



The Hebrew University of Jerusalem
Faculty of Science
Racah Institute of Physics

Hybrid Logical-Physical Qubit Interaction for Quantum Metrology

אינטראקציה היברידית לוגית-פיזית עבור
חישה קוונטית

Nadav Carmel

Thesis submitted in partial fulfillment of the requirements
for the Master of Sciences degree
in Physics

Under the supervision of **Prof. Nadav Katz**

June 2023



האוניברסיטה העברית בירושלים
הפקולטה למתמטיקה ולמדעי הטבע
מכון רקח

Hybrid Logical-Physical Qubit Interaction for Quantum Metrology

אינטראקציה היברידית לוגית-פיזית עבור
חישה קוונטית

מוגש על ידי
נדב כרמל

עבודת גמר לתואר מוסמך בפיזיקה

עבודה זו הונחתה על ידי
פרופ' נדב כץ

אייר, ה'תשפ"ג

תקציר

השימוש במערכות קוונטיות עבור חישוב קוונטי וחישה קוונטית הוא תחום מחקר יחסית חדש. מדידת אופרטורים היא התכלית של תיאוריה שנקראת חישה קוונטית, שהוקрат את היכולת שלנו להשתמש במערכות קוונטיות למטרות חישה מדוקפת.

מחשב קוונטי מורכב מביטים קוונטיים (קיובייטים), מערכות בעלות שתי רמות אנרגיה שניתן לשאור ולתפעל לפי רצוניינו ולכך מאפשרות את ההרצה של אלגוריתמים מורכבים. משפהה של אלגוריתמים שימושים אותן למטרות חישה קוונטיות נקבעת הערכת פאזה קוונטית, והיא מאפשרת את המדידה של ערך עצמי של אופרטור בהינתן המצב העצמי המתאים.

בעזרת גרסאות איטרטיביות של הערכת פאזה קוונטית, ניתן למדוד בדיק ארביטרاري את הערך העצמי המדובר, לפחות בתיאוריה. אך בעולם האמתי ישנן שגיאות (מדידה, שגיאות בשערים, או שגיאות שנובעות מחשיאה של הקיובייטים לעולם החיצון) שמכבילות את היכולת שלנו למדוד ערכים בדיק גבוה. תיקון שגיאות קוונטי הוא תחום של מחשוב קוונטי (תחום מדעי המחשב) שספק שיטות כלליות להתגבר על שגיאות ורעים, ממש כמו תיקון שגיאות במחשבים קלאסיים. משפט הסוף עבור חישוב קוונטי מצין שבבhinתן שהרעיון בשערים קוונטיים הוא מתחתי לשף מסוים, ניתן לבצע ביעילות חישובים גדולים כרצונו במחשב קוונטי בעזרת שיטות חישוב עמידות לשגיאות. הרעיון של שיטות חישוב עמידות אלה הוא לעצב שערים בצורה

כזו שומרת את המצב הקונטני בקוד, וגם מבטיחה שם קרתה לכל היותר שגיאה אחת בהפעלת שער, אז יהיה לכל היותר קיוביט אחד שיש בו שגיאה בסוף החישוב, אפילו אם השער יוצר שגיאה.

בחירה-בדיעבד היא שיטה נוספת המשמשת להתרמודדות עם שגיאות בחישוב קוונטי, שבה אני בוחרים בדיעבד את המידע שמתאים לציפויו שלנו.

במחקר זה אנחנו מציגים רעיונות חדשים שימושיים את התהומות של חישה קוונטיות, תיקון שגיאות קוונטי ובחירה-בדיעבד, בנסיבות רעש חיצוני, ע"י שימוש בקוד חמוץ הקיובייטים, שהוא הקוד הקטן ביותר שמאפשר תיקון של כל סוג שגיאות:

- אנחנו מציגים אלגוריתם הערכת פאזה איטרטיבי חדש, שמאפשר מדידה של מספר ספרות בכל איטרציה. האלגוריתם הזה הוא למעשה מעשה הכללה של אלגוריתם קיים.
- אנחנו חוקרים ומסבירים איך בבחירה-בדיעבד על החישון יכולה להועיל למטרות חישה קוונטיות באlgorigitmic הערכת פאזה.
- אנחנו מגדירים את הקונספט שלבחירה-בדיעבד לוגית, ומספקים פיתוח תיאורטי שմסביר את התפקיד שלו נגד שגיאות כלליות.
- אנחנו מגדירים תכוונה של קודים קוונטיים לתקן שגיאות שמאפשרת את האינטראקציה בין קיובייטים משכבות לוגיות שונות, ומוסאים את הנسبות הכלליות שבנהן אינטראקציה כזו נותנת יתרון.
- לבסוף, אנחנו מדגימים את השימוש בבחירה-בדיעבד לוגית עבור שני אלגוריתמים של הערכת פאזה קוונטי, ומוסאים את הסף הרלוונטי שבו השימוש בשיטה הנ"ל משפר את התוצאות.

Abstract

The use of quantum systems for quantum computing and quantum sensing is a relatively new area of research, both theoretically and experimentally. Measuring an operator is the purpose of a theory called Quantum Sensing, exploring our ability of using quantum systems for accurate sensing.

A quantum computer is made of quantum bits (qubits), two-level quantum systems that can be controlled and entangled at will, and thus can perform algorithms. One family of algorithms used for quantum sensing is called Quantum Phase Estimation (QPE), enabling the measurement of an eigenvalue of an operator, given it's corresponding eigenstate.

The eigenvalue can, in theory, be measured to arbitrary accuracy using a small number of qubits, with iterative versions of QPE. But in the real world, there are errors (measurement errors, gate errors, qubit decoherence...) limiting our ability of precise measurement. Quantum Error Correction (QEC) is a field of quantum computing (computer science) that gives general methods for overcoming errors and noise, much like the classical field of error correction in classical computers. The threshold theorem for quantum computations states that provided the noise in individual quantum gates is below a certain constant threshold it is possible to

efficiently perform an arbitrarily large quantum computation using methods of fault tolerance quantum computation. The idea of fault-tolerant quantum computation is to design the gates acted on the quantum state, in a way that keeps the state in the code and assures that if one error occurred in the process of applying the gate, then we have at most one faulty qubit in each code block at the end, even if the gate creates entanglement.

Post selection is another technique used for coping with errors, in which we choose to work only with the portion of the data that fits our expectations.

In this work we present some new ideas in the field of quantum sensing, quantum error correction and post selection, in the presence of T1,T2 qubit relaxation, using the five-qubit code (the smallest error correcting code):

- We present a new QPE algorithm, which is a generalization of IPEA, enabling the measurement of more than one digit in each iteration.
- We explore and explain how Sensor Post Selection (SPS) may be effective for quantum sensing with QPE algorithms.
- We define the notion of Logical Post Selection (LPS) and give theoretical derivation of it's advantage against the depolarizing channel.
- We define a property of quantum codes enabling an interaction between qubits of different logical layers, and find under which circumstances it gives an advantage.
- Finally, we demonstrate the use of logical ancilla and physical sensor qubits for QPE and find the relevant thresholds for quantum sensing.

Acknowledgements

This work has been a most meaningful experience for me in the last year - a journey of self-revelation, development as a researcher and as a person. I owe the success of that journey to quite a few people, primarily of course to my advisor, Prof. Nadav Katz. Growing up, asked who is a role-model for me, I always had a hard time finding the right people to answer that definition. Over the years I have gathered a small collection of individuals who can be role models for me, and Nadav is definitely one of them. In my initial acquaintance with Nadav, I was amazed at how much a person can be professional in his field. Nadav is a physicist with very broad knowledge, sharp intuition and plenty of interesting ideas. In time I got to know him better, and I saw a man who was not only full of passion for research, but also for practical work - his investment in the development of the quantum field at the Hebrew University and in Israel has served as an example for a person who cares, on a large scale. Most of all I enjoyed the conversations with Nadav at the meta level, as he instilled in me motivation and explained to me the true meaning of being a researcher. In addition to being a professional, caring and passionate person, he is also an excellent teacher, always looking for opportunities to enrich his students and broaden their knowledge and understanding. I thank him

mainly for actively developing my research abilities, which will definitely serve me in the future.

I also wish to thank Prof. Michael Ben-Or for his help in understanding the field of quantum error correction, and to Dr. Tuvia Gefen for two fruitful discussions on quantum sensing.

I'd like to thank my friends from Talpiot, helping me to continue through a lot of struggles and reminding me about passion and hard work.

My commanding officers from the program over the years had a large influence on me, pushing me towards being a researcher. I'd especially like to thank two of them, to Omer Deutsch who fought to give me the appropriate conditions to carry out my research and to Daniel Yaron, my personal commanding officer over the last year, who was very understanding and cared a lot of my improvement as a researcher and as a person.

Contents

1	Introduction	1
1.1	Wait, But What Is A Qubit?	1
1.1.1	The Bloch Sphere	2
1.1.2	Quamtum Gates	3
1.1.3	Physical Implementations	6
1.2	Quantum Noise	8
1.2.1	Shot Noise	8
1.2.2	Decoherence	9
1.3	Distance Measures for Quantum Information	12
1.4	Quamtum Sensing	14
1.4.1	The Quantum Sensing Protocol	15
1.4.2	Fisher Information & Quantum Cramer Rao Bound .	19
1.4.3	Algorithmic Quantum Sensing	22
1.5	Quantum Error Correction	31
1.5.1	Stabilizer Formalism	31

1.5.2	Fault-Tolerant Quantum Computation	33
1.5.3	Quantum Error Correction for Quantum Metrology .	36
2	Research Question	37
3	Methods & Theoretical Derivations	39
3.1	n-Iterative Phase Estimation Algorithm	39
3.2	Iterative Phase Estimation Algorithm Under the Presence of Decoherence	40
3.3	The Simulation	43
3.3.1	Gate-Based Evolution	46
3.3.2	Kraus-Based Decoherence	46
3.3.3	Measurements	48
3.4	Sensor Post Selection	50
3.4.1	The Potential of Sensor Post Selection	50
3.4.2	SPS Simulation Methods	51
3.5	Logical Post Selection	53
3.5.1	Scaling of the Error Probability	54
3.5.2	Hybrid Logical-Physical Entanglement	58
3.6	Logical Post Selection for Quantum Metrology	61
3.6.1	Gate Implementations	62
3.6.2	LPS Simulation Methods	69
3.7	Measurement Errors	75

<i>CONTENTS</i>	ix
4 Results	79
4.1 Sanity Checks for the Simulation	81
4.1.1 T_1 & T_2 Extractions	82
4.1.2 Ramsey Experiment	83
4.1.3 Convergence Tests	85
4.1.4 From Decoherence to Gate Fidelity	86
4.2 Sensor Post Selection for Quantum Sensing	88
4.2.1 Target: Accuracy	88
4.2.2 Resource: Minimal Number of Trials	89
4.3 Logical-Physical Interaction - General Numeric Exploration	92
4.4 Logical Post Selection for Quantum Metrology	97
4.4.1 Gate Selection	97
4.4.2 Kitaev's Approach - with Accelerated Hamiltonians .	101
4.4.3 IPEA - without Accelerated Hamiltonians	107
5 Conclusion & Outlook	117
A Code Guide	119
A.1 About	119
A.2 Installation	119
A.2.1 required packages	119
A.2.2 Download	120
A.3 The Simulators	120

A.3.1	Creating a Quantum Register	121
A.3.2	Simulating a Quantum Circuit	124
A.3.3	Visualizing The State History of a Qubit	125

List of Figures

1.1	Bloch Sphere Representation of a Qubit	2
1.2	Basic Gates on Bloch Sphere	4
1.3	Frequently Used Gates	5
1.4	Models of Closed and Open Quantum Systems	9
1.5	Basic Steps of Quantum Sensing	17
1.6	Standard Quantum Phase Estimation	23
1.7	AQFT Implementation	26
1.8	Kitaev's Iterative Quantum Phase Estimation	27
1.9	Iterative Phase Estimation Algorithm	28
1.10	Histogram of IPEA results in the presence of noise	30
1.11	CZ Gadget	35
3.1	n-Iterative Phase Estimation Algorithm	39
3.2	Histogram of IPEA results for different number of trials	42
3.3	General Circuit Simulation	44
3.4	SPS Simulations - Circuit-Level	52

3.5	Logical-Physical Entangling Gate for the 5-Qubit Code	59
3.6	Logical State Preparation	63
3.7	Logical Hadamard	64
3.8	Logically Controlled Rotation	64
3.9	Logical-Physical Entangling Gate for Quantum Sensing	65
3.10	Logical-Physical Entangling Gate for Quantum Sensing - Fault-Tolerant Implementation	65
3.11	Logical $R_z(\theta)$ Rotation	66
3.12	Syndrome Extraction Circuits	68
3.13	IPEA and Kitaev QPE	70
3.14	Decision Tree for Post Selection	76
4.1	Decision Tree for Accelerated Hamiltonian	80
4.2	Decision Tree for non-Accelerated Hamiltonian	81
4.3	Verifying Decoherence	83
4.4	Ramsey with Dephasing	84
4.5	Convergence Tests	85
4.6	Map from T_1, T_2 to gate fidelity	87
4.7	SPS for Accuracy	89
4.8	SPS for Minimal Number of Trials	91
4.9	Logical-Physical Entanglement with Constant Gate Fidelity	94
4.10	Logical-Physical Entanglement with Constant Circuit Depth	95
4.11	Logical-Physical Entanglement	96

4.12 Logical $R_z(\theta)$ Rotation - Results	98
4.13 Gate Selection for State Preparation	100
4.14 LPS for Accuracy	102
4.15 Results of Kitaev Fidelity in Linear and Log Scales	104
4.16 LPS for Minimal Number of Trials	105
4.17 Confirming Error Probability Scaling	107
4.18 IPEA - Good Sensor & Noisy Ancilla	110
4.19 IPEA - Dephasing Sensor $ +\rangle$ & Good Ancilla	110
4.20 IPEA - Relaxing Sensor $ +\rangle$ & Good Ancilla	111
4.21 IPEA - Relaxing Sensor $ 1\rangle$ & Good Ancilla	111
4.22 Scaling of the Success probability as a function of noise for different n 's. For each sub-figure we see a cross-over where the probability of success is higher when using LPS.	113
4.23 A two dimensional map of the success probability difference between physical and logical control methods for different desired precision. It is evident that for gate fidelities higher than approximately 0.985 for each desired precision, the probability of success is higher when using LPS, and gets closer to the traditional approach as the number of post selected trials for each digits is larger.	113
4.24 Histogram of IPEA results in the presence of Noise with LPS	114
4.25 Mean of Histogram attained from of IPEA	115
4.26 Accurate Sensing with IPEA	116

A.1	The Bell State	125
A.2	Verifying Decoherence	129
A.3	All Defined Gates	131

List of Tables

1.1	Generator Table for the 5-Qubit Code	34
1.2	Recovery Table for the 5-Qubit Code	34
3.1	Gate-Hamiltonian for Simulation	46
3.2	Logical Post Selection - Error Probability Scaling	57
4.1	Thresholds Accelerated Hamiltonian	103
4.2	Assumptions for IPEA	109

1 Introduction

1.1 Wait, But What Is A Qubit?

The *bit* is the fundamental unit of classical information. It can be in one of two states - 0 or 1. The bit can be thought of as a pure mathematical concept, and it has numerous physical realizations, usually two distinct values of voltage, current, light intensity, magnetization or polarization, etc. Quantum information theory is built upon an analogous fundamental unit - the *qubit*.

In this section we introduce the concept of a qubit, discuss its mathematical properties and give examples for a few of its physical implementations.

In contrast with the bit which has to be in one of its two states, a qubit can be in a normalized complex superposition of the states:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

With $\alpha, \beta \in \mathbb{C}$, $|\alpha|^2 + |\beta|^2 = 1$. However, when we 'read' (measure) the qubit we can read only one of our two computational basis states - $|0\rangle$ with probability $|\alpha|^2$ and $|1\rangle$ with probability $|\beta|^2$. Thus if we have a global phase $e^{i\theta}$ multiplying the state, it has no physical measurable effect.

1.1.1 The Bloch Sphere

One particularly useful way to think of qubits is the following geometrical interpretation: The qubit has two degrees of freedom, and thus it can be represented as a point on the unit sphere:

$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\varphi} \sin \frac{\theta}{2} |1\rangle$$

Up to a global phase.

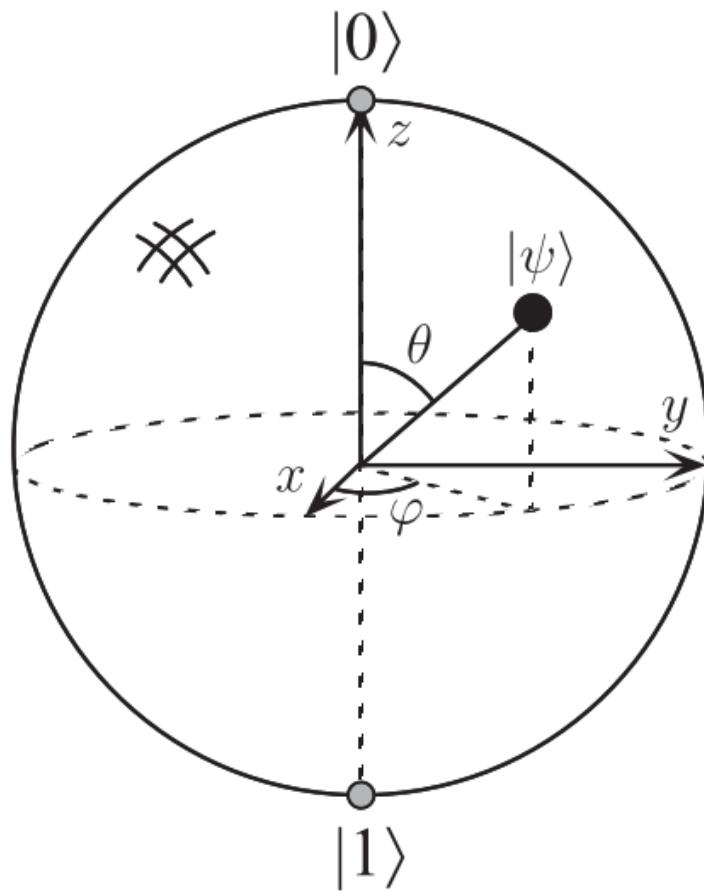


Figure 1.1: Bloch sphere representation of a qubit from [32].

We can also represent the density matrix ρ as a function of the point it creates on Bloch sphere:

$$\rho = \frac{I + \vec{r} \cdot \vec{\sigma}}{2} = \frac{1}{2} \begin{bmatrix} 1 + r_z & r_x - ir_y \\ r_x + ir_y & 1 - r_z \end{bmatrix}$$

1.1.2 Quantum Gates

In perfect analogy to the classical logic gates, there are quantum logic gates. While the only classical single bit gate is the NOT gate, there are a continuum of single qubit gates moving the qubit from one point on the Bloch sphere to another. This is true for two qubit gates as well, and in this section we introduce the most frequently used quantum gates. All frequently used gates are defined in figure 1.3.

In addition to the defined gates of figure 1.3 we define the useful gates

$$R_x(\theta) = e^{-i\frac{\theta}{2}\sigma_x}, R_y(\theta) = e^{-i\frac{\theta}{2}\sigma_y}, R_z(\theta) = e^{-i\frac{\theta}{2}\sigma_z} \quad (1.1)$$

They are particularly useful since all single qubit gates can be described as rotations on the Bloch sphere. Bloch Representation of some of them are in figure 1.2.

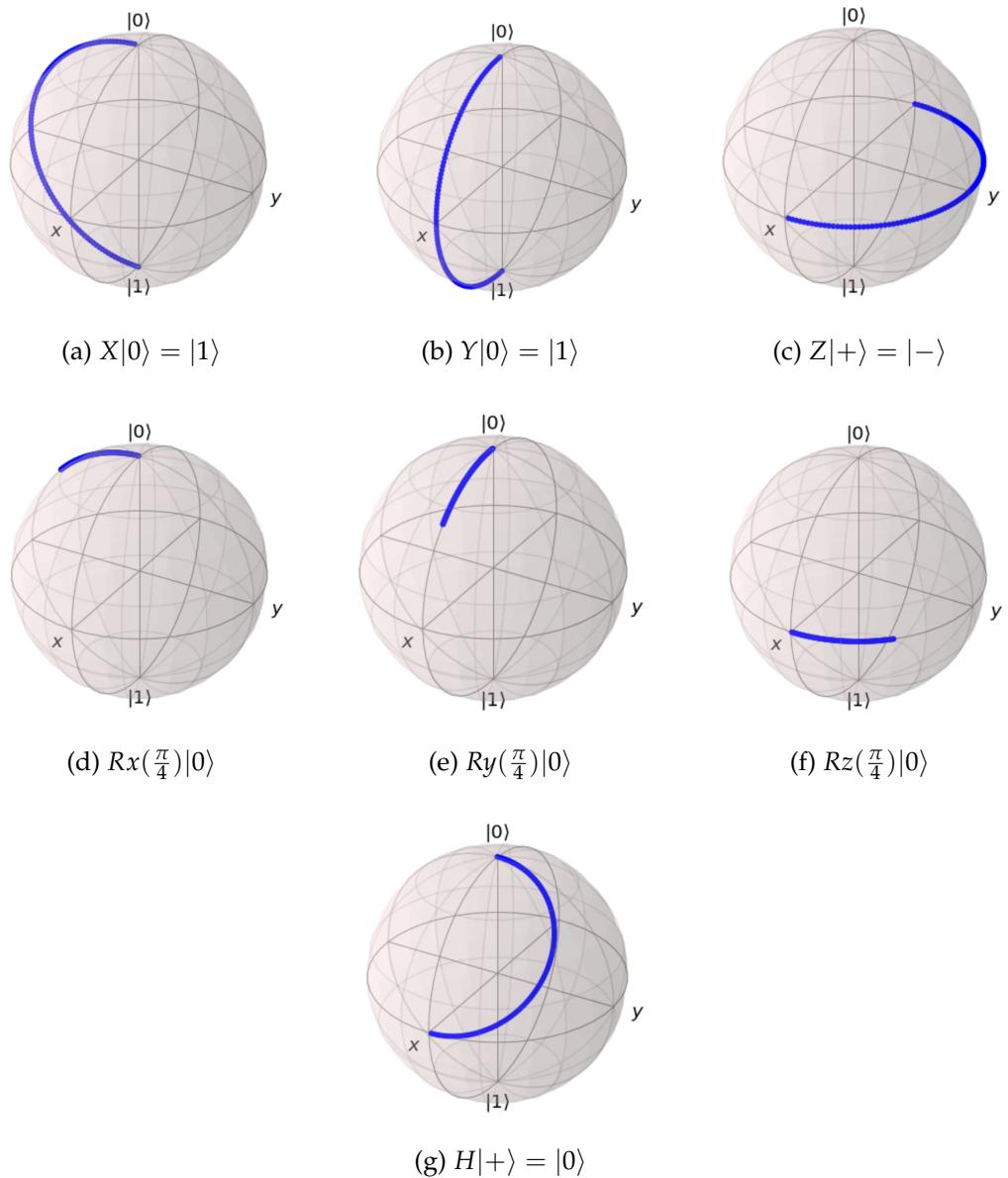


Figure 1.2: Basic gates on Bloch sphere. The simulation acts correctly for the defined gates.

