by the Josephson coupling E_J between the box and the reservoir. The number of the Cooper pairs on the island can be tuned by the voltage V_g applied to the gate coupled to the box through the gate capacitor C_g.

1.3 Hamiltonian of the One Josephson Charge Qubit Model

The Hamiltonian of a Josephson charge qubit can be written as

$$\mathbf{H} = 4E_c(\hat{n} - n_g)^2 - E_j \cos \hat{\Theta}, \tag{1.1}$$

where n is the number operator of (excess) Cooper-pair charges on the island, and Θ, the phase of the superconducting order parameter of the island (phase difference across a Josephson junction), is its quantum-mechanical conjugate, $n = -i\hbar \frac{\partial}{\partial(\hbar\Theta)}$, E_j
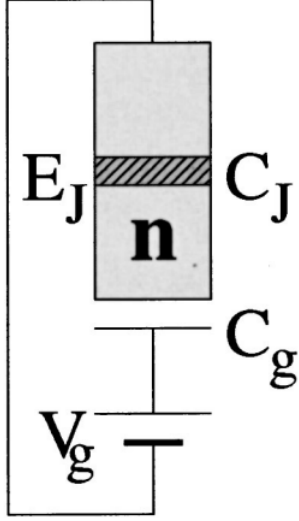
Figure 1.1: Schematic illustration of a Josephson charge qubit; from [1]. The bold "n" is the number of Cooper pairs that tunnel through the insulator into the island or "box". C_J is the junction capacitance, C_g is the gate capacitance. E_J is the junction energy and V_g is the gate voltage.

is the Josephson coupling energy.

$$E_C = \frac{e^2}{2(C_g + C_J)} \tag{1.2}$$

is the single electron charging energy and depends on the total capacitance of the island. The dimensionless gate charge

$$n_g = \frac{C_g V_g}{2e} \tag{1.3}$$

accounts for the effect of the gate voltage. We will vary the gate voltage as our external control.

Since we will control the tunneling Cooper pairs, we choose the number states describing the number of the Cooper pairs on the island as our basis, the so called the charge basis. In order to express the Hamiltonian by the basis, we introduce the relation between n and Θ:

$$[\hat{\Theta}, \hat{n}] = i \tag{1.4}$$

with $\hat{\Theta} \in [0, 2\pi]$. One can use Eq.(1.4) to derive $[e^{i\hat{\Theta}}, \hat{n}] = -e^{i\hat{\Theta}}$ and $[e^{-i\hat{\Theta}}, \hat{n}] = e^{-i\hat{\Theta}}$. This is similar to the commutation relation between the creation operator, the annihilation operator and the number operator of a harmonic oscillator. So one can regard $e^{\pm i\hat{\Theta}}$ as creation and annihilation operators, respectively. Inserting an identity $\mathbf{I} = \sum_n |n\rangle\langle n|$ onto the both sides of the Hamiltonian, Eq.(1.1), we obtain

$$\mathbf{H} = \sum_n \{4E_c(n - n_g)^2 |n\rangle\langle n| - \frac{1}{2}E_j(|n+1\rangle\langle n| + |n\rangle\langle n+1|)\}. \tag{1.5}$$

One may have seen different form of the Hamiltonian from the form of Eq.(1.3), but they are actually the same just because sometimes people would like to shift identities to make the equation look more simple than it's original form.

Usually, one considers the charge states corresponding to $n = 0, 1$ as the computational states of the qubit to perform a gate operation. However, as in [2], we know that the dynamics of the Josephson charge qubit system is affected by the other number state, which is the leaky qubit effect, or so called leakage. That is, the consideration including other states into the calculations may have large difference to the case by considering only the computational states $\{|0\rangle, |1\rangle\}$. To study the leakage effect, in this thesis, we consider the charge states $\{|-1\rangle, |0\rangle, |1\rangle, |2\rangle\}$. Many related papers point out that there is no need to consider even more charge states.[2, 3, 4]. One will see that the control problem is not physical enough if one does not consider the leakage effect. One may ask what $n = -1$ state is. What is minus number of Cooper pairs? We have to see this in a bigger picture. Since the number of Cooper pairs on the island should be a huge amount instead of just one or two, the number of Cooper pairs on the island we discussed before is the change in the number of the Cooper pairs as compared with the equilibrium number, i.e., the number of the excess Cooper pairs.

Note that some people might use energy eigenstates like Figure 1.2 to construct their two-level system. Here we just use the charge basis which are eigenstates of the number operator with eigenvalue describing the number of Cooper-pair on the island.
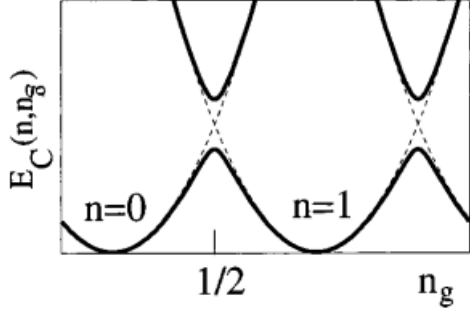
Figure 1.2: The charge energy of the Cooper-pair box as a function of n_g. From [1]. Since n_g is not an actual number of charge, it can be a float number. When $n_g = \frac{1}{2}$, the system with states corresponding to the energy splitting caused by E_j can be approximate as a two-level system. The dashed lines means the changing energy with definite Cooper pairs n on the island, and the solid line stands for the energy eigenstates.

1.4 Power Spectrum and Noise Fluctuation

There must exist some noise in every quantum system to prevent us from perfectly controlling the quantum system. For Josephson charge qubit, there are some fluctuation of the voltage that will cause noise.

Let's rewrite Eq.(1.1) as in [7]

$$\mathbf{H}_{box} = 4E_C(n - \frac{Q_g}{2e})^2 - E_j \cos \Theta, \tag{1.6}$$

where $Q_g = C_g V_g$ is the gate charge which is controlled by gate voltage and $E_C = \frac{e^2}{2(C_j + C_g)}$.

By properly choosing the right parameters such as voltage, temperature, frequency, and the strength of control pulses, only two charge states $|n = 0\rangle$ and $|n = 1\rangle$ play an important role. In the spin notation, the number operator becomes $n = \frac{1}{2}(1 - \sigma_z)$, while $\cos \Theta = \frac{1}{2}\sigma_x$, and the effective spin Hamiltonian after inserting $V_g + \delta V$, where δV denotes the voltage fluctuation, into Eq. (1.6) reads

$$\mathbf{H} = -\frac{1}{2}B_z(V_g)\sigma_z - \frac{1}{2}B_x\sigma_x - \frac{1}{2}X\sigma_z + \mathbf{H_{bath}}, \tag{1.7}$$

where the parameters are $B_z = \frac{2e}{C_j + C_g}(e - C_g V_g)$, $B_x = E_j$, and $B_z(V_g + \delta V) =$
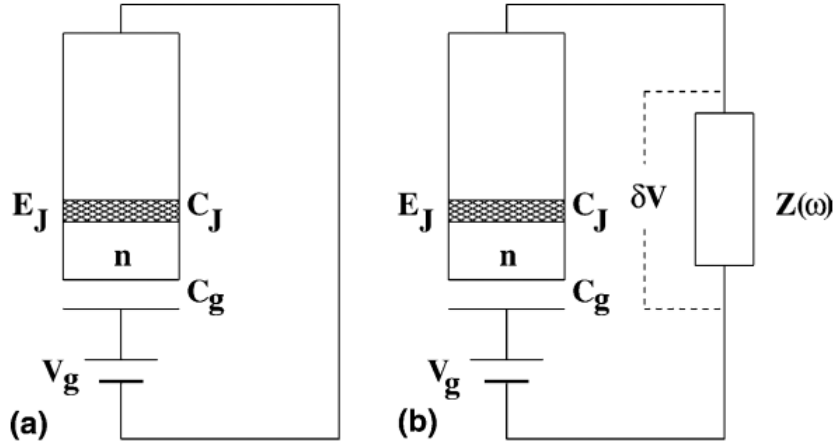
Figure 1.3: From [7]. (a) Circuit of a Josephson charge qubit. (b) Circuit representing noise coming from fluctuation denoted by δV.

$B_z(V_g) + X$. The sum of the voltage $V_g + \delta V$ couples to the charge in the circuit.

The coupling term involving X, which comes from the fluctuation of voltage represented as the interaction with a bath results in decoherence and dissipation. Here

$$X = -(\frac{2eC_t}{C_j})\delta V. \tag{1.8}$$

Since we assume interaction with the bath or environment is sufficient weak, [19, 20, 21], we can represent the environment as a set of harmonic oscillators with a coupling linear in the oscillator coordinates and/or momenta to the system; by appropriate canonical transformation and related tricks, it is possible to ensure that the "coupling is only to the coordinates". That is

$$\mathbf{H_{bath}} = \sum_i \left[\frac{1}{2m_i}p_i^2 + \frac{m_i\omega_i}{2}x_i^2 \right] \tag{1.9}$$

and the coupling part of the environment is

$$\delta V = \sum_i \lambda_i x_i, \tag{1.10}$$

where λ_i is the coupling coefficient with each coordinate x_i of the oscillators, where

$$x_i = \frac{1}{\sqrt{2m\omega_i}}(a_i + a_i^\dagger) \tag{1.11}$$

We will see later in Sec. 2.2 that X represents the coupling part of δn there:

$$X = \sum_i \bar{\lambda}_i x_i = \sum_i \bar{\lambda}_i (a_i + a_i^\dagger), \tag{1.12}$$

where $\bar{\lambda}_i$ absorbs the rest of the parameters of Eq. (1.8).

From the Hamiltonian of eigenbasis, one can easily see the energy splitting of the system. Although one may not always choose eigenbasis, this representation can let one see the relation between the power spectrum and the relaxation rate from the Fermi golden rule. The Hamiltonian, Eq.(1.7), written in the qubit energy eigenbasis becomes

$$\mathbf{H} = -\frac{1}{2} \triangle E \sigma_z - \frac{1}{2} X \left[\cos(\eta)\sigma_z + \sin(\eta)\sigma_x \right] + \mathbf{H_{bath}} \tag{1.13}$$

with the energy difference $\triangle E = \sqrt{B_x^2 + B_z^2}$, $\tan\eta = \frac{B_x}{B_z}$, $X_{tran} = -\frac{1}{2}X\sin\eta$, and $V(t) = X_{tran}(t)$. Why do we want to connect the power spectrum with the relaxation rate? The reason is that we normally measure the relaxation rate to define how much the environment affects our system, and the noise power spectrum and the bath spectral density are related to the bath correlation function. The following equations tell us how the relaxation rate is related to the noise power spectrum.

First we know from the Fermi golen rule that the transition rate between two different states $|a\rangle$ and $|b\rangle$ is

$$\Gamma_{ba} = \frac{1}{\hbar^2} \int_{-\infty}^{\infty} dt \langle \mathbf{V_{ab}}(t)\mathbf{V_{ba}}(0) \rangle_B e^{-i\omega_{ba}t}, \tag{1.14}$$

where $\mathbf{V}$ denotes the interaction Hamiltonian which is assumed to be weak. This equation will be discuss in the next section. And we know there must be two directions

of the transition

$$\Gamma_{\downarrow} = \Gamma_{10} = \frac{\sin^2 \eta}{4\hbar^2} \langle \mathbf{X}^2_{\omega = \frac{\triangle \mathbf{E}}{\hbar}} \rangle, \qquad (1.15)$$

$$\Gamma_{\uparrow} = \Gamma_{01} = \frac{\sin^2 \eta}{4\hbar^2} \langle \mathbf{X}^2_{\omega = -\frac{\triangle \mathbf{E}}{\hbar}} \rangle, \qquad (1.16)$$

where $\langle \mathbf{X}^2_{\omega} \rangle = \int dt e^{i\omega t} \langle \mathbf{X}(t)\mathbf{X}(0) \rangle$. Combining these two rates, we obtain the relaxation rate

$$\Gamma_1 = \Gamma_{\downarrow} + \Gamma_{\uparrow} = \frac{\pi \sin^2 \eta}{2\hbar^2} S_U(\omega = \frac{\triangle E}{\hbar}) \qquad (1.17)$$

with the noise power spectrum $S_U(\omega) = \frac{1}{2\pi}(\langle \mathbf{X}^2_{\omega} \rangle + \langle \mathbf{X}^2_{-\omega} \rangle)$. The relation between the noise power spectrum and the bath spectral density will be discussed in Sec. 2.3 . We can describe the evolution of the density matrix of the qubit via the equation of motion:

$$Tr\hat{\rho} = \rho_{00} + \rho_{11} = 1, \qquad (1.18)$$

$$\dot{\rho}_{00} = -\dot{\rho}_{11} = -\Gamma_{\uparrow}\rho_{00} + \Gamma_{\downarrow}\rho_{11}, \qquad (1.19)$$

1.5 Transition Rate

Since different papers have their own definition of the noise spectrum, power spectrum, spectral density, with or without 2π of the Fourier transformation, we can check their definition of the transition rate, and relate it to the noise terms mentioned above. Then we can compute the environment effect using the noise power spectrum obtained from the experimental data.

Return to how we get the transition rate from the Fermi golden rule [18]. Consider a simple model

$$\mathbf{H} = (\mathbf{H_S} + \mathbf{H_B}) + \mathbf{V} = \mathbf{H_0} + \mathbf{V}, \qquad (1.20)$$

where $\mathbf{V}$ is the interaction Hamiltonian describing weak coupling the between system and the bath.

Assume that the system has two eigenstates $|a\rangle$ and $|b\rangle$ and their corresponding eigenenergies are E_a and E_b, respectively. Similarly, we assume that the bath mode γ has eigenstates $|\gamma\rangle$ and eigenenergy E_γ. The Hamiltonian $\mathbf{H_S}$ and $\mathbf{H_B}$ written in their eigenbasis become

$$\mathbf{H_S} = E_a|a\rangle\langle a| + E_b|b\rangle\langle b|, \tag{1.21}$$

$$\mathbf{H_B} = \sum_\gamma E_\gamma|\gamma\rangle\langle\gamma| \tag{1.22}$$

with $\gamma = \cdots \alpha, \beta \cdots$. Thus, we have

$$\mathbf{H_0}|a\alpha\rangle = (E_a + E_\alpha)|a\alpha\rangle, \tag{1.23}$$

where $|a\alpha\rangle = |a\rangle \otimes |\alpha\rangle$ is the product state of the system and the bath. Set $|a\alpha\rangle$ the initial state and $|b\beta\rangle$ the final state. Consider the case where the system energy decreases and the bath energy rises.

$$
\begin{aligned}
\Gamma_{fi} &= \frac{2\pi}{\hbar}\sum_{i,f}p_f|\langle i|\mathbf{V}|f\rangle|^2\delta(E_f - E_i) \\
&= \frac{2\pi}{\hbar}\sum_{a,b,\alpha,\beta}p_{b\beta}|\langle a\alpha|\mathbf{V}|b\beta\rangle|^2\delta((E_b + E_\beta) - (E_a + E_\alpha)) \\
&= \frac{1}{\hbar^2}\int_{-\infty}^{\infty}dt\sum_{a,b,\alpha,\beta}p_{b\beta}\langle b\beta|\mathbf{V}|a\alpha\rangle\langle a\alpha|\mathbf{V}|b\beta\rangle e^{-\frac{i}{\hbar}[(E_b - E_a)+(E_\beta - E_\alpha)]t},
\end{aligned}
\tag{1.24}
$$

where p_f is the probability of the final state $|f\rangle$. Therefore, the transition rate system from the system eigenstate $|a\rangle$ to $|b\rangle$ is

$$\Gamma_{ba} = \frac{1}{\hbar^2}\int_{-\infty}^{\infty}dt\sum_{\alpha,\beta}p_\beta\langle\beta|e^{\frac{i}{\hbar}E_\alpha t}\mathbf{V_{ba}}e^{-\frac{i}{\hbar}E_\beta t}|\alpha\rangle\langle\alpha|\mathbf{V_{ba}}|\beta\rangle e^{-i\omega_{ba}t} \tag{1.25}$$

Define

$$\mathbf{V_{ab}} = \langle a | \mathbf{V} | b \rangle. \tag{1.26}$$

In interaction picture, one can get

$$\mathbf{V_{ab}}(t) = e^{\frac{i}{\hbar} \mathbf{H_B} t} \mathbf{V_{ab}} e^{-\frac{i}{\hbar} \mathbf{H_B} t}. \tag{1.27}$$

The bath correlation function is defined as

$$C_{ba}(t) = \langle \mathbf{V_{ab}}(t) \mathbf{V_{ba}}(0) \rangle_B, \tag{1.28}$$

where

$$\langle \cdots \rangle_B = \sum_\beta \langle \beta | \cdots | \beta \rangle \tag{1.29}$$

means an equilibrium thermal average over the bath states. Then, we have

$$\Gamma_{ba} = \frac{1}{\hbar^2} \int_{-\infty}^{\infty} dt \langle \mathbf{V_{ab}}(t) \mathbf{V_{ba}}(0) \rangle_B e^{-i\omega_{ba} t}. \tag{1.30}$$

So we can use this form to the relation between the relaxation rate and the bath correlation function.

Chapter 2

Superconducting Qubit and Noise

2.1 Overview

The basic element of quantum information processing is qubit. There are many physical systems that can implement qubits. Here we choose the Josephson superconducting charge qubit system to study. Like other qubits, the Josephson charge qubit has its own weakness; especially it is susceptible to charge or voltage noise and has the problem of qubit state leakage. However it is still a good physical system for quantum computation, if we can perform quantum gate operation with high fidelity taking into account the influence of noise and environment. So the goal of this thesis is to find control field pulse to achieve high-fidelity controlled-not (CNOT) gate operation by the tool called the Krotov optimal control method, which will be introduced in Chapter 3. In this Chapter, we first present the Hamiltonian of two Josephson charge qubits. Then we describe the decoherence effect from the environment and the leakage states.

2.2 Hamiltonian of Two Coupled Josephson Charge Qubits
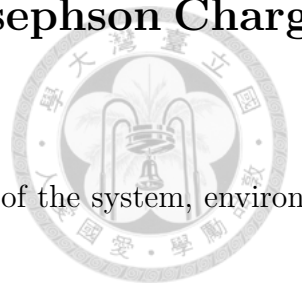
The total Hamiltonian contain three parts: the Hamiltonian of the system, environment (bath), and their interaction

$$\mathbf{H_T} = \mathbf{H_S} + \mathbf{H_{bath}} + \mathbf{H_{int}}. \tag{2.1}$$

Since our theoretical model is based on the experimental setup in [1, 5], we follow the form of their Hamiltonian. So the system Hamiltonian of two capacitively coupled charge qubits is

$$
\begin{aligned}
\mathbf{H_S} = &\sum_{n_1,n_2=N_L}^{N} \{ E_{c1}(n_1 - n_{g1}(t))^2 + E_{c2}(n_2 - n_{g2}(t))^2 \\
&+ E_m(n_1 - n_{g1}(t))(n_2 - n_{g2}(t))\} |n_1, n_2\rangle\langle n_1, n_2| \\
&- \frac{1}{2}E_{j1} \sum_{n_1=N_L}^{N-1} \sum_{n_2=N_L}^{N} \{(|n_1 + 1\rangle\langle n_1| + |n_1\rangle\langle n_1 + 1|) \otimes |n_2\rangle\langle n_2|\} \\
&- \frac{1}{2}E_{j2} \sum_{n_1=N_L}^{N} \sum_{n_2=N_L}^{N-1} \{|n_1\rangle\langle n_1| \otimes (|n_2 + 1\rangle\langle n_2| + |n_2\rangle\langle n_2 + 1|)\},
\end{aligned} \tag{2.2}
$$

where n_1, n_2 denote the charge states of the two qubits, and n_{g1}, n_{g2} are the gate charges, N_L means the lowest charge state, and the N is the highest charge state. We choose the values of the parameters according to the experimental setup in Refs.[1, 5] as

$$
\begin{cases}
\frac{E_{c1}}{h} = 140.2 & \text{GHz,} \\
\frac{E_{c2}}{h} = 162.2 & \text{GHz,} \\
\frac{E_{j1}}{h} = 10.9 & \text{GHz,} \\
\frac{E_{j2}}{h} = 9.9 & \text{GHz,} \\
\frac{E_m}{h} = 23.0 & \text{GHz.}
\end{cases} \tag{2.3}
$$

The control parameters are

$$n_{g\nu}(t) = n_{g\nu}^0 + \overline{n}_{g\nu}(t) \tag{2.4}$$

with $\nu \in \{1,2\}$. The values of $n_{g1}^0 = 0.24$, $n_{g2}^0 = 0.26$ are chosen according to Refs.[1, 5].

We consider two cases for the Hamiltonian Eq. (2.1). The first case is to treat each of the qubits as a two-level system with $N = 1$, and $N_L = 0$. The second case is to include the leakage state for the qubits with $N = 2$, and $N_L = -1$. Also, one can split Eq. (2.2) into

$$\mathbf{H_S} = \mathbf{H_d} + n_{g1}(t) \times \mathbf{H_1} + n_{g2}(t) \times \mathbf{H_2}, \tag{2.5}$$

where $\mathbf{H_d}, \mathbf{H_1}, \mathbf{H_2}$ are time independent. In the case of $N = 1, N_L = 0$, the Hamiltonians may be written in terms of Pauli matrices σ_i as

$$
\begin{aligned}
\mathbf{H_d} &= -(\frac{E_m}{4} + \frac{E_{c1}}{2})(\sigma_z^{(1)} \otimes \mathbf{I}) - \frac{E_{j1}}{2}(\sigma_x^{(1)} \otimes \mathbf{I}) \\
&\quad -(\frac{E_m}{4} + \frac{E_{c2}}{2})(\mathbf{I} \otimes \sigma_z^{(2)}) - \frac{E_{j2}}{2}(\mathbf{I} \otimes \sigma_x^{(2)}) \\
&\quad +\frac{E_m}{4}(\sigma_z^{(1)} \otimes \sigma_z^{(2)}),
\end{aligned}
\tag{2.6}
$$

$$\mathbf{H_1} = E_{c1}(\sigma_z^{(1)} \otimes \mathbf{I}) + \frac{E_m}{2}(\mathbf{I} \otimes \sigma_z^{(2)}) \tag{2.7}$$

$$\mathbf{H_2} = \frac{E_m}{2}(\sigma_z^{(1)} \otimes \mathbf{I}) + E_{c2}(\mathbf{I} \otimes \sigma_z^{(2)}) \tag{2.8}$$

where $\sigma_i^{(\nu)}$ are Pauli matrices acting on the Hilbert space of the νth qubit. In the case of $N = 2$, and $N_L = -1$, we choose to use the original form of Hamiltonian in number basis states. After appropriate identity shift

$$\mathbf{H_S}' = \mathbf{H_S} - \{...\}\mathbf{I} \tag{2.9}$$

and $\mathbf{H_1} = \frac{\partial \mathbf{H_S}'}{\partial n_{g1}}$, $\mathbf{H_2} = \frac{\partial \mathbf{H_S}'}{\partial n_{g2}}$, we write $\mathbf{H_d} = \mathbf{H_S'} - n_{g1}(t)\mathbf{H_1} - n_{g2}(t)\mathbf{H_2}$, and it's

possible to get the form as

$$\mathbf{H_1} = E_{c1}(\mathbf{S_1}) + \frac{E_m}{2}(\mathbf{S_2}),$$

(2.10)

$$\mathbf{H_2} = \frac{E_m}{2}(\mathbf{S_1}) + E_{c2}(\mathbf{S_2}).$$

(2.11)

where $\mathbf{S_1}$ and $\mathbf{S_2}$ are system operators similar to those in Eqs. (2.7) and (2.8).

In order to describe the decoherence effect, we focus on the charge noise and the interaction between the system and the bath is

$$\mathbf{H_{int}} = \delta n_{g1}\mathbf{S_1} + \delta n_{g2}\mathbf{S_2}$$

(2.12)

The symbol δn_g denotes the fluctuation of the control ,like Eq.(1.10), or other forms of charge noise. Each of the qubit couples to its own independent charge noise environment or independent voltage fluctuation source. We model each of the independent environments as a large set of harmonic oscillators, each of which interacts weakly with its corresponding qubit. So the bath Hamiltonian is

$$\mathbf{H_{bath}} = \sum_{\nu=1}^{2} \sum_{i} [\frac{1}{2m_i^{(\nu)}}(p_i^{(\nu)})^2 + \frac{m_i^{(\nu)}\omega_i^{(\nu)}}{2}(x_i^{(\nu)})^2]$$

(2.13)

with

$$x_i^{(1)} = \frac{1}{\sqrt{2m_i^{(1)}\omega_i^{(1)}}}(a_i + a_i^\dagger),\, x_i^{(2)} = \frac{1}{\sqrt{2m_i^{(2)}\omega_i^{(2)}}}(b_i + b_i^\dagger).$$

(2.14)

The bath operators δn_{g1} and δn_{g2} are

$$\delta n_{g1} \equiv \sum_i \lambda_i(a_i + a_i^\dagger) \equiv \mathbf{Y_1} \otimes \mathbf{I} = \Gamma_1,$$

(2.15)

$$\delta n_{g2} \equiv \sum_j \lambda_j(b_j + b_j^\dagger) \equiv \mathbf{I} \otimes \mathbf{Y_2} = \Gamma_2,$$

(2.16)

where $\mathbf{Y_1}$ and $\mathbf{Y_2}$ are bath operators acting on the Hilbert space of the first and

14

second qubits, respectively. The system-bath coupling Hamiltonian is then

$$\mathbf{H_{int}} = \mathbf{S_1} \otimes \Gamma_1 + \mathbf{S_2} \otimes \Gamma_2. \tag{2.17}$$

The master equation for the reduced density matrix $\tilde{\rho}(t)$ of the two qubit system in the interaction picture can be written as [23]

$$
\begin{aligned}
\dot{\tilde{\rho}}(t) =& -\sum_{i,j} \int_0^t dt' \{ [\mathbf{\tilde{S}_i}(t)\mathbf{\tilde{S}_j}(t')\tilde{\rho}(t') - \mathbf{\tilde{S}_j}(t')\tilde{\rho}(t')\tilde{S}_j(t)] < \tilde{\Gamma}_i(t)\tilde{\Gamma}_j(t') >_R \\
&+ [\tilde{\rho}(t')\mathbf{\tilde{S}_j}(t')\mathbf{\tilde{S}_i}(t) - \mathbf{\tilde{S}_i}(t)\tilde{\rho}(t')\mathbf{\tilde{S}_j}(t')] < \tilde{\Gamma}_j(t')\tilde{\Gamma}_i(t) >_R \},
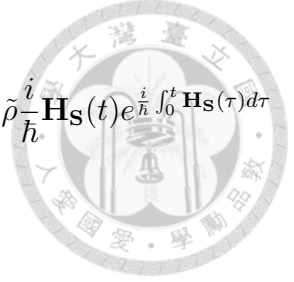\end{aligned}
\tag{2.18}
$$

where

$$\tilde{\rho}(t) = e^{\frac{i}{\hbar}\int_0^t \mathbf{H_S}(\tau)d\tau} \rho(t) e^{-\frac{i}{\hbar}\int_0^t \mathbf{H_S}(\tau)d\tau} \tag{2.19}$$

Transforming back to the Schrodinger picture with

$$\rho(t) = e^{-\frac{i}{\hbar}\int_0^t \mathbf{H_S}(\tau)d\tau} \tilde{\rho}(t) e^{\frac{i}{\hbar}\int_0^t \mathbf{H_S}(\tau)d\tau} \tag{2.20}$$

15

we obtain

$$
\begin{aligned}
\dot{\rho}(t) &= -\frac{i}{\hbar}\mathbf{H_S}(t) + e^{-\frac{i}{\hbar}\int_0^t \mathbf{H_S}(\tau)d\tau}\dot{\tilde{\rho}}e^{\frac{i}{\hbar}\int_0^t \mathbf{H_S}(\tau)d\tau} + e^{-\frac{i}{\hbar}\int_0^t \mathbf{H_S}(\tau)d\tau}\tilde{\rho}\frac{i}{\hbar}\mathbf{H_S}(t)e^{\frac{i}{\hbar}\int_0^t \mathbf{H_S}(\tau)d\tau} \\
&= -\frac{i}{\hbar}[\mathbf{H_S}(t),\rho(t)] + U^\dagger(t)\dot{\tilde{\rho}}U(t) \\
&= -\frac{i}{\hbar}[\mathbf{H_S}(t),\rho(t)] \\
&\quad -\frac{1}{\hbar^2}U^\dagger(t)\sum_{i=1}^2 \int_0^t dt'\{[\tilde{\mathbf{S}}_\mathbf{i}(t)\tilde{\mathbf{S}}_\mathbf{i}(t')\tilde{\rho}(t') - \tilde{\mathbf{S}}_\mathbf{i}(t')\tilde{\rho}(t')\tilde{\mathbf{S}}_\mathbf{j}(t)] < \tilde{\Gamma}_i(t)\tilde{\Gamma}_j(t') >_R \\
&\quad +[\tilde{\rho}(t')\tilde{\mathbf{S}}_\mathbf{i}(t')\tilde{\mathbf{S}}_\mathbf{i}(t) - \tilde{\mathbf{S}}_\mathbf{i}(t)\tilde{\rho}(t')\tilde{\mathbf{S}}_\mathbf{i}(t')] < \tilde{\Gamma}_i(t')\tilde{\Gamma}_i(t) >_R\}U(t) \\
&= -\frac{i}{\hbar}[\mathbf{H_S}(t),\rho(t)] \\
&\quad -\frac{1}{\hbar^2}\sum_{i=1}^2\{\mathbf{S_i}U^\dagger(t)\int_0^t dt'(\tilde{\mathbf{S}}_\mathbf{i}(t')\tilde{\rho}(t') < \tilde{\Gamma}_i(t)\tilde{\Gamma}_i(t') >_R -\tilde{\rho}(t')\tilde{\mathbf{S}}_\mathbf{i}(t) < \tilde{\Gamma}_i(t')\tilde{\Gamma}_i(t) >_R) \\
&\quad -U^\dagger(t)\int_0^t dt'(\tilde{\mathbf{S}}_\mathbf{i}(t')\tilde{\rho}(t') < \tilde{\Gamma}_i(t)\tilde{\Gamma}_i(t') >_R -\tilde{\rho}(t')\tilde{\mathbf{S}}_\mathbf{i}(t') < \tilde{\Gamma}_i(t')\tilde{\Gamma}_i(t) >_R)U(t)\mathbf{S_i}\} \\
&= -\frac{i}{\hbar}[\mathbf{H_S}(t),\rho(t)] - \frac{1}{\hbar^2}\sum_{i=1}^2\{[\mathbf{S_i},D^{(i)}(t)] - [\mathbf{S_i},(D^{(i)}(t))^\dagger]\}. \quad (2.21)
\end{aligned}
$$

Since we are dealing with two independent bathes, so there is no cross term. If there is no environment effect, there will only be the first term left in Eq.(2.21). The 2nd and 3rd terms describe the environment effect and the dissipators $D^{(i)}(t)$ are defined as [24]

$$
\begin{aligned}
D^{(i)}(t) &= -\frac{1}{\hbar^2}U^\dagger(t)\int_0^t dt' U(t')\mathbf{S_i}\rho(t')U^\dagger(t') < \tilde{\Gamma}_i(t)\tilde{\Gamma}_i(t') >_R U(t) \\
&= -\frac{1}{\hbar^2}\int_0^t \mathcal{U}_S(t-t')\mathbf{S_i}\rho(t')C(t-t')dt' \quad (2.22)
\end{aligned}
$$

The definition of the bath correlation function $C(t-t') = < \tilde{\Gamma}_i(t)\tilde{\Gamma}_i(t') >_R$ will be discussed in the next section. To overcome the difficulty of solving the time-nonlocal integral-differential equations (2.21) and (2.22), we follow the approach in [24] to introduce auxiliary density matrices in the extended Liouville space. The procedure is to use many exponential terms to fit the bath correlation function, and the total

number of the exponential terms is determined by the accuracy of the fitting error. Usually, we choose error to be less than 10^{-5}. So we can write

$$C(t - t') = C(\tau) = \sum_j C_j(t - t') = \sum_j C_j(0)e^{\gamma_j \tau}, \qquad (2.23)$$

where $C_j(0)$ and γ_j are complex. In this case, the dissipators,

$$D^{(i)}(t) = \sum_j K_j^{(i)}(t), \qquad (2.24)$$

where

$$
\begin{aligned}
K_j^{(i)}(t) &= -\frac{1}{\hbar^2} \int_0^t \mathcal{U}_S(t - t')\mathbf{S_i}\rho(t')C_j(t - t')dt' \\
&= -\frac{1}{\hbar^2} \int_0^t A \times C_j(t - t')dt' \qquad (2.25)
\end{aligned}
$$

with $A = \mathcal{U}_S(t - t')\mathbf{S_i}\rho(t')$. Differentiating Eq.(2.25) with respect to time, we obtain

$$
\begin{aligned}
\frac{d}{dt}K_j^{(i)}(t) &= -\frac{1}{\hbar^2}\mathcal{U}_S(0)\mathbf{S_i}\rho(t)C_j(0) - \frac{1}{\hbar^2}\int_0^t \frac{d}{dt}\{\mathcal{U}_S(t - t')\mathbf{S_i}\rho(t')C_j(t - t')\}dt' \\
&= -\frac{1}{\hbar^2}C_j(0)\mathbf{S_i}\rho(t) - \frac{1}{\hbar^2}\int_0^t dt'\{[-\frac{i}{\hbar}\mathbf{H_s}(t)A + \frac{i}{\hbar}A\mathbf{H_s}(t)]C_j(t - t') + A\frac{dC_j(t - t')}{dt}\} \\
&= -\frac{1}{\hbar^2}C_j(0)\mathbf{S_i}\rho(t) - \frac{1}{\hbar^2}\int_0^t dt'\{-\frac{i}{\hbar}[\mathbf{H_s}(t), A]C_j(t - t') + A\frac{dC_j(t - t')}{dt}\} \\
&= -\frac{1}{\hbar^2}C_j(0)\mathbf{S_i}\rho(t) - \frac{1}{\hbar^2}\{\mathcal{L}_s(t)K_j^{(i)}(t) + \gamma_j K_j^{(i)}(t)\}. \qquad (2.26)
\end{aligned}
$$

Eq.(2.21) with Eq.(2.24) and Eq.(2.26) form a set of coupled time-local differential equations. Now with the above coupled time-local equations, one may solve the evolution of the model. Remember that one can always returns to the Schrodinger (von-Neumann) equation of the closed system by setting $D = 0$ in Eq.(2.21). Since $\rho(t)$ and $K_j(t)$ are coupled and mutually needed for solving the differential equations.

One could combine them into a new vector as

$$\overrightarrow{\rho}(t) = \begin{pmatrix} \rho(t) \\ K_j^\dagger(t) \\ K_j(t) \\ \vdots \end{pmatrix}.$$ (2.27)

In this way, one may rewrite the coupled equations of motion in a very concise form as

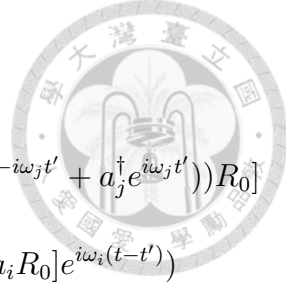$$\frac{d}{dt}\overrightarrow{\rho}(t) = \Lambda(t)\overrightarrow{\rho}(t),$$ (2.28)

2.3 Spectral Density and Correlation Function

Generally speaking, there are two processes relevant for the observed decoherence: energy relaxation (the decay of probability of some states) and dephasing effect (the decay of the off-diagonal terms of the density matrix). We will discuss the bath correlation function and the bath spectral density in the following

In the interaction picture, the bath operator takes the form

$$\tilde{\Gamma}_1(t) = e^{\frac{i}{\hbar}\mathbf{H_B}t}\Gamma_1 e^{-\frac{i}{\hbar}\mathbf{H_B}t}.$$ (2.29)

So the bath correlation function becomes

$$
\begin{aligned}
C(t-t') &= C(\tau) = Tr_B[\tilde{\Gamma}_1(t)\tilde{\Gamma}_1(t')R_0] \\
&= Tr_B[(\sum_i \lambda_i(a_i e^{-i\omega_i t} + a_i^\dagger e^{i\omega_i t}))(\sum_j \lambda_j(a_j e^{-i\omega_j t'} + a_j^\dagger e^{i\omega_j t'}))R_0] \\
&= \sum_i |\lambda_i|^2 \{(Tr_B[a_i a_i^\dagger R_0]e^{-i\omega_i(t-t')} + Tr_B[a_i^\dagger a_i R_0]e^{i\omega_i(t-t')}) \\
&= \int_0^\infty d\omega' J(\omega')[(n(\omega') + 1)e^{-i\omega_i(t-t')} + n(\omega)e^{i\omega_i(t-t')}] \\
&= \int_0^\infty d\omega' J(\omega')[(n(\omega') + 1)\{\cos[\omega'(t - t')] - i\sin[\omega'(t - t')]\} \\
&\quad + n(\omega)\{\cos[\omega'(t - t')] + i\sin[\omega'(t - t')]\}] \\
&= \int_0^\infty d\omega' J(\omega')\{(2n(\omega') + 1)\cos[\omega'(t - t')] - i\sin[\omega'(t - t')]\} \\
&= \int_0^\infty d\omega' J(\omega')\{\left(\frac{1 + e^{-\frac{\hbar\omega}{k_B T}}}{1 - e^{-\frac{\hbar\omega}{k_B T}}}\right)\cos[\omega'(t - t')] - i\sin[\omega'(t - t')]\} \\
&= \int_0^\infty d\omega' J(\omega')\{cos[\omega'(t - t')]\coth(\frac{\hbar\omega'}{2k_B T}) - i\sin[\omega'(t - t')]\},
\end{aligned}
$$

where $J(\omega')$ is the bath spectral density. Depending on the choice of the bath spectral density, instead of the frequency integration limits of 0 and ∞, we may introduce the lower bound and upper bound frequencies, also called infrared and ultraviolet cutoff frequencies, ω_r and ω_c, respectively. Rewrite the bath correlation function concisely:

$$
\begin{aligned}
< \tilde{\Gamma}_1(\tau)\tilde{\Gamma}_1(0) > &= < (\sum_i \lambda_i(a_i e^{-i\omega_i \tau} + a_i^\dagger e^{i\omega_i \tau}))(\sum_j \lambda_j(a_j + a_j^\dagger)) >_B \\
&= \int_{\omega_r}^{\omega_c} d\omega' J(\omega')(\cos(\omega'\tau)\coth(\frac{\hbar\omega'}{2k_B T}) - i\sin(\omega'\tau)) \qquad (2.30)
\end{aligned}
$$