Hadamard		$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Pauli-X		$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y		$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z		$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Phase		$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
$\pi/8$		$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$
controlled-NOT		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
swap		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
controlled-Z		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$

Figure 1.3: Frequently used quantum gates from [32]. Gates are defined in the basis $(0, 1)$ for single qubit gates and in the basis $(00, 01, 10, 11)$ for two qubit gates.

1.1.3 Physical Implementations

In order to realize a physical qubit one needs to satisfy four basic requirements:

- The quantum system at stake should have two distinct states (possibly degenerate) which are easily separated from other energy levels or states of the system.
- Controllable unitary evolution should be possible, allowing the implementation of quantum gates.
- One has to be able to prepare an initial state.
- One has to be able to measure the qubits.

In this section we discuss two leading implementations of qubits: Superconducting circuits and Ion traps.

Superconducting Circuits

One prominent platform for constructing a multi-qubit quantum processor involves superconducting qubits [27, 28, 44], in which information is stored in the quantum degrees of freedom of nanofabricated, anharmonic oscillators constructed from superconducting circuit elements. In contrast to other platforms, superconducting qubits are macroscopic in size and are thus easier to work with.

One remarkable feature of superconducting qubits is that their energy-level spectra are governed by circuit element parameters and thus are configurable; they can be designed to exhibit “atom-like” energy spectra with

desired properties. Therefore, superconducting qubits are also often referred to as artificial atoms, offering a rich parameter space of possible qubit properties and operation regimes, with predictable performance in terms of transition frequencies, anharmonicity, and complexity.

Ion Traps

Spins provide potentially good representations for qubits [32]. Spin is a strange (but very real!) concept but since the energy difference between different spin states is typically very small compared with other energy scales (such as the kinetic energy of typical atoms at room temperature), the spin states of an atom are usually difficult to observe, and even more difficult to control. In carefully crafted environments, however, exquisite control is possible. Such circumstances are provided by isolating and trapping small numbers of charged atoms in electromagnetic traps, then cooling the atoms until their kinetic energy is much lower than the spin energy contribution. After doing this, incident monochromatic light can be tuned to selectively cause transitions which change certain spin states depending on other spin states. This is the essence of how trapped ions can be made to perform quantum computation. A quantum register made up of trapped ions is comprised of the spin states of the qubits, and one extra qubit is represented by the collective motional state of the trapped ions, enabling qubit-qubit entangled interaction.

1.2 Quantum Noise

Up until now we have discussed the mathematical properties of a qubit and more importantly, we have mentioned some of its physical implementations. In this section we describe two types of noise, one is the famous *shot noise* due to the discrete nature of quantum systems, and one is related to the fact that a qubit can not be perfectly isolated from the environment.

1.2.1 Shot Noise

Shot noise is the most fundamental limitation on our ability to extract information from a quantum system. Suppose a qubit is in some unknown state, and our goal is to assess the state in our computational basis. Measuring the qubit will collapse it to one of our computational basis vectors, with some probability. When we refer to shot noise, we usually mean to refer to the fundamental uncertainty of assessing the right probabilities: We will always deal with some standard deviation due to the fact our experiment results have some probability distribution. By the law of large numbers, this standard deviation is proportional to the square root of the number of measurements, and thus the relative error will approach zero as the inverse square of the number of measurements. We use shot noise to write the corrected error due to lost information, in equations 3.10, 3.11.

1.2.2 Decoherence

Another kind of noise is arising from the fact our principal quantum system (say, a qubit) couples to the environment in some unknown way. There are two complementary ways to describe this quantum noise: The discrete description, called the *operator-sum representation* of the noise, and the continuous description with *master equations*. We use the discrete description of noise to write our simulations, and so we will focus on the operator-sum representation of noise. For more information on master equations, please consult ref. [39].

A natural way to describe the dynamics of an open quantum system (our principal system) is to describe the dynamics of the closed quantum system, comprised of our principal system and the environment. Suppose our principal system is sent to a black-box U operating on the system and the environment, as in figure 1.4. For the purpose of this short derivation we assume the input state for this black-box is a tensor product state (a more detailed derivation could be found in ref. [32]) and we define the quantum operation acting on the system as

$$\mathcal{E}(\rho) = \text{Tr}_{\text{env}}[U(\rho \otimes \rho_{\text{env}})U^\dagger] \quad (1.2)$$

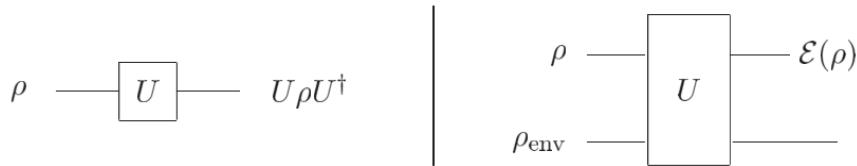


Figure 1.4: Models of closed (left) and open (right) quantum systems. An open quantum system consists of two parts, the principal quantum system and the environment. The figure is taken from [32].

The *operator-sum representation* of equation 1.2 is just writing the same equation in terms of the operators of the principal system's Hilbert space alone. Let $|e_k\rangle$ be an orthonormal basis for the Hilbert space of the environment, and suppose the environment starts from a pure state $\rho_{env} = |e_0\rangle\langle e_0|$. Equation 1.2 can thus be rewritten as

$$\begin{aligned}\mathcal{E}(\rho) &= \sum_k \langle e_k | U[\rho \otimes |e_0\rangle\langle e_0|]U^\dagger | e_k \rangle \\ &= \sum_k E_k \rho E_k^\dagger\end{aligned}\tag{1.3}$$

Where $E_k = \langle e_k | U | e_0 \rangle$ is an operator on the state space of the principal system. This is the operator sum representation. We now describe three noise processes using the operator sum representation, focusing on dephasing and amplitude damping. We sometimes refer to the operation elements E_k as the **Kraus operators** of the noise process.

The Depolarizing Channel

The depolarizing channel is of use to us for performing the error probability analysis of section 3.5.1, because the ability to error-correct the depolarizing channel automatically implies the ability to error-correct an arbitrary single qubit quantum operation. It is defined as applying each of the Pauli operators with probability $p/3$:

$$\mathcal{E}(\rho) = (1 - p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z)$$

Amplitude Damping

Amplitude damping, which we sometimes refer to as energy dissipation, energy relaxation, T_1 process, or just relaxation, is the effect due to loss

of energy from the system to the environment. The Kraus operators for energy relaxation are as follows:

$$E_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{bmatrix}, E_1 = \begin{bmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{bmatrix} \quad (1.4)$$

E_1 is the lowering operator multiplied with the probability to lose a photon. E_0 leaves $|0\rangle$ unchanged but reduces the population of $|1\rangle$ by a factor of $1 - p$. Physically it means that no energy was lost to the environment and so the environment believes it is more likely that the system is in the ground state rather than the excited state. Replacing p with something time-dependent, like $p = 1 - e^{-t/T_1}$ (thus defining T_1) and applying on the quantum state $\rho = \begin{pmatrix} a & b \\ b^* & c \end{pmatrix}$ gives:

$$\mathcal{E}(\rho) = \begin{bmatrix} 1 - (1-p)(1-a) & b\sqrt{1-p} \\ b^*\sqrt{1-p} & c(1-p) \end{bmatrix} = \begin{bmatrix} 1 - e^{-t/T_1}(1-a) & be^{-t/2T_1} \\ b^*e^{-t/2T_1} & ce^{-t/T_1} \end{bmatrix} \quad (1.5)$$

Phase Damping

Phase damping is a purely quantum mechanical process, which describes the loss of information without the loss of energy. The energy eigenstates of a quantum system do not change as a function of time, but rather accumulate phase proportional to the energy eigenvalue. When an eigenvector evolves in time for a not precisely known amount of time, partial information about this quantum phase - relative phase between energy eigenstates - is lost. We sometimes refer to phase damping as dephasing, or T_2 process.

Here we describe a simple model for this noise process. Suppose we have a qubit $|\psi\rangle = a|0\rangle + b|1\rangle$ upon which a random phase kick $R_z(\theta)$ is acted, with θ being a random variable which has a Gaussian distribution with

mean 0 and variance 2λ . The output density matrix from this process is given from averaging over θ :

$$\begin{aligned}\rho &= \frac{1}{\sqrt{4\pi\lambda}} \int_{-\infty}^{\infty} R_z(\theta) |\psi\rangle\langle\psi| R_z^\dagger(\theta) e^{-\theta^2/4\lambda} d\theta \\ &= \begin{bmatrix} |a|^2 & ab^* e^{-\lambda} \\ a^* b e^{-\lambda} & |b|^2 \end{bmatrix}\end{aligned}$$

We define the rate at which the coherences are lost as t/T_2 . This process will make the following transformation to the point on Bloch's sphere:

$$(r_x, r_y, r_z) \rightarrow (\sqrt{1-\lambda}r_x, \sqrt{1-\lambda}r_y, r_z)$$

meaning the point will just get closer to the Z axis. Choosing the probability $p = 1 - e^{-t/T_2}$ gives the following possible representation to the Kraus operators:

$$K_0 = \sqrt{p} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, K_1 = \sqrt{p} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, K_2 = \sqrt{1-p} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (1.6)$$

Such that

$$\mathcal{E}(\rho) = \begin{bmatrix} a & (1-p)b \\ b^*(1-p) & c \end{bmatrix} = \begin{bmatrix} a & be^{-t/T_2} \\ b^*e^{-t/T_2} & c \end{bmatrix} \quad (1.7)$$

1.3 Distance Measures for Quantum Information

How close are two quantum states? For this purpose we have to define a distance measure, a metric, which has to satisfy three ground rules:

- $D(x, y) = D(y, x)$
- $D(x, y) = 0 \rightarrow x = y$

- $D(x, z) \leq D(x, y) + D(y, z)$

In this section we try to answer this question, defining the two distance measures widely used in this work - the *Distance*, a variation to *Trace Distance* of ref.[32], which gives us information of the possible outcomes of a measurement, and the *Fidelity* which is not precisely a metric but a more quantum measure, which approaches 1 for similar states and 0 for distant states.

Distance

We define the Distance between two quantum states ρ, σ to be

$$D(\rho, \sigma) = \text{Tr}|\rho - \sigma|^2 \quad (1.8)$$

With $|A| = \sqrt{A^\dagger A}$ the positive square root of $A^\dagger A$. It is possible to prove that our definition of *Distance* behaves the same as the widely used *Trace Distance* defined as $\frac{1}{2}\text{Tr}|\rho - \sigma|$ [32].

A useful interpretation to the above definition is given using the geometrical picture of the Bloch sphere: Suppose

$$\rho = \frac{I + \vec{r} \cdot \vec{\sigma}}{2}, \sigma = \frac{I + \vec{s} \cdot \vec{\sigma}}{2}$$

Using our definition the distance is $D(\rho, \sigma) = \frac{1}{2}|\vec{r} - \vec{s}|^2$, just half the square of the ordinary Euclidean distance.

Fidelity

We use the regular definition of *Fidelity*,

$$F(\rho, \sigma) = \text{Tr}\sqrt{\rho^{1/2}\sigma\rho^{1/2}} \quad (1.9)$$

The fidelity of two quantum states is, as mentioned, not a metric. It resembles an inner product between two quantum states, and it is indeed defined in such a way:

$$F(|\psi\rangle, \rho) = \sqrt{\langle\psi|\rho|\psi\rangle}, F(|\psi\rangle, |\varphi\rangle) = \langle\psi|\varphi\rangle$$

We can see it approaches 1 as the states are closer and approaches 0 when they are not.

1.4 Quamtum Sensing

Quantum sensing is one of the few areas of study intended to have real-world consequences in the NISQ era. [31, 35]. "Quantum Sensing" is typically used to describe one of the following [16]:

- Use of a quantum object to measure a physical quantity (classical or quantum). The quantum object is characterized by quantized energy levels.
- Use of quantum coherence to measure a physical quantity.
- Use of quantum entanglement to improve the sensitivity or precision of a measurement, beyond what is possible classically ("Quantum Metrology").

In this study, we mainly focus on quantum metrology.

The Hamiltonian of the quantum sensor consists of three parts: The internal Hamiltonian of the probe, the signal Hamiltonian and the control Hamiltonian used to manipulate the probe and read out the data:

$$\hat{H}(t) = \hat{H}_0 + \hat{H}_V(t) + \hat{H}_{control}(t)$$

Typically, and in our scenarios, the internal Hamiltonian \hat{H}_0 describes approximately (up to higher energy levels) the energy eigenstates $|0\rangle, |1\rangle$:

$$\hat{H}_0 = E_0|0\rangle\langle 0| + E_1|1\rangle\langle 1| = \frac{1}{2}\hbar\omega_0\sigma_z$$

With $\hbar\omega_0$ being the energy difference. Note that it is not strictly necessary that $\omega_0 \neq 0$ but it will be the case in most scenarios.

The signal Hamiltonian is a function of some external parameter V that causes a change in the internal energy $\hbar\omega_0$ of the form $\gamma = \partial^q E / \partial V^q$ usually with $q = 1$ or $q = 2$. The signal Hamiltonian is a sum of two qualitatively different contributions, the parallel (commuting with the internal Hamiltonian) part $\hat{H}_{V\parallel}(t)$ and the transverse (non-commuting) component $\hat{H}_{V\perp}(t)$. We can write these two parts as:

$$\begin{aligned} \hat{H}_{V\parallel}(t) &= \frac{1}{2}\gamma V_{\parallel}(t)\{|1\rangle\langle 1| - |0\rangle\langle 0|\} \\ \hat{H}_{V\perp}(t) &= \frac{1}{2}\gamma\{V_{\perp}(t)|1\rangle\langle 0| + V_{\perp}^{\dagger}(t)|0\rangle\langle 1|\} \end{aligned} \quad (1.10)$$

These contributions have different effects on the probe: The parallel one leads to shifts in the energy levels and an associated change in the transition frequency ω_0 and the transverse one can cause transitions between energy levels, thus changing the transition rate.

1.4.1 The Quantum Sensing Protocol

In this subsection we describe the basic methodology for performing measurements with quantum systems. This methodology always consists of three elementary steps: initialization of the sensor, interaction with the signal and readout of the final state. The key quantity is the *quantum phase* picked up by the quantum sensor due to the interaction. After that ba-

sic protocol, algorithmic quantum sensing may be applied to recover the measured quantity.

The basic steps for the quantum sensing protocol appears in figure 1.5. Here we elaborate on the steps:

1. The quantum sensor is initialized to some initial state, usually $|0\rangle$.
2. The quantum sensor is initialized to the sensing state.
3. The probe accumulates phase according to the interaction with the signal.
4. the state of the probe is transformed to one that is informatively measured in the readout basis, which is usually the computational basis.
5. The final state is measured.
6. We repeat steps 1 through 5 and average for the probabilities.
7. We usually repeat steps 1 through 6 for a set of different scenarios, and estimate the wanted information of the signal from them.

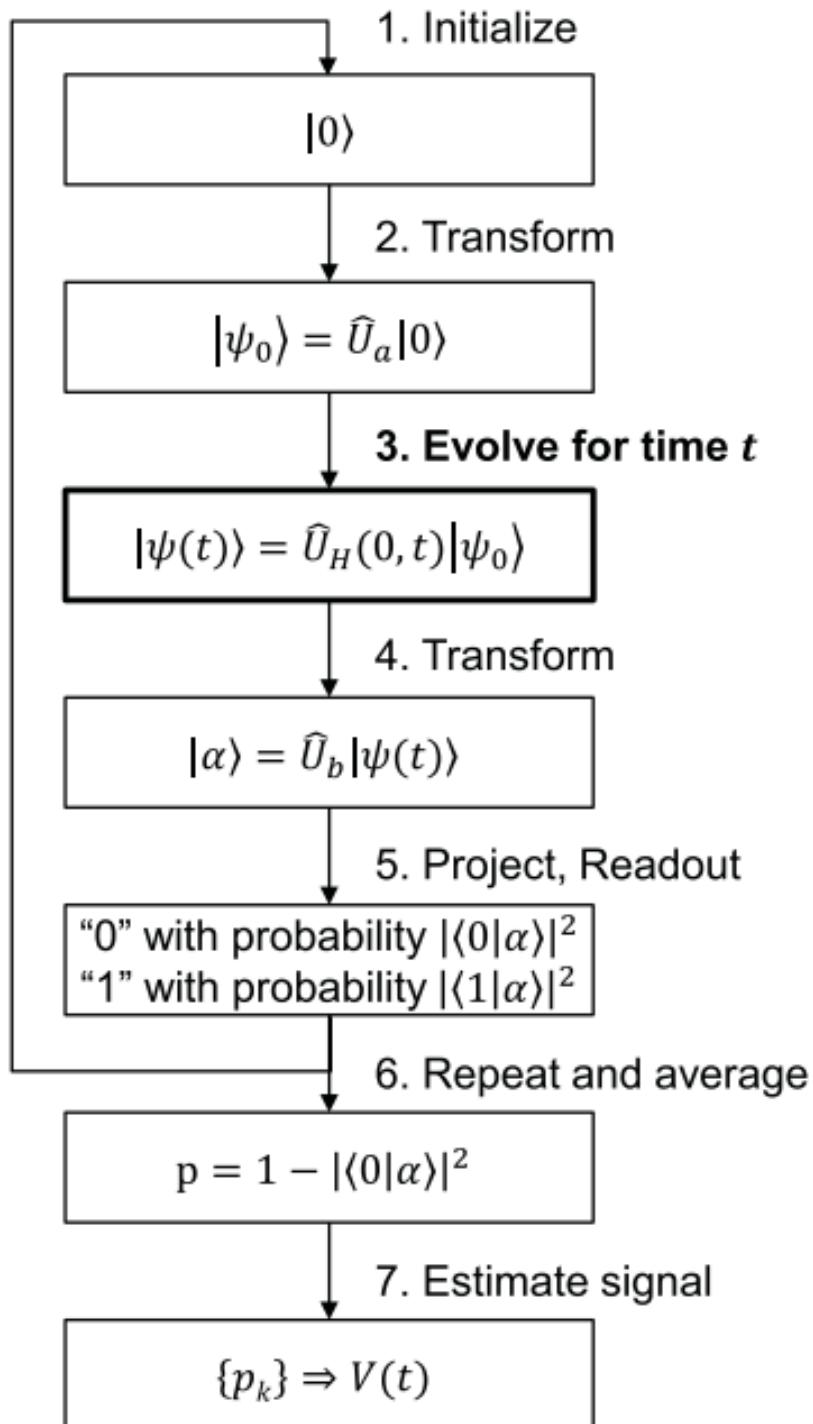


Figure 1.5: Basic steps of the quantum sensing protocol. The figure is taken from [16].

A good example for using this scheme is the Ramsey experiment.

Ramsey Measurement

The Ramsey experiment is used to estimate the static energy splitting $\hbar\omega_0$, or T_2 in noisy experiments. Here we discuss the ideal case with no noise. We discuss the noisy version at section 4.1.2.

The steps for the Ramsey experiment are as follows:

1. The quantum sensor is initialized to the ground state, $|0\rangle$.
2. Using a $\pi/2$ pulse, or Hadamard gate, the quantum sensor is transformed to the superposition state

$$|\psi(0)\rangle = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

3. The sensor evolves under the internal Hamiltonian \hat{H}_0 for time t and accumulates a relative phase,

$$|\psi(t)\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{-i\omega_0 t}|1\rangle)$$

4. Using another $\pi/2$ pulse, or Hadamard gate, the state $|\psi(t)\rangle$ is converted to a measurable state

$$|\alpha\rangle = \frac{1}{2}(1 + e^{-i\omega_0 t})|0\rangle + \frac{1}{2}(1 - e^{-i\omega_0 t})|1\rangle$$

5. The final state is read-out, giving the transition probability

$$p = 1 - |\langle 0|\alpha\rangle|^2 = \sin^2(\omega_0 t/2)$$

Recording p as a function of t enables us to estimate the energy splitting $\hbar\omega_0$.

1.4.2 Fisher Information & Quantum Cramer Rao Bound

For detailed explanation on Fisher information and Quantum Fisher Information, please consult ref. [31]. Fisher information is all about estimating the amount of information a probability distribution holds. For us, the Fisher information is some quantity that helps us to quantify how precise we can get measuring another quantity.

To quantify the distinguishability of two neighboring probability distributions, the Fisher information is defined by [48]:

$$F(\{P_x(\theta)\}) = \sum_x \frac{1}{P_x(\theta)} \left(\frac{\partial P_x(\theta)}{\partial \theta} \right)^2$$

where x is the label of measurement results, $P_x(\theta)$ is the probability of obtaining x when the parameter is equal to θ , satisfying $P_x(\theta) \geq 0$ and $\sum_x P_x(\theta) = 1$. The Fisher information satisfies

$$F(\{P_x(\theta)\}) \leq \text{Tr}[\rho_\theta L_\theta^2] = J(\rho_\theta)$$

Here, L_θ is the Symmetric Logarithmic Derivative, defined by $\partial_\theta \rho_\theta = \frac{1}{2}(L_\theta \rho_\theta + \rho_\theta L_\theta)$ and $J(\rho_\theta)$ is the quantum Fisher information.

Another way to calculate the Fisher information, theoretically, is given the density matrix ρ and an observable A which generates a unitary rotation of the state with parameter θ , $\rho(\theta) = e^{-i\theta A} \rho_0 e^{i\theta A}$, the quantum Fisher information is defined to be:

$$J(\rho, A) = 2 \sum_{k,l} \frac{(\lambda_k - \lambda_l)^2}{(\lambda_k + \lambda_l)} |\langle k | A | l \rangle|^2$$

Where λ_k and $|k\rangle$ are the eigenvalues and eigenvectors of the density matrix ρ , respectively.

Using the notion of quantum Fisher information and pure mathematical considerations, one can write down the Quantum Cramer-Rao Bound:

$$\delta\theta \geq \frac{1}{\sqrt{N}J(\rho_\theta)} \quad (1.11)$$

Where N is the number of repetitions.

Using Entanglement to Saturate QCRB

The (quantum) Cramer-Rao bound contains a factor $1/\sqrt{N}$ which accounts for the fact that we can simply decrease the standard deviation of our estimates by averaging over N independent repetitions of the same experiment. The scaling

$$\delta\theta \propto \frac{1}{\sqrt{N}}$$

is called the standard quantum limit (SQL) or shot-noise limit. But the approach of just performing independent repetitions of the same experiment does not make use of one of the most crucial properties of quantum mechanics, namely *entanglement*. Here we repeat the Ramsey measurement, with N entangled probes:

1. The quantum sensor is initialized to the ground state, $|0..0\rangle$.
2. The quantum sensor is transformed to the GHZ state

$$|\psi(0)\rangle = \frac{1}{\sqrt{2}}(|0..0\rangle + |1..1\rangle)$$

3. The sensor evolves under the internal Hamiltonian \hat{H}_0 for time t and accumulates an enhanced relative phase,

$$|\psi(t)\rangle = \frac{1}{\sqrt{2}}(|0..0\rangle + e^{-iN\omega_0 t}|1..1\rangle)$$

4. The state $|\psi(t)\rangle$ is converted to a measurable state

$$|\alpha\rangle = \left[\frac{1}{2}(1 + e^{-iN\omega_0 t})|0_1\rangle + \frac{1}{2}(1 - e^{-iN\omega_0 t})|1_1\rangle \right] |0..0\rangle_{2,N}$$

5. The final state is read-out, giving the transition probability

$$p = 1 - |\langle 0|\alpha\rangle|^2 = \sin^2(N\omega_0 t/2)$$

This factor of N enhancement lies at the heart of quantum advantage in sensing. You might rightfully ask why this is the case, because we also got a factor N enhancement from just performing N separate repetitions. But it turns out that the fact that we enhanced the signal by a factor of N actually gives us a factor N^2 improvement in the Fisher information! By enhancing the signal by a factor of N we actually changed the rate with which it changes – the derivative with respect to θ – by a factor of N compared to a single repetition. Because the Fisher information contains the square of the derivative, this effectively gives us the enhancement by a factor of N^2 , leading to the new scaling of the standard deviation:

$$\delta\theta \propto \frac{1}{N}$$

Which is actually the most fundamental limit attainable when exploiting quantum mechanical effects and is called the *Heisenberg limit*.

It is important to note that this scaling is achievable if the probability distribution of sampling the desired observable is not biased, according to [16]. **In this work we see that this is not the case - in the presence of noise the probability distribution attained from algorithmic quantum sensing is biased, and we give a way of making it less biased, thus approaching to the Heisenberg limit.**

1.4.3 Algorithmic Quantum Sensing

Degen *et al.* [16] defines quantitatively the Dynamic Range of a quantum sensor, and finds it scales as the square root of the measurement time. This is where algorithmic quantum sensing is important - It gives a way to enlarge the dynamic range of a quantum sensor to scale as the measurement time by giving an appropriate weighting to different quantum measurements, thus approaching the Heisenberg limit. Here we give a detailed explanation on the most fundamental family of algorithmic sensing protocols, called Quantum Phase Estimation (QPE). Most algorithmic sensing protocols are variations of this family of algorithms, and QPE has numerous applications in other areas of study [14, 33, Santagati et al., Kais], the most famous one being Shor's algorithm.

In the following section we define the problem of quantum phase estimation and give the textbook algorithm, and in the subsequent three sections we give detailed explanation on an implementation of QPE and the two iterative versions of quantum phase estimation that we use - iterative phase estimation algorithm (IPEA) and Kitaev's iterative approach. In the IPEA subsection I will also give a showcase of the histogram of results, in the presence of noise.

Algorithm: Quantum phase estimation

Inputs: (1) A black box which performs a controlled- U^j operation, for integer j ,
 (2) an eigenstate $|u\rangle$ of U with eigenvalue $e^{2\pi i \varphi_u}$, and (3) $t = n + \lceil \log(2 + \frac{1}{2\epsilon}) \rceil$ qubits initialized to $|0\rangle$.

Outputs: An n -bit approximation $\widetilde{\varphi}_u$ to φ_u .

Runtime: $O(t^2)$ operations and one call to controlled- U^j black box. Succeeds with probability at least $1 - \epsilon$.

Procedure:

1. $|0\rangle|u\rangle$ initial state
2. $\rightarrow \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle|u\rangle$ create superposition
3. $\rightarrow \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle U^j |u\rangle$ apply black box
 $= \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} e^{2\pi i j \varphi_u} |j\rangle|u\rangle$ result of black box
4. $\rightarrow |\widetilde{\varphi}_u\rangle|u\rangle$ apply inverse Fourier transform
5. $\rightarrow \widetilde{\varphi}_u$ measure first register

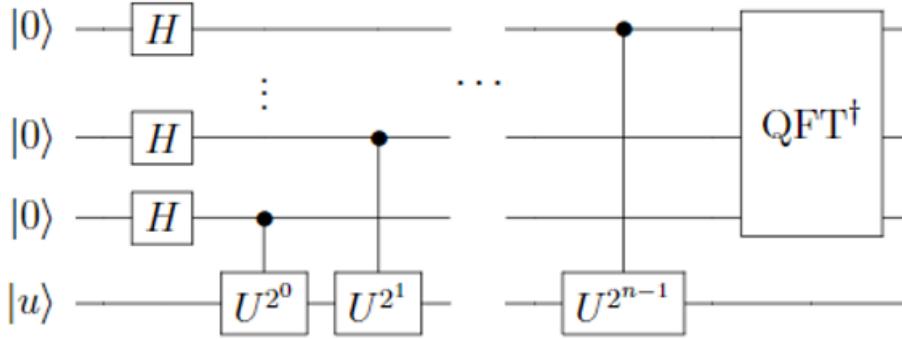


Figure 1.6: the standard QPE procedure, based on inverse QFT. Each digit of the phase is measured separately. The figure is taken from [32].

Quantum Phase Estimation

Quantum phase estimation is a family of algorithms. Suppose a unitary operator U has an eigenvector $|u\rangle$ with eigenvalue $e^{2\pi i \phi}$, where the value

of ϕ is unknown. The goal of the phase estimation algorithm is to estimate ϕ . To perform the estimation, we assume we have available black boxes capable of preparing the state $|u\rangle$ and performing controlled- U^{2^j} operations for some positive integer j . The algorithm uses two quantum registers, one for the measured operator U and one for ancilla qubits needed for the computation. Phase estimation was first introduced by Kitaev [26]. The standard approach, based on inverse QFT, is depicted in figure 1.6 [32]. Stage 4 in this approach, applying iQFT on the state

$$\frac{1}{2^{\frac{n}{2}}} \sum_{k=0}^{2^n-1} e^{2\pi i \theta k} |k\rangle$$

brings us to the multi-qubit state

$$\frac{1}{2^{\frac{n}{2}}} \sum_{k=0}^{2^n-1} e^{2\pi i \theta k} \left(\frac{1}{2^{\frac{n}{2}}} \sum_{x=0}^{2^n-1} e^{-\frac{2\pi i k x}{2^n}} |x\rangle \right) = \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{k=0}^{2^n-1} e^{-\frac{2\pi i k}{2^n} (x - 2^n \theta)} |x\rangle$$

We can approximate the value of $\theta \in [0, 1]$ by rounding $2^n \theta$ to the nearest integer. This means that $2^n \theta = a + 2^n \delta$, where a is the nearest integer to $2^n \theta$ and the difference $2^n \delta$ satisfies $0 \leq |2^n \delta| \leq \frac{1}{2}$. We can now write the state:

$$\sum_{x=0}^{2^n-1} \left(\frac{1}{2^n} \sum_{k=0}^{2^n-1} e^{-\frac{2\pi i k}{2^n} (x-a)} e^{2\pi i \delta k} \right) |x\rangle$$

Performing a measurement in the computational basis on the first register yields the closest integer $|a\rangle$ with probability

$$\Pr(a) = \left| \frac{1}{2^n} \sum_{k=0}^{2^n-1} e^{-\frac{2\pi i k}{2^n} (x-a)} e^{2\pi i \delta k} \right|^2$$

Since we are measuring $|a\rangle$, $x = a$ meaning the expression will reduce to

$$\Pr(a) = \frac{1}{2^{2n}} \left| \sum_{k=0}^{2^n-1} e^{2\pi i \delta k} \right|^2 = \begin{cases} 1 & \delta = 0 \\ \frac{1}{2^{2n}} \left| \frac{1-e^{2\pi i 2^n \delta}}{1-e^{2\pi i \delta}} \right|^2 & \delta \neq 0 \end{cases}$$

For $\delta = 0$ the approximation is precise, and we always measure the accurate value of the phase. For $\delta \neq 0$ since $|\delta| \leq \frac{1}{2^{n+1}}$ the algorithm yields the correct result with probability $\Pr(a) \geq \frac{4}{\pi^2} \approx 0.405$ - see any textbook on QPE for complete proof.

It can be realized in many different ways, starting with Kitaev's original approach, using Quantum Fourier Transform (QFT) or using approximate QFT (AQFT). Here we will present three different approaches for doing QPE: an approach using multiple qubits, with constant precision phase shift operators [3], Kitaev's original approach [26], and an approach called Iterative Phase Estimation Algorithm (IPEA) [18].

Quantum Phase Estimation with Constant Precision Phase Shift Operators

The standard QPE method does not use post processing and so is more resilient to noise in the measurement. Its disadvantage is that it uses a large number of qubits and a large number of gates, thus increasing the level of noise. The standard procedure is given in figure 1.6. The constant precision phase shift operators approach has the same advantages and disadvantages as the previous one, but Ahmadi [3] has shown that it can be implemented using a reduced number of gates, and thus it is more resilient to noise. The inverse QFT implemented in this approach is given in figure 1.7, with R_k^{-1} being $\begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{pmatrix}$.

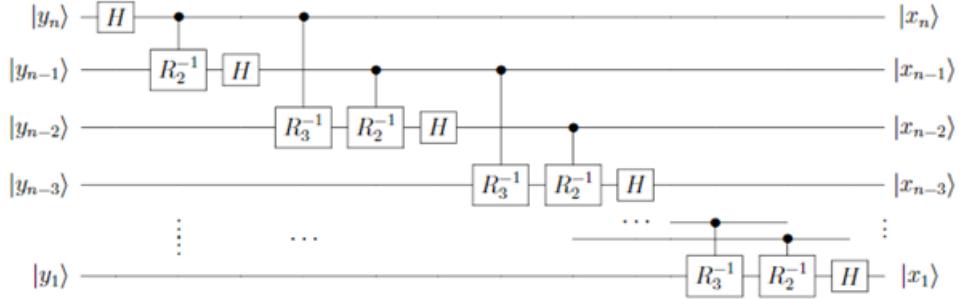


Figure 1.7: Approximate QFT with only two controlled phase shift operations replacing the inverse QFT part in the text-book QPE. The figure is taken from [3].

Kitaev's Quantum Phase Estimation

From now on we refer to the precision by m and to other quantities by n depending on the context (the number of qubits in the measured eigenvalue or the number of post selected trials). Defining $\alpha_k = 2^{k-1}\tilde{\phi}$ and using the circuit from figure 1.8 we get for using $K = I$ the relation $\cos(2\pi\alpha_k) = 2P(0|k) - 1$ and for using $K = S$ the relation $\sin(2\pi\alpha_k) = 1 - 2P(0|k)$. We have enough information to extract α_k . Once we obtain all the α_k for k from 1 to m , we can retrieve $\tilde{\phi}$ using the following algorithm:

Algorithm 1: Kitaev Estimator

Result: $\tilde{\phi} = 0.\phi_1\phi_2\dots\phi_{m+2}$ the $(m+2)$ -bit approximation to the phase ϕ

```

1 Estimate all  $\alpha_k$  using the circuit in figure 1.8;
2 Set  $\beta_m = 0.\phi_m\phi_{m+1}\phi_{m+2}$  where  $\beta_m$  is the closest octant  $\{\frac{0}{8}, \frac{1}{8}, \dots, \frac{7}{8}\}$ 
   to  $\alpha_m$ ;
3 for  $j = m - 1$  to 1 do
4    $\phi_j = \begin{cases} 0 & \text{if } |0.0\phi_{j+1}\phi_{j+2} - \alpha_j|_{mod1} < 1/4 \\ 1 & \text{if } |0.1\phi_{j+1}\phi_{j+2} - \alpha_j|_{mod1} < 1/4 \end{cases}$ ;
5 end
```

This algorithm can be used today on Amazon's and IBM's quantum computers – the tasks are independent, the algorithm is built for the ability to estimate probabilities, and tasks can be launched in parallel.

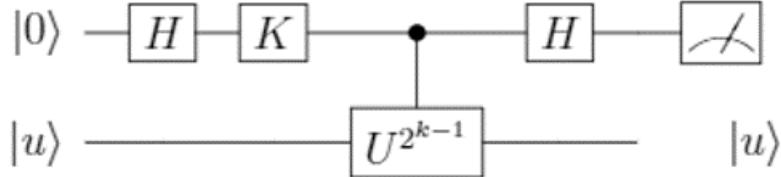


Figure 1.8: Measuring the k 'th digit using Kitaev's iterative approach. Get measurement histograms for $K = S$ and $K = I$, and then extract the digit using classical computation as described in the algorithm 1.

Iterative Phase Estimation Algorithm

Iterative Phase Estimation [18] uses only one ancilla qubit to perform the phase estimation, and so it has great importance in the age of Noisy In-

termediate Scale Quantum (NISQ) computers, since we cannot currently use many qubits simultaneously. (Note that quantum error correction increases the number of qubits at least 5 times fold). Its disadvantage is that it is a dynamic quantum algorithm, in which future states depend on outcomes of measurements that happen during the circuit. Implementing this kind of circuit is hard, and relatively new work [40] has demonstrated that it is possible.

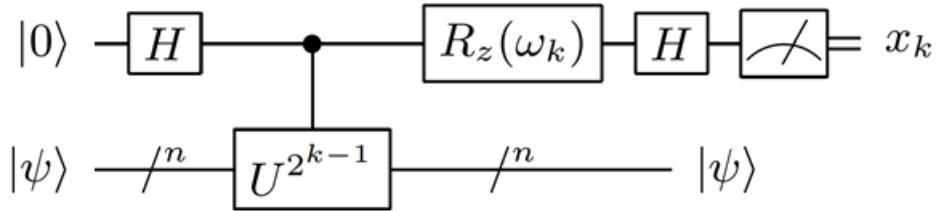


Figure 1.9: The k 'th iteration of the iterative phase estimation algorithm. The feedback angle depends on the previously measured bits through $\omega_k = -2\pi(0.0x_{k+1}x_{k+2}\dots x_m)$, and $\omega_m = 0$. The figure is taken from [18].

So how does it work? The circuit for iterative phase estimation is given in figure 1.9. First, suppose we want to measure a phase ϕ that has no more than m bits, $\phi = (\phi_1\phi_2\dots\phi_m000\dots)$. Then, if we label the system's state after the action of the j 'th gate by $|\psi_j\rangle$: After the first Hadamard gate

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|\psi\rangle$$

Then we apply the controlled- U^{2^j} operation and we get

$$|\psi_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i 2^{k-1}(\phi_1\phi_2\dots\phi_m000\dots)}|1\rangle)|\psi\rangle$$

Then we rotate the $|1\rangle$ state by $\omega_k = -2\pi(0.0x_{k+1}x_{k+2}\dots x_m)$, and

$$|\psi_3\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i(2^{k-1}(0.\phi_1\phi_2\dots\phi_m000\dots)-(0.0\phi_{k+1}\dots\phi_m))}|1\rangle)|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i(0.\phi_k)}|1\rangle)|\psi\rangle$$

Finally, after performing the last Hadamard

$$|\psi_4\rangle = \frac{1}{2}(\cos(\pi(0.\phi_k))|0\rangle + (\sin(\pi(0.\phi_k))|1\rangle))|\psi\rangle$$

up to a global phase. So, in the measurement, if $\phi_k = 0$ we get $|0\rangle$ with probability 1, and if $\phi_k = 1$ we get $|1\rangle$ with probability 1. This means that the whole measurement is deterministic. Now, suppose we want to measure only up to the n 'th digit, Dobsicek [18] has shown that this procedure does that with an error probability $\epsilon < 1 - 8/\pi^2$ which is independent of n . How does that work? Suppose we want to measure ϕ to a precision of m digits, $\tilde{\phi} = (\phi_1\phi_2\dots\phi_m)$. Both answers $\tilde{\phi}$ and $\tilde{\phi} + 2^{-m}$ are considered valid. Defining the general remainder $0 \leq \delta < 1$ as $\phi = \tilde{\phi} + \delta 2^{-m}$, the probability to measure ϕ_m correctly is $\cos^2(\frac{\pi\delta}{2})$, and if ϕ_m was measured correctly the probability to measure ϕ_{m-1} correctly is $\cos^2(\frac{\pi\delta}{4})$. Note that with some remainders ($\delta \rightarrow 1$), the ‘right’ digits may actually be measured with probability $P = 0$. The conditional probability to measure the k 'th bit correctly is $\cos^2(\pi\delta 2^{k-m-1})$. Thus the probability to extract $\tilde{\phi}$ is

$$P(\delta) = \frac{\sin^2(\pi\delta)}{2^{2m} \sin^2(\frac{\pi\delta}{2^m})}$$

and the probability to extract $\tilde{\phi} + \delta 2^{-m}$ is $P(1-\delta)$, so the overall success probability of the ideal algorithm is $P(\delta) + P(1-\delta) > \frac{8}{\pi^2}$, which is independent of the precision m .

Earlier I have mentioned that the probability distribution resulting from quantum phase estimation is biased in the presence of decoherence, making the Heisenberg limit impossible to achieve due to a systematic error of

the mean of the histogram of results. In figure 1.10 I give a showcase, measuring the angle $\theta = 1/3$ up to 4 digits of accuracy with IPEA. The noise is dephasing, as discussed in section 1.2.2 and the simulation is implemented as discussed in section 3.3 and in appendix A.

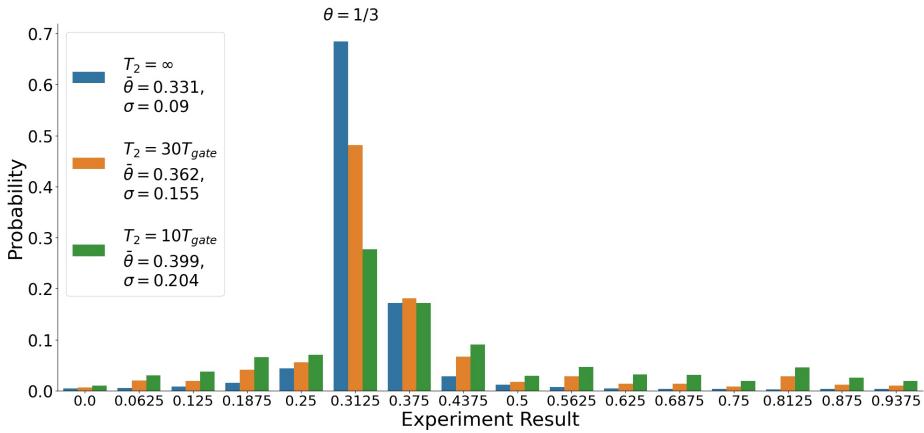


Figure 1.10: A showcase of the probability distribution of all possible results of IPEA, measuring $R_x(2\pi/3)$ up to 4 binary digits, in the presence of decoherence. It is apparent that the histogram of results is more biased the larger the noise.

1.5 Quantum Error Correction

Generally, the purpose of quantum error correction is to do quantum information processing reliably – protecting quantum information from noise. The threshold theorem for quantum computations states that provided the noise in individual quantum gates is below a certain constant threshold it is possible to efficiently perform an arbitrarily large quantum computation using methods of fault tolerance quantum computation.

The basic idea of the theory of quantum error correction is to encode quantum states by a unitary operation into a *quantum error correcting code* (QECC), formally defined as a subspace C of some larger Hilbert space. After encoding the code is subjected to noise, following which a syndrome measurement is performed to diagnose the type of error which occurred, that is, the *error syndrome*. Once this has been determined, a *recovery* operation is performed, to return the quantum system to the original state of the code [32].

1.5.1 Stabilizer Formalism

The stabilizer formalism [32] is ideally suited to the description of quantum codes. In this section we describe how this may be done and use it to introduce the five qubit code which is the smallest code that can be used to protect against the effects of arbitrary errors on a single qubit.

First, we define the pauli group G_n on n qubits: For a single qubit, we define $G_1 \equiv \{\pm I, \pm iI, \pm X, \pm iX, \pm Y, \pm iY, \pm Z, \pm iZ\}$. This set of matrices forms a group under the operation of matrix multiplication. The Pauli group on n qubits, G_n , is the group generated by the operators described

above applied to each of the n qubits in the tensor product Hilbert space $(\mathbb{C}^2)^{\otimes n}$. We can now define stabilizers a little more precisely. Suppose S is a subgroup of G_n and define V_S to be the set of n qubit states which are fixed by every element of S (for arbitrary operator $O \in S$ and state $|\psi\rangle \in V_S$ we have $O|\psi\rangle = |\psi\rangle$). V_S is the vector space stabilized by S , and S is said to be the stabilizer of the space V_S , since every element of V_S is stable under the action of elements in S . Now, every group may be described by its *generators*. A set of elements g_1, \dots, g_l in a group G is said to generate the group G if every element of G can be written as a product of elements from the list g_1, \dots, g_l , and we write $G = \langle g_1, \dots, g_l \rangle$.

The basic idea is very simple: an $[n, k]$ stabilizer code is the code that encodes k logical qubits onto n physical qubits. It is defined to be the vector space V_S stabilized by a subgroup S of G_n such that $-I \notin S$ and S has $n - k$ independent and commuting generators, $S = \langle g_1, \dots, g_{n-k} \rangle$. A method for measuring generators for the 5 qubit code is given in the Methods Chapter. For now, suppose we can measure a generator g with the result denoted by '0' indicating a state stabilized by g and a result denoted by '1' indicating a state such that $g|\psi\rangle = -|\psi\rangle$. The idea is that any error E on a single qubit (we have $4n$ such errors, for the single qubit errors X, Y, Z, I) can be detected (collapsing the state to a state with that error) and corrected by using a code-specific table with (error-syndrome, recovery operation) pairs.

As mentioned earlier, the 5 qubit code is the smallest QECC possible. It's

basis states are:

$$\begin{aligned}
 |0_L\rangle = \frac{1}{4} & [|00000\rangle + |10010\rangle + |01001\rangle + |10100\rangle \\
 & + |01010\rangle - |11011\rangle - |00110\rangle - |11000\rangle \\
 & - |11101\rangle - |00011\rangle - |11110\rangle - |01111\rangle \\
 & - |10001\rangle - |01100\rangle - |10111\rangle + |00101\rangle] \\
 \\
 |1_L\rangle = \frac{1}{4} & [|11111\rangle + |01101\rangle + |10110\rangle + |01011\rangle \\
 & + |10101\rangle - |00100\rangle - |11001\rangle - |00111\rangle \\
 & - |00010\rangle - |11100\rangle - |00001\rangle - |10000\rangle \\
 & - |01110\rangle - |10011\rangle - |01000\rangle + |11010\rangle]
 \end{aligned}$$

It's generators are in table 1.1 on page 34 And the Recovery Table 1.2 on page 34.