The noise power spectrum is defined as the Fourier transform of the symmetric bath

correlation function:

$$
\begin{aligned}
S_U(\omega) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} < \Gamma_1(\tau)\Gamma_1(0) + \Gamma_1(0)\Gamma_1(\tau) >_B e^{-i\omega\tau} d\tau \\
&= \frac{1}{2\pi} \int_{-\infty}^{\infty} [\int_0^{\infty} d\omega' J(\omega') \cos(\omega'\tau) \coth(\frac{\hbar\omega'}{2k_B T}) \times 2] e^{-i\omega\tau} d\tau \\
&= \frac{1}{2\pi} \int_0^{\infty} d\omega' J(\omega') \coth(\frac{\hbar\omega'}{2k_B T}) \times 2 \times \frac{1}{2} \{\pi\delta(\omega - \omega') + \pi\delta(\omega' - \omega)\} \\
&= \frac{1}{2\pi} \cdot 2\pi J(\omega) \coth(\frac{\hbar\omega}{2k_B T}) \\
&= J(\omega) \coth(\frac{\hbar\omega}{2k_B T}). \quad\quad (2.31)
\end{aligned}
$$

The result of the following calculation has been used to derive Eq.(2.31)

$$
\begin{aligned}
\int_{-\infty}^{\infty} \cos(\omega'\tau) e^{-i\omega\tau} d\tau &= \int_{-\infty}^{\infty} (\frac{e^{i\omega'\tau} + e^{-i\omega'\tau}}{2}) e^{-i\omega\tau} d\tau \\
&= \frac{1}{2} [\int_{-\infty}^{\infty} e^{i(\omega'-\omega)\tau} + e^{-i(\omega'+\omega)\tau} d\tau] \\
&= \frac{1}{2} [\int_0^{\infty} e^{i(\omega'-\omega)\tau} d\tau - \int_0^{-\infty} e^{i(\omega'-\omega)\tau} d\tau \\
&\quad + \int_0^{\infty} e^{-i(\omega'+\omega)\tau} d\tau - \int_0^{-\infty} e^{-i(\omega'+\omega)\tau} d\tau] \\
&= \frac{1}{2} [\int_0^{\infty} e^{i(\omega'-\omega)\tau} d\tau + \int_0^{\infty} e^{i(\omega'-\omega)\tau} d\tau \\
&\quad + \int_0^{\infty} e^{-i(\omega'+\omega)\tau} d\tau + \int_0^{\infty} e^{-i(\omega'+\omega)\tau} d\tau] \\
&= \frac{1}{2} \{[\pi\delta(\omega - \omega') + i\frac{P}{\omega - \omega'}] + [\pi\delta(\omega' - \omega) + i\frac{P}{\omega' - \omega}] \\
&\quad + [\pi\delta(\omega' + \omega) + i\frac{P}{\omega' + \omega}] + [\pi\delta(-\omega' - \omega) + i\frac{P}{-\omega' - \omega}]\} \\
&= \frac{1}{2} \{\pi\delta(\omega - \omega') + \pi\delta(\omega' - \omega)\}
\end{aligned}
$$

The decoherence effect is discuss in a number of theoretical papers and the re-laxation of the excited states of the charge qubits off the degeneracy point had been discussed. The decoherence effects still lack of systematic studies. Especially the classical 1/f noise, we can only use it as a experimental data.

Figure 2.1 is the experimentally measured noise spectrum reported in [8]. The reduced noise spectrum S_U was derived from the measured relaxation rate Γ_1 for two
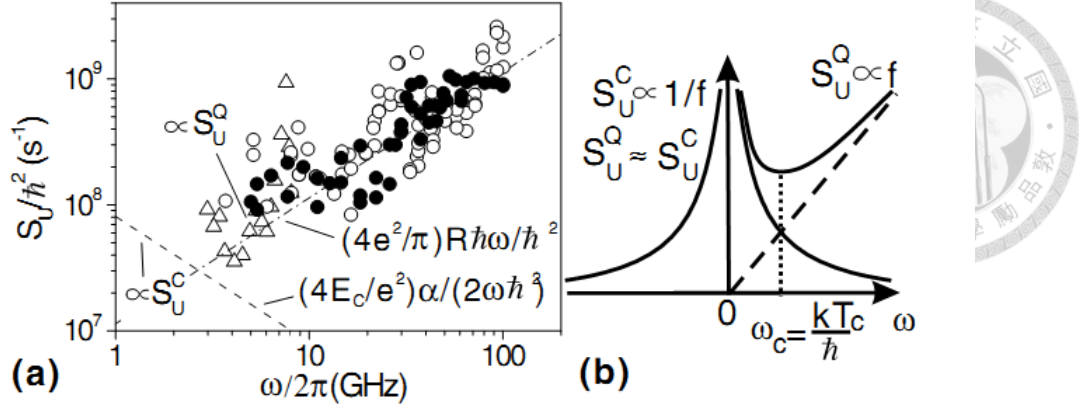
Figure 2.1: The noise spectrum of the experimental data from [8]. (a) The two dashed lines are Ohmic noise S_U^Q and 1/f noise S_U^C . (b) Simple diagram and the cross point ω_C. ω_C is renamed as ω_{cross} in the texts since it has be used to denote the cutoff frequency.

measured samples. The pattern of the $S_{\triangle E}$ can be seen as a combination of two linear lines representing as Ohmic noise and 1/f noise, respectively. The cross point of the two lines indicate that the noise contributions from 1/f and f dependencies are the same at $\omega_{cross} = 2\pi \times 2.6GHz$. This also defines an effective temperature $T_{cross} = \frac{\hbar\omega_{cross}}{k_B} = 120mK$, close to the electron temperature. Theoretical studies to construct a model to explain the measured noise spectra have been reported.

Instead of integrating the correlation function from zero to infinite, we choose a ultraviolet cutoff frequency and an infrared cutoff frequency as the upper bound and lower bound for the integration limits. The Ohmic noise spectral density is proportional to the frequency, so it dominates over the 1/f noise at high frequencies or low temperatures as shown below. For $\hbar\omega \gg 2k_BT$

$$
\begin{aligned}
S_U(\omega) &= J_f(\omega)\coth(\frac{\hbar\omega}{2k_BT}) \approx J_f(\omega) \\
&= S_U^Q(\omega) = \frac{4e^2}{\pi}R\hbar\omega
\end{aligned}
\tag{2.32}
$$

The symbol Q of $S_U^Q(\omega)$ means quantum. The other noise is the 1/f noise. Just like its name, the 1/f noise is proportional to the inverse of the frequency of the bath

oscillators. For $\hbar\omega \ll 2k_BT$

$$
\begin{aligned}
S_U(\omega) &= J_{1/f}(\omega)\coth(\frac{\hbar\omega}{2k_BT}) \approx J_{1/f}(\omega)\frac{2k_BT}{\hbar\omega} \\
&= S_U^C(\omega) = (\frac{4E_C}{e})^2\frac{\alpha}{2\omega}
\end{aligned}
\tag{2.33}
$$
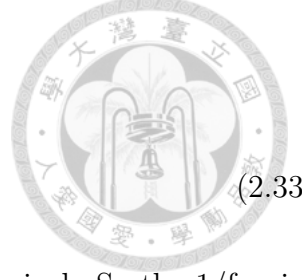
with $\alpha \approx (1.3 \times 10^{-3}e)^2$. The symbol C of $S_U^C(\omega)$ means classical. So the 1/f noise dominates at low frequencies or high temperatures as shown below. One may use the following equations to get the above equations.

$$
\coth(x) = \frac{1}{\tanh(x)} = \frac{\cosh(x)}{\sinh(x)} = \frac{e^x + e^{-x}}{e^x - e^{-x}} = \frac{1+e^{-2x}}{1-e^{-2x}}
$$

$$
x \gg 1 \Rightarrow \coth(x) \approx (1+e^{-2x})(1+e^{-2x}) \approx 1 + 2e^{-2x} + e^{-4x} \approx 1
$$

$$
x \ll 1 \Rightarrow e^{-2x} = 1 + (-2x) + \frac{(-2x)^2}{2!} + \dots
$$

$$
\therefore \coth(x \to \infty) \approx \frac{2}{2x} = \frac{1}{x}
$$

2.4 Diagram of simulation

The cross point of the ohmic and 1/f power spectrum is around $\omega_{cross} = 2\pi \times 2.6 GHz$.

So we define that the "high" and "low" frequency cutoff according to ω_{cross}, i.e.

$$
\begin{cases}
high & (\omega_c > \omega_{cross}) \\
low & (\omega_c < \omega_{cross})
\end{cases}
$$

We plot the simulated bath correlation functions using the parameters of $T = 10$ mK, $\omega_c = 2\pi \times 1$ GHz, $t_f = 55$ ps, and $dt = 0.1$ ps.

As one can see in Figure 2.3, the value of the 1/f noise correlation function is bigger than the Ohmic correlation function when the cutoff frequency is low. Also, the shape of the Ohmic correlation function is steeper and the 1/f noise remains flat. The blue lines in Figure 2.3 and Figure 2.4 indicate the real parts of the correlation
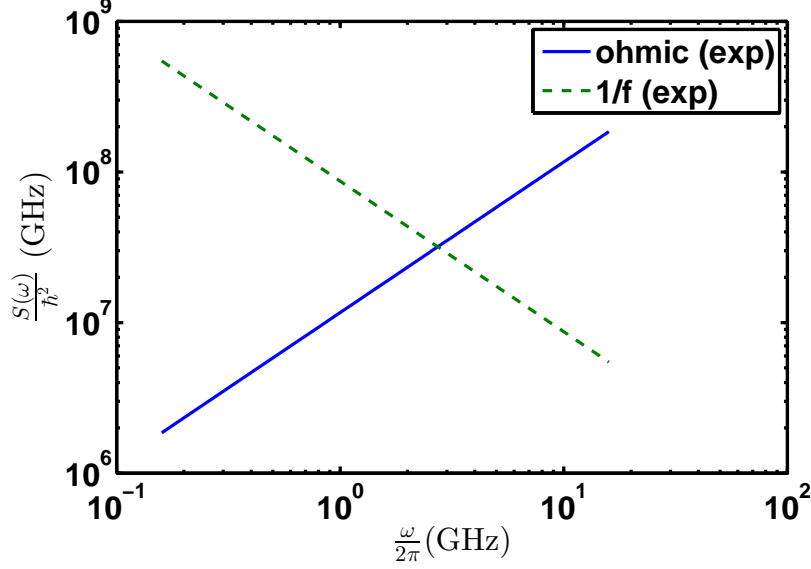
Figure 2.2: The diagram of power spectrum from experimental data [8]. The cross point is $\omega_{cross} = 2\pi \times 2.6$ GHz.

functions and the green lines indicate the imaginary parts of the correlation functions.

On the contrary, the bath spectral density with a high cutoff frequency gives different responses. We plot the bath correlations in Figure 2.4 for the parameters of $T = 10$ mK, $\omega_c = 2\pi \times 100$ GHz, $t_f = 55$ ps, and $dt = 0.1$ ps.

Both the shapes of the Ohmic and the 1/f noise correlation functions are steeper than their corresponding cases at low cutoff frequency. However, the values of the Ohmic correlation functions dominates over the 1/f noise correlation function at short time (within the time scale of the gate operation time). Therefore, we consider the Ohmic correlation function in the high cutoff frequency region.

Figure 2.3: Correlation functions (blue line: real part; green line: imaginary part) for low cutoff frequency. The parameters used are $T = 10$ mK, $\omega_c = 2\pi \times 1$ GHz, $t_f = 55$ ps, and $dt = 0.1$ ps.
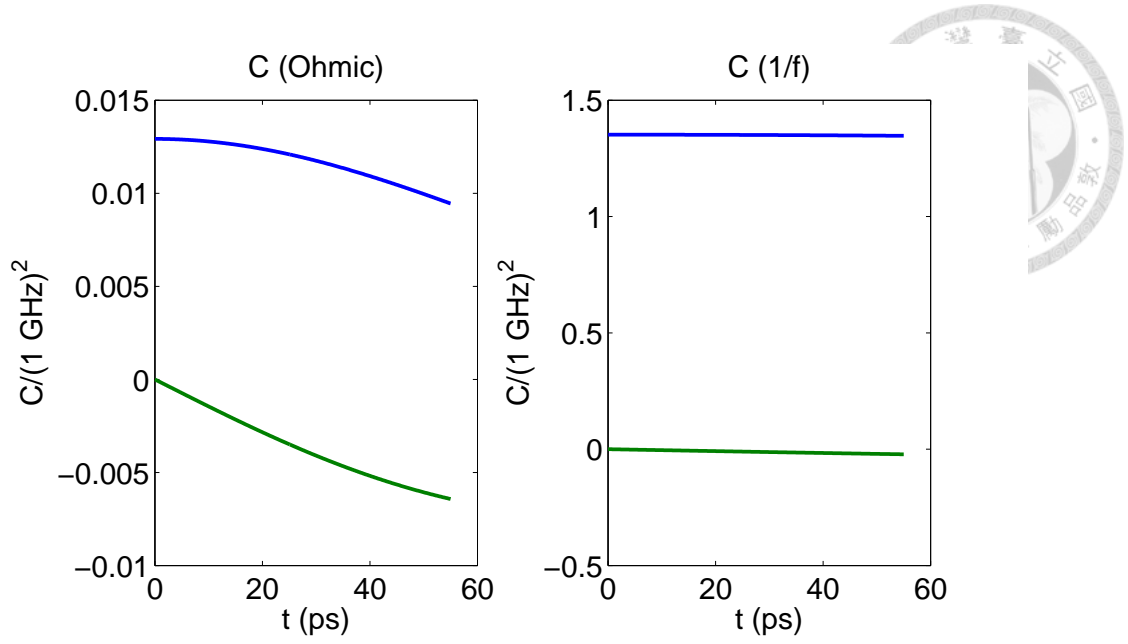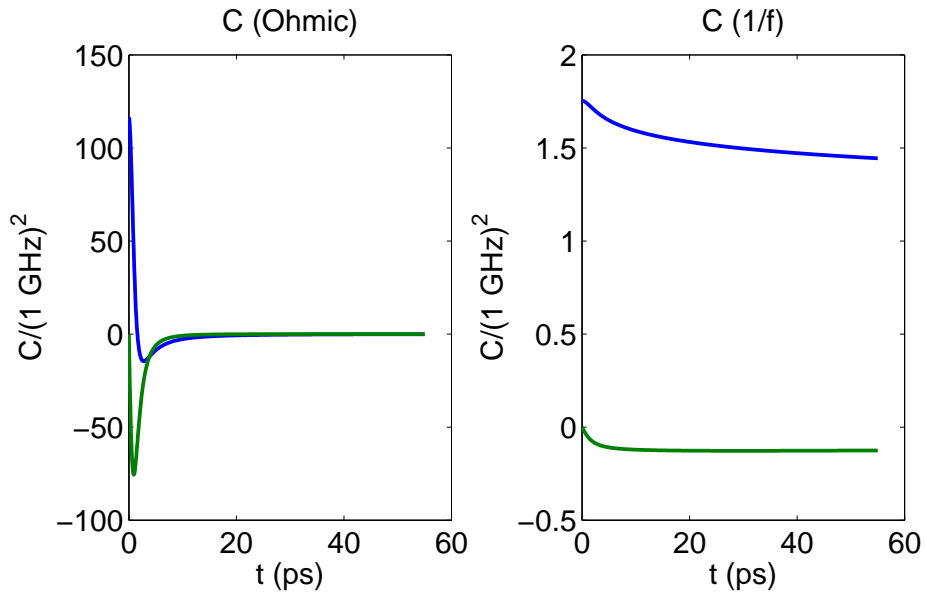

Figure 2.4: Correlation functions (blue line: real part; green line: imaginary part) for high cutoff frequency. The parameters used are $T = 10$ mK, $\omega_c = 2\pi \times 100$ GHz, $t_f = 55$ ps, and $dt = 0.1$ ps..

Chapter 3

Optimal Control

3.1 Introduction

The optimal control method can be applied to many different fields. But they all have the same purpose that is to get a extreme accuracy result. There are many way to do the optimal control such as GRAPE, Newton method, quasi-Newton method, and Krotov method. The optimal control has an advantage to design a unitary quantum gate in a closed system. In this thesis, we also use optimal control method to deal with open systems, especially, non-Markovian open quantum systems. Some details regarding the implementation of the optimal control are put in Appendices. Different fidelity/error measures that are adopted as the objective function for optimization and error values. We will discuss two commonly used error measures in Sec. 3.2 and will conclude in Chapter 4 why we use the error J_2 defined in Eq. (3.2) for the cases of open quantum systems. Since we are dealing with the gate operation, we need to perform the operation independent of initial states. So I introduce the state-independent optimal control and the superoperator concept latter.

3.2 Error/Fidelity Definition

According to [22], we can simply definition two kinds definition of error/fidelity.

$$J_1 = 1 - \frac{1}{N}Tr(\mathbf{Q}^\dagger \mathbf{G}(T)) \tag{3.1}$$

$$J_2 = \frac{1}{2N}Tr[(\mathbf{Q} - \mathbf{G}(T))^\dagger (\mathbf{Q} - \mathbf{G}(T))] \tag{3.2}$$

with $\mathbf{Q}$ is the target superoperator (CNOT gate), and $\mathbf{G}(T)$ is the final optimized propagator in a superoperator form, and N is the dimension of the superoperator. The concept of superoperator is presented in section 3.5.2 since it also relates to the concept of computational states.

In a closed system, the decoherence effect is not considered yet, the propagator should be unitary since the system is "perfect". Thus in a closed system, we have

$$\mathbf{G}(t)^\dagger \mathbf{G}(t) = \mathbf{I}, \tag{3.3}$$

and $t \in [0, T]$. Expanding J_2 , one can obtain the result that $J_2 = J_1$ in a "closed" system.

However, we cannot promise Eq.(3.3) in a (non-Markovian) open system. And since $\mathbf{G}(T)$ is no longer a unitary matrix, then the definition of J_1 may not be a better measure of gate error. If we consider leakage levels, we will not have a unitary result for the gate operation since we project the operation matrix onto the computational basis states and let the other elements be zeros.

So when we consider an open system or leakage levels, we will use the error definition of J_2. However, to prove the optimal control based on the Krotov' s method in the next section. We will still use J_1 as a part of the cost function.

Note that, when we replace the J_1 into J_2 in the cost function, the optimization procedure can still work and the update rule just like Eq. (3.23) is obtained. However, this change of target might cause the original derivation of Krotov unable to be done and make the differential derivation more like the gradient ascent method. But the

step by step update is still more like a Krotov type. So even the original proof of Krotov might be a problem when we choose J_2 as our error function, the result is still monotonically convergent if we choose the right penalty function λ.
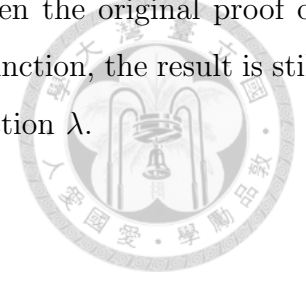
3.3 One Qubit Optimal Control

There are many paper [9, 10, 11, 12, 13] describe the standard way of the Krotov optimal control method. However, some other papers use a different approach and obtain the same result,[14, 15]. Here we choose the most comprehensive parts in these approaches and combine them together to show the optimal iteration update equation.