1.5.2 Fault-Tolerant Quantum Computation

A perfect quantum code $[|n, k, d|]$ encodes k logical qubits into n physical qubits, and it can detect and correct errors in at most $\lfloor (d - 1)/2 \rfloor$ qubits. Let us look at a simple code, with $k = 1$: one logical qubit is encoded in n physical qubits. The 5-qubit code and the 7-qubit code are such codes, with $d = 3$. They can correct errors occurring in at most one physical qubit. The idea of Fault-Tolerant quantum computation is to design the gates acting on the quantum state, in a way that keeps the state in the code and assures that if one error occurred in the process of applying the gate, then we have at most one faulty physical qubit at the end, even if the gate creates entanglement. One can answer this challenge using two

Generator Table for the 5-Qubit Code	
g_1	$X_1Z_2Z_3X_4$
g_2	$X_2Z_3Z_4X_5$
g_3	$X_1X_3Z_4Z_5$
g_4	$Z_1X_2X_4Z_5$

Table 1.1: Generator Table for the 5-Qubit Code

Error Syndrome $\langle g_1, g_2, g_3, g_4 \rangle$	Recovery Operator
0000	I
0001	X_1
0010	Z_3
0011	X_5
0100	Z_5
0101	Z_2
0110	X_4
0111	Y_5
1000	X_2
1001	Z_4
1010	Z_1
1011	Y_1
1100	X_3
1101	Y_2
1110	Y_3
1111	Y_4

Table 1.2: Recovery Table for the 5-Qubit Code

approaches. One approach is to design the gates transversely (acting on each qubit independently), or in any other way that promises a fault tolerant application of the gate in one try [32]. The second approach is trying to know when errors propagate from one qubit to another, and if one such error did propagate – start over, again and again until you have a successful implementation of the gate with error in at most one physical qubit at the end. The second approach is relatively new, and is called ‘flag Fault-Tolerance’ [9] , because we use additional ‘flag’ qubits that indicate us if there was some uncorrectable error. Measuring the flags collapses the state to some subspace of the Hilbert space in which there was no error.

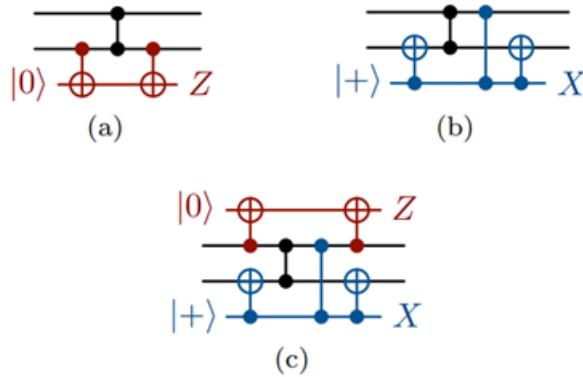


Figure 1.11: CZ gadgets to catch correlated faults. (a) An extra qubit can be used to catch XX, XY , YX and YY faults after the CZ gate. (b) A similar circuit catches ZZ faults. (c) In combination, these gadgets can catch all two-qubit correlated faults

Implementing the relevant gates fault-tolerantly in the 5-qubit code remains an open problem for future work. Our approach for this future work is to use the second approach (flag fault-tolerance) and a trick introduced by Chao and Reichardt [9]. They call this trick a *CZ gadget*: Adding

two flag qubits that indicate for correlated errors. Nielsen had proved [32] that any entangling gate can be implemented using only CNOTs and single qubit gates, and note that we can use the circuit identity $\hat{\phi} = \hat{H}_1 \otimes \hat{H}_2$ to use only CZs and single qubit gates. The CZ gadget is depicted in figure 1.11, and by Chao and Reichardt's theorem 1, it caches all single qubit correlated errors XX,XY,YX,YY,ZZ.

1.5.3 Quantum Error Correction for Quantum Metrology

Much effort has been made to recover the Heisenberg-limit scaling using quantum error correction [4, 8, 20, 30, 37, 48, Unden et al.]. All those efforts focus on sequential quantum metrology, encoding the sensor as a logical qubit and using smart ways to correct errors and not correcting the signal itself. In the next section we introduce our research question, giving a new approach of using quantum error correction for quantum metrology.

2 Research Question

In the field of quantum metrology, we are most interested in finding ways to recover the Heisenberg-limit scaling, promising that the error in estimating our observable scales as one over the number of measurements, probing time or number of probes [48]. Recent results indicate that this limit cannot be recovered in the presence of general Markovian noise if the Hamiltonian lies in the span of the noise operators.[29, 47]. Much effort has been made to recover the Heisenberg-limit scaling using quantum error correction [4, 8, 20, 30, 37, 48, Unden et al.]. All those efforts focus on sequential quantum metrology. Recently the use of algorithmic quantum sensing had caught the attention of the community [24, 40], and improving the performance of Quantum Phase Estimation (QPE) for quantum sensing can lead to breakthroughs in other fields, as a result of the algorithm being the basis for an enormous number of other uses [14, 33, Santagati et al., Kais], Shor's algorithm is one of them. Thus a lot of study has been done on how QPE performs under the presence of noise [11, 18, 19].

This Heisenberg-limit scaling is achievable if the probability distribution is not biased, according to [16]. In this work we see that this is not the case - in the presence of decoherence the probability distribution is biased, and

we give a way of making it less biased, thus approaching to the Heisenberg scaling in the number of measurements and probing time (which are aligned with each other in iterative quantum phase estimation) .

The existence of ancilla qubits opens a new realm to study - doing error correction or error detection on the ancillas instead of on the sensor. The will of making only the ancilla logical, while letting the sensor remain a physical sensor, enforces the need of hybrid logical-physical interaction, which is a new idea to our knowledge. Although we could focus on making this interaction fault-tolerant [Aharonov], for the purpose of quantum sensing we do not need the ability of computing endlessly - we only need one successful run. Thus it is possible to apply only error detection on the ancillas, and post selecting the results [5]. This way we essentially encode some of the noise in redundant degrees of freedom of the Hilbert space, enabling us to get rid of more of it. To our knowledge, we are the first ones approaching the problem this way.

3 Methods & Theoretical Derivations

3.1 n-Iterative Phase Estimation Algorithm

This section is a short side track from the main thrust of the thesis - it shows our idea for a slight modification to the IPEA, enabling it to measure few digits at a time. We call it the n-IPEA (see figure 3.1) and it is based on the textbook approach and on the IPEA. This algorithm gives a way to measure the next n digits of the phase, x_k, \dots, x_{k+n-1} , given the previous digits x_{k+n}, \dots, x_m where m is the desired precision. This version enables to measure a varying number of digits in each iteration, using $n + r$ qubits with r the number of sensor qubits.

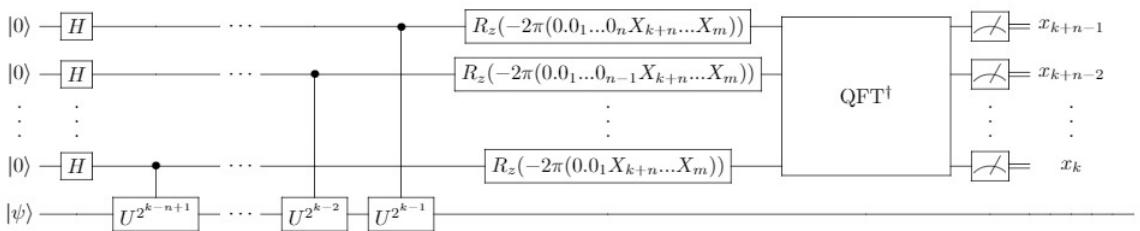


Figure 3.1: n-Iterative Phase Estimation Algorithm. The first few digits may be measured as described in the IPEA, with one ancilla.

The Idea is this: Suppose we want to measure the digits x_k, \dots, x_{k+n-1} , given x_{k+n}, \dots, x_m , when the eigenvalue of $|\psi\rangle$ with the operator U is $e^{2i\pi\phi}$. Using n ancilla qubits the system's state after the first Hadamards is

$$|\phi\rangle = \frac{1}{2^{n/2}}(|0\rangle + |1\rangle)^{\otimes n} \otimes |\psi\rangle$$

And after the controlled operators sequence, the state becomes

$$|\phi\rangle = \frac{1}{2^{n/2}}\left(\sum_{j=0}^{2^n-1} e^{2i\pi\phi j}|j\rangle\right) \otimes |\psi\rangle$$

Continuing the analysis for the simple $n = 2$ case, this is exactly

$$\frac{1}{2}(|00\rangle + e^{2i\pi(0.\phi_k\dots\phi_m)}|01\rangle + e^{2i\pi(0.\phi_{k+1}\dots\phi_m)}|10\rangle + e^{2i\pi(0.\phi_k\phi_{k+1}\dots\phi_m+0.\phi_{k+1}\dots\phi_m)}|11\rangle) \otimes |\psi\rangle$$

After the rotations, the state becomes

$$|\phi\rangle = \frac{1}{2}(|00\rangle + e^{2i\pi(0.\phi_k\phi_{k+1})}|01\rangle + e^{2i\pi(0.\phi_{k+1})}|10\rangle + e^{2i\pi(0.\phi_k\phi_{k+1}+0.\phi_{k+1})}|11\rangle) \otimes |\psi\rangle$$

which is exactly the system state for a textbook QPE before the reverse QFT stage. The generalization to n -digit case is trivial, and so we have a functional n-IPEA.

3.2 Iterative Phase Estimation Algorithm Under the Presence of Decoherence

As stated earlier, the motivation of my work is to improve sensing capabilities using quantum algorithms, specifically quantum phase estimation. Figure 1.10 shows the affect of noise on the histogram of all possible results. But QPE is not only use for sensing purposes - It can also be used for quantum computation scenarios, in which it is not the error that is

of interest, but the probability to measure the right phase after a single run. In such cases, where the sensing or computation time has a lower importance, it is convenient to have an oracle stating whether the result is reliable or not. Implementation of IPEA is possible both in two types of systems:

1. Ensemble systems, where it is convenient to apply the whole algorithm in parallel on a large number of qubits and measure result statistics.
2. Superconducting systems, where it is convenient to measure the digit in each iteration a number of times and take a majority vote.

Here we first explain how to calculate an upper bound to the error probability of the ensemble scenario, and then we show our simulated error probability for the superconducting scenario. In both cases the success probability is defined to be:

$$P_{\text{success}} = \sum_{\phi \in [\tilde{\theta}, \tilde{\theta} + 2^{-m}]} p(\phi) \quad (3.1)$$

Where θ is the measured phase, $\tilde{\theta}$ is the best m -bit approximation to it from below, and ϕ is a possible result of the algorithm. $p(\phi)$ is the probability to measure ϕ and $P_{\text{error}} = 1 - P_{\text{success}}$.

Assume an experiment contains n runs of the whole algorithm, and the result is taken to be a majority vote on the most probable interval. Then

$$\begin{aligned} P_{\text{error}} &\leq P(\text{less than } n/2 \text{ trials in the interval}) \\ &= P(0 \text{ successes in } n \text{ trials}) + P(1 \text{ successes in } n \text{ trials}) \\ &\quad + P(\lfloor n/2 \rfloor \text{ successes in } n \text{ trials}) \end{aligned}$$

Each of the probabilities in the above formula is distributed binomially, and so it is possible to exactly compute P_{error} [7].

Now assume the experiment contains n runs for each bit and the result for the bit is taken to be a majority vote. This has the largest impact only on the first (least significant) digits. There are only two possible results - 0 is measured or 1 is measured. Thus the probability to measure 0 is updated in our simulation to be the probability to measure 1 less than half of the times, and vice versa:

$$p(0) \leftarrow \sum_{k=0}^{\lfloor n/2 \rfloor} \binom{n}{k} p_1^k p_0^{n-k} \quad p(1) \leftarrow \sum_{k=0}^{\lfloor n/2 \rfloor} \binom{n}{k} p_0^k p_1^{n-k} \quad (3.2)$$

And the error probability is defined as in Eq.3.1.

The histogram of results for this case is given in figure 3.2, in analogy to the one given at 1.10.

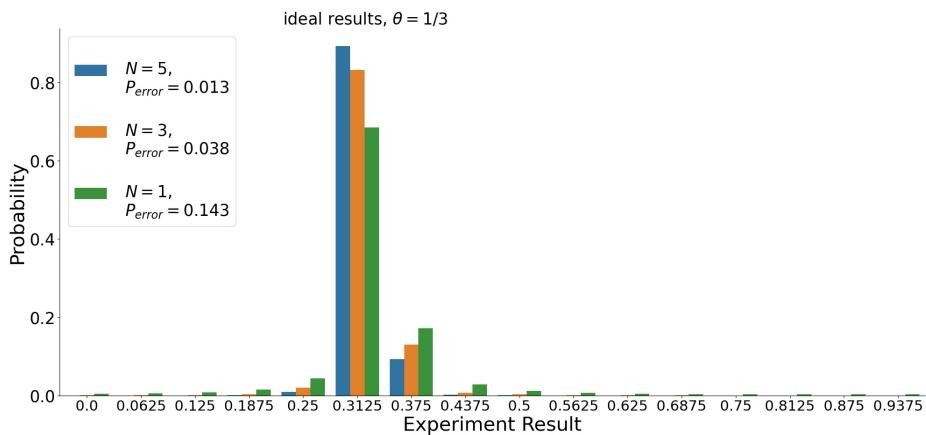


Figure 3.2: A showcase of the probability distribution of all possible results of IPEA, measuring $R_x(2\pi/3)$ up to 4 binary digits, while each digit is measured n times and the digit is decided by a majority vote. It is apparent that the probability of a precise measurement increases with n .

3.3 The Simulation

For a full documentation on the code package see appendix A.

In this Work, we use a full density matrix simulation, similar to Ref.[12]. We save the quantum state of n qubit register as a 2-d matrix of dimensions $[2^n, 2^n]$. Operators are saved as a $[2^n, 2^n]$ matrix, and if the quantum state was initially ρ then performing the operation U on the density matrix is equivalent to updating the density matrix $\rho \rightarrow U\rho U^\dagger$.

The noise in the simulation is based on Kraus operators (more of them in section 3.3.2). The simulation is thus made up of many small time steps, with repeated application of Kraus based decoherence in one small time-step and gate-based evolution in the next small time-step. A description of the simulation is given in figure 3.3 along with algorithm 2. In the algorithm, we take the generator of each unitary gate G_i to be H_i , such that $G_i = e^{i\frac{dt}{T}H_i}$.

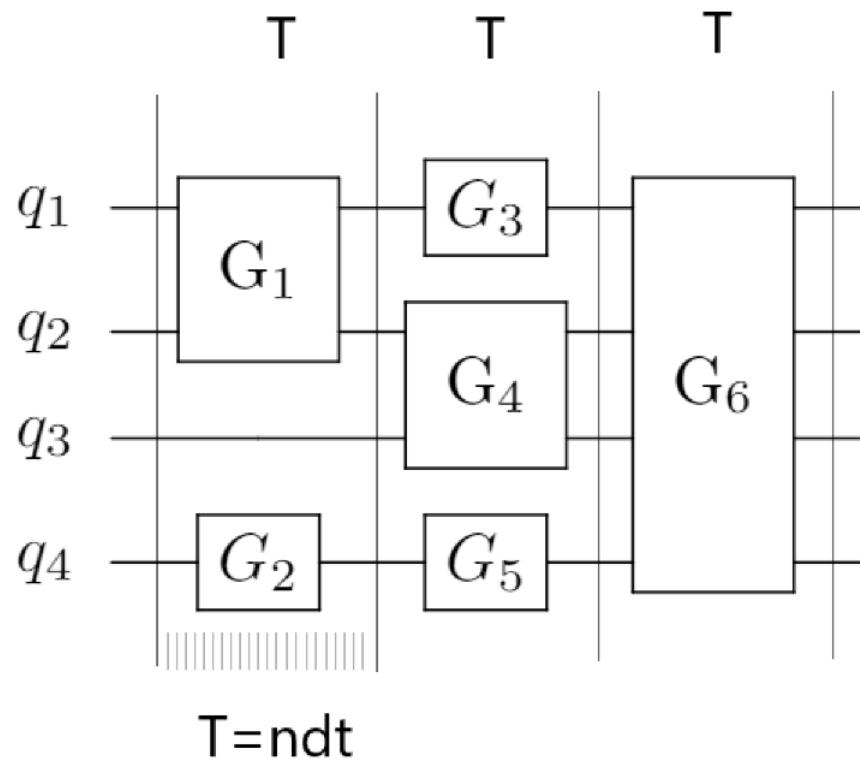


Figure 3.3: Simulation of a general circuit. Each gate-step T has length T_{gate} and is made up of $n = 20$ small time-steps of size T_{gate}/n .

Algorithm 2: Noisy Circuit Simulation

Result: Density matrix of the register after a noisy quantum circuit.

```

1 for gate-step  $T$  do
2    $U_{dt}^T = e^{i \frac{dt}{T} \sum_i H_i}$  ;
3   for  $t$  do
4      $\rho_{t+1} = U_{dt}^T \rho_t (U_{dt}^T)^\dagger$  ;
5     for qubit do
6       amplitude damping ;
7       dephasing ;
8     end
9   end
10 end

```

Main Parameters

The main parameters used in each simulation are the number of qubits N , the time $T_{gate} = n \cdot dt$ ($n = 20$) of each gate-step, the dephasing time of qubit q , T_2^q and the energy relaxation time T_1^q of the same qubit. From these parameters we define the error rates for each process and qubit:

$$p_{decay}^q = 1 - e^{-\frac{dt}{T_1^q}}, \text{ and } p_{dephase}^q = 1 - e^{-\frac{dt}{T_2^q}}$$

Defining the pauli operator σ_i^q acting on qubit q as a tensor product of σ_i in the q index and Identity operators in all other indexes, the base Hamiltonian $H_0 = \bigotimes_{q=1}^N \frac{\hbar\omega_{01}}{2} \sigma_z^q$ represents the free evolution of the quantum register, with $\omega_{01} = 2\pi \cdot 6$ [GHz]. In our research we work in the rotating frame, but simulation is also possible outside of the rotating frame with our code package.

3.3.1 Gate-Based Evolution

In each gate-step, possibly many gates act upon the register. Thus, we start with the base Hamiltonian $H = H_0$ and for each gate G in the gate-step we find its corresponding Hamiltonian H_G using table 3.1 and update the Hamiltonian to be $H \rightarrow H + H_G$.

Now, we define the evolution operator U to be $e^{iH\frac{dt}{T_{gate}}}$, and we apply this evolution as in step 4 of algorithm 2 for a total of T_{gate}/dt times with decoherence step between each application of U .

Gate	Hamiltonian
P^q	$-\frac{\pi}{2}\sigma_P^q$
$R_P^q(\theta)$	$-\frac{\theta}{2}\sigma_P^q$
H^q	$\frac{\pi}{2} \cdot \frac{\sigma_X^q + \sigma_Z^q}{\sqrt{2}}$
$CNOT(q_1, q_2)$ - q_2 is control	$\ln\left(\frac{1}{2} \cdot ((I - \sigma_Z^{q_2})\sigma_X^{q_1} + I + \sigma_Z^{q_2})\right)$
$CZ(q_1, q_2)$ - q_2 is control	$\ln\left(\frac{1}{2} \cdot ((I - \sigma_Z^{q_2})\sigma_Z^{q_1} + I + \sigma_Z^{q_2})\right)$
$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ - single qubit (q) gate	$\ln\left(\frac{a+d}{2}I + \frac{a-d}{2}\sigma_Z^q + \frac{c+b}{2}\sigma_X^q + \frac{c-b}{2i}\sigma_Y^q\right)$

Table 3.1: Gate and Gate Hamiltonian table for the simulation, with $P \in \{X, Y, Z\}$

3.3.2 Kraus-Based Decoherence

In each small time-step dt we do the Kraus-based decoherence on every qubit in the register. There are three options:

- We are given only decay time T_1^q , assuming the pure dephasing time $(T_2^q)^*$ is infinite, for each qubit q .

- We are given only pure dephasing time T_2^q , assuming the decay time T_1^q is infinite, for each qubit q .
- We are given both the decay time T_1 and the pure dephasing time T_2 for each qubit q .

Using the relation between the decay time T_1 , the dephasing time T_2 and the pure dephasing time T_2^* , $T_2 = (\frac{1}{2T_1} + \frac{1}{T_2^*})^{-1}$, we can see that if the pure dephasing time is assumed to be infinite then we have $T_2 = 2T_1$. This is embedded in the amplitude damping process, with it's Kraus operators.

So, for the second option (given T_2^q (pure dephasing time) assuming T_1^q is infinite) we iterate over the qubit q and update the register state to be:

$$\rho \rightarrow (1 - \frac{P_{dephase}^q}{2})\rho + \frac{P_{dephase}^q}{2}\sigma_Z^q\rho\sigma_Z^q$$

This is true because if we take the Kraus operators of the phase damping (=phase flip) channel, $K_1 = \sqrt{p}(\begin{smallmatrix} 1 & 0 \\ 0 & 0 \end{smallmatrix})$, $K_2 = \sqrt{p}(\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix})$, $K_3 = \sqrt{1-p}(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix})$. We get $\sum_i K_i \rho K_i^\dagger = (1 - \frac{p}{2})\rho + \frac{p}{2}\sigma_Z\rho\sigma_Z$.

For the first option (given T_1^q assuming $(T_2^q)^*$ is infinite) we define the Kraus operators of the amplitude damping channel $M_1 = (\begin{smallmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{smallmatrix})$, $M_2 = (\begin{smallmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{smallmatrix})$. Which, expressed by pauli operators, look like

$$M_1^q = \frac{\sqrt{1-P_{decay}^q}}{2}(I - \sigma_Z^q) + \frac{1}{2}(I + \sigma_Z^q),$$

$$M_2^q = \frac{\sqrt{P_{decay}^q}}{2}(\sigma_X^q + i\sigma_Y^q)$$

And we iterate over the qubit q , updating the register to be:

$$\rho = M_1^q \rho (M_1^q)^\dagger + M_2^q \rho (M_2^q)^\dagger$$

Finally, for the third option - given both pure dephasing rate and energy

relaxation rate, we update the state of the register as follows:

$$\rho_{temp} = M_1^q \rho (M_1^q)^\dagger + M_2^q \rho (M_2^q)^\dagger$$

$$\rho = (1 - \frac{P_{dephase}^q}{2})\rho_{temp} + \frac{P_{dephase}^q}{2}\sigma_Z^q \rho_{temp} \sigma_Z^q$$

3.3.3 Measurements

There are two kinds of measurements we perform - post selection measurements and probabilistic measurements. Here, we first refer to the post selection measurements.

Post selection measurements are done on the sensor qubit for SPS, on flag qubits for fault-tolerance, and once again on the sensor qubit (that acts as an ancilla qubit for syndrome measurement) for LPS. To perform post selection measurements, we collapse the register state as defined below according to the preferred measurement outcomes. One could say we choose the system's trajectory.

First, to add flag qubits to the simulation, we expand the register state with a tensor product to the two additional qubit sub-spaces. Next, we perform the entangling operations with the flags, and finally project on the trivial flag state $|0..0\rangle$ using the operator defined below.

For probabilistic measurements (e.g. Error correction measurements), to decide measurement outcome on the qubit group $A = \{q_{k_1}, \dots, q_{k_n}\}$, renaming with $B = \{q_1, \dots, q_N\}/A$, we trace out B to get $\rho^A = Tr_B(\rho)$. Then, we define P as the diagonal of ρ^A and P' as the cumulative sum of P. we take a random number $0 < x < 1$ and find the first index i such that $x < P'[i]$. The result of the measurement is the binary string of $i - 1$.

To collapse the quantum state to a state after measuring qubits in the group A, we use the following projector:

$$P_m = \bigotimes_{q=1}^N \begin{cases} I_2 & \text{if } q \notin A \\ \frac{I+\sigma_Z^q}{2} & \text{if } q \in A \text{ and measurement result is } |0\rangle \\ \frac{I-\sigma_Z^q}{2} & \text{if } q \in A \text{ and measurement result is } |1\rangle \end{cases}$$

And after this projection operation $\rho \rightarrow P_m \rho P_m^\dagger$ we trace out the flag qubits.

The procedure described above can cause numeric errors when the state decohere for a long time, because the projection operation as described is not trace preserving. To have a valid density matrix, for each projection, say the k 'th projection, we first save the state's trace as Ps_k and then normalize the state. To calculate the portion of information that have been lost due to post selection, we use the following reasoning:

- after one projection, we have lost $1 - Ps_1$ information and remain with a state with trace Ps_1 .
- after the second projection, we have lost $Ps_1 \cdot (1 - Ps_2)$ more information.
- after the third projection, we have lost $Ps_1 \cdot Ps_2 \cdot (1 - Ps_3)$ more information.
- after the k 'th projection, we have lost $Ps_1 \cdot \dots \cdot Ps_{k-1} \cdot (1 - Ps_k)$ more information.

Overall, this is the amount of *lost information*:

$$l_i = \sum_i \left(\left(\prod_{k=1}^{i-1} Ps_k \right) \cdot (1 - Ps_i) \right) \quad (3.3)$$

3.4 Sensor Post Selection

Up until now we have introduced general background, and discussed technical details of the simulation. In this section and the next two we discuss more thoroughly the core concept we use to improve algorithmic quantum metrology - Post Selection (PS). Post selection is a technique used for coping with errors, in which we choose to work only with the portion of the data that fits our expectations. According to ref.[5], post-selection offers a non-classical advantage in metrology. We begin by presenting the possible advantages of performing post selection on the sensor qubit, or as we call it - Sensor Post Selection (SPS). We then discuss in detail the simulation done for checking SPS performance, in the theoretic circuit-level.

3.4.1 The Potential of Sensor Post Selection

Suppose we have the measured operator $U = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$ with eigenvectors $|u\rangle$ and $|v\rangle$, 1 being the eigenvalue of $|u\rangle$. Define some error in U by $U \rightarrow U + \vec{\Delta} \cdot \vec{\sigma} \stackrel{\text{def}}{=} U'$ with $\vec{\sigma} = (\sigma_x, \sigma_y, \sigma_z)$. In this case, we have:

$$U'|u\rangle = (e^{i\theta} - \Delta_z)|u\rangle + (\Delta_x - i\Delta_y)|v\rangle$$

Measuring the sensor qubit and post selecting the measured eigenstate, we can thus eradicate Δ_x and Δ_y errors. Since Δ_z is defined in the eigenbasis of the operator, it translates to a rotation with some different angle θ - this is noise in the signal. We accept this error Δ_z as a part of U that we do not strive to correct.

Perhaps the biggest challenge in QPE algorithms is the high power of the measured operator. Doing the naive thing and applying the operator again

again is not practical, because there exists a precision for which we will have to apply the operator for times larger than the qubit's decoherence time. To overcome this challenge one can think of experimental ways (for example, applying a measured magnetic field with different angles) or use VFF [13] as a mean to accelerate the Hamiltonian. Those methods can cause constant errors in U . Now suppose Δ_z is some constant systematic error. Take again the same faulty $U \rightarrow U + \vec{\Delta} \cdot \vec{\sigma} \stackrel{\text{def}}{=} U'$, and suppose the ideal operator has two eigenvalues λ_0 for $|0\rangle$ and λ_1 for $|1\rangle$. We have the two equations:

$$U'|1\rangle = (\lambda_1 - \Delta_z)|1\rangle + (\Delta_x - i\Delta_y)|0\rangle$$

$$U'|0\rangle = (\lambda_0 + \Delta_z)|0\rangle + (\Delta_x + i\Delta_y)|1\rangle$$

Doing QPE and post selecting the sensor for each one, we get the results $\tilde{\lambda}_1, \tilde{\lambda}_0$ and we can use the following relations to extract the real eigenvalues:

$$\tilde{\lambda}_1 + \tilde{\lambda}_0 = \lambda_1 + \lambda_0$$

$$2 = \lambda_1^2 + \lambda_0^2$$

Where the second equation is a result of normalizing $U \otimes U|+\rangle$.

3.4.2 SPS Simulation Methods

In this work, we check the affect SPS has on a noisy register running the Kitaev Iterative Phase Estimation 1.4.3, with $T_2/T_{gate} = 10^{15} \approx \infty$ on the two qubits and $T_1^{ancilla}/T_{gate} = 10^{15} \approx \infty$ on the control qubit. Since it is interesting to find the efficiency of SPS for the worst case possible we put the sensor in the excited eigenstate of R_z - $|1\rangle$ and allow it to have bit-flip errors. Meaning - We assume only T_1 relaxation on the sensor qubit, and

change the value T_1^{sensor} / T_{gate} in the range $[10, 10^3]$. This translates to the worst case single gate fidelity as described in 4.1.4 by applying the identity gate on the excited state and calculating the fidelity of the result with the excited state. We average the results on 10 different evenly spaced phases ϕ in the range $[0, 2\pi)$.

We compare three algorithms:

- *Traditional* - The simple, two-qubit Kitaev Iterative Phase Estimation as in figure 1.8, without acceleration.
- *Single SPS* - The simple, two-qubit Kitaev Iterative Phase Estimation, with Sensor Post Selection after all of the applications of the controlled operator, as in figure 3.4 (a).
- *Multiple SPS* - The simple, two-qubit Kitaev Iterative QPE, with Sensor Post Selection after the each application of the controlled operator, as in figure 3.4 (b).

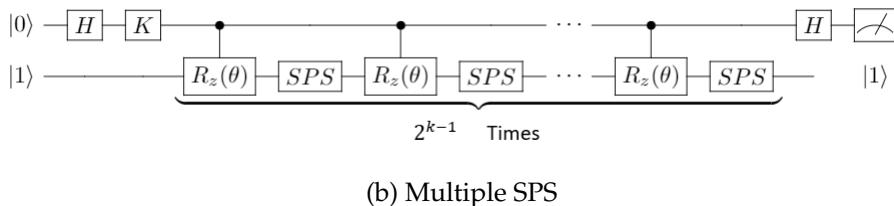
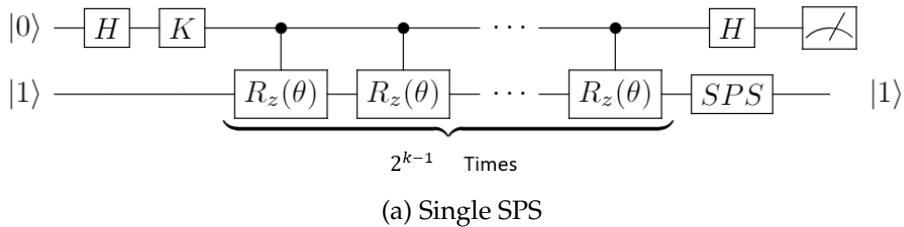


Figure 3.4: Kitaev Iterative QPE as in figure 1.8, without operator acceleration. (a) Single Sensor Post Selection, (b) Multiple Sensor Post Selection.

3.5 Logical Post Selection

In the previous section we have introduced the idea of using post selection on the sensor to improve sensing sensitivity. In this section we define the notion of Logical Post Selection (LPS), discuss the scaling of the error probability after performing LPS and develop insights on the subject of hybrid logical-physical entanglement. In the next section we describe briefly the implementation of logical post selection for quantum metrology.

Here we develop a new point of view on post selection. In the act of sensor post selection, we select only the fraction of the data that indicates that the sensor was in a certain state before the measurement. One could say we have selected to work only in a subspace of the Hilbert space, by projecting the system's state to the subspace defined by the sensor being in the initial eigenstate. One could further depict the action of post selection as encoding some of the noise in redundant degrees of freedom of the Hilbert space, in orthogonal subspaces to the one we work in, and throw this noise away by the act of post selection, thus throwing some of the information collected.

Intuitively, encoding the noise in a bigger redundant Hilbert space should result in more noise being separated from the actual 'pure' data. This can be done by using an error-detection or an error-correction code, and instead of correcting the noise we can just detect it and post select data fitting our desires. Although we could focus on making the operations fault-tolerant [Aharonov], or doing error correction, for the purpose of quantum sensing (and a variety of other purposes like state distillation and error mitigation [42, Czarnik et al., Piveteau et al.]) we do not need the ability

of computing endlessly - we only need one successful run. Thus it is possible to apply only error detection on the ancillas, and post selecting the results [5].

Another incentive to use error detection instead of error correction is graphically depicted in figure 4.14 of the Results Chapter. This could be explained as follows: For small gate fidelity the act of error correction can spoil the results, and sometimes destroy them entirely. The reason for that is in table 3.2. In the realm of small gate fidelity and non fault-tolerant gates, the assumption that we have at most one faulty qubit is invalid, and we have to assume it is possible to also have two faulty qubits with some, non-negligible probability. In this case, we do neglect the possibility of three faulty qubits.

We define logical post selection by the act of measuring the logical state's error syndrome, and post selecting only results with the trivial syndrome. In the case of the 5-qubit code, that would be the syndrome '0000'.

In this work we use the simple 5-qubit-code as our error detection (correction) code, due to two interesting attributes it exhibits:

- LPS leads to an improved scaling of the error probability, as described in the next subsection 3.5.1.
- It enables an easy logical-physical interaction, as described in the following subsection 3.5.2 .

3.5.1 Scaling of the Error Probability

In this subsection we dive deeper into understanding the notion of logical post selection, and analyse the new scaling of the error probability.

The incentive of developing a deeper understanding appears in table 3.2, where we can see that errors in two or less qubits will result in measuring a non-trivial syndrome, enabling the detection of such errors. Errors in 3 and more qubits can result in measuring the trivial syndrome, making it impossible to detect some of them. In our error analysis we follow the explanation by Nielsen and Choung,[32] (Chap.10).

Here, we present a simple error analysis of LPS for the five qubit code. Suppose the probability of an error in one faulty qubit is p , then the probability to have at most two faulty qubit is $(1 - p)^5 + 5p(1 - p)^4 + 10p^2(1 - p)^3$, and so the probability that there was an error and we couldn't recognize it is $10p^3 - 15p^4 + 6p^5$. To improve the results using LPS we have to require $10p^3 - 15p^4 + 6p^5 < p$ which gives us, as expected, $0 \leq p \leq \frac{1}{2}$.

Now, for a slightly better analysis. We assume the depolarising channel with probability p acts on the state, giving

$$\rho \rightarrow (1 - p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z) \quad (3.4)$$

For a simple one physical qubit case, taking a pure state $\rho = |\psi\rangle\langle\psi|$, we get process fidelity of

$$F = \sqrt{\langle\psi|\rho|\psi\rangle} = \sqrt{(1 - p) + \frac{p}{3}[\langle\psi|X|\psi\rangle^2 + \langle\psi|Y|\psi\rangle^2 + \langle\psi|Z|\psi\rangle^2]}$$

This expression gets the lowest fidelity for $|\psi\rangle = |0\rangle$, with:

$$F = \sqrt{1 - \frac{2}{3}p} = 1 - \frac{p}{3} + O(p^2)$$

Now, for the logical qubit. Assume we encode one qubit of information into n physical qubits, each goes through a depolarizing channel ϵ with probability p , as in equation 3.4. Then the channel's action on a state ρ becomes:

$$\epsilon^{\otimes n}(\rho) = (1 - p)^n\rho + \sum_{j=1}^n \sum_{k=1}^3 (1 - p)^{n-1} \frac{p}{3} \sigma_k^j \rho \sigma_k^j$$

$$+ \sum_{j_1=1}^n \sum_{j_2=1}^n \sum_{k_1=1}^3 \sum_{k_2=1}^3 (1-p)^{n-2} \frac{p^2}{9} \sigma_{k_1}^{j_1} \sigma_{k_2}^{j_2} \rho \sigma_{k_2}^{j_2} \sigma_{k_1}^{j_1} + \dots$$

With σ_k^j being the k 'th pauli operator acting on the j 'th qubit. The first element represents one faulty qubit and the second represents two faulty qubits, and the dots represent errors in more than 2 qubits. Now, after performing LPS, each element in this sum will be returned to the state ρ given ρ was in the code:

$$(R \otimes \varepsilon^{\otimes n})(\rho) = [(1-p)^n \rho + np(1-p)^{n-1} + \binom{n}{2} p^2 (1-p)^{n-2}] \rho$$

And finally, the fidelity F remains:

$$\begin{aligned} F &\geq \sqrt{(1-p)^{n-2}(1 + (n-2)p + (\frac{n(n-1)}{2} - n + 1)p^2)} \\ &= 1 - \frac{1}{12}n(n^2 - 3n + 2)p^3 + O(p^4) \end{aligned} \quad (3.5)$$

Giving a p^3 dependence and confirming our intuition from table 3.2.

Error Syndrome $\langle g_1, g_2, g_3, g_4 \rangle$	Possible Cause
0000	IIII
0001	XIII, IYYYI, IIZIX, IXIZI, IYXII, IZIIZ, IIIXY
0010	IIIIX, IIXYI, IYYII, IZIXI, XIZII, YXIII, ZIIIZ
0011	IIZII, IIIXZ, IITYI, IZIY, XIII, YYIII, ZXIII
0100	IIIXI, IIZIZ, IXYII, IZIIX, XIIIY, YYIII, ZIXII
0101	IIIY, IIYZI, IXYI, IZZII, XIIXI, YIXII, ZYIII
0110	IZIII, IIIXX, IIXZI, IIIZY, XIII, YIYII, ZIIYI
0111	IIIIZ, IIZXI, IXXII, IYIZI, XZIII, YIIYI, ZIYII
1000	IIXII, IIIYX, IXIIZ, IZIZI, XYIII, YIIIY, ZIIXI
1001	IYIII, IIIZZ, IYYIX, IIZYI, XIXII, YIIXI, ZIIIY
1010	IIYI, IIXIX, IXIY, IYZII, XIYII, YIII, ZZIII
1011	IYII, IIIZY, IXIXI, IYIIX, XIYI, YZIII, ZIII
1100	ZIII, IIIZX, IXXXI, IIYIZ, IXZII, IYIIY, IZIYI
1101	YIII, IIIYZ, IIXIY, IIIZI, XIIX, IYIXI, IZYII
1110	IIIIZ, IIYT, IYIIZ, IZXII, XXIII, YIZII, ZIII
1111	IXIII, IIIYY, IIXIZ, IIYXI, XIIZI, YIII, ZIZII

Table 3.2: Error syndrome and possible cause for the 5 qubit code. Causes with errors in more than 2 qubits are neglected. It is apparent that every possible two-qubit error results in a non-trivial syndrome, enabling the detection of such errors. This means that if the probability of error in one qubit is p , then the probability of error after error detection is proportional to p^3 at the lowest power.

3.5.2 Hybrid Logical-Physical Entanglement

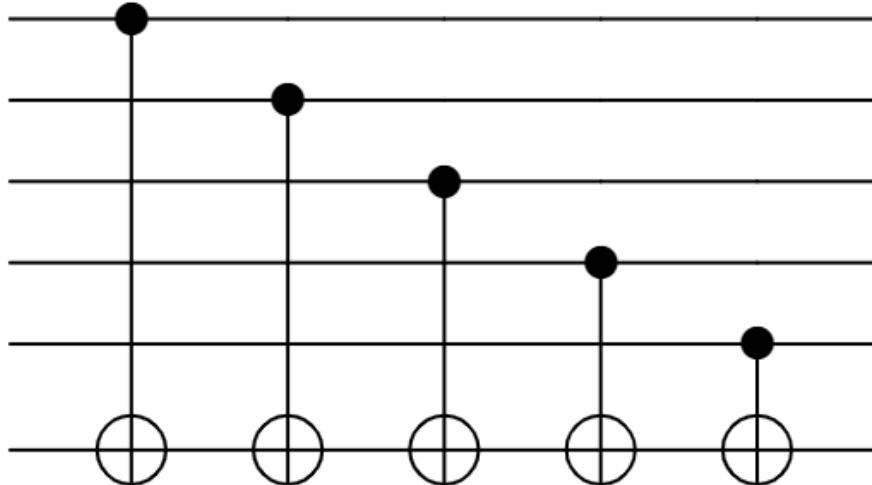
As explained earlier, some of the noise can be encoded in redundant degrees of freedom. We hope to achieve this encoding by using a logical qubit interacting with a physical qubit, and describing this interaction is what we do in this subsection. Assumptions on the noise vulnerability of both qubits, logical and physical, will be made throughout the work, and we will make sure to point out our assumptions in each stage.

We start by remarking that any multi-qubit gate can be decomposed into single qubit gates and CNOT gates [32], and so our focus should be on understanding how to implement the CNOT gate between a logical and physical qubits, as control and target respectively. Some QECC have a nice attribute: The logical ground state, $|0_L\rangle$, is made up of a sum of quantum states with even number of 1's, and the logical excited state, $|1_L\rangle$, is made up of a sum of quantum states with odd number of 1's, as for example, for the 5-qubit code, in figure 3.5 (a). In the case of one logical layer, this attribute allows us to implement the CNOT gate in a semi-transversal manner, as in figure 3.5 (b). This can be generalized trivially for any number of logical layers, provided that the quantum code used for each layer has this attribute.

We have thus given a way of implementing logical-physical qubit interaction for the five qubit code, used in this work. In the case of quantum phase estimation implementation, the first five qubits in figure 3.5 (b) are the ancilla qubits composing the logical ancilla, and the sixth qubit (first from the bottom) is the sensor.

$$\begin{aligned}
 |0_L\rangle = \frac{1}{4} & \left[|00000\rangle + |10010\rangle + |01001\rangle + |10100\rangle \right. \\
 & + |01010\rangle - |11011\rangle - |00110\rangle - |11000\rangle \\
 & - |11101\rangle - |00011\rangle - |11110\rangle - |01111\rangle \\
 & \left. - |10001\rangle - |01100\rangle - |10111\rangle + |00101\rangle \right] \\
 |1_L\rangle = \frac{1}{4} & \left[|11111\rangle + |01101\rangle + |10110\rangle + |01011\rangle \right. \\
 & + |10101\rangle - |00100\rangle - |11001\rangle - |00111\rangle \\
 & - |00010\rangle - |11100\rangle - |00001\rangle - |10000\rangle \\
 & \left. - |01110\rangle - |10011\rangle - |01000\rangle + |11010\rangle \right]
 \end{aligned}$$

(a) 5-Qubit Code Basis States



(b) Logically Controlled CNOT

Figure 3.5: (a) The logical basis states for the 5-qubit code. This code has the aforementioned attribute, allowing the application of a logically-controlled CNOT gate (b) with the first 5 qubits acting as a logical qubit and the sixth qubit as the target qubit.

Logical-Physical Interaction - Simulation Method

As mentioned earlier, each multi-qubit interaction can be reduced to single qubit gates and simple CNOT or CZ entangling gates. Thus it is interesting to ask how does the threshold value (in which logical control is better than physical control) depend on the number of entangling gates? To answer this question, we take $T_1/T_{gate} = \infty$ for all qubits, and a varying T_2/T_{gate} for all qubits. We also take a varying number of CNOT gates, in the range [1,200]. We start with the initial state $|++\rangle$ which is an eigenstate of the CNOT gate, where the first + in the tensor product may represent the logical qubit. For a specific number N_{gates} of CNOT gates, we apply N_{gates} CNOT gates and then apply noisy LPS. We calculate and save the fidelity of the output state and $|++\rangle$, and plot a three dimensional graph (figure 4.11) where one axis is the number of entangling gates, the second axis is the worst case single gate fidelity (extracted from figure 4.6), and the third is the fidelity difference between the logical and the traditional control. Note that we apply noisy syndrome extraction, so the circuit depth has a constant overhead of approximately 20 gates.

The fidelity is calculated as follows: The ideal (noiseless) state ρ_i , right before measurement, is a two qubit matrix (4x4). We define the logical state for a logical register with one sensor (6 qubits) at state ρ' as:

$$\rho = \begin{pmatrix} \langle 0_L 0 | \rho' | 0_L 0 \rangle & \langle 0_L 0 | \rho' | 0_L 1 \rangle & \langle 0_L 0 | \rho' | 1_L 0 \rangle & \langle 0_L 0 | \rho' | 1_L 1 \rangle \\ \langle 0_L 1 | \rho' | 0_L 0 \rangle & \langle 0_L 1 | \rho' | 0_L 1 \rangle & \langle 0_L 1 | \rho' | 1_L 0 \rangle & \langle 0_L 1 | \rho' | 1_L 1 \rangle \\ \langle 1_L 0 | \rho' | 0_L 0 \rangle & \langle 1_L 0 | \rho' | 0_L 1 \rangle & \langle 1_L 0 | \rho' | 1_L 0 \rangle & \langle 1_L 0 | \rho' | 1_L 1 \rangle \\ \langle 1_L 1 | \rho' | 0_L 0 \rangle & \langle 1_L 1 | \rho' | 0_L 1 \rangle & \langle 1_L 1 | \rho' | 1_L 0 \rangle & \langle 1_L 1 | \rho' | 1_L 1 \rangle \end{pmatrix} \quad (3.6)$$

Fidelity was calculated in comparison to the ideal state, $F(\rho_i, \rho) = \text{Tr}(\sqrt{\sqrt{\rho}\rho_i\sqrt{\rho}})$. Note that this is not necessarily a pure state. Performing LPS in our simu-

lation is done by projecting the state onto the code, forcing each stabilizer to measure '0'. In addition, when we perform error correction and the resulting state is not within the code, the reduction operation of Eq. 3.6 is not trace preserving. Due to the projective nature of these operations, to calculate the fidelity or distance between two states, where one of them is not a pure density matrix, we save the trace of each one and normalize them before hand. We define the lost information to be the fraction of information lost due to post selection, $l_i = 1 - \text{Tr}(\rho')$ where ρ' is the reduced 2-qubit density matrix defined in Eq.3.6. We quantify the noise by worst-case single gate fidelity, calculated by putting a qubit in it's most susceptible state to the applied noise - for example, the state $|+\rangle$ for dephasing and $|1\rangle$ for amplitude damping.