Consider the equation of motion

$$\frac{d}{dt}\overrightarrow{\rho}(t) = \Lambda(t)\overrightarrow{\rho}(t), \tag{3.4}$$

where $\Lambda(t)$ is a matrix operator (superoperator) acting on a column vector $\overrightarrow{\rho}(t)$. Let the propagator (superoperator) of $\overrightarrow{\rho}(t)$ be $\mathbf{G}(t)$. That is

$$\overrightarrow{\rho}(t) = \mathbf{G}(t,0)\overrightarrow{\rho}(0) = \mathbf{G}(t)\overrightarrow{\rho}(0). \tag{3.5}$$

So we can rewrite the equation of motion

$$\frac{d}{dt}\mathbf{G}(t)\overrightarrow{\rho}(0) = \Lambda(t)\mathbf{G}(t)\overrightarrow{\rho}(0), \tag{3.6}$$

and get

$$\frac{d}{dt}\mathbf{G}(t) = \Lambda(t)\mathbf{G}(t), \tag{3.7}$$

which will be Schrodinger equation in our control problem if the environment is excluded. Suppose that the superoperator of the Hamiltonian $\Lambda^k(t)$ has the form

$$\Lambda^k(t) = \Lambda_d + \epsilon^k(t)\Lambda_1, \tag{3.8}$$

where k is the number of iteration, Λ_d and $\mathbf{\Lambda_1}$ are time-independent, and $\epsilon(t)$ is the control parameter. The cost function for the propagator is chosen to be

$$
\begin{aligned}
J(\epsilon^{k+1}) = \; & \mathrm{ReTr}(\mathbf{Q}^\dagger \mathbf{G^{k+1}}(T)) \\
& - \int_0^T \mathrm{ReTr}\{[\frac{d}{dt}\mathbf{G}^{k+1}(t) - \Lambda^{k+1}(t)\mathbf{G}^{k+1}(t)]\mathbf{B}(t)\}dt \\
& - \lambda \int_0^T |\epsilon^{k+1}(t) - \epsilon^k(t)|^2 dt
\end{aligned}
\tag{3.9}
$$

The first term in the cost function is the error function, J_1. One may replace it with J_2 and the proof for error function J_2 is put in Appendix C. The second term is the evolution condition Eq.(3.6) with the Lagrange multiplier $\mathbf{B}^k(t)$ which is the backward propagating superoperator. The third term is the augmented Lagrange multiplier with penalty function λ. This term makes sure that we can get the update rule of control pulses.

Using the relation

$$
\int_0^T \mathrm{ReTr}\{\frac{d}{dt}[\mathbf{G}^{k+1}(t)\mathbf{B}(t)] - \mathbf{G}^{k+1}(t)\frac{d}{dt}\mathbf{B}(t)\}dt
$$
$$
= \mathrm{ReTr}\{\mathbf{G}^{k+1}(t)\mathbf{B}(t)\}|_0^T - \int_0^T \mathrm{ReTr}\{\mathbf{G}^{k+1}(t)\frac{d}{dt}
\tag{3.10}
$$

we can get

$$
\begin{aligned}
J(\epsilon^{k+1}) = \; & \mathrm{ReTr}(\mathbf{Q}^\dagger \mathbf{G}^{k+1}(T)) - \mathrm{ReTr}\{\mathbf{G}^{k+1}(t)\mathbf{B}(t)\}|_0^T \\
& + \int_0^T \mathrm{ReTr}\{\mathbf{G}^{k+1}(t)\frac{d}{dt}\mathbf{B}(t)\}dt \\
& + \int_0^T \mathrm{ReTr}\{\Lambda^{k+1}(t)\mathbf{G}^{k+1}(t)\mathbf{B}(t)\}dt \\
& - \lambda \int_0^T |\epsilon^{k+1}(t) - \epsilon^k(t)|^2 dt.
\end{aligned}
\tag{3.11}
$$

Next, we focus on the difference of the cost function between two neighboring itera-

tions:

$$
\begin{aligned}
J(\epsilon^{k+1}) - J(\epsilon^k) = &\ \mathrm{ReTr}(\mathbf{Q}^\dagger \Delta \mathbf{G}(T)) - \mathrm{ReTr}(\Delta \mathbf{G}(t)\mathbf{B}(t))|_0^T \\
&+ \int_0^T \mathrm{ReTr}\{\Delta \mathbf{G}(t)\frac{d}{dt}\mathbf{B}(t)\}dt \\
&+ \int_0^T \mathrm{ReTr}\{\Lambda^k(t)\Delta \mathbf{G}(t)\mathbf{B}(t)\}dt \\
&+ \int_0^T \mathrm{ReTr}\{\Delta \Lambda(t)\mathbf{G}^{k+1}(t)\mathbf{B}(t)\}dt \\
&- \lambda \int_0^T [\epsilon^{k+1}(t) - \epsilon^k(t)]^2 dt \\
&+ \lambda \int_0^T [\epsilon^k(t) - \epsilon^{k-1}(t)]^2 dt,
\end{aligned}
\tag{3.12}
$$

where $\Delta \mathbf{G}(t) = \mathbf{G}^{k+1}(t) - \mathbf{G}^k(t)$. And here we insert the boundary condition and the evolution equation of $\mathbf{G}(t)$ and $\mathbf{B}(t)$ to let some terms of Eq.(3.12) to vanish.

$$
\begin{cases}
\mathbf{B}(T) = \mathbf{Q}^\dagger, \\
\frac{d}{dt}\boldsymbol{B}(t) = -\mathbf{B}(t)\Lambda^k(t).
\end{cases}
\tag{3.13}
$$

and we define the iteration number of $\mathbf{B}(t)$ by the iteration number of $\Lambda^k(t)$. So we can rewrite it as

$$
\begin{cases}
\mathbf{B}^k(T) = \mathbf{Q}^\dagger, \\
\frac{d}{dt}\boldsymbol{B}^k(t) = -\mathbf{B}^k(t)\Lambda^k(t).
\end{cases}
\tag{3.14}
$$

But we can also use another way to get the boundary condition of $\mathbf{B}(t)$, and this method is simpler. Just differential the Eq.(3.11) with the forward propagator $\mathbf{G}$ and you will get the same result of the backward propagator $\mathbf{B}(t)$.

$$
\frac{\partial \mathbf{J}}{\partial \mathbf{G}} = 0 \Rightarrow
\begin{cases}
\mathbf{B}(T) = \mathbf{Q}^\dagger, \\
\frac{d}{dt}\boldsymbol{B}(t) = -\mathbf{B}(t)\Lambda(t).
\end{cases}
\tag{3.15}
$$

Notice that we ignore the iteration number of $\Lambda(t)$ in this method since it will be

wrong. Because of we use the difference of $\Lambda(t)$ in Eq.(3.12). That is,

$$\Delta\Lambda(t) = \Lambda^{k+1}(t) - \Lambda^k(t). \tag{3.16}$$

But the differential method give us the same form and much simpler procedure than Eq.(3.12). So we use this method when we consider another error definition J_2. Also we consider the projection the whole matrix into computational states in Appendix C. We already have the evolution equation Eq.(3.7). Using the initial condition of the forward propagator $\mathbf{G}$ gives

$$\mathbf{G}(0) = \mathbf{I} \Rightarrow \Delta\mathbf{G}(0) = 0. \tag{3.17}$$

Let

$$\triangle a(t) = \epsilon^{k+1}(t) - \epsilon^k(t), \tag{3.18}$$

$$\triangle b(t) = \epsilon^k(t) - \epsilon^{k-1}(t), \tag{3.19}$$

and then from Eq.(3.8) we have

$$\Delta\Lambda(t) = \Delta a(t)\Lambda_1. \tag{3.20}$$

Substituting the boundary condition of $\mathbf{B}^k(t)$ into the difference of the cost function, we obtain

$$\begin{aligned}
\Delta J &= J(\epsilon^{k+1}) - J(\epsilon^k) \\
&= + \int_0^T \mathrm{ReTr}\{\Delta a(t)\Lambda_1 \mathbf{G}^{k+1}(t)\mathbf{B}^k(t)\}dt \\
&\quad - \lambda \int_0^T [\Delta a(t)]^2 dt \\
&\quad + \lambda \int_0^T [\Delta b(t)]^2 dt.
\end{aligned} \tag{3.21}$$

Taking the derivation with respect to $\triangle a(t)$, and requiring it to vanish

$$
\begin{aligned}
\frac{\partial(\Delta J)}{\partial(\Delta a(t))} &= \int_0^T ReTr\{\mathbf{B}^k(t)\Lambda_1 \mathbf{G}^{k+1}(t)\}dt \\
&\quad -2\lambda \int_0^T \Delta a(t)dt = 0,
\end{aligned} \tag{3.22}
$$

we obtain an update rule for the control field

$$
\begin{aligned}
\epsilon^{k+1}(t) &= \epsilon^k(t) + \Delta a \\
&= \epsilon^k(t) + \frac{1}{2\lambda} ReTr\{\mathbf{B}^k(t)\Lambda_1 \mathbf{G}^{k+1}(t)\}.
\end{aligned} \tag{3.23}
$$

3.4 Two Qubit Optimal Control

If now we have two independent control fields on the two qubits, respectively, in the operator $\Lambda^k(t)$, where

$$
\Lambda^k(t) = \Lambda_d + \epsilon_1^k(t)\Lambda_1 + \epsilon_2^k(t)\Lambda_2, \tag{3.24}
$$

and the update can be easily generalized to be

$$
\epsilon_1^{k+1}(t) = \epsilon_1^k(t) + \frac{1}{2\lambda} ReTr\{\mathbf{B}^k(t)\Lambda_1 \mathbf{G}^{k+1}(t)\} \tag{3.25}
$$

$$
\epsilon_2^{k+1}(t) = \epsilon_2^k(t) + \frac{1}{2\lambda} ReTr\{\mathbf{B}^k(t)\Lambda_2 \mathbf{G}^{k+1}(t)\} \tag{3.26}
$$

3.5 State-independent Optimal Control

3.5.1 Computational States and Leakage States

Instead of the optimization of state-selective transition which is frequently used in chemical reaction, we focus on the state independent optimal control for quantum gate operations. Consider a Hilbert space which is the tensor product of the system

and bath Hilbert spaces. The Hamiltonian of the whole system can be written as

$$\mathbf{H} = \mathbf{H}_S \otimes \mathbf{H}_B, \tag{3.27}$$

where $\mathbf{H_S}$ is the system Hamiltonian acting on the system Hilbert space and $\mathbf{H_B}$ is the environment Hamiltonian acting on the environment Hilbert space.

If we consider the leakage levels and two independent baths coupled to the two qubits, respectively, we may write down

$$\mathbf{H}_S = \mathbf{H}_{\text{com}} \oplus \mathbf{H}_{\text{leak}}, \tag{3.28}$$

and

$$\mathbf{H}_B = \mathbf{H}_{B1} \otimes \mathbf{H}_{B2}, \tag{3.29}$$

where $\mathbf{H}_{\text{com}}$ is the qubit Hamiltonian acting on the Hilbert space of the computational basis states and $\mathbf{H}_{\text{leak}}$ is the leakage part of the Hamiltonian.

Combining these we can get

$$\mathbf{H} = (\mathbf{H}_{\text{com}} \oplus \mathbf{H}_{\text{leak}}) \otimes (\mathbf{H}_{B1} \otimes \mathbf{H}_{B2}). \tag{3.30}$$

So how does we denote the evolution of state without telling you the initial one? If we consider a state evolution with a time-independent Hamiltonian, we have

$$|\psi_S(t)\rangle = e^{-\frac{i}{\hbar}\mathbf{H}t}|\psi_S(0)\rangle, \tag{3.31}$$

$$\rho_S(t) = e^{-\frac{i}{\hbar}\mathbf{H}t}\rho_S(0)e^{\frac{i}{\hbar}\mathbf{H}t}. \tag{3.32}$$

Now we transform it to superoperator form representation:

$$\rho_S(t) = \xi_t\{\rho_S(0)\}. \tag{3.33}$$

In this case, the superoperator is functionally dependent on externally applied field $\epsilon(t)$, i.e.

$$\xi_t = \xi_t[\epsilon(t)].$$

Since the density matrix is a positive operator, it must remain positive under the gate operation. So the map of the superoperator ξ must be completely positive. We can express it by Kraus operators K_m:

$$\rho_S(t) = \xi_t[\epsilon(t)]\{\rho_S(0)\} = \sum_m \mathbf{K}_m[\epsilon](t)\rho_S(0)\mathbf{K}_m^\dagger[\epsilon](t) \tag{3.34}$$

with $\mathbf{K}_m[\epsilon](t)$ depending on the propagator of the composite system and on $\epsilon(t)$.

So we want to know when we can find the target control parameter using the optimal control theory. How do we know that the superoperator $\xi_t[\epsilon^*]$ is close to the target mapping ξ_D? The target gate operation we want is a unitary operation. In the state vector representation we can write

$$|\psi_S(t_f)\rangle = \mathbf{U}_D(t_f)|\psi_S(0)\rangle.$$

In the density matrix representation, we have

$$\rho_S(t_f) = \mathbf{U}_D(t_f)\rho_S(0)\mathbf{U}_D^\dagger(t_f).$$

In terms of the state-independent superoperator, $\rho_S(t_f) = \xi_f\{\rho_S(0)\}$, where

$$\xi_f\{\dots\} = \mathbf{U}_D(t_f)(\dots)\mathbf{U}_D^\dagger(t_f) \tag{3.35}$$

It is useful to formulate a cost functional within the language of process tomography, so that one can easily see the difference between the target superoperator or if it is

unitary or not. By expanding the Kraus operator, we can rewrite the mapping.

$$\mathbf{K}_m(t) = \sum_n \alpha_{mn}\overline{\mathbf{K}}_n \tag{3.36}$$

with $\alpha_{mn} \in C$ and $\overline{K}_n \in A$, where A denotes a complete basis set of $M \times M$ matrices. The total number of orthonormal basis states M is

$$M = M_{\text{com}} + M_{\text{leak}} \tag{3.37}$$

where M_{com} is the number of the computational states and M_{leak} is the number of the leakage states. So we can write the final density matrix

$$\rho(t_f) = \xi_{t_f}\{\rho(0)\} = \sum_{m,n} \overline{\mathbf{K}}_m\rho(0)\overline{\mathbf{K}}_m^{\dagger}\chi_{mn}(t_f) \tag{3.38}$$

with $\chi_{mn} = \sum_k \alpha_{km}\alpha_{kn}^{\star}$.

Considering the Josephson charge qubits we talk about before. We have

$$\begin{cases} M_{\text{com}} = 2 \quad \text{(computational)} \\ M_{\text{leak}} = 2 \quad \text{(leakage)} \end{cases} \tag{3.39}$$

with basis states $\{|0>, |1>\}$ and $\{|-1>, |2>\}$ respectively.

For one qubit, the dimension of the representation of χ is $M^2 \times M^2 = 4 \times 4 = 16$. For two qubits, because of the tensor product of the Hilbert spaces of two different qubits, we have

$$\begin{cases} M_{\text{com}}^1 \times M_{\text{com}}^2 = 4 \quad \text{(computational)} \\ M_{\text{leak}}^1 \times M_{\text{leak}}^2 = 4 \quad \text{(leakage)} \end{cases} \tag{3.40}$$

So the dimension of the representation of χ is $M^2 \times M^2 = 16 \times 16 = 256$.

Since the dimension of the total Hamiltonian superoperator matrix will become so big when you consider the leakage states. It will take a lot of time for the evolution of

the propagator. The matrix χ is usually termed as the process tomography matrix. In order to compute the element of χ, we choose a fixed set of operators $\{\sigma_j\} = B$ (and choose $B = A$). We can write down the time evolution with respect to the mapping ξ.

$$\sigma_j(t_f) \equiv \xi_{t_f}\{\sigma_j\} \tag{3.41}$$

For one-qubit operations with two leakage levels or for two-qubit gates, the tensor product of Pauli matrices $\tau_i \otimes \tau_j$ with $i, j \in \{x, y, z, 0\}$ and $\tau_0 = I$ is a convenient choice for A, B. Recently, people use process tomography method has been used to deal with super conducting qubit problem [2].

The mapping is linear, so one can rewrite it by

$$\xi_{t_f}\{\sigma_j\} = \sum_k c_{jk}\sigma_k \tag{3.42}$$

The mapping ξ_{t_f} is determined by the coefficients c_{jk}, and the experimental determination of the mapping involves quantum state tomography.

Relating c_{jk} to χ_{mn} yields

$$\sum_k c_{jk}\sigma_k = \sum_{m,n} \sigma_m\sigma_j\sigma_n\chi_{mn}(t_f). \tag{3.43}$$

By using

$$\langle \sigma_i, \sigma_k \rangle = i\delta_{ik}, \tag{3.44}$$

we can extract c_{ij}:

$$c_{ij} = \langle \sigma_i, \sigma_j(t_f) \rangle = \sum_{m,n}\langle \sigma_i, \sigma_m\sigma_j\sigma_n \rangle\chi_{mn}(t_f) \equiv \sum_{m,n} B_{imjn}\chi_{mn}(t_f). \tag{3.45}$$

So we can solve the system of linear equations $\langle \sigma_i, \sigma_j(t_f) \rangle = \sum_{m,n} B_{imjn}\chi_{mn}(t_f)$, which involves the inversion of B.

Define the operator

$$\hat{\chi} = \sum_{m,n} (\bar{\mathbf{K}}_n^\star \otimes \bar{\mathbf{K}}_n) \chi_{mn} \tag{3.46}$$

and the cost functional

$$J \equiv \left\| P\hat{\chi} - \hat{\chi}^D \right\|^2 = Tr\{ \left[P\hat{\chi} - \hat{\chi}^D \right] \left[P\hat{\chi} - \hat{\chi}^D \right]^\dagger \}, \tag{3.47}$$

where P is the projection operator that project onto the M_C-dimensional computational Hilbert space. Ref.[2] deals with the one-qubit operation with two leakage states. The projector is

$$P_{ij} = \sum_{k=6,7,10,11} \delta_{i,k} \delta_{j,k} \tag{3.48}$$

with $i, j \in \{1, 2, \cdots, M^2 = 16\}$. How to explain the values of k ? For our two-qubit operation, we consider that each qubits has two leakage states. We use the computational tensor product state basis, e.g.

$$|00\rangle = |0\rangle \otimes |0\rangle. \tag{3.49}$$

The CNOT operation in the computational state basis reads

$$\text{CNOT} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{3.50}$$

The corresponding transitions are

$$
\begin{cases}
|00\rangle \rightarrow |01\rangle, \\
|01\rangle \rightarrow |00\rangle, \\
|10\rangle \rightarrow |10\rangle, \\
|11\rangle \rightarrow |11\rangle.
\end{cases}
\tag{3.51}
$$

By considering the leakage states from $n = -1 \sim 2$, the target CNOT matrix must be rewritten. Begin with the first qubit state in $|-1\rangle$ and combine the second qubit state as a product state. Then we have the basis states

$$
|-1-1\rangle, |-10\rangle, |-11\rangle, |-12\rangle, |0-1\rangle, |00\rangle, |01\rangle, |02\rangle
$$
$$
, |1-1\rangle, |10\rangle, |11\rangle, |12\rangle, |2-1\rangle, |20\rangle, |21\rangle, |22\rangle.
$$

Checking with the Eq. (3.51), we can see that the tributed states are No.6,7,10,11. Thus, the values of k in the expression of the projector, Eq.(3.48), should be exactly those numbers with the other vanishing since we concern only the CNOT operation in the computational state basis.