3.6 Logical Post Selection for Quantum Metrology

In the previous section we introduced the idea of Logical Post Selection and gave a way to implement an entangling gate between a physical sensor qubit and a logical ancilla qubit. In this section we demonstrate the use of LPS and this hybrid interaction for the purpose of algorithmic quantum sensing.

The field of error correction for quantum metrology is a new, exciting growing field, which suffers one major issue: It is hard to correct noise in the sensed signal without 'correcting' the signal itself. We offer an approach to possibly overcome some of that noise by encoding the ancilla instead of the sensor, letting some of the noise from the sensor be propa-

gated to the ancilla, and using post selection to purify the system's state.

In the following we focus on two specific QPE algorithms: The Iterative Phase Estimation Algorithm (IPEA) 5 and Kitaev's Iterative Phase Estimation 1.4.3.

3.6.1 Gate Implementations

Looking at the gate sequence defining each of the above algorithms, we can easily deduce our focus should be on implementing the following set of gates or procedures:

- Logical state preparation.
- Logical Hadamard gate.
- Logically-controlled operator. In this work we focus, for simplicity, on single qubit Z and X rotations, $R_z(\theta)$ and $R_x(\theta)$ respectively.
- Logical $R_z(\theta)$ rotation for phase kicks.
- Syndrome extraction procedures.

We will focus on the non-fault-tolerant implementations of these gates, and offer directions for making some of them fault-tolerant, meaning - implement them in such a way that errors will not spread throughout the circuit, possibly by using some more post selection.

Logical State Preparation

First we need to prepare the logical $|0\rangle$ state. State preparation can be done by first preparing the $|+\rangle^{\otimes 5}$ state by applying a transverse Hadamard gate,

and then continue following the procedure depicted in figure 3.6 (a). Note that the procedure is taken from Chao and Reichardt [9] with small adjustments. For fault-tolerant state preparation, insert flags as in 3.6 (b). Under the assumption that ancilla qubits are not perfect and have all the same decoherence and relaxation times, a simulation of the two methods was done and depicted in figure 3.4 on page 52, and it is clear that in the realm of small gate fidelity it is better to use the non fault-tolerant method. For perfect ancilla qubits we also use the non fault-tolerant method, for faster simulation times.

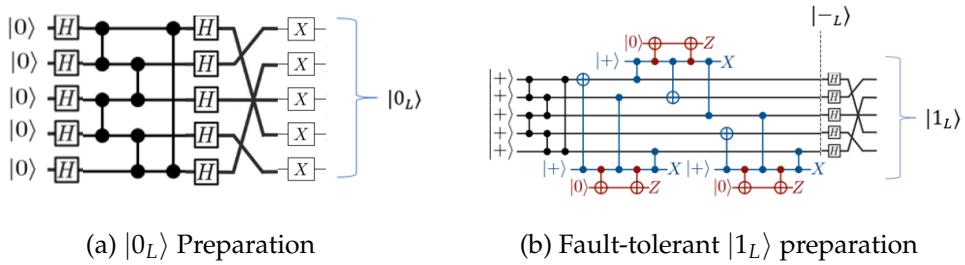


Figure 3.6: State preparation for the 5-qubit code. (a) Prepare the logical 5-qubit code ground state as appearing in section 1.5.1. (b) Flag fault tolerant preparation of the logical 5-qubit code excited state as appearing in section 1.5.1. logical ground state may be achieved by applying transverse X gate.

Logical Hadamard

The Hadamard gate can be implemented as depicted in figure 3.7 [45]. It is a transversal application of Hadamard followed by simple qubit permutation. If the permutation is done by relabelling the qubits, the logical Hadamard is transversal and thus automatically fault tolerant.

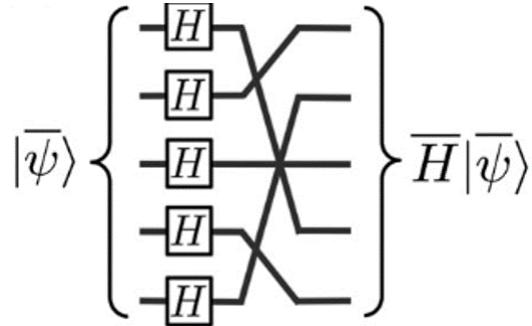


Figure 3.7: Logical Hadamard

Logically Controlled Operator

In the case of interest, where the measured operator is a single qubit operator, the logically controlled operator can be realized as a series of CNOTs and single qubit gates, as explained by Nielsen [32]. We focus for simplicity on single qubit Z and X rotations, $R_z(\theta)$ and $R_x(\theta)$ respectively. We denote them $R_p(\theta)$ with $p \in \{X, Z\}$. The implementation of controlled $R_p(\theta)$ rotation is depicted in figure 3.8. We have given here a symmetric representation of the rotation to fight error echoing.

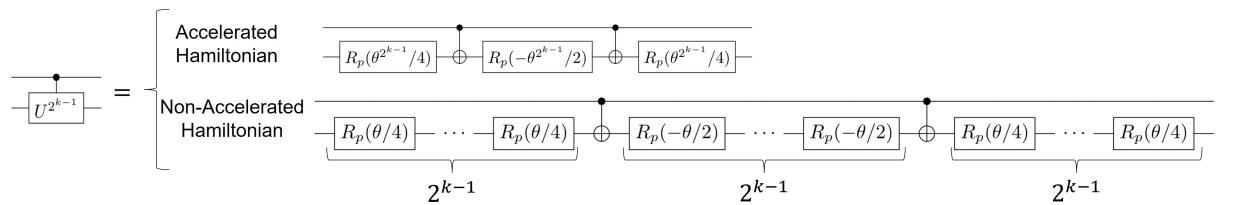


Figure 3.8: The controlled operation in this work is a single qubit rotation gate where $p = x$ or $p = z$. In the figure are implementations of accelerated and non-accelerated controlled signal Hamiltonians.

The CNOT gate can be traditional or logical. In figure 3.9 I give a way to implement the logically controlled CNOT gadget. Whether or not it is

possible to make this gadget fault-tolerant (Perhaps with using Flag Fault-Tolerance?) remains an interesting question for future work, although we give here a possible **untested** implementation using flag fault-tolerance. The idea is to detect possible errors propagating from the sensor to each ancilla qubit - see figure 3.10.

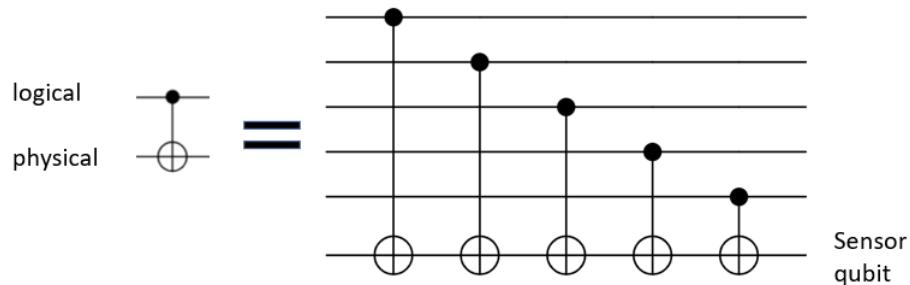


Figure 3.9: non-Fault-Tolerant Logically Controlled CNOT Gate. Errors can propagate through qubits undetected.

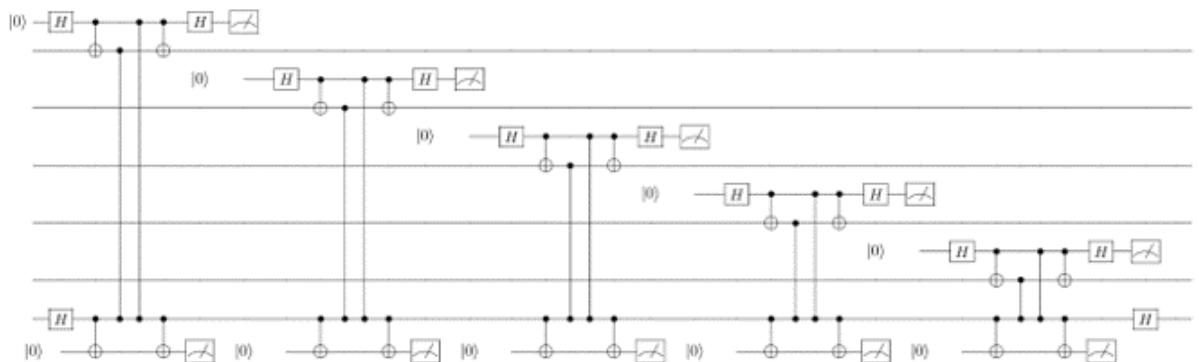


Figure 3.10: Fault-Tolerant Logically Controlled CNOT Gate. Errors can propagate through qubits, but a portion of them will be detected.

Logical Z rotation

Yoder *et al.* has shown that the logical Z rotation may be implemented according to figure 4.12(a) [45]. Note that we use $K = SH$. We give here two possible **untested in QPE** implementations using flag fault-tolerance. The implementations were tested and found to improve fidelity of the whole gate when all qubits (including the flag qubits) decohere but at the cost of a large amount of lost information.

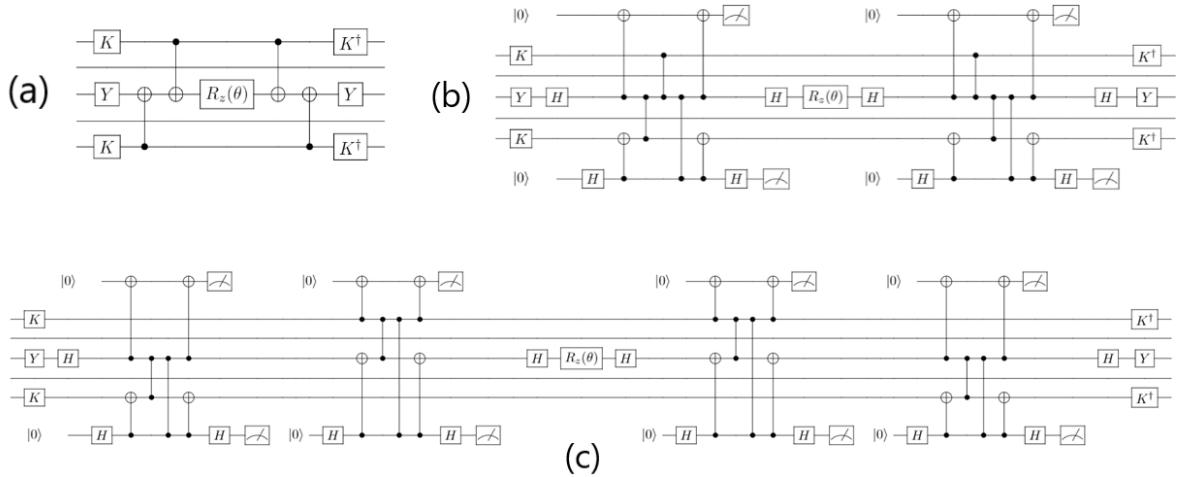


Figure 3.11: Logical $R_z(\theta)$ implementation for the five qubit code (a) by Yoder *et al.* Note that we use $K = SH$. (b) Improved flag-fault-tolerant implementation. (c) Naive flag fault-tolerant implementation.

Syndrome Measurement

First, note that state measurement can be done by measuring the logical Z operator, which serves as a parity check. Syndrome measurement was simulated non fault-tolerantly. Chao and Reichardt [9] give ways of mea-

suring syndromes fault tolerantly using flags. the syndromes were extracted as in figure 3.12. Note that we use the sensor qubit as a measurement qubit, and so before the syndrome extraction we need to reset it to the ground state.

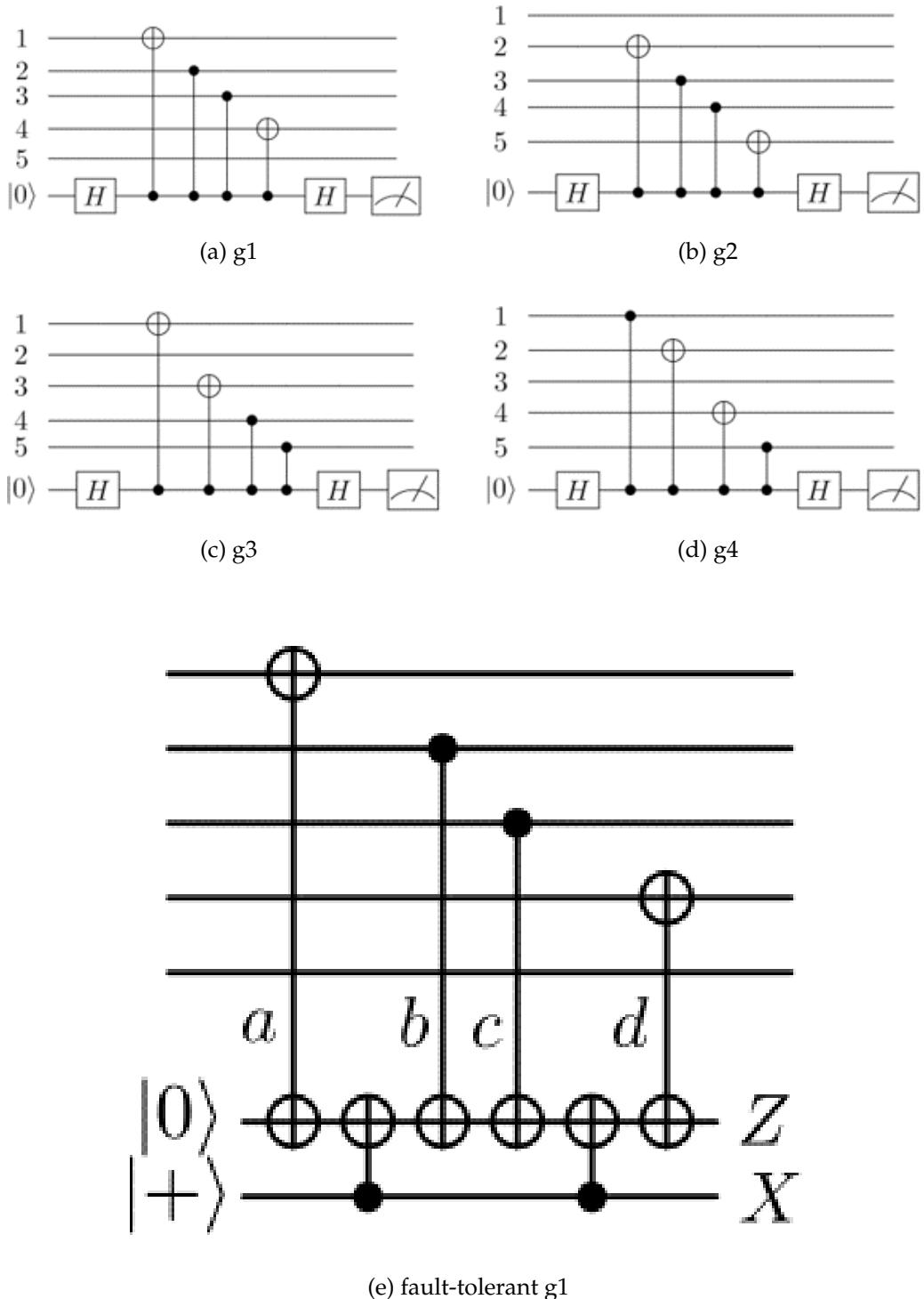


Figure 3.12: Syndrome extraction circuits for the five qubit code. (e) is a fault-tolerant example of extracting $g1$ syndrome from Chao and Reichardt.

3.6.2 LPS Simulation Methods

Up until now we have described in this Chapter how we simulate SPS and how we simulate the general case of logical control 3.5.2. In this subsection we describe the simulations done in a big bulk of our research: Simulations of logical post selection for algorithmic quantum metrology, done by using Kitaev's iterative phase estimation algorithm (Kitaev's approach) and the iterative phase estimation algorithm (IPEA). The goal of those simulations is to find the circumstances in which algorithmic quantum sensing with logical ancillary qubits is better than the traditional approach, with physical ancillary qubits.

For this end we make a set of assumptions defining the rules of the game:

- All measurements are perfect, meaning there are no classical measurement errors. All errors are thus quantum.
- We study the effects of pure dephasing and relaxation independently, in each simulation we take one of T_2 , T_1 to be infinite.
- We may assume 'good' qubits, meaning some qubits are much more resiliant to noise than others.
- We may assume doing SPS or not.
- We may assume that we have access to accelerated Hamiltonians, meaning we have available black boxes capable of performing controlled- U^{2^j} operations for some positive integer j with high fidelity. Great study has been done by D.Aharonov on the subject of accelerated Hamiltonians [2, 6]. This can be achieved via situation-specific methods like applying a sensed magnetic field in different angles, or by

using general methods like QAQC [25], or VFF [13].

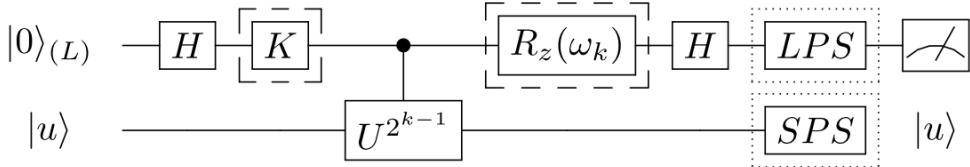


Figure 3.13: Iterative versions of Quantum Phase Estimation. The dashed K gate appears only in Kitaev's iterative version, where K can be $K = S$ and $K = I$. The dashed R_z gate appears only in Iterative Phase Estimation Algorithm (IPEA), where the feedback angle depends on the previously measured bits through $\omega_k = -2\pi(0.0x_{k+1}x_{k+2}\dots x_m)$, and $\omega_m = 0..$. The dotted LPS (Logical Post Selection) and SPS (Sensor Post Selection) gates appear in circuits as described in the main text and other figures.

Kitaev's Iterative Approach - With Accelerated Hamiltonians

Algorithmic quantum sensing using the quantum phase estimation algorithm is especially powerful under the assumption of accelerated Hamiltonians - having a black-box that can apply high powers of the time evolution operator with high fidelity. For our study we first assume accelerated Hamiltonians and we simulate one iteration of Kitaev's approach 3.13, with logical post selection (LPS) and sensor post selection (SPS). We average over 10 angles uniformly distributed in the range $[0, 2\pi)$ and calculate (after post selection) the fidelity with the state of the two-qubit quantum register after an ideal run right before the measurement (and thus assume a perfect measurement). We do that for three different algorithms:

- *Traditional* - The simple, two-qubit circuit for iterative QPE (Kitaev

and IPEA). This algorithm requires 2 qubits.

- *Logical 1LPS* - Using physical sensor and one logical layer of ancilla. All non transversal gates are not fault tolerant. Logical Post Selection (LPS) happens once after the whole procedure. sensor post selection (SPS) happens once after the controlled operation. This algorithm requires 6 qubits.
- *Logical 1EC* - Same as the above, except an error correction step happens once after the syndrome measurement at the end of the iteration, adapted from table 1.2.

The results are plotted in figure 4.14, where we include the fidelity of the error-corrected state. Note that the error-corrected state shows a stochastic behaviour, due to it's 'random' nature - a mistake in the syndrome extraction leads to the application of a faulty correction operator, and note that we have only one round of error correction in the circuit. In addition to the fidelity we define our 'Trace Distance' to be the distance from a perfect measurement:

$$D = \sqrt{(P_0 - P_0^{ideal})^2 + (P_1 - P_1^{ideal})^2} \quad (3.7)$$

Here we see an interesting result: there is a threshold of noise for which the logical control is better then using the traditional approach. The problem with this result is that it does not give any weight to the lost information, and one has to define a 'resource' and give the final results as a function of that resource. We follow [40] and set our resource to be the number of measurements done in one iteration to evaluate the probabilities correctly.

The relevant sizes for the problem are the ideal probability for the ideal circuit without noise P_i , the ideal probability for the noisy circuit P_n and the

estimate to that probability \tilde{P}_n obtained by $m = N(1 - li)$ successful trials, with li being the lost information obtained from equation 3.3 . We define the distance between the ideal circuit's probability and the noisy circuit's ideal probability by $|d| = |P_n - \tilde{P}_n| = D/\sqrt{2}$. Following reference [3] we demand that the error in estimating the ideal probability be confined, such that the probability that the algorithm succeeds is (by the addition rule for statistical and systematic errors)

$$Pr \left(\sqrt{|\tilde{P}_n - P_n|^2 + \frac{D^2}{2}} < \frac{2 - \sqrt{2}}{4} \right)$$

This equals to

$$Pr \left(|\tilde{P}_n - P_n|^2 < \left(\frac{2 - \sqrt{2}}{4} \right)^2 - \frac{D^2}{2} \right)$$

So, the first conclusion is that the algorithm fails for T_2 such that

$$D(T_2) > \frac{\sqrt{2} - 1}{2} \quad (3.8)$$

We conclude that the probability for the algorithm to fail (by applying Chernoff bound) is

$$\begin{aligned} Pr \left(|\tilde{P}_n - P_n| \geq \sqrt{\left(\frac{2 - \sqrt{2}}{4} \right)^2 - \frac{D^2}{2}} \right) \leq \\ 2e^{-2((\frac{2-\sqrt{2}}{4})^2 - \frac{D^2}{2})m} \end{aligned}$$

and we demand that the probability to succeed is

$$Pr \left(|\tilde{P}_n - P_n| < \sqrt{\left(\frac{2 - \sqrt{2}}{4} \right)^2 - \frac{D^2}{2}} \right) \geq 1 - \epsilon$$

This gives us a Minimum of

$$N > \frac{\ln(\frac{2}{\epsilon})}{2(1 - li(T_2))((\frac{2-\sqrt{2}}{4})^2 - \frac{(D(T_2))^2}{2})} \quad (3.9)$$

trials for the algorithm to succeed with probability of success $p \geq 1 - \epsilon$. Calculating this value for noise of different strengths and for a number of different ϵ 's we get the expected result of Fig. 4.16, showing no improvement of the logical approach over the physical one.

Overall we have assumed perfect measurements, accelerated hamiltonians, infinite relaxation time for all qubits, we did SPS and we assumed we have no access to good qubits.