3.5.2 Superoperator and Column Representation

We transform the density matrix into a vector using the matlab function *reshape*. The label of vector is first from the top to bottom of the first column of the matrix and then jumps to the top of next column. The procedure is repeated until all the elements are rearranged into the vector. That is, splitting the matrix into columns and then combine all the columns into one vector.

Let's consider a simple example of a density matix below and assume the index

numbers with bar on the top are in the computational states:

$$
\rho = \begin{pmatrix} 1 & 5 & 9 & 13 \\ 2 & \overline{6} & \overline{10} & 14 \\ 3 & \overline{7} & \overline{11} & 15 \\ 4 & 8 & 12 & 16 \end{pmatrix}.
\tag{3.52}
$$

Using the matlab *reshape* function will transform the matirx of Eq.(3.52) into

$$
reshape(\rho, \text{ row} = 16, \text{ column} = 1) \rightarrow \text{col}(\rho) = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ \overline{6} \\ \overline{7} \\ 8 \\ 9 \\ \overline{10} \\ \overline{11} \\ 12 \\ 13 \\ 14 \\ 15 \\ 16 \end{pmatrix}.
\tag{3.53}
$$

When do we need this column representation? Evolving a density matrix, unlike pure state, we need to multiply the evolution operator both in the front and behind, i.e., $\rho(t) = \mathbf{U}(t)\rho(0)\mathbf{U}^{\dagger}(t)$. More generally, we will encounter the operation of

$$
\overline{\rho} = A\rho B
\tag{3.54}
$$

in the master equation for the reduced system density matrix in open quantum systems. Define a superoperator by combining the equation Eq. (3.38) and Eq. (3.46)

$$\hat{\chi} = \mathbf{B}^T \otimes \mathbf{A}. \tag{3.55}$$

The operation of Eq.(3.54) is transformed as

$$\text{col}(\overline{\rho}) = \hat{\chi}\text{col}(\rho) = (\mathbf{B}^T \otimes \mathbf{A})\text{col}(\rho). \tag{3.56}$$

The benefit of doing this is that (a) the operation part is complete in front of the density matrix and (b) this will make the form of equation of motion simpler and easier to compute by a computer.

But remember that the dimension of the superoperator is now bigger than ever.

$$\text{dimension} \rightarrow \begin{cases} \mathbf{A} & (M \times M), \\ \mathbf{B} & (M \times M), \\ \hat{\chi} & (M^2 \times M^2). \end{cases}$$

Let's consider the two-qubit CNOT operator with two leakage levels. In this

column vector representation,

$$
CNOT =
\begin{pmatrix}
0 & \cdots & \cdots & & & & & & & & & 0 \\
\vdots & \ddots & & & & & & & & & & \\
\vdots & & \ddots & & & & & & & & & \\
& & & \overline{0} & \overline{1} & & \overline{0} & \overline{0} & & & & \\
& & & \overline{1} & \overline{0} & & \overline{0} & \overline{0} & & & & \\
& & & & & & & & & & & \\
& & & \overline{0} & \overline{0} & & \overline{1} & \overline{0} & & & & \\
& & & \overline{0} & \overline{0} & & \overline{0} & \overline{1} & & & & \\
& & & & & & & & & \ddots & & \vdots \\
& & & & & & & & & & \ddots & \vdots \\
0 & \cdots & \cdots & & & & & & \cdots & \cdots & & 0
\end{pmatrix}
\tag{3.57}
$$

with the values of CNOT operator in the computational states, specified with bars on the tops shown above. The operator in the number state basis which including leakage states is a 16×16 matrix. But when we consider it in the superoperator form, it is going to be a large 256×256 matrix. We will use this column vector and superoperator form to implement our numerical calculations on a computer.

Chapter 4

Results

4.1 Overview

In this Chapter, we present our optimal control results for four different cases, namely, closed/open system with/without leakage states.

But first we give an example of how we monitor the whole optimal procedure and how we determine whether the result is good or bad.

The plot in the first row of Fig. 4.1 shows two different error measures, J_1 and J_2, versus iteration number. As the number of iteration increases, the error J_1 and error J_2 both decrease. The colorful matrix bar figure shows the values of the elements of the final propagator in the column vector representation. Since this result is for a closed system with no leakage states, two errors J_1 and J_2 are the same and the propagators during this optimal procedure are all unitary. The plots in the bottom row show two control field pulses or sequence, where "control n1" represents $\overline{n}_{g1}$ and "control n2" represents $\overline{n}_{g2}$ as in Eq.(2.4).
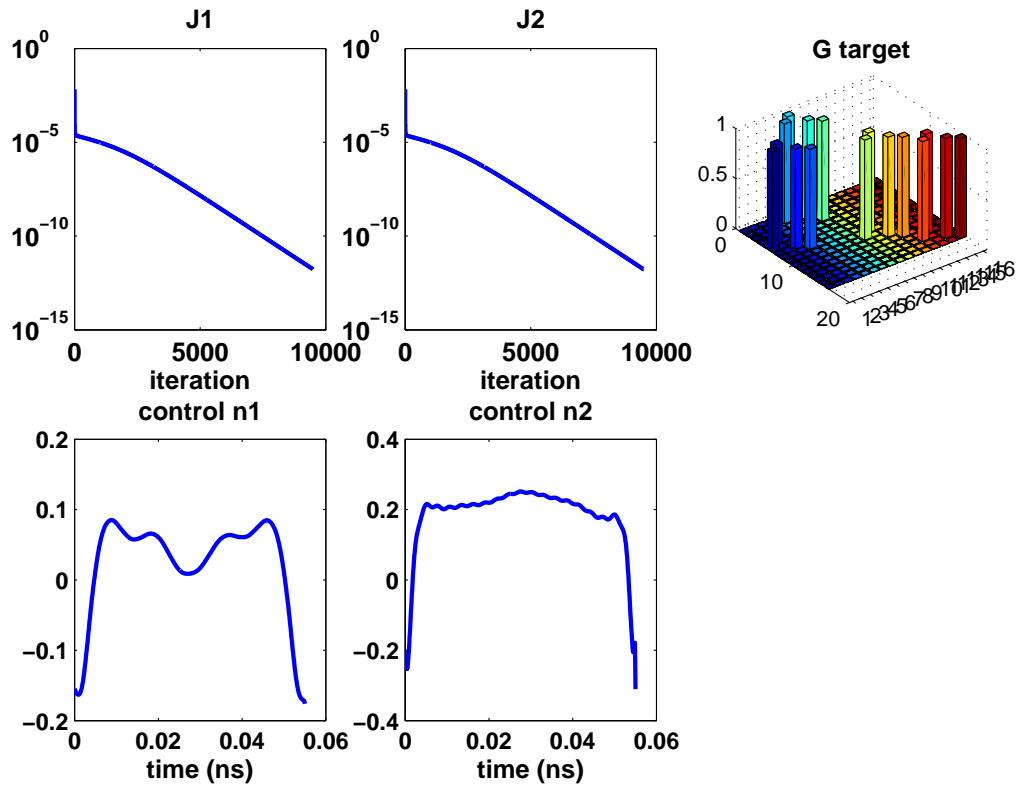
Figure 4.1: A standard diagram of analyzing a whole optimal control procedure. The first row shows two errors J_1 and J_2 versus iteration number, and a color bar matrix representing the elements of superoperator at the final operation time. The second row shows the control pulses of the final optimal results.
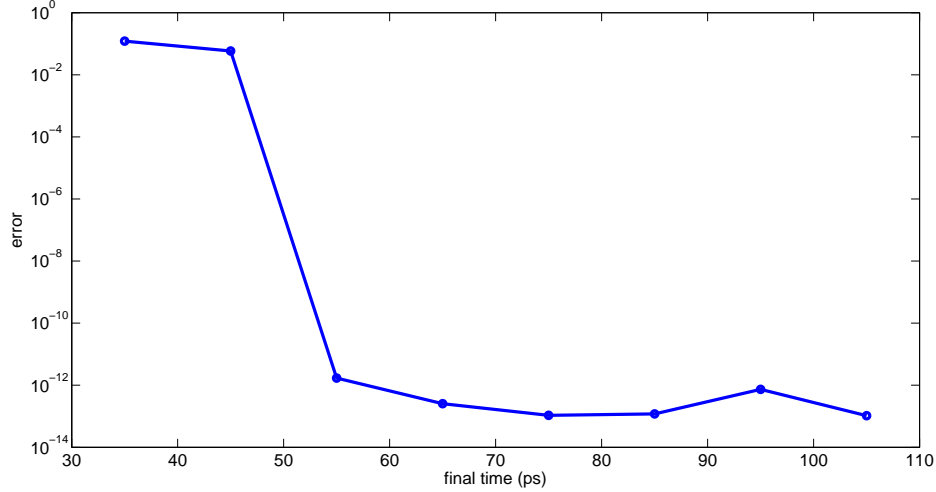
Figure 4.2: Optimal control CNOT gate error versus final gate operation time for a closed system without leakage states.

4.2 Closed System with No Leakage Levels

4.2.1 Overview

We know that performing a gate operation needs time, and there is a limit on how fast gate can be performed.Since errors J_1 and J_2 are the same for unitary operations in a closed system, one can use either of them as a error measure. We can see that the minimum time we need to perform this CNOT operation.

From Fig. 4.2, we can see that if the CNOT gate operation time is shorter than 55ps, the error grows rapidly. This is become there is a limit on how large the parameters in the tunable terms of the Hamiltonian.

In the following analysis (with leakage or not), we will take the CNOT operation time of 55ps as a reference standard.

4.2.2 Pulse Shape

Figures 4.3~4.5 shows the CNOT gate error versus the iteration number, the matrix element of the find propagator in the superoperator form, and control field pulses for gate operation time of 35ps, 45ps, and 55ps, respectively. One can see that the gate

error J_1 and J_2 for $t_f = 35$ps and $t_f = 45$ps can not be formed when the number of iterations is above about 100. In contrast the gate error can still be improved with the increasing iteration number for the case of $t_f = 55$ps. As a result, it has a much lower value of gate errors then the other two cases.

This can also be seen when one compares the values of the color bars representing the matrix element of the superoperator in there three cases with that of the target CNOT operation in Fig. 4.1.

Significant difference in color bar values between $t_f = 35$ps, $t_f = 45$ps cases and the target CNOT case can be observed, while the agreement between the $t_f = 55$ps case and the target CNOT case is excellent.

Figure 4.6 shows the optimal control pulses of a CNOT gate for various operation times. One can clearly see that the control pulse shapes depend on the values of the gate operation times.

4.3 Open System with No Leakage Levels

4.3.1 Overview

In the case of open system, we want to know how much the decoherence effect of the environment could affect the optimal results, and how well can we perform a optimal CNOT gate control under the decoherence.

The CNOT gate errors for different temperatures and different cutoff frequencies are plot in Fig. 4.8. The best result with error $J_2 \sim 10^{-5}$ is at the lowest temperature $10mK$ and the lowest cutoff $\omega_c = 2\pi \times 1$ GHz. This does make sense because the decoherence effect should be small when the temperature and the cutoff frequency are both small. Since this result is for the CNOT gate operation time of $55ps$, this operation time is fast so that the decoherence effect has not fully dominated. If one compares the optimized pulses in the closed system case with their corresponding optimal control pulses obtained in the open system cases for the CNOT gate with $t_f = 55ps$, one will find that the control pulses shapes do not change much. That is
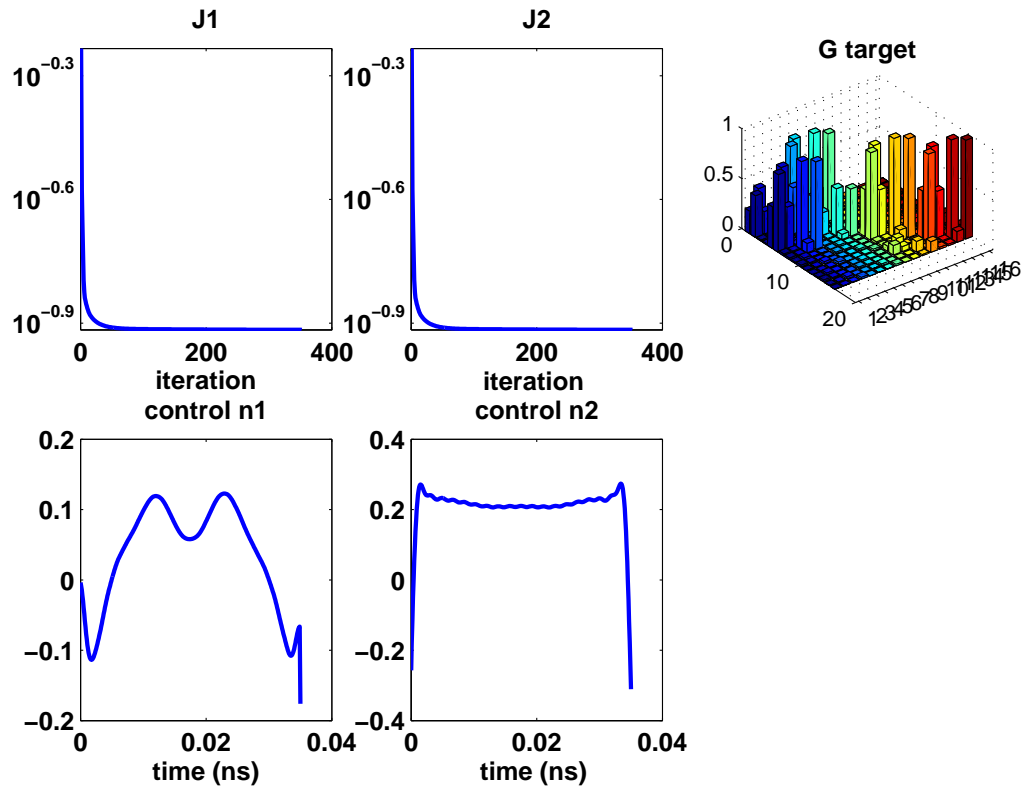
Figure 4.3: Result of the CNOT gate operation for a operation time $t_f = 35ps$. The first row shows two errors J_1 and J_2 versus iteration number, and a color bar matrix representing the elements of superoperator at the final operation time. The second row shows the control pulses of the final optimal results.
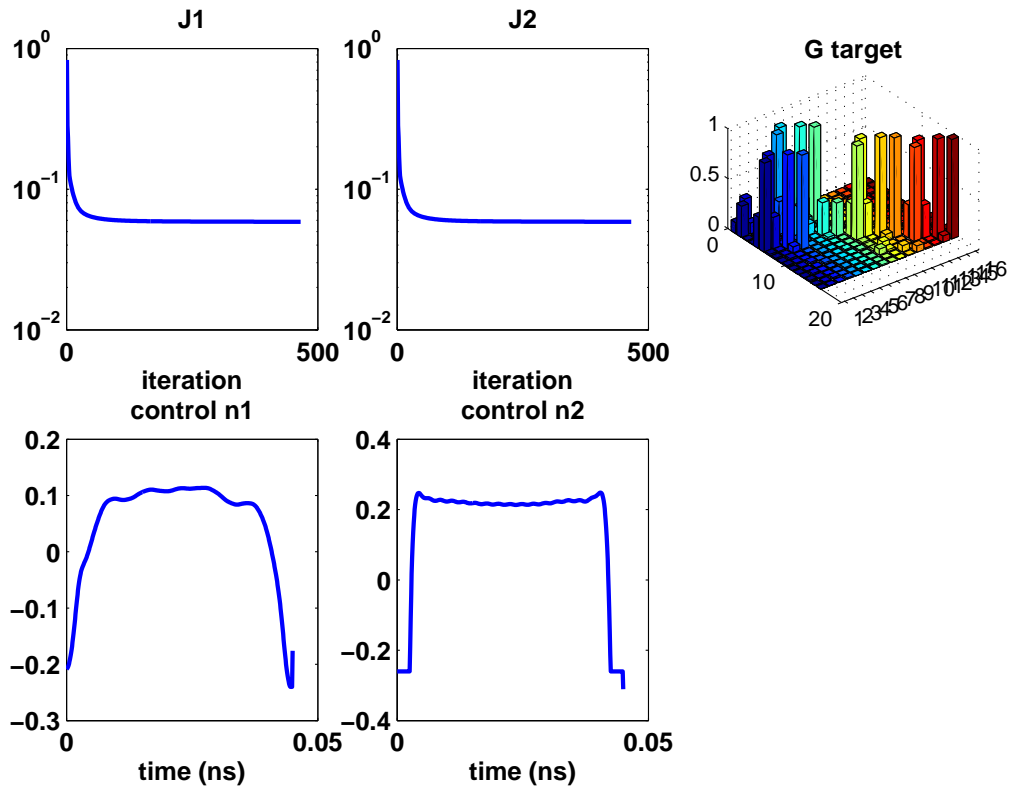
Figure 4.4: Result of the CNOT gate operation for a operation time $t_f = 45ps$. The first row shows two errors J_1 and J_2 versus iteration number, and a color bar matrix representing the elements of superoperator at the final operation time. The second row shows the control pulses of the final optimal results.
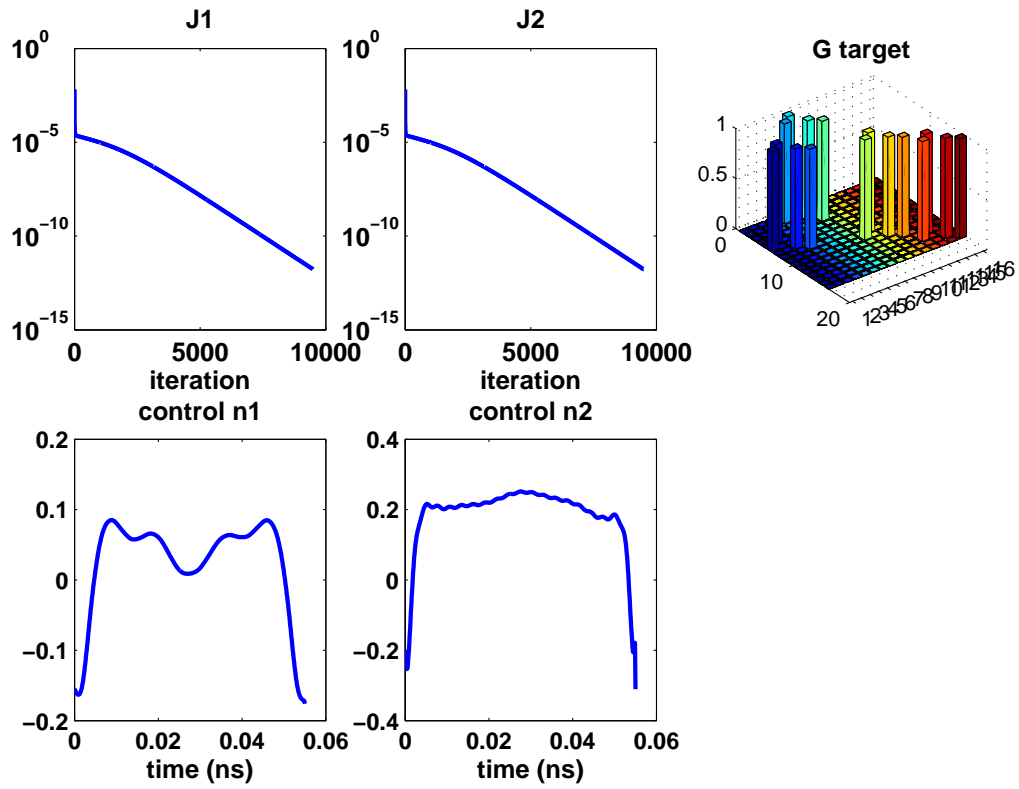
Figure 4.5: Result of the CNOT gate operation for a operation time $t_f = 55ps$. The first row shows two errors J_1 and J_2 versus iteration number, and a color bar matrix representing the elements of superoperator at the final operation time. The second row shows the control pulses of the final optimal results.
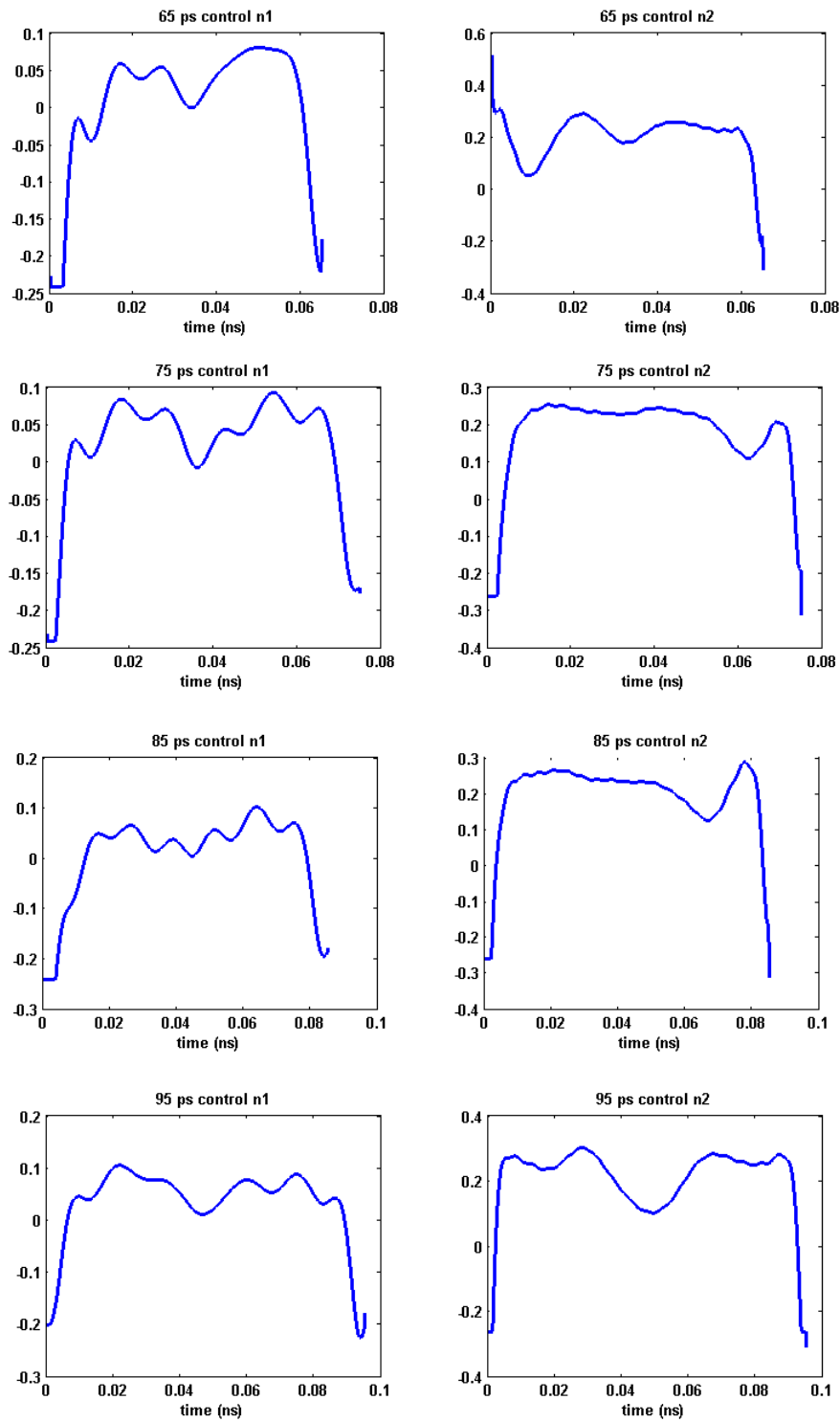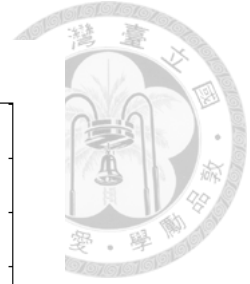
Figure 4.6: CNOT gate control pulse for different operation times in a closed system with no leakage states.
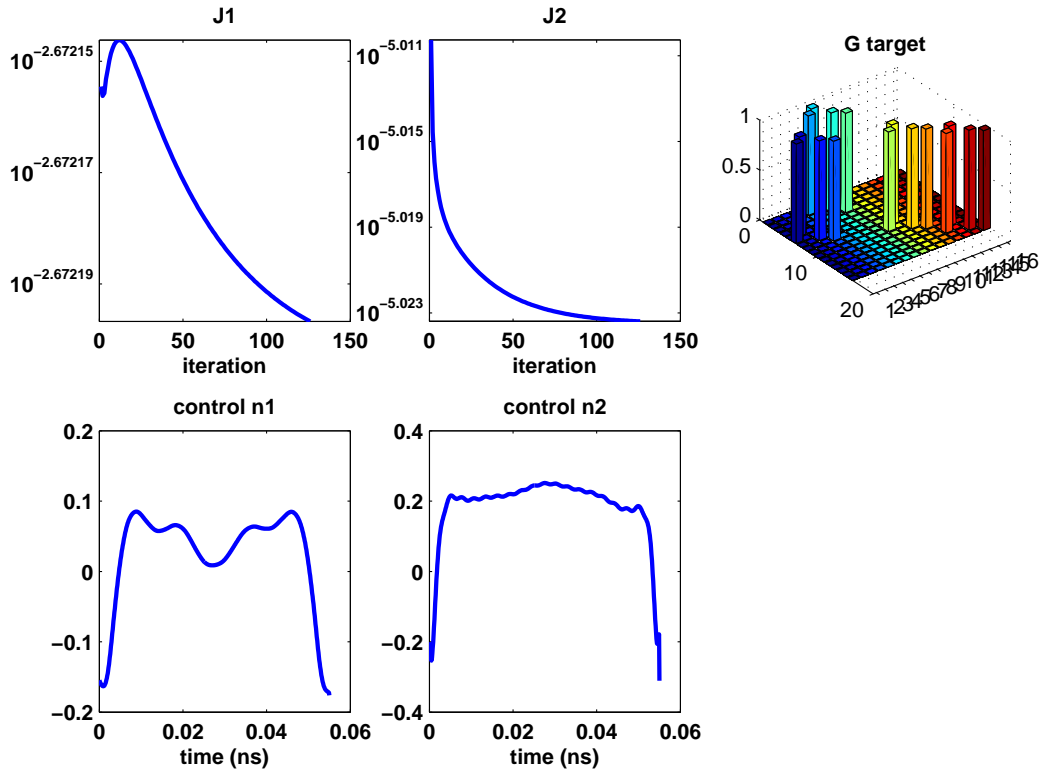
Figure 4.7: $10mK$ open system optimal control diagram. Notice that the J_1 does not monotonically converge since we use the algorithm which optimize J_2.
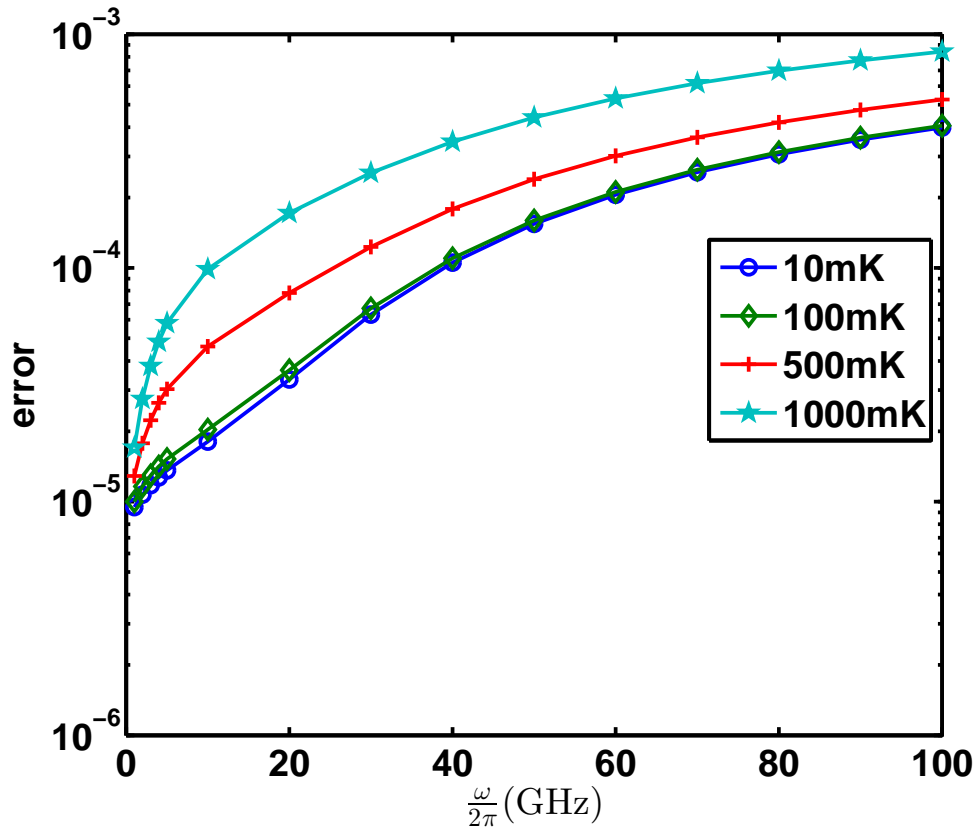
Figure 4.8: Operation CNOT gate error J_2 as a function of the cutoff frequency ω_c for different temperature.
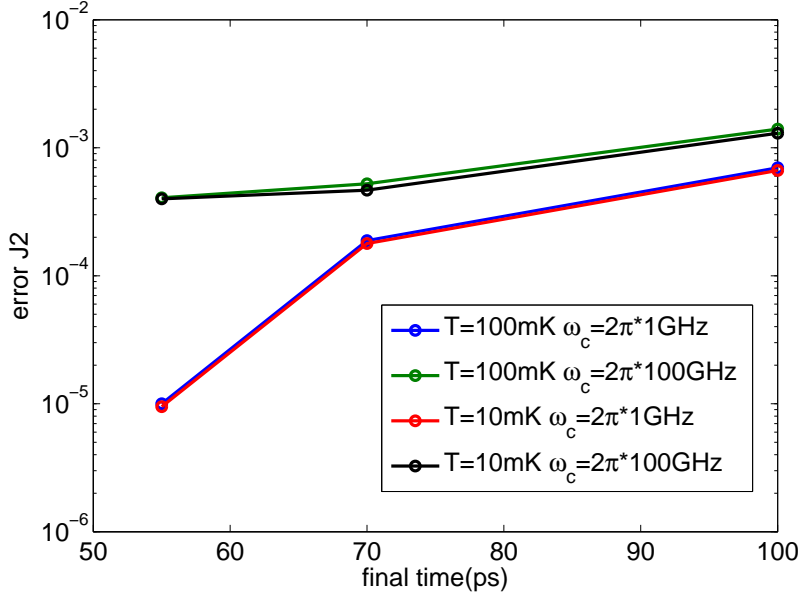
Figure 4.9: Open system CNOT gate error J_2 as a function of final time t_f for different temperatures and cutoff frequencies.

the advantage of the fast operation time.

Figure 4.9 shows the open system CNOT gate error J_2 versus gate operation time for two different cutoff frequencies and two different temperature. As the gate operation time gets larger, the error becomes bigger due to the larger decoherence effect. The gate error for small cutoff frequency of $\omega_c = 2\pi \times 1$ GHz is more sensitive to the length of gate operation time t_f But when $t_f > 70ps$, the dependence is similar to that of $\omega_c = 2\pi \times 100$ GHz.

One can see that the cutoff frequency has a big effect on the gate error when the gate operation time is $55ps$. However, for a given cutoff frequency, the gate error is almost unchanged for the two different temperatures of 10 mK and 100 mK.

4.4 Closed System with Leakage Levels

4.4.1 Overview

The behavior of the CNOT gate error as a function of gate operation time with leakage states shown in Fig.4.10 is similar to that in Fig. 4.2. However, the leakage states
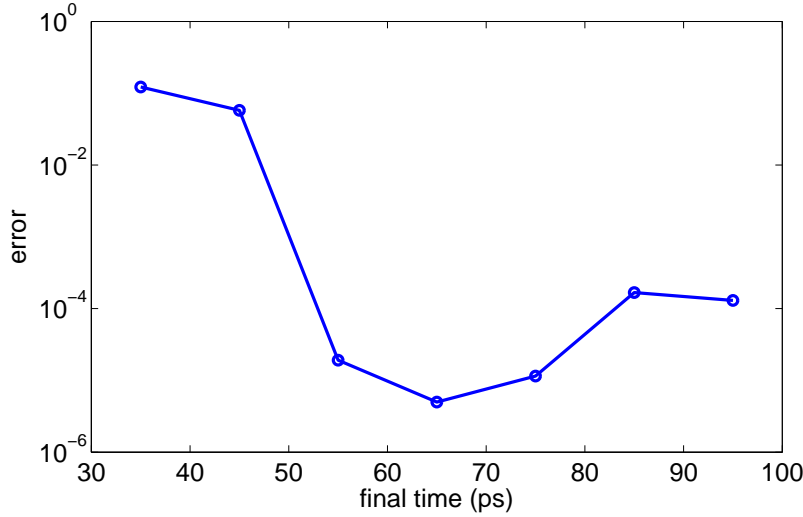
Figure 4.10: The closed system CNOT gate optimal diagram with leakage level.

iteration number	5000	10000	15000	20000
error	1.9137e-005	1.2016e-005	7.6831e-006	5.8017e-006

Table 4.1: Comparison of the iteration numbers and error. Since the error didn't massively decrease and the time is too much, we only choose 5000 iteration number to analyze in this section.

make the error bigger than Figure 4.2. This is fine since the errors are only around $10^{-4} \sim 10^{-5}$. And the important part of this is we perform a CNOT gate optimal control with leakage levels. So now we can compare our result with the previous papers which they claim they use leakage levels.

Since the leakage levels make the dimension of the superoperator much larger, the time we need to complete the optimal control calculation is massively increased. Table 4.1 shows the gate error J_2 at different iteration numbers. What we see is that when the error reach a small value , it does not decrease very much as the number of iteration increases. But the time we need increases significantly. Besides, when the final time of the optimal gate operation increases, the time needed becomes very expense. So the data used in Figure 4.10 is for the iteration number 5000. But Table 4.1 does show that the error can be lower when the iteration number increases.
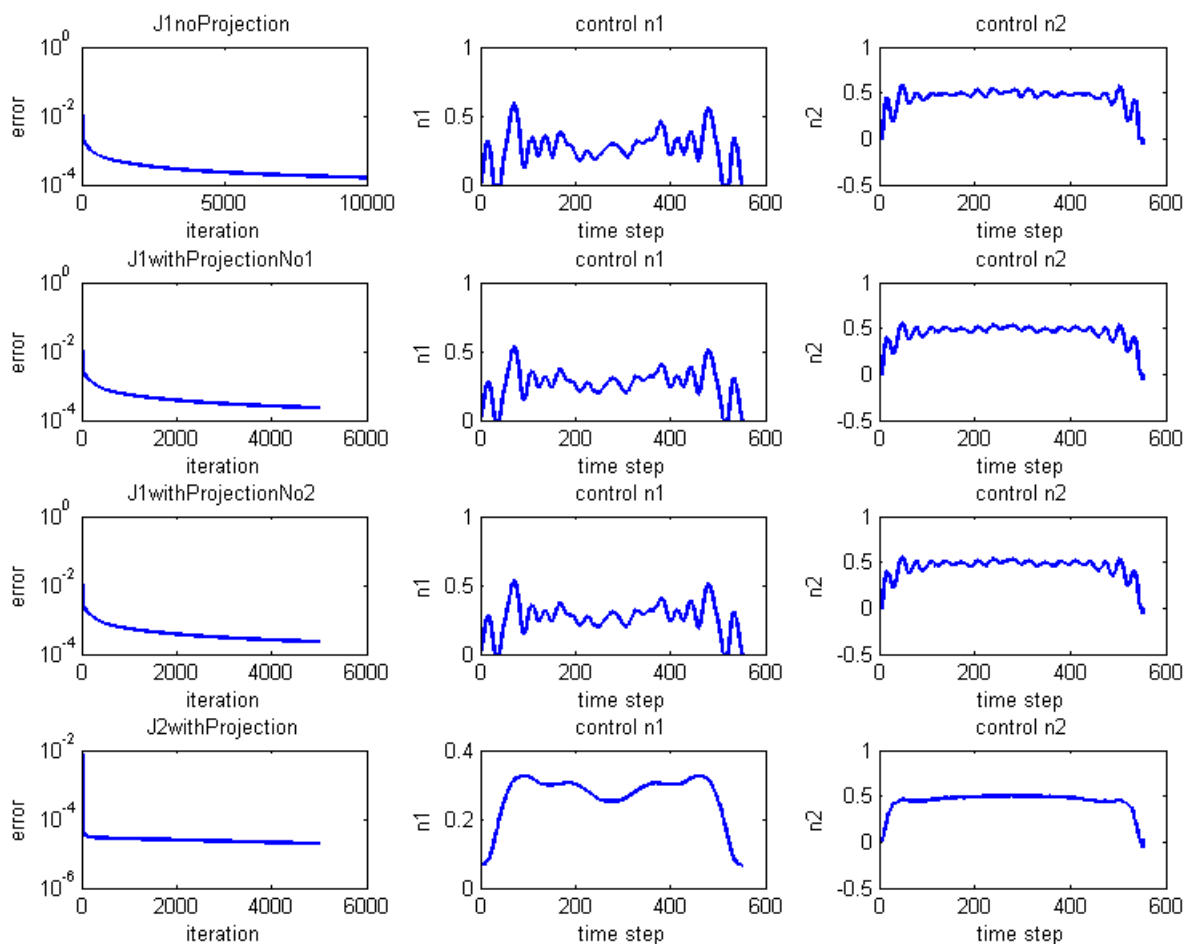
Figure 4.11: There are three columns and rows, they represent different optimal situation and their control pulse shape. The first row is the optimization of CNOT gate with leakage but the algorithm is made to optimize J_1. The second and the third rows are basically the same. The second is the algorithm is changed to adapt the leakage levels but still made for J_1, and the error is decided by pulling out the computational elements and reconstruct it into a small matrix and then compute error. The third one is the same but compute the error by using the big matrix with leakage levels only set the non-computational elements into zeros. The fourth one is made to optimize J_2 with leakage levels. The dimension of time step is 0.1ps.
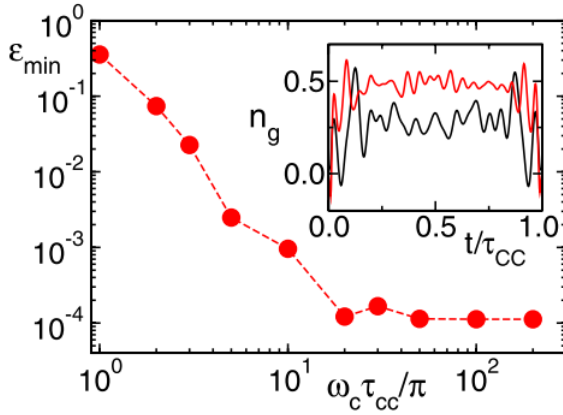
Figure 4.12: The error and the control pulse from [3]. The pulse shape is very similar to these obtained through the optimization of the error function J_1 in Fig. 4.11.