IPEA - Without Accelerated Hamiltonians

Our previous results show that for deep quantum circuits the LPS gadget has a potential of un-biasing the resultant probability distribution over all possible results. Thus we can give up on the non-trivial assumption of accelerated Hamiltonians and use the IPEA (Fig.3.13 (a) with LPS and without SPS).

We pick the irrational phase $\phi = 2\pi/\sqrt{3}$ and evaluate it for up to nine binary digits of accuracy, requiring a maximum of around $3 * 2^8 \approx 800$ consecutive applications of gates including syndrome extraction, approximately 40 times deeper circuit than our previous simulation of Kitaev's approach.

In this part of the research we do only LPS (no SPS) and compare the traditional and the logically-assisted approaches, in four main scenarios:

- Perfect Sensor & Noisy Ancilla with Dephasing - We measure $R_z(\frac{2\pi}{\sqrt{3}})$ with sensor qubit initialized in the excited state $|1\rangle$, assuming all qubits are susceptible to T_2 noise with infinite T_1 . See figure 4.18.
- Noisy Sensor & Perfect Ancilla with Dephasing - we measure $R_x(\frac{2\pi}{\sqrt{3}})$

with sensor qubit initialized in the eigenstate $|+\rangle$, assuming perfect ancillas (with infinite T_1, T_2) and allow only dephasing to occur to the sensor. See figure 4.19.

- Noisy Sensor & Perfect Ancilla with Amplitude Damping - Same as the above, but instead of infinite T_1 we assume infinite pure dephasing rate T_2 and allow T_1 processes (which have some dephasing in them [28]). See figure 4.20.
- Noisy Sensor & Perfect Ancilla with Amplitude Damping - we measure $R_z(\frac{2\pi}{\sqrt{3}})$ with sensor qubit initialized in the eigenstate $|1\rangle$, assuming perfect ancillas (with infinite T_1, T_2) and allow only amplitude damping to occur to the sensor. See figure 4.21.

Using IPEA here is comfortable since it gives us an assessment of the standard deviation of the phase, which is just the inverse of the extracted Fisher information [31]. We compare the standard deviation difference $\sigma - \sigma_{ideal}$ for the traditional and the logical ancillas approaches, and we define the standard deviation to be, in general,

$$\sigma = \frac{\sigma'}{\sqrt{N(1 - l_i)}} \quad (3.10)$$

Where σ' is the standard deviation of the histogram of phase results, N is the total number of trials and l_i is the lost information.

We compute the mean $\bar{\theta} = \text{Arg}(\sum_{\phi} P(\phi)e^{i\phi})$ and standard deviation $\sigma = \sqrt{-2 \ln |\sum_{\phi} P(\phi)e^{i\phi}|}$ of the resulting circular probability distribution (due to the periodicity of the phase). Measuring the average of a gaussian-like distribution [33] gives a scaling of the mean's error of $1/\sqrt{n}$ where n stands for the number of post selected trials. The total measurement error

should be defined as the maximum of the digital error 2^{-m} (where m is the desired precision) and the statistical error approximated in the limit of large n by $\sigma / \sqrt{N(1 - l_i)}$ where N is the total number of algorithm trials. We define the minimal number of trials needed for the statistical error to reach to the digital error to be

$$N_{min} = 2^{2m} \frac{\sigma^2}{1 - l_i} \quad (3.11)$$

Now we focus on the case of "Noisy Sensor & Perfect Ancilla with Dephasing" and explore the resulting probability distribution. First we show that for a specific case, LPS does improve the results. Next, we plot the mean and both the standard deviation of the histogram and the one taking lost information into account 3.10. We find thresholds indicating LPS improvement of the results, and Finally we plot the minimum number of whole algorithm run required to reach the desired digital accuracy by equation 3.11.

3.7 Measurement Errors

Measurement errors are classical errors resulting from a faulty identification of the quantum state. In our study we choose to focus on quantum errors occurring mid-circuit, and we assume perfect measurements and perfect measurement procedure (i.e., if a parity measurement is required to measure the digit at the end of the algorithm, we assume no quantum errors occur during the parity measurement time, and we also assume no classical error is made in labeling the quantum state).

Here we give a detailed explanation of how to cope with measurement errors, even though we have not tested the following methods by simu-

lation. There are two kinds of measurements occurring in the algorithm: syndrome measurements and digit measurements at the end of the algorithm. To measure a digit, one can perform a parity check on the operator $\bar{Z} = Z_1Z_2Z_3Z_4Z_5$ resulting in only one measurement with error probability P_m . To measure a syndrome, one must make four measurements - one for each stabilizer, with a faulty syndrome resulting with probability $1 - (1 - P_m)^4$.

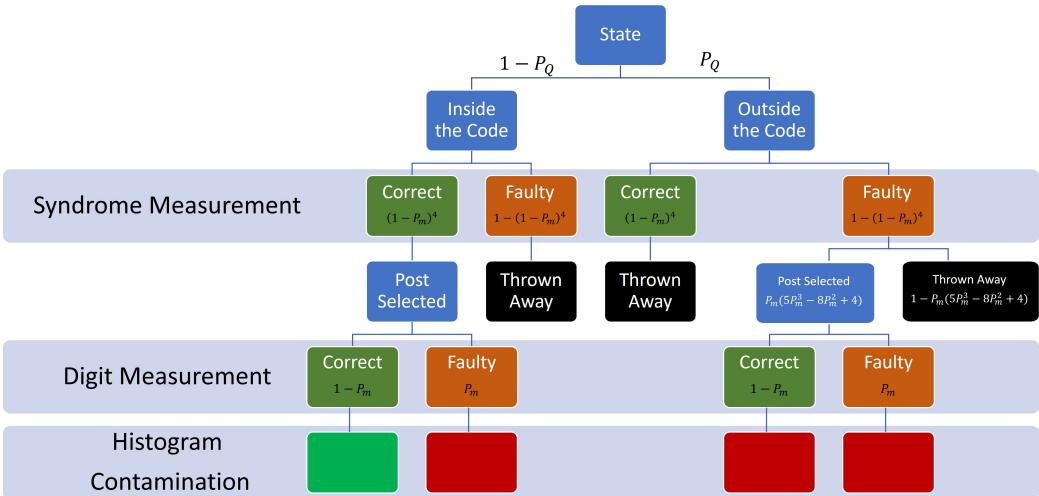


Figure 3.14: Decision tree for Logical Post Selection. For simplicity, we assume (only in this analytic derivation) that no quantum errors can happen during syndrome and digit measurements. The figure is drawn to highlight the scenarios resulting in histogram contamination, where a green mark in the last line is the result of a path with no contamination of the results and a red mark in the last line is the result of a path that results in histogram contamination.

The full decision tree of measurements with errors is depicted in figure 3.14, where P_Q is the probability of a quantum error resulting in logical state outside of the code. LPS can be thought as occurring right after syn-

drome measurement. The only non-trivial route in the decision tree is where the state is outside of the code, and a faulty syndrome measurement is made. In this scenario, there is a chance the run is post selected: Suppose the real quantum state should give the syndrome '0001' but a faulty measurement on the last stabilizer gives the trivial syndrome '0000'. Summing up all these possible scenarios (mistakes in one, two, three or four stabilizers) results in a probability that the state is outside of the code but detected as if it was inside the code:

$$4P_m(1 - P_m)^3 + 12P_m^2(1 - P_m)^2 + 4P_m^3(1 - P_m) + P_m^4 = P_m(5P_m^3 - 8P_m^2 + 4)$$

And we note that the left hand side is strictly positive for $0 < P_m < 1$.

The only possible path that does not contaminate the probability distribution (or histogram) of the algorithm is the one with no errors at all. In all other paths where the run is post selected, the histogram is contaminated. looking at the decision tree and assuming small error probabilities, the probability of contamination is:

$$P_m(1 - P_Q)(1 - P_m)^4 + P_mP_Q(1 - (1 - P_m)^4)(5P_m^3 - 8P_m^2 + 4) \approx P_m + P_mP_Q$$

It is apparent that histogram contamination can happen as a result of two scenarios: Faulty digit measurement of a state within the code, or a faulty syndrome measurement of a state outside of the code. The former results in probability of approximately P_m and the latter results in probability of approximately P_mP_Q , and under the assumption of small error probabilities we have $P_mP_Q \ll P_m$ meaning a faulty digits measurement is the error that should be treated with highest priority.

Such measurement error can be treated with classical error correction techniques, such as repeated measurements: Repeating the measurement (or

the whole iteration) 3 times and taking majority vote, gives error with probability $P_m^2(1 - P_m) + P_m^3 = P_m^2$. Adding n layers of repetitions like so (overall 3^n measurements) gives measurement error with probability $P_m^{(2^n)}$. In this technique, we beat measurement errors exponentially with the number of repetitions.

We note again that the above approach works fine in theory, but in practice we do care about the number of repetitions, and while we assume for simplicity that we have no measurement errors this problem remains an interesting question for future work.

4 Results

In this Chapter we show and analyze our results to the experiments (simulations) described in the last Chapter.

Throughout this work we separate two scenarios:

- Target: Accuracy - In this scenario we give no consideration to the lost information. In the presence of decoherence, quantum phase estimation will converge (given an infinite amount of trials) to biased measurement probabilities P_0, P_1 different from the ideal. In some cases we may be interested in getting this systematic error as low as possible, and we wouldn't care about how much information we have lost by post selecting. Algorithmic quantum sensing approaches the Heisenberg limit only for unbiased distributions, and using logical post selection allows us to make the distribution less biased.
- Resource: Minimal Number of Trials - In this scenario we consider the lost information in our calculations, in various ways. In these more common scenarios of quantum sensing, the lost information and number of trials does play a role, and a strong evidence for that is the fact that sensitivity is often defined per root Hertz.

The simulations in the context of quantum metrology of this work can be roughly divided into two categories. The first one is where we simulate accelerated Hamiltonians with Kitaev's QPE circuits, and the measured quantity is the fidelity with the ideal (not noisy) circuit, or the distance at the end of the circuit and before the measurement. The second category is where we simulate non accelerated Hamiltonians with IPEA, and there are numerous measured quantities - all describe features of the probability distribution of all possible algorithm's results.

Decision trees describing our results in a very rough manner are depicted in figures 4.1 and 4.2. Note that in the latter figure there is a complicated dependence of the threshold on circuit depth and Worst Case Single Gate Fidelity, and the given thresholds are for the lowest WCGSF that gives an advantage to LPS.

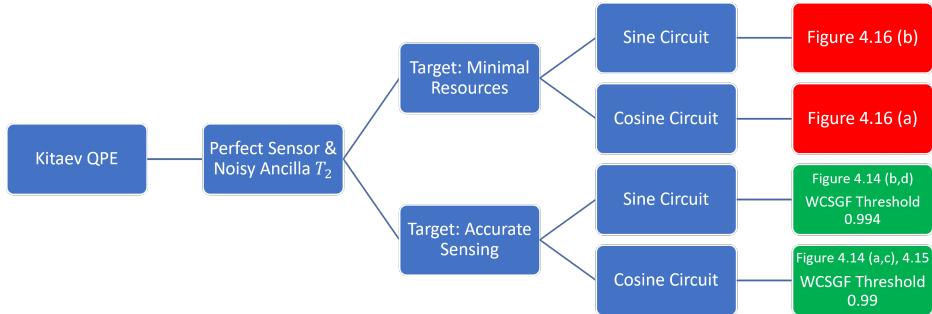


Figure 4.1: Decision tree for accelerated Hamiltonian. Simulation is done with Kitaev's QPE circuits, and the measured quantity is the fidelity with the ideal (not noisy) circuit, or the distance at the end of the circuit and before the measurement. For sensing with minimal time the LPS gadget does not do any good, but it does for sensing with high accuracy.

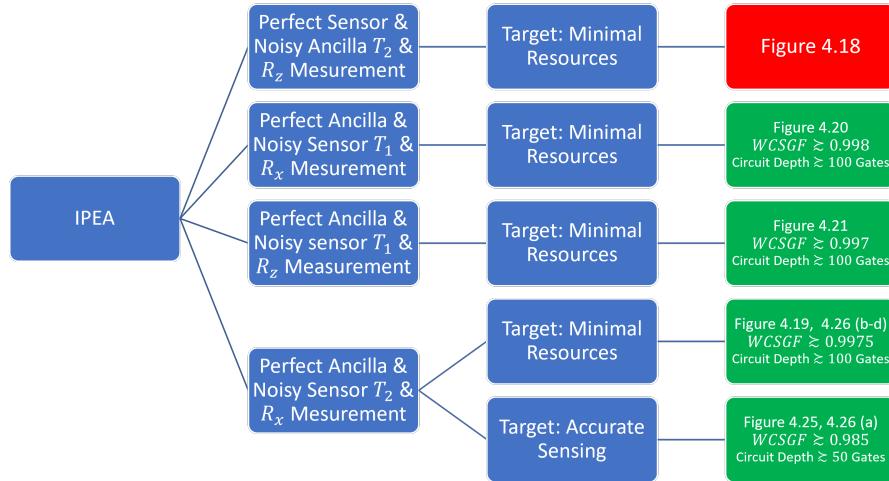


Figure 4.2: Decision tree for non-accelerated Hamiltonian. Simulation is done with IPEA, and there are numerous measured quantities - all describe features of the probability distribution of all possible algorithm's results. Note that there is a complicated dependence of the threshold on circuit depth and Worst Case Single Gate Fidelity, and the given thresholds are for the lowest WCSGF that gives an advantage to LPS. Circuit depth is given for U 's sensing time - i.e., for a forth power of U each application is of the length of 16 gates, and 3 such applications are required for ancilla-controlled operation, which gives a circuit depth of approximately 50 gates.

4.1 Sanity Checks for the Simulation

This section discusses sanity checks and numeric error estimation done for evaluating the simulation's performance. A basic sanity check that does not appear in this section is the gate sanity checks plotted in figure A.3 of the appendices. To check correctness of our simulation we simulate dephasing and amplitude damping, and in addition apply these processes

to a noisy Ramsey experiment, comparing the simulation to analytic solutions. Then we show representative convergence tests and justify our choice of the simulation step size. In the third subsection of this section we describe our map from T_1 and T_2 of qubits to the worst-case gate fidelity.

4.1.1 T_1 & T_2 Extractions

Our simulation of decoherence is based on Kraus operators, and trotterization. Here we do T_2 relaxation on the state $|+\rangle\langle+| = \frac{1}{2}(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix})$, and expect to see that the state evolves under dephasing as

$$\rho(t) = \frac{1}{2} \left(\begin{smallmatrix} 1 & e^{-t/T_2} \\ e^{-t/T_2} & 1 \end{smallmatrix} \right) \quad (4.1)$$

We plot (fig.4.3 (a)) the coherence and see how close is the graph of $\rho_{10}(t)$ to e^{-t/T_2} . Next we check our amplitude damping on the state $|+\rangle\langle+| = \frac{1}{2}(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix})$, and we expect to see that the state evolves under amplitude damping as

$$\rho(t) = \frac{1}{2} \left(\begin{smallmatrix} 2 - e^{-t/T_1} & e^{-t/2T_1} \\ e^{-t/2T_2} & e^{-t/T_1} \end{smallmatrix} \right) \quad (4.2)$$

We plot (fig.4.3 (b,c,d)) all elements of the density matrix to see that they are well-behaved. Overall the decoherence simulation is as expected and plotted in figure 4.3

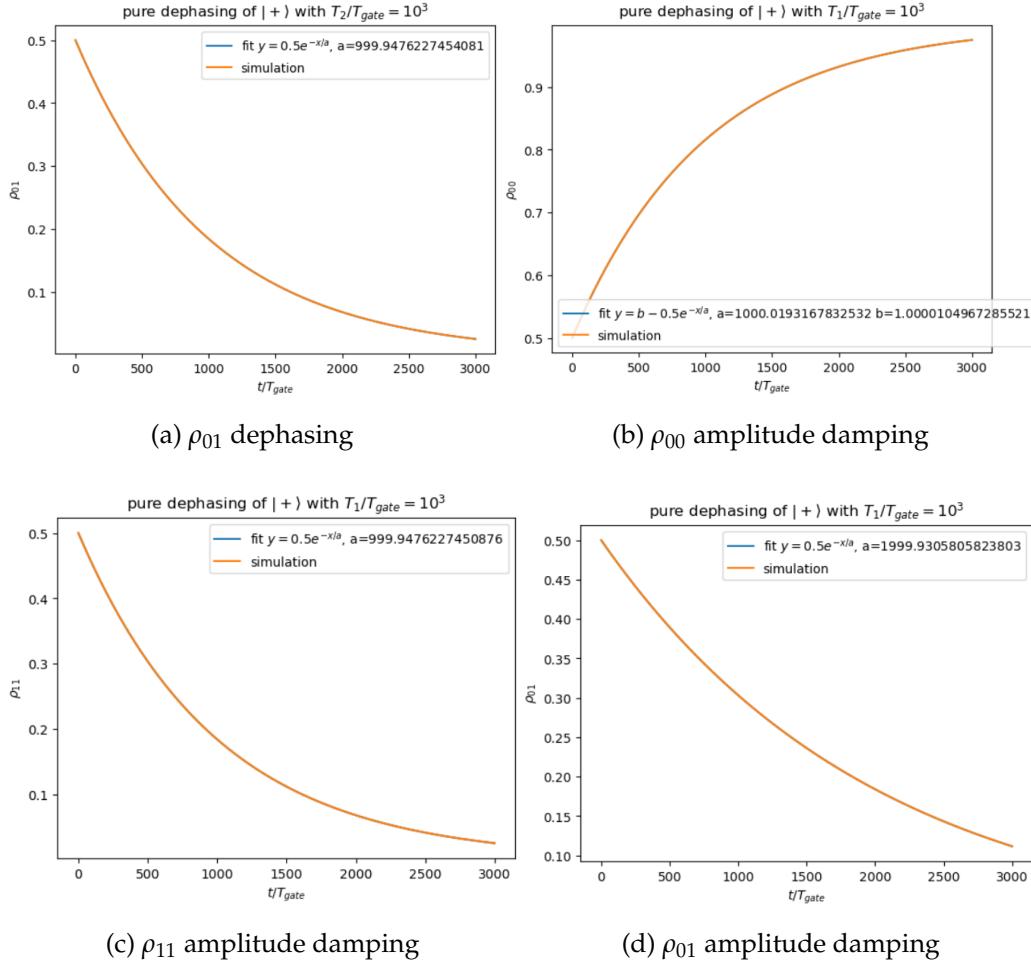


Figure 4.3: Relaxation for three decoherence times. Fit and simulation are the same.

4.1.2 Ramsey Experiment

Another sanity check we do is implementing the Ramsey experiment as described in the introduction, 1, but in the presence of T_2 noise. We do that for energy splitting $\omega_0 T_{gate} = \pi/20$ and noise $T_2/T_{gate} = 500$ and

expect to measure the excited state $|1\rangle$ with probability:

$$P(1) = \frac{1}{2}(1 + e^{-\frac{t}{T_2}} \sin \omega_0 t) \quad (4.3)$$

We collect data and perform a fit to the above probability. We expect the envelope to be $\frac{1}{2}(1 \pm e^{-\frac{t}{T_2}})$ with T_2 being approximately 500 and the data to oscillate with angular velocity of approximately $\pi/20 = 0.157$. The data and fits are plotted in figure 4.4 and act as expected, meaning our simulation is built properly.

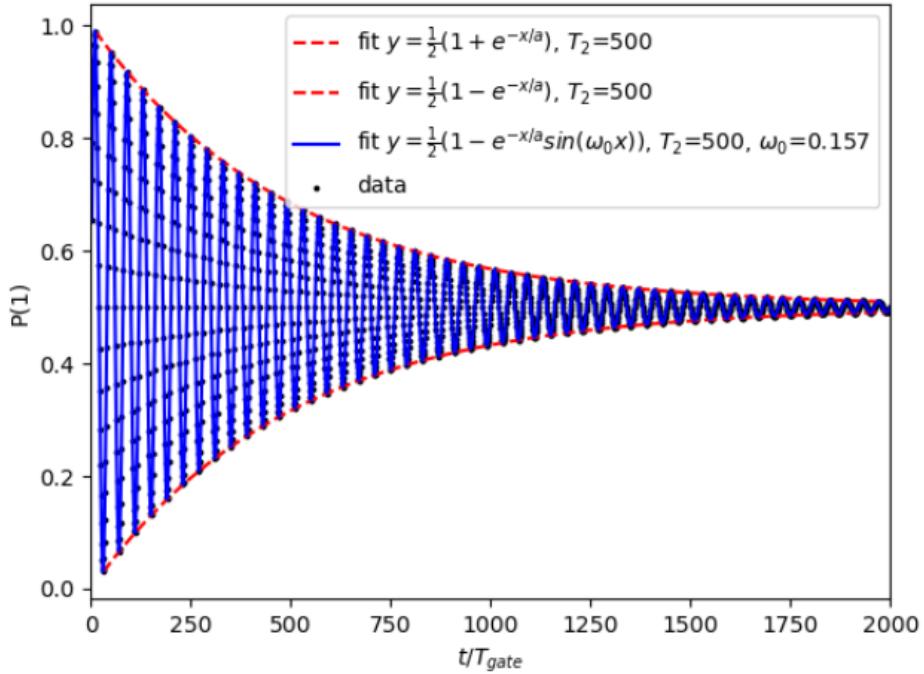


Figure 4.4: Simulation of Ramsey experiment with $T_{gate}\omega_0 = \pi/20$ and $T_2/T_{gate} = 500$. Probability develops as equation 4.3. Fit and simulation are the same.

4.1.3 Convergence Tests

A final sanity check that needs to be done in any simulation is that the results are converged as a function of the step size. We have conducted a very large number of these tests. Figure 4.5 displays the essence of those tests and show that we have chosen converged parameters.

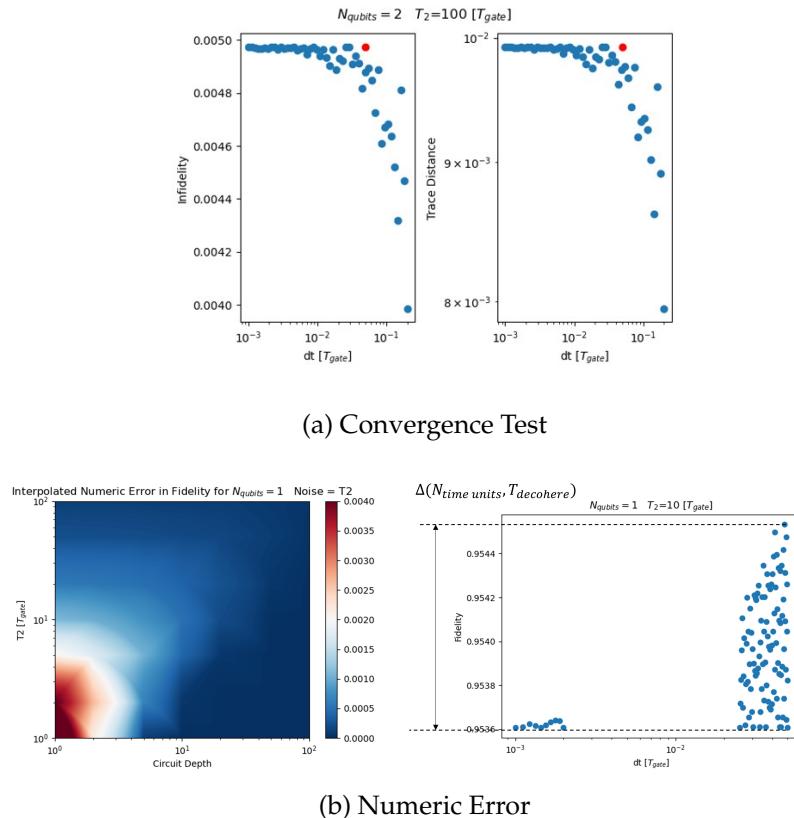


Figure 4.5: Here is an example of a convergence test, one of many, that has been done to check convergence. (a) shows the selected time step dt of the simulation, and that it is converged. (b) defines the ‘numeric error’, and shows that for large circuit depths (as has been done in this work) the numeric error is essentially neglectable.