4.4.2 Pulse Shape

Figure 4.10 shows the control field pulses obtained by the optimization of different error functions.

The result of the first row is to optimize error function J_1 in the closed system with leakage states but without projecting J_1 onto the computational states. The second and the third are basically the same as they both optimize the error function J_1 with the second one making the projection onto the computational state subspace and the third one setting the non-computational basis subspace elements to zero. This lead to the question regarding which optimal control result is closer to the real target CNOT operation. The proper choice is to optimize J_2 with the propagator and target CNOT gate projected onto the computational subspace.

The reason to make the projection operation is that we do not care what the evolution of the leakage states is when we construct the two-qubit CNOT operation. The consequence of projecting the actual evolution operator (or propagator) at the final time will make the projected propagator no longer unitary. So the error functions J_1 and J_2 optimized with projected propagator are different. We choose J_2 as the distance between target gate and actual propagator takes the difference of every element in the matrix into account.

One can see from Fig. 4.11 that the control pulse shapes obtained by optimizing

J_1 are quite different from those obtained by optimizing J_2. Figure 4.12 shows the gate error and control pulses reproduced from [3]. The algorithm used in Ref.[3] is to optimized J_1, so the control pulses are very similar to our results for J_1 in Fig. 4.11. But as stated, optimization of J_2 is a better choice to define error when the actual propagator is not unitary.

4.5   Open System with Leakage Levels

Considering the open system effect by introducing the auxiliary density matrix make the dimension of the matrix of the propagator big. For example, fitting the bath correlation function with 1 exponential terms for the lowest cutoff frequency $\omega_c = 2\pi \times 1GHz$ has 1280×1280 superoperators, which is very difficult to deal with using matlab. The result is in Figure 4.14. And Figure 4.13 shows the optimal control pulses obtained in the closed system with leakage states. We evaluate the error by evolving the open system with the closed-system optimal control pulses. One iteration takes about 45 minutes. The error affect one iteration with the parameters used is small: $J_2 = 7.5266 \times 10^{-5}$. That is, the closed system optimal control are good enough to achieve a high-fidelity CNOT gates.

The gate errors for different temperature obtained by evolving the non-Markovian master equation with the optimal control pulses in closed system case are presented in Fig. 4.14. Since the optimal control calculation for two qubits with leakage states, interacting with a non-Markovian bath is quite computationally expensive. We choose to take the closed-system control pulses for the open system. The gate error obtained in one iteration will be taken as the error for the open system. The gate error obtained this way also same as a measure about how good the closed-system optimal control pulses are against the environment induced decoherence. We find that the error is only around $10^{-3} \sim 10^{-4}$ even at high cutoff frequency for the parameters used in our calculations. This indicates that the closed-system optimal control pulses are pretty robust against decoherence for the model we investigate.

In Figure 4.15, we insert the different error J_2 of closed system into $T = 10mK$
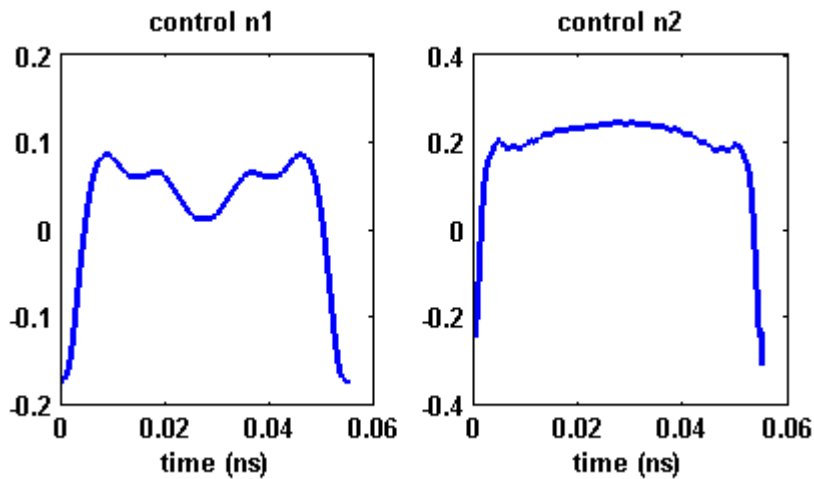
Figure 4.13: Optimized pulses obtain in a closed system with leakage states for $t_f = 55ps$, which are used to evolve the open system with leakage states.

open system. It shows that the error can be further improved if we have the extreme low error result in closed system. Since the dimension is so big and the iteration of closed system is slow enough, we only demonstrate the possibility of the improvement in open system.

Compare to the gate error for the optimal control case of the open system without leakage states shown in Fig. 4.8, the presence of the leakage states seems to prevent the gate error at low cutoff frequencies from decreasing. But keep in mind that we do not actually perform optimal control algorithm for the gate error shown in Fig. 4.14.
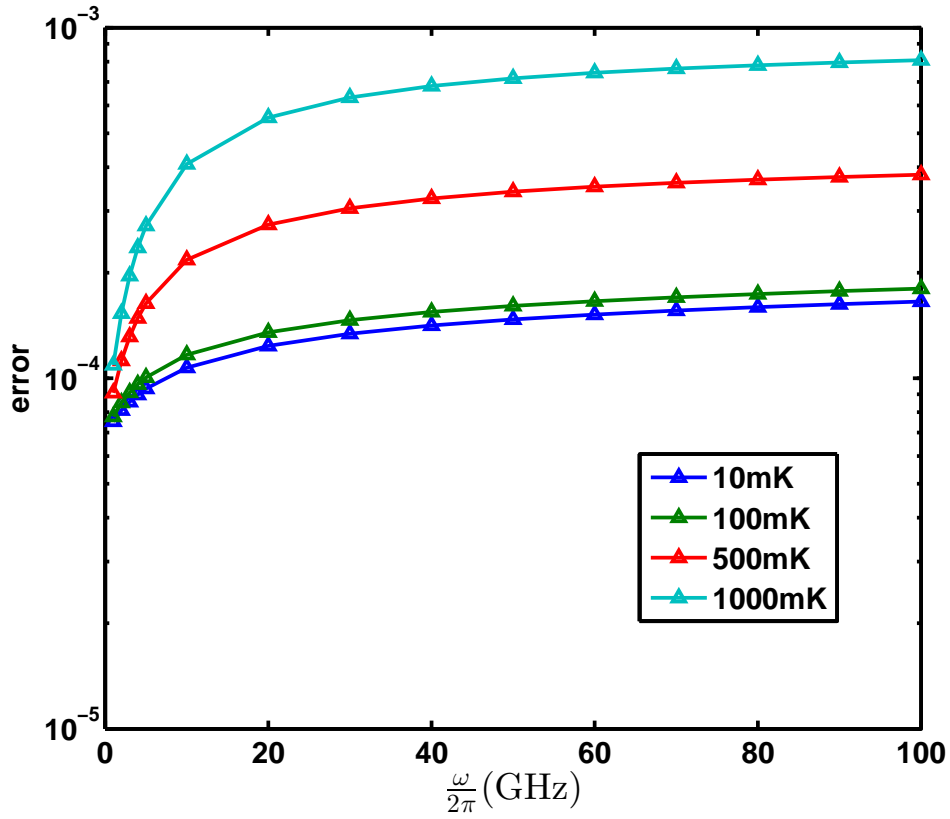
Figure 4.14: The error J_2 as a function of cutoff frequency ω_c for different temperatures at $t_f = 55ps$ for the open quantum system with leakage states. The gate error is obtained by inserting the optimal control pulses of the closed system into the non-Markovian master equation of the open system.
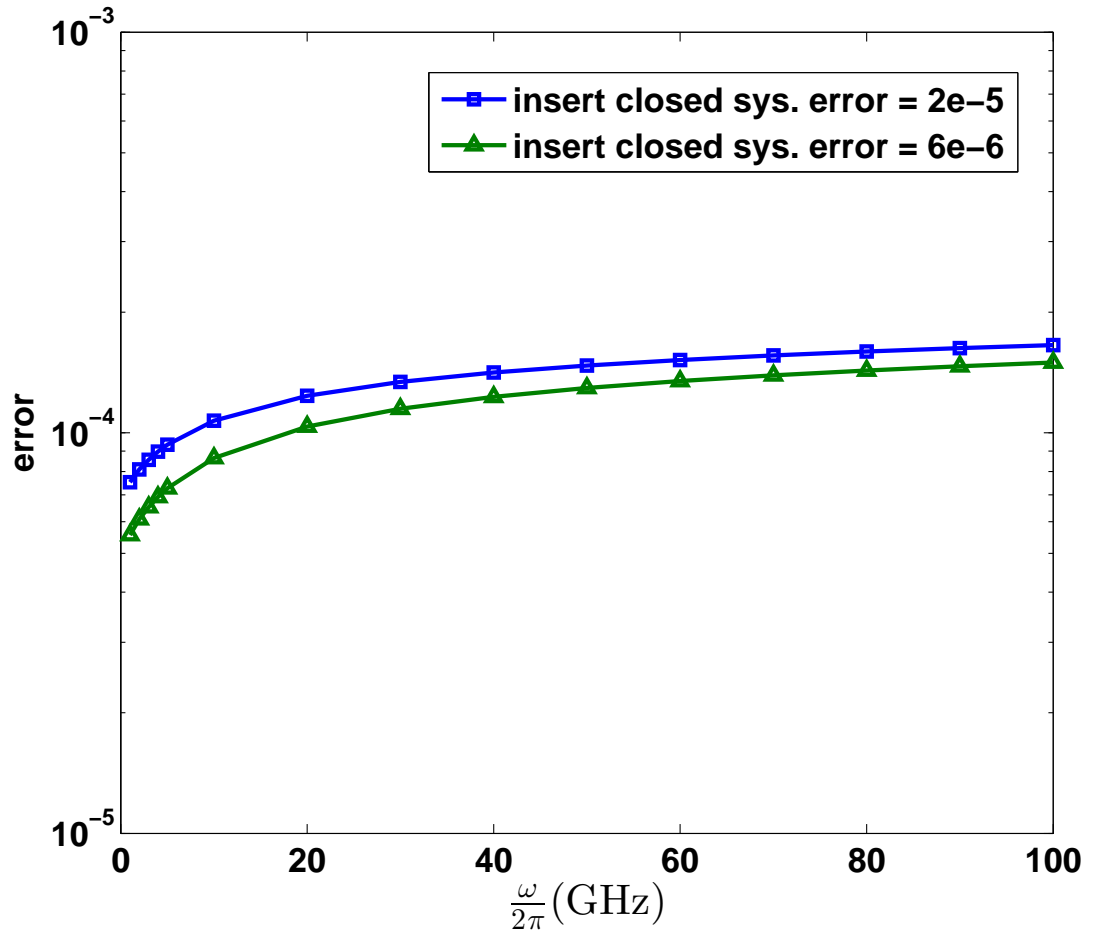
Figure 4.15: Demonstrate the effect of different insert errors into open system. In $10mK$ environment , the error J_2 is possible to get lower level in open system if we insert lower error pulse shapes of closed system. The blue line insert the pulse shapes of error $J_2 = 2 \times 10^{-5}$ of closed system into open system, and the green line insert the pulse shapes of error $J_2 = 6 \times 10^{-6}$ of closed system into open system.

Chapter 5

Conclusion

We give a short conclusion in this Chapter. In order to perform optimal control of gate operations in closed and open quantum system, we choose the Krotov optimal control method. Krotov optimal control method is effective when we are dealing with closed system with no leakage level. Using $J_2 = \frac{1}{2N} Tr[(\mathbf{Q} - \mathbf{G}(T))^\dagger (\mathbf{Q} - \mathbf{G}(T))]$ is a better way to define the error than $J_1 = 1 - \frac{1}{N} Tr(\mathbf{Q}^\dagger \mathbf{G}(T))$ since the closed system result will make $J_2 = J_1$ and in the open system result $\mathbf{G}(T)$ will not be unitary and the error function J_2 take the distance of every matrix element in the $\mathbf{Q} - \mathbf{G}(T)$ into account. However, the definition of J_2 will cause the derivation of Krotov optimal control not able to be monotonically convergence anymore. But since the Krotov method only guarantee the convergence of the cost function and we only need the convergence of the error. So the try and error of penalty become a practical issue during the simulation. Using optimal control to optimize J_2 in the closed system with no leakage states can give extreme low error, e.g. 10^{-10}. When we insert this control parameters (closed system control pulses) into non-Markovian open system and do the optimal control again, we find that the iteration just stops at very low numbers, e.g. 10 with error J_2 about $10^{-3} \sim 10^{-5}$, depending on the bath parameters used in the calculations. This shows that the optimal control is efficient and robust in the closed system. The optimal control of error J_2 in the closed system with leakage states is hard to reach low error (but possible if the iteration increases). With 5000 to 10000 iterations we could only reach error $J_2 \sim 10^{-5}$. With the same iteration

number, we could reach error of 10^{-10} for the closed system without leakage states. For the case of open system with leakage states, we simply use the optimal control pulses obtained in the closed system to evolve the open system to obtain the gate error as the optimal control calculation in this case is rather computationally expensive. The result of the gate error is around $10^{-3} \sim 10^{-4}$ for the bath parameters extracted from the experimental noise spectrum.

The reason why the improvement for the open system optimal control is not great may be the following. In our charge qubit model, it is assumed that we have control only on the σ_z terms or on the operators with diagonal matrix elements in the charge state basis in our Hamiltonian, Thus we may be able to correct the environment induced relaxation efficiently. As a result, the optimal control can not improve much the gate error for our open system model. If control over both σ_z and σ_x terms or over both diagonal and off-diagonal matrix element operators in the Hamiltonian were allowed, the optimal control theory might be able to do a much better job on gate error improvement.

Appendix A

RK4 and expm

In this appendix, we compare the Runge-Kutta method (RK4) with matrix exponential ($expm$) method of solving the differential equation of the propagator:

$$\frac{d\mathbf{G}(t)}{dt} = \mathbf{\Lambda}(t)\mathbf{G}(t). \tag{A.1}$$

This problem of solving the differential equation is hard when you want to directly solve it. However, we can use the assumption that the superoperator Hamiltonian can be treated as time independent in every small time step dt.

That is,

$$\mathbf{G}(t + dt) = e^{\mathbf{\Lambda}(t)dt}\mathbf{G}(t), \tag{A.2}$$

which is exact in every small time step.

This can lead us to the evolution of the propagator G

$$\mathbf{G}(t_f) = e^{\Lambda(t_f - dt)dt} \cdots \cdots e^{\Lambda(2dt)dt} e^{\Lambda(dt)dt} e^{\Lambda(0)dt}\mathbf{G}(0). \tag{A.3}$$

The definition of matrix exponential is

$$
\begin{aligned}
e^{\mathbf{A}} &= \mathbf{I} + \frac{\mathbf{A}}{1!} + \frac{\mathbf{A}^2}{2!} + \frac{\mathbf{A}^3}{3!} + \cdots\cdots \\
&= \sum_{i=0}^{\infty} \frac{\mathbf{A}^i}{i!}.
\end{aligned}
\tag{A.4}
$$

But the above definition cannot be directly turned into codes when the norm of A is bigger than 1. Since the Taylor series do not converge. One may use the Pade approximation to calculate the matrix exponential.

Another method to solve the differential equation is Runge-Kutta method (RK4).

For a differential equation $\frac{dy}{dt} = f(t, y)$ with the initial condition $y(0) = y_0$, y_{n+1} at the time $t_{n+1} = t_n + h$ can be written as

$$
y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),
\tag{A.5}
$$

where h is a small time step

$$
\begin{aligned}
k_1 &= f(t_n, y_n) \\
&= (\Lambda_d + n_{g1}(t_n)\Lambda_1 + n_{g2}(t_n)\Lambda_2)\mathbf{G}(t_n),
\end{aligned}
\tag{A.6}
$$

$$
\begin{aligned}
k_2 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) \\
&= (\Lambda_d + n_{g1}(t_n + \frac{h}{2})\Lambda_1 + n_{g2}(t_n + \frac{h}{2})\Lambda_2)(\mathbf{G}(t_n) + \frac{h}{2}k_1),
\end{aligned}
\tag{A.7}
$$

$$
\begin{aligned}
k_3 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2) \\
&= (\Lambda_d + n_{g1}(t_n + \frac{h}{2})\Lambda_1 + n_{g2}(t_n + \frac{h}{2})\Lambda_2)(\mathbf{G}(t_n) + \frac{h}{2}k_2),
\end{aligned}
\tag{A.8}
$$

$$
\begin{aligned}
k_4 &= f(t_n + h, y_n + hk_3) \\
&= (\Lambda_d + n_{g1}(t_n + h)\Lambda_1 + n_{g2}(t_n + h)\Lambda_2)(\mathbf{G}(t_n) + hk_3),
\end{aligned}
\tag{A.9}
$$

We have used $\frac{d\mathbf{G}}{dt} = \Lambda^{(k)}\mathbf{G}$ for $\frac{dy}{dt} = f(t, g)$ to obtain each last line of Eq.(A.6)~(A.9).

Since we already make our control be constant in each of the small time intervals. By comparing the *expm* of matlab and our own rk4. We figure out that every time step of the differential equation must solve separately. That is, choose $n_{g1}(t_n + \frac{h}{2})$ and $n_{g1}(t_n + h)$ both as $n_{g1}(t_n)$ and choose $n_{g2}(t_n + \frac{h}{2})$ and $n_{g2}(t_n + h)$ both as $n_{g2}(t_n)$. Which means that the control does not change at all in this time interval.

Remember that RK4 gives only approximated results. When the norm of $\mathbf{\Lambda}(t)dt$ is too big, using RK4 to approximate the matrix exponential result would fail. Why is this issue important? Consider the closed system where is no environment noise. Theoretically, the optimal control of a two-qubit CNOT gate should be perfect and only bounded by the computer precision limit. Somehow the error of RK4 may cause problem such as error re-bounce in early iteration or make the final error lower than it used to be when using *expm*.

The lowest final time we choose with good fidelity is55*ps*. We divide it into 551 time steps in the closed system problem. Other paper [6] choose only 55 time steps. In the first time step, the propagator elements only have error around 10^{-6} as compared to the answer of *expm*. However, the initial error could cause a slightly difference after all 551 points and make the propagator non-unitary. The propagator should be unitary under the condition of being in a closed system. And this causes the values of two definitions of the error functions different.

For example:

$$\Lambda(t) = \mathbf{A} + n_{g1}(t)\mathbf{B} + n_{g2}(t)\mathbf{C}, \tag{A.10}$$

$$\text{where} \mathbf{A} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{pmatrix} + i \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{pmatrix}$$

and $\mathbf{B} = \mathbf{C} = \begin{pmatrix} 1 & 2 & 0 & 0 \\ 5 & 6 & 0 & 0 \\ 0 & 0 & 11 & 12 \\ 0 & 0 & 15 & 16 \end{pmatrix} + i \begin{pmatrix} 1 & 2 & 0 & 0 \\ 5 & 6 & 0 & 0 \\ 0 & 0 & 11 & 12 \\ 0 & 0 & 15 & 16 \end{pmatrix}$ with $dt = 10^{-3}$. We may also tune the norm of the matrix and make $\mathbf{A} \to a\mathbf{A}$, $\mathbf{B} \to a\mathbf{B}$, $\mathbf{C} \to a\mathbf{C}$, and $\mathbf{\Lambda}(t) \to a\mathbf{\Lambda}(t)$, with $a = 1, 10, 100, 1000$.

That is

$$a\mathbf{\Lambda}(t) = a\mathbf{A} + n_{g1}(t)a\mathbf{B} + n_{g2}(t)a\mathbf{C}. \tag{A.11}$$

Compute the differential equation

$$\frac{d\mathbf{G}(t)}{dt} = a\mathbf{\Lambda}(t)\mathbf{G}(t) \tag{A.12}$$

and compare each element of the matrix at the final time for different values of "a". When the norm of the $a\mathbf{\Lambda}(t)$ is too big, e.g. $a = 100, 1000$, the RK4 method fails.

Appendix B

Derivatives of Matrix, Traces

The following matrix properties are extracted from [17]. These properties and techniques are necessary for our calculations.

B.1 Matrix Multiplication

A matrix multiplication is simple.

$$\mathbf{A}\mathbf{B} = \mathbf{C},$$

and the dimension of these real matrices are

$$\begin{cases} \mathbf{A} & (m \times n), \\ \mathbf{B} & (n \times p), \\ \mathbf{C} & (m \times p). \end{cases}$$

The elements of matrix $\mathbf{C}$ can be represented by those of the original two matrices:

$$C_{ij} = (\mathbf{A}\mathbf{B})_{ij} = \sum_{r=1}^{n} A_{ir} B_{rj}.$$

B.2 Derivatives of Matrix

We want to perform a derivative of a matrix. Remember that the derivative can always be reduced to the element derivative in matrix derivative. First, the basic assumptions of the matrix element derivative can be written in a formula as

$$\frac{\partial X_{kl}}{\partial X_{ij}} = \delta_{ik}\delta_{lj}.$$

B.3 Derivatives of Trace

$$\mathbf{C} = Tr(\mathbf{AB}),$$

$$C_{ij} = Tr(\mathbf{AB})_{ij} = \sum_{r=1}^{n} A_{ir}B_{ri},$$

$$D_{ir} = \frac{\partial \mathbf{C}}{\partial A_{ir}} = \frac{\partial Tr(\mathbf{AB})}{\partial A_{ir}} = \frac{\partial}{\partial A_{ir}}\left[\sum_{r=1}^{n} A_{ir}B_{ri}\right] = B_{ri}.$$

In a matrix form

$$\mathbf{D} = \frac{\partial \mathbf{C}}{\partial \mathbf{A}} = \frac{\partial Tr(\mathbf{AB})}{\partial \mathbf{A}} = \mathbf{B}^{T}.$$

This mathematical tool of matrix derivative is important because in optimal control problems, we always want to reach some maximum or minimum points and a density matrix is usually used. A lot of equations and constraints are also expressed by matrix operators. Since we want to find the maximum or the minimum point, the derivative of a matrix trace is very important.

B.3.1 Derivation of Trace REAL Matrix

Here we list some useful formula below. Assume that in this section the matrices are REAL.

B.3.1.1 First Order

$$\frac{\partial Tr(\mathbf{X})}{\partial \mathbf{X}} = \mathbf{I},$$

$$\frac{\partial Tr(\mathbf{XA})}{\partial \mathbf{X}} = \mathbf{A}^T,$$

$$\frac{\partial Tr(\mathbf{AXB})}{\partial \mathbf{X}} = \mathbf{A^T B^T},$$

$$\frac{\partial Tr(\mathbf{AX^T B})}{\partial \mathbf{X}} = \mathbf{BA},$$

$$\frac{\partial Tr(\mathbf{X^T A})}{\partial \mathbf{X}} = \mathbf{A},$$

$$\frac{\partial Tr(\mathbf{AX^T})}{\partial \mathbf{X}} = \mathbf{A}.$$

B.3.1.2 Second Order

$$\frac{\partial Tr(\mathbf{X}^2)}{\partial \mathbf{X}} = 2\mathbf{X}^T,$$

$$\frac{\partial Tr(\mathbf{X^2 B})}{\partial \mathbf{X}} = (\mathbf{XB} + \mathbf{BX})^T,$$

$$\frac{\partial Tr(\mathbf{X^T BX})}{\partial \mathbf{X}} = \mathbf{BX} + \mathbf{B^T X},$$

$$\frac{\partial Tr(\mathbf{XBX^T})}{\partial \mathbf{X}} = \mathbf{XB^T} + \mathbf{XB},$$

$$\frac{\partial Tr(\mathbf{AXBX})}{\partial \mathbf{X}} = \mathbf{A^T X^T B^T} + \mathbf{B^T X^T A^T},$$

$$\frac{\partial Tr(\mathbf{X^T X})}{\partial \mathbf{X}} = 2\mathbf{X}.$$

B.3.2 Derivation of Trace COMPLEX Matrix

The generalization to the case for complex matrices is straightforward. One can split a complex matrix into a real part and a imaginary part, and just carries out the calculation as in the case for real matrices. But putting them together into a formula requires to pay more attention on the combination formula.

B.3.2.1 Generalized Complex Derivative:

$$\frac{\partial Tr(\mathbf{AB})}{\partial \mathbf{B}} = \frac{1}{2} \left(\frac{\partial Tr(\mathbf{AB})}{\partial \mathbf{B_{real}}} - i\frac{\partial Tr(\mathbf{AB})}{\partial \mathbf{B_{imag}}} \right).$$

B.3.2.2 Useful Formula

$$\frac{\partial Tr(\mathbf{AX})}{\partial \mathbf{X}} = \frac{1}{2} \left(\frac{\partial Tr(\mathbf{AX})}{\partial \mathbf{X_{real}}} - i\frac{\partial Tr(\mathbf{AX})}{\partial \mathbf{X_{imag}}} \right) = \frac{1}{2} \left(\mathbf{A^T} + \mathbf{A^T} \right) = \mathbf{A^T},$$

$$\frac{\partial Tr(\mathbf{AX}^\dagger)}{\partial \mathbf{X}} = \frac{1}{2} \left(\frac{\partial Tr(\mathbf{AX}^\dagger)}{\partial \mathbf{X_{real}}} - i\frac{\partial Tr(\mathbf{AX}^\dagger)}{\partial \mathbf{X_{imag}}} \right) = \frac{1}{2} \left(\mathbf{A} - \mathbf{A} \right) = 0,$$

$$\frac{\partial Tr(\mathbf{AX}^\star)}{\partial \mathbf{X}} = \frac{1}{2} \left(\frac{\partial Tr(\mathbf{AX}^\star)}{\partial \mathbf{X_{real}}} - i\frac{\partial Tr(\mathbf{AX}^\star)}{\partial \mathbf{X_{imag}}} \right) = \frac{1}{2} \left(\mathbf{A^T} - \mathbf{A^T} \right) = 0.$$

Appendix C

Leakage Optimal Calculation

This is the basic calculation on the target function which is in the cost function that one wants to minimize. The derivative of the target function is directly related to the initial condition of the backward propagator. Surprisingly, when we consider the leakage effect, the result of the derivative is still very simple.

Before we start, some basic knowledge and notations are given as follows

Let

$$\mathbf{G}(T) = \mathbf{G_1} + i\mathbf{G_2},$$

The projection operator acting on $\mathbf{G}$ is written as

$$P[\mathbf{G}(T)] = \mathbf{G_{pro1}} + i\mathbf{G_{pro2}}.$$

We have

$$\frac{\partial f(\mathbf{G}(T))}{\partial \mathbf{G}(T)} = \frac{1}{2}\{\frac{\partial f(\mathbf{G}(T))}{\partial \mathbf{G_1}} - i\frac{\partial f(\mathbf{G}(T))}{\partial \mathbf{G_2}}\}$$

First, I present the calculation:

$$\mathbf{R} = \mathbf{Q} - P[\mathbf{G}(T)],$$

$$
\begin{aligned}
ReTr(\mathbf{R}^\dagger \mathbf{R}) &= ReTr(\{\mathbf{Q} - \mathbf{G_{pro}}\}^\dagger \{\mathbf{Q} - \mathbf{G_{pro}}\}) \\
&= ReTr(\mathbf{Q}^\dagger \mathbf{Q} - \mathbf{Q}^\dagger \mathbf{G_{pro}} - \mathbf{G_{pro}^\dagger} \mathbf{Q} + \mathbf{G_{pro}^\dagger} \mathbf{G_{pro}}) \\
&= ReTr\{\mathbf{Q}^\dagger \mathbf{Q} - \mathbf{Q}^\dagger (\mathbf{G_{pro1}} + i\mathbf{G_{pro2}}) \\
&\quad - [(\mathbf{G_{pro1}})^{\mathbf{T}} - i(\mathbf{G_{pro2}})^{\mathbf{T}}]\mathbf{Q} \\
&\quad + [(\mathbf{G_{pro1}})^{\mathbf{T}} - i(\mathbf{G_{pro2}})^{\mathbf{T}}](\mathbf{G_{pro1}} + i\mathbf{G_{pro2}})\} \\
&= ReTr\{\mathbf{Q}^\dagger \mathbf{Q} - \mathbf{Q}^\dagger \mathbf{G_{pro1}} - i\mathbf{G_{pro2}}\mathbf{Q}^\dagger - (\mathbf{G_{pro1}})^{\mathbf{T}}\mathbf{Q} \\
&\quad + i(\mathbf{G_{pro2}})^{\mathbf{T}}\mathbf{Q} + (\mathbf{G_{pro1}})^{\mathbf{T}}\mathbf{G_{pro1}} \\
&\quad + i(\mathbf{G_{pro1}})^{\mathbf{T}}\mathbf{G_{pro2}} - i(\mathbf{G_{pro2}})^{\mathbf{T}}\mathbf{G_{pro1}} + (\mathbf{G_{pro2}})^{\mathbf{T}}\mathbf{G_{pro2}}\}.
\end{aligned}
$$

The derivative with respect to the real part $\mathbf{G_1}$:

$$
\begin{aligned}
\frac{\partial}{\partial \mathbf{G_1}} ReTr(\{\mathbf{Q} - P[\mathbf{G}(T)]\}^\dagger \{\mathbf{Q} - P[\mathbf{G}(T)]\}) &= Re\{0 - P[\mathbf{Q^\star}] - 0 - P[\mathbf{Q}] \\
&\quad + 0 + P[2\mathbf{G_1}] + iP[\mathbf{G_2}] \\
&\quad - iP[\mathbf{G_2}] + 0\} = \alpha.
\end{aligned}
$$

The derivative with respect to the imaginary part $\mathbf{G_2}$:

$$
\begin{aligned}
\frac{\partial}{\partial \mathbf{G_2}} ReTr(\{\mathbf{Q} - P[\mathbf{G}(T)]\}^\dagger \{\mathbf{Q} - P[\mathbf{G}(T)]\}) &= Re\{0 - 0 - iP[\mathbf{Q^\star}] - 0 \\
&\quad + iP[\mathbf{Q}] + 0 + iP[\mathbf{G_1}] \\
&\quad - iP[\mathbf{G_1}] + P[2\mathbf{G_2}]\} = \beta.
\end{aligned}
$$

Combining them together, we obtain the result

$$
\begin{aligned}
\frac{\partial}{\partial \mathbf{G}(T)} ReTr(\{\mathbf{Q} - P[\mathbf{G}(T)]\}^\dagger \{\mathbf{Q} - P[\mathbf{G}(T)]\}) &= \frac{1}{2}(\alpha - i\beta) \\
&= \frac{1}{2}\{P[(-\mathbf{Q^\star} - \mathbf{Q}) - i(-i\mathbf{Q^\star} + i\mathbf{Q})] \\
&\quad + P[(2\mathbf{G_1}) - i(2\mathbf{G_2})]\} \\
&= \frac{1}{2}\{P[-2\mathbf{Q^\star} + 2\mathbf{G^\star}(T)]\} \\
&= P[-\mathbf{Q^\star} + \mathbf{G^\star}(T)]
\end{aligned}
$$

One may notice that if we remove the projection operator, the result reduces to

70

the original one. The reason is the following. The derivative of the matrix with or without the projection operator doesn't change the derivative result much.

$$
\begin{aligned}
M_{ij} &= \frac{\partial Tr(P[\mathbf{A}]\mathbf{B})}{\partial A_{ij}} = \frac{\partial Tr(\mathbf{A_{pro}}\mathbf{B})}{\partial A_{ij}} \\
&= \frac{\partial}{\partial A_{ij}} \sum_i \sum_j A_{pro,ij} B_{ji} = \begin{cases} B_{ji} & \text{computational,} \\ 0 & \text{otherwise.} \end{cases}
\end{aligned}
$$
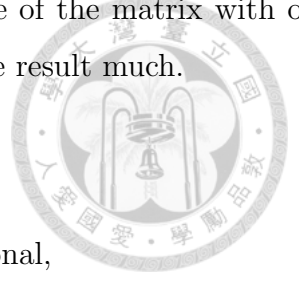
In matrix form:

$$
\mathbf{M} = \frac{\partial Tr(P[\mathbf{A}]\mathbf{B})}{\partial \mathbf{A}} = \frac{\partial Tr(\mathbf{A_{pro}}\mathbf{B})}{\partial \mathbf{A}} = P[\mathbf{B^T}].
$$

$$
\begin{aligned}
N_{ji} &= \frac{\partial Tr(P[\mathbf{A}]^{\mathbf{T}}\mathbf{B})}{\partial A_{ji}} = \frac{\partial Tr((\mathbf{A_{pro}})^{\mathbf{T}}\mathbf{B})}{\partial A_{ji}} \\
&= \frac{\partial}{\partial A_{ij}} \sum_i \sum_j A_{pro,ji} B_{ji} = \begin{cases} B_{ji} & \text{computational,} \\ 0 & \text{otherwise.} \end{cases}
\end{aligned}
$$

In matrix form:

$$
\mathbf{N} = \frac{\partial Tr(P[\mathbf{A}]^{\mathbf{T}}\mathbf{B})}{\partial \mathbf{A}} = \frac{\partial Tr((\mathbf{A_{pro}})^{\mathbf{T}}\mathbf{B})}{\partial \mathbf{A}} = P[\mathbf{B}].
$$

$$
\begin{aligned}
O_{ji} &= \frac{\partial Tr(P[\mathbf{A}]^{T} P[\mathbf{A}])}{\partial A_{ji}} = \frac{\partial Tr((\mathbf{A_{pro}})^{\mathbf{T}}\mathbf{A_{pro}})}{\partial A_{ji}} \\
&= \frac{\partial}{\partial A_{ij}} \left(\sum_i^{i \neq j} \sum_j^{i \neq j} A_{pro,ji} A_{pro,ji} + \sum_u^{i=j=u} A_{pro,uu} A_{pro,uu} \right) \\
&= \begin{cases} 2A_{uu} & i = j = u \text{(computational)}, \\ 2A_{ji} & i \neq j \text{(computational)}, \\ 0 & i = j = u \text{(non-computational)}, \\ 0 & i \neq j \text{(non-computational)}. \end{cases}
\end{aligned}
$$

In matrix form:

$$\mathbf{O} = \frac{\partial Tr(P[\mathbf{A}]^{\mathbf{T}} P[\mathbf{A}])}{\partial \mathbf{A}} = \frac{\partial Tr((\mathbf{A_{pro}})^{\mathbf{T}} \mathbf{A_{pro}})}{\partial \mathbf{A}} = P[2\mathbf{A}].$$
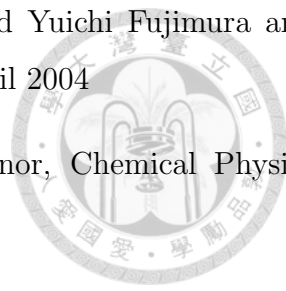
Bibliography

[1] Yuriy Makhlin and Gerd Schon and Alexander Shnirman, Rev. of Mod. Phys. Vol. 73, No. 2 (2001).

[2] Robert Roloff and Walter Potz, PRB, 79, 224516(2009)

[3] Simone Montangero and Tommaso Calarco and Rosario Fazio, PRL, 99, 170501(2007)

[4] Yu.A.Pashkin et al, Nature(London), Vol. 421, p.823(2003)

[5] T.Yamamoto and Yu.A.Pashkin and O.Astafiev and Y.Nakamura and J.S. Tsai, Nature, Vol. 425, p.941(2003)

[6] A. Sporl and T. Schulte-Herbruggen and S. J. Glaser and V. Bergholm and M. J. Storcz adn J. Ferber and F. K. Wilhelm, PRA, 75, 012302(2007)

[7] Yuriy Makhlin and Gerd Schon and Alexander Shnirman,Chemical Phys. 296(2004) 315-324

[8] O.Astafiev and Yu.A.Pashkin and Y.Nakamura and T.Yamanoto and J.S.Tsai, PRL, 93, 267007(2004)

[9] Jose P. Palao and Ronnie Kosloff, PRA, 68, 062308(2003)

[10] R. Roloff and M. Wenin and W. Potz, arXiv:0910.0362v1 [quant-ph] 2 Oct 2009

[11] Jose P. Palao and Ronnie Kosloff, PRL, 89, 188301(2002)

[12] G.-Q. Li and U. Kleinekath¨ofer, Eur. Phys. J. B 76 , 309–319 (2010)

[13] Ruixue Xu and YiJing Yan and Yukiyoshi Ohtsuki and Yuichi Fujimura and Herschel Rabitz J. Chem. Phys., Vol. 120, No. 14, 8 April 2004

[14] Allon Bartana and Ronnie Kosloff and David J. Tannor, Chemical Physics 267(2001) 195-207

[15] Matthew Grace, Constantin Brif, Herschel Rabitz, Ian A Walmsley, Robert L Kosut and Daniel A Lidar, J. Phys. B: At. Mol. Opt. Phys. 40 (2007) S103–S125

[16] Yu. A. Pashkin and O. Astafiev and T. Yamamoto and Y. Nakamura and J. S. Tsai, Quantum Inf Process (2009) 8:55–80

[17] Kaare Brandt Petersen and Michael Syskind Pedersen, The Matrix Cookbook, version 12/14/2008

[18] Andrei Tokmakoffm, MIT Department of Chemistry, 3/12/2009, Quantum Relaxation Ch10

[19] A. J. Leggett and S. Chakravarty and A.T. Dorsey and Matthew P. A. Fisher and Anupam Garg and W. Zwerger, Rev. Mod. Phys. Vol. 59, No1(1987)

[20] Leggett, A.J., 1984b, PRB, 30, 1208

[21] A.O Caldeira and A.J Leggett, Ann. Phys, 149, 374

[22] M. Wenin and W. Pötz, PRB 78, 165118 (2008)

[23] H.J. Carmichael, Statistical Methods in Quantum Optics 1

[24] Bin Hwang and Hsi-Sheng Goan , PRA 85, 032321 (2012)