4.1.4 From Decoherence to Gate Fidelity

For our study, we need a way to speak in the language of gate fidelity instead of in terms of T_2/T_{gate} , T_1/T_{gate} where T_{gate} is the time required to apply the average gate. We study the effect of dephasing and amplitude damping independently, and so we create a map between the coherence times to the worst case gate fidelity, both for a single qubit and a two-qubit gates. For **dephasing** we set the qubit(s) in the (tensor product) state $| + (+) \rangle$ and let them dephase for a varying time, and we then calculate the fidelity according to $F = \sqrt{\langle +(+)|\rho| + (+)\rangle}$ assuming they end up in the density matrix state ρ . For **amplitude damping** we apply the same procedure, starting from the state $|1(1)\rangle$ and calculating the worst case gate fidelity by $F = \sqrt{\langle 1(1)|\rho|1(1)\rangle}$. Both maps are available in figure 4.6.

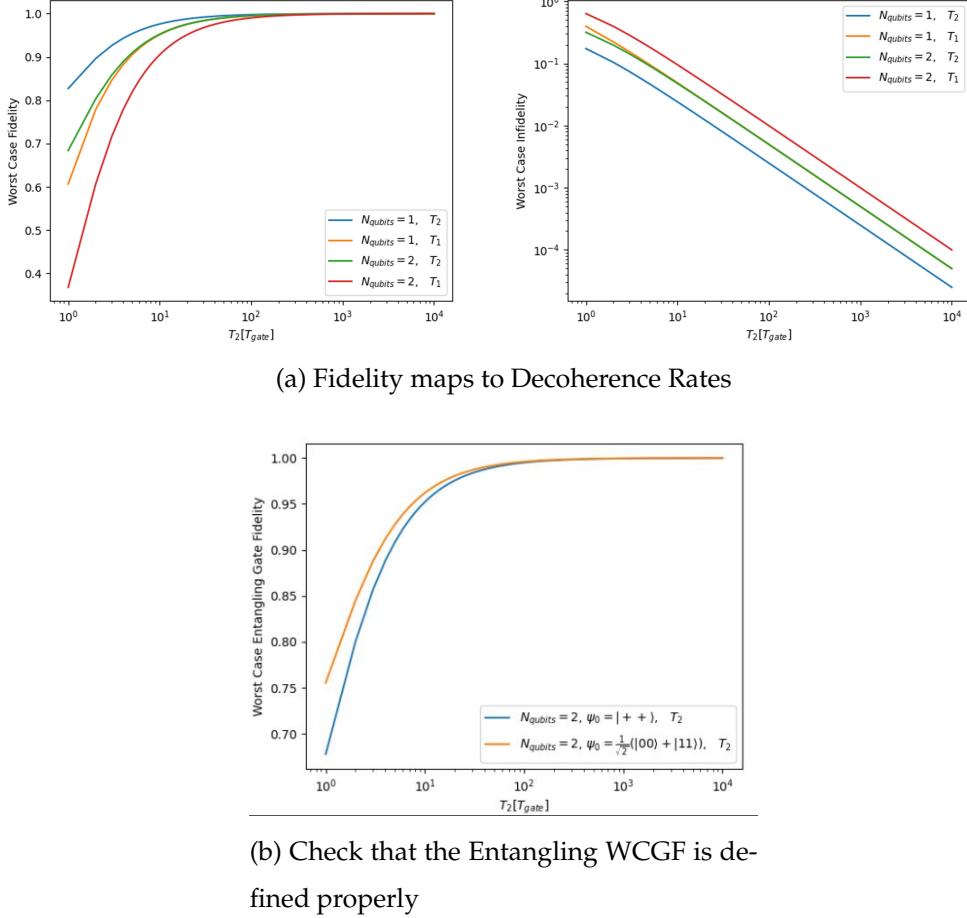


Figure 4.6: (a) Maps from coherence times to worst case gate fidelity for single and two-qubit gates, extracted by putting the qubits in their most vulnerable state to dephasing (amplitude damping) and letting them decohere. (b) A check that indeed $|++\rangle$ is more vulnerable to dephasing than $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. The Infidelity is calculated as $1 - F$.

4.2 Sensor Post Selection for Quantum Sensing

As discussed in section 3.4 post selection can sometimes be used to enhance the sensitivity of a quantum sensor. In this section we check whether or not this is the case for Kitaev's iterative approach of quantum phase estimation, for the simple sensor post selection schemes of figure 3.4.

4.2.1 Target: Accuracy

In some cases when we are interested in measuring something, the number of measurements required is of no interest, and we aim to be as accurate as possible measuring the observable.

A simulation has been done as described in section 3.4.2, and the results are depicted in figure 4.7. As we can see and as expected, SPS helps with the cost of losing information. A weird phenomenon I cant explain right now is the existence of the threshold in which Single SPS is better than Multiple SPS. And yet, the two methods perform much better than the traditional approach. Another interesting thing to notice is the existence of a threshold for the trace distance, which represents the actual information that can be extracted from the state after measurements. This quantity is better for estimating algorithmic sensing performance and so in the next subsection we continue working with this quantity.

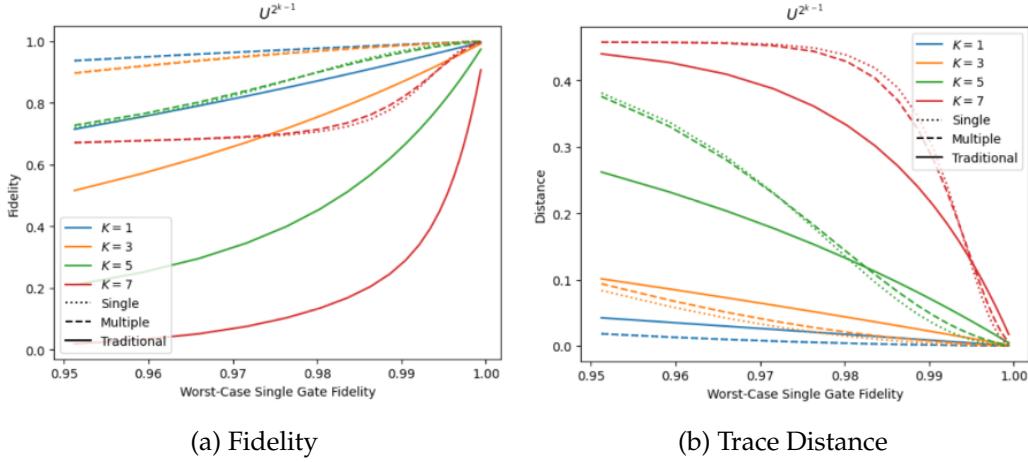


Figure 4.7: Kitaev Iterative QPE as in figure 3.4, without Hamiltonian acceleration. (a) Fidelity of the entire algorithm, (b) Trace Distance ($\sqrt{\sum_i d_i^2}$ with $d_i = \rho_{ii} - \rho'_{ii}$, ρ is noisy density matrix and ρ' is ideal density matrix) for the entire algorithm. It is evident that for each circuit depth there is a cross-over in distance between ideal application of the algorithm and the regular approach, and that the fidelity of post selected trails is always better, as expected.

4.2.2 Resource: Minimal Number of Trials

In this subsection our resource is the minimal number of trials required for Kitaev's algorithm ($K = S$ for taking the worst case) to not fail according to equation 3.9, which is valid to any run of Kitaev's phase estimation with noise and lost information. Results are depicted in figure 4.8.

For $K = 1$ we see no advantage for using SPS, and for any K higher the performance in terms of the defined resource is getting worse and worse, as expected due to the high percentage of lost information caused by the large circuit depth.

To conclude, Sensor Post Selection as defined earlier does not help us in the common sensing scenarios but rather only in those we aim for accuracy at the expanse of measurement time.

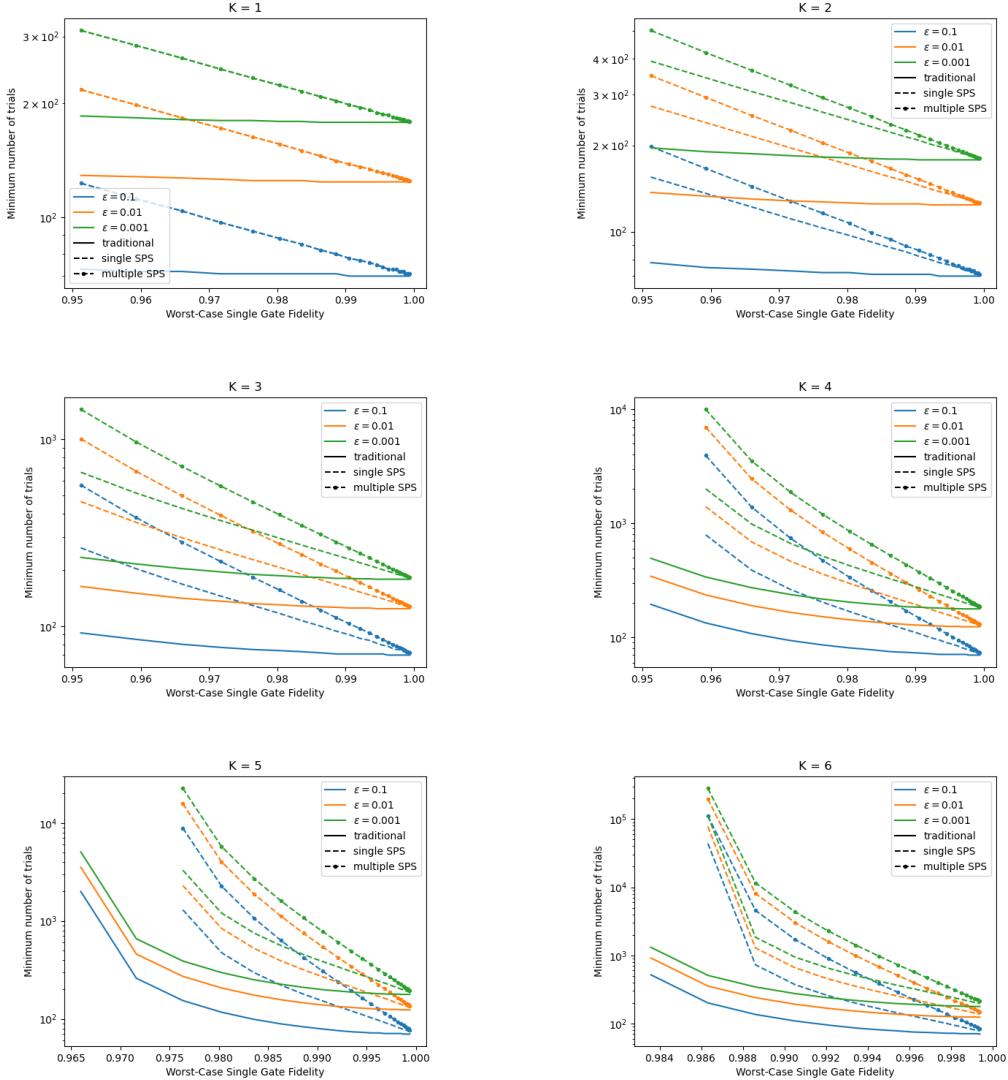


Figure 4.8: Minimal Number of Trials that is required for Kitaev's algorithm with unaccelerated Hamiltonian to work according to equation 3.9 is plotted for each value of K from 1 to 6, where we raise the measured operator to the power 2^{K-1} . It is evident that it is never better to use post selection of the sensor if our desire is sensing with a minimal number of trials.

4.3 Logical-Physical Interaction - General Numeric Exploration

In the previous section we have explored the use of sensor post selection for algorithmic quantum sensing. We saw that for some needs it might help and for others it can only harm. This section and the next focuses on mapping the world of logical post selection in the presence of noise, finding scenarios in which it might be helpful to use logical post selection and pointing at the ones where it is not.

In this section we acquire insights on the general use of logical-physical interaction, in terms of the circuit's depth (or it's number of entangling gates, N_{gates}) and the qubit's decoherence time T_2 , which translates to worst-case gate fidelity as described in section 4.1.4. A simulation has been performed as described in section 3.5.2.

So, for what number of entangling gates does the logical control contribute? does this number even exist? Picking a constant T_2/T_{gate} ratio we can depict (fig.4.9 (c)) the fidelity as a function of the number of entangling gates. We can see that for each gate fidelity, it is always better to use traditional CNOT gate when we have to apply only one such gate. But there is a range of number of gates in which it is better to use logical control (fig.4.9 (c)). This range stops at some point, in which the gate overhead is not profitable. The cost, the amount of lost information, is large, and depicted in fig.4.9 (b). Looking at fig.4.9 (c), we suspect that for infinitely good gates, using logical control is better for infinitely deep circuits.

A similar graph is extracted for constant number of CNOT gates N_{gates} , and varying T_2/T_{gate} ratio, and depicted in figure 4.10 (c). We can see that

for each number of gates, there is a threshold of worst case single gate fidelity for which the logical gadget is better than the traditional approach. This threshold was extracted and depicted in fig.4.10 (a), and we can see it extracts the same graph as in fig.4.9 (a). Here as well, the cost (lost information) is high and depicted in figure 4.10 (b).

Finally, a three dimensional view of the above is depicted in figure 4.11. Although this simulation is done for varying T_2 , we expect similar behaviour for T_1 noise. We remind that these results were obtained with gate overhead of about 20 gates for syndrome extraction, and that they were obtained for both noisy logical and physical qubits, and thus we expect the following limit behaviour:

- Noisy Physical & Perfect Logical - logical control will always be better starting from a certain circuit depth.
- Perfect Physical & Noisy Logical - logical control will never improve the fidelity. Adding noisy sources shouldn't help, intuitively.

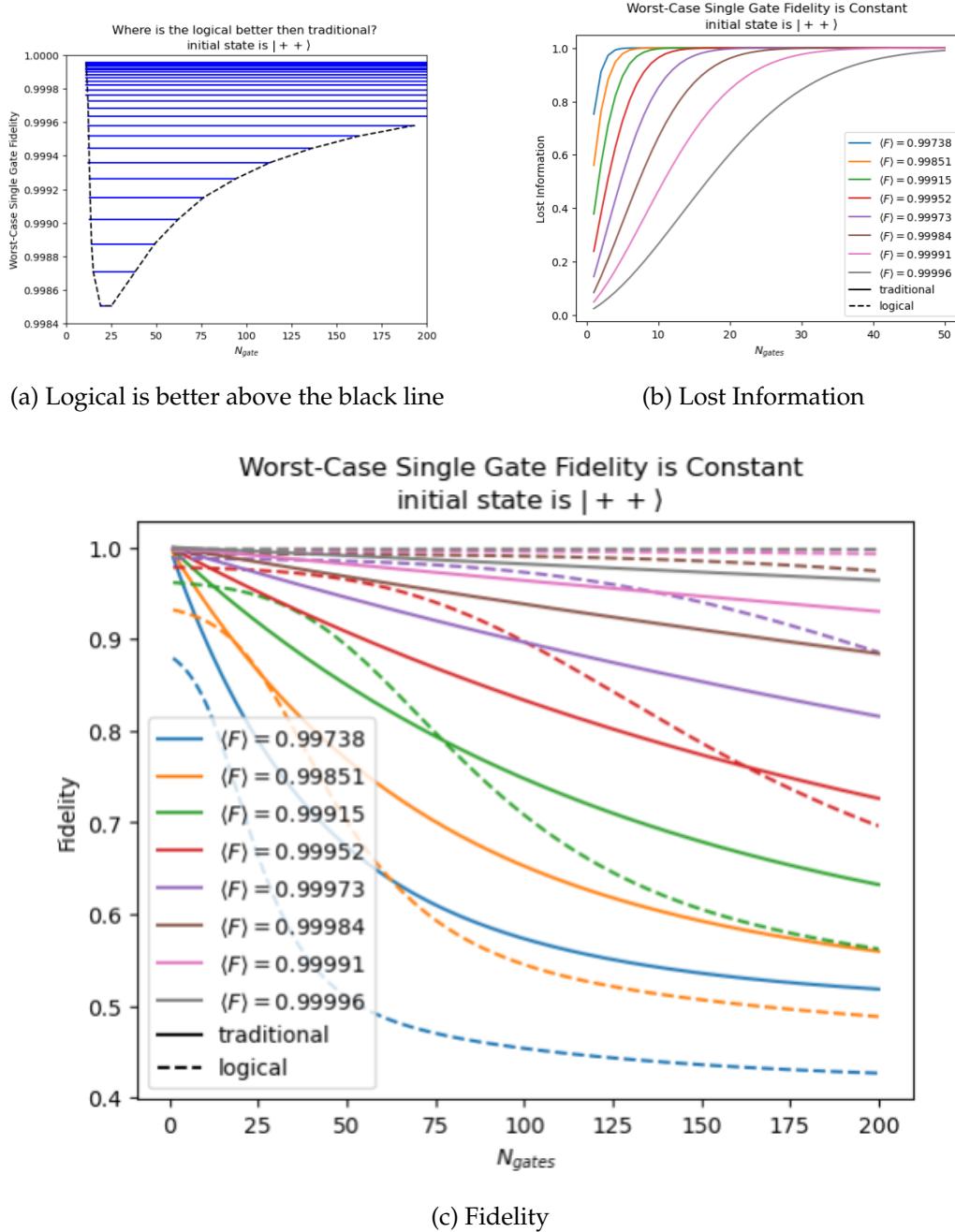


Figure 4.9: Simulation of logical CNOT gate compared to the traditional CNOT gate, for a varying number N_{gates} of CNOT gates and constant T_2/T_{gate} ratio. It is evident that for gate fidelities above a certain threshold there is a cross-over in the circuit depth, where it is better to use logical post selection.

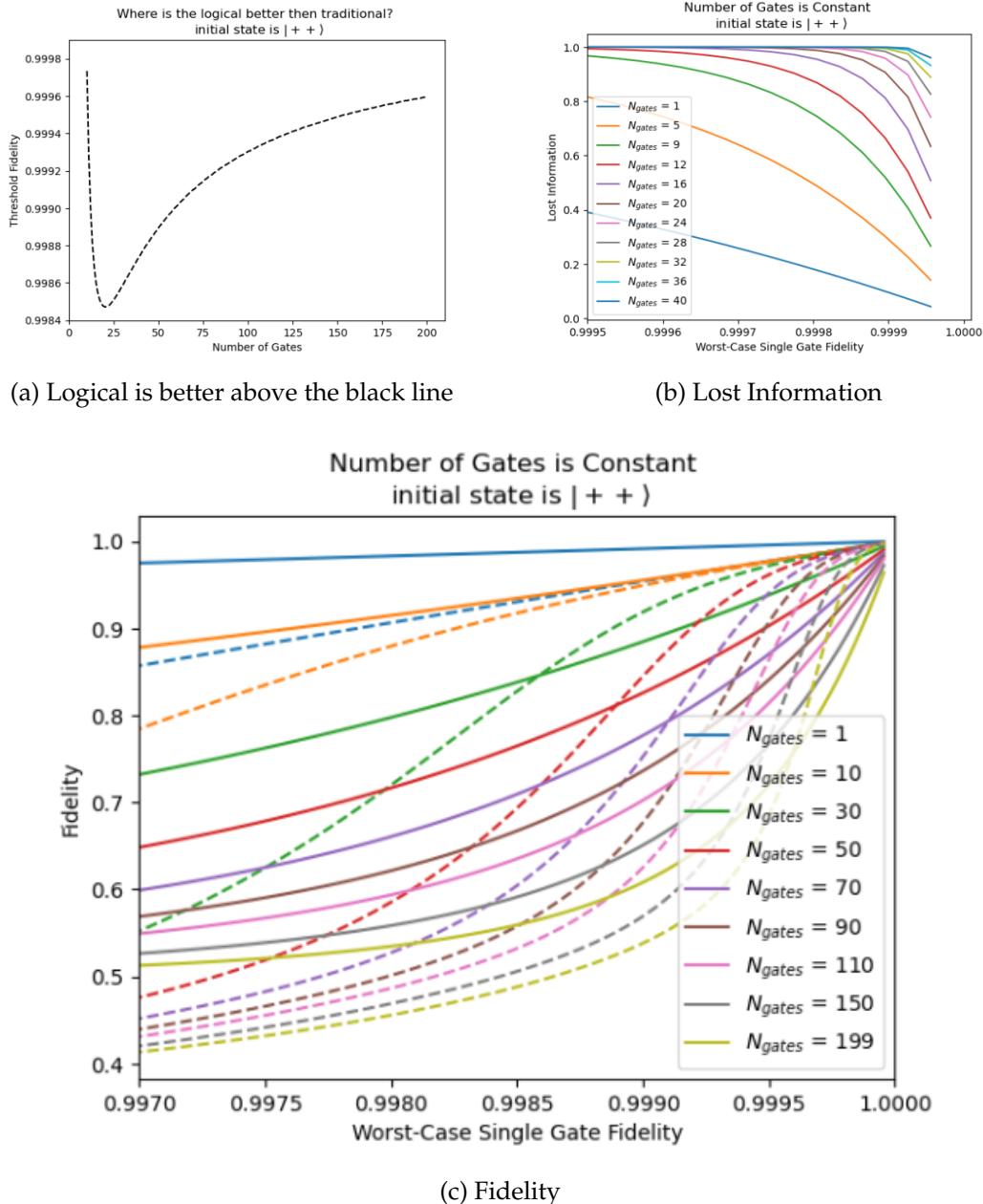


Figure 4.10: Simulation of logical CNOT gate compared to the traditional CNOT gate, for a varying T_2/T_{gate} ratio and a constant number N_{gates} of CNOT gates. It is evident that for circuit depths above a certain threshold there is a cross-over in the worst case single gate fidelity, where it is better to use logical post selection.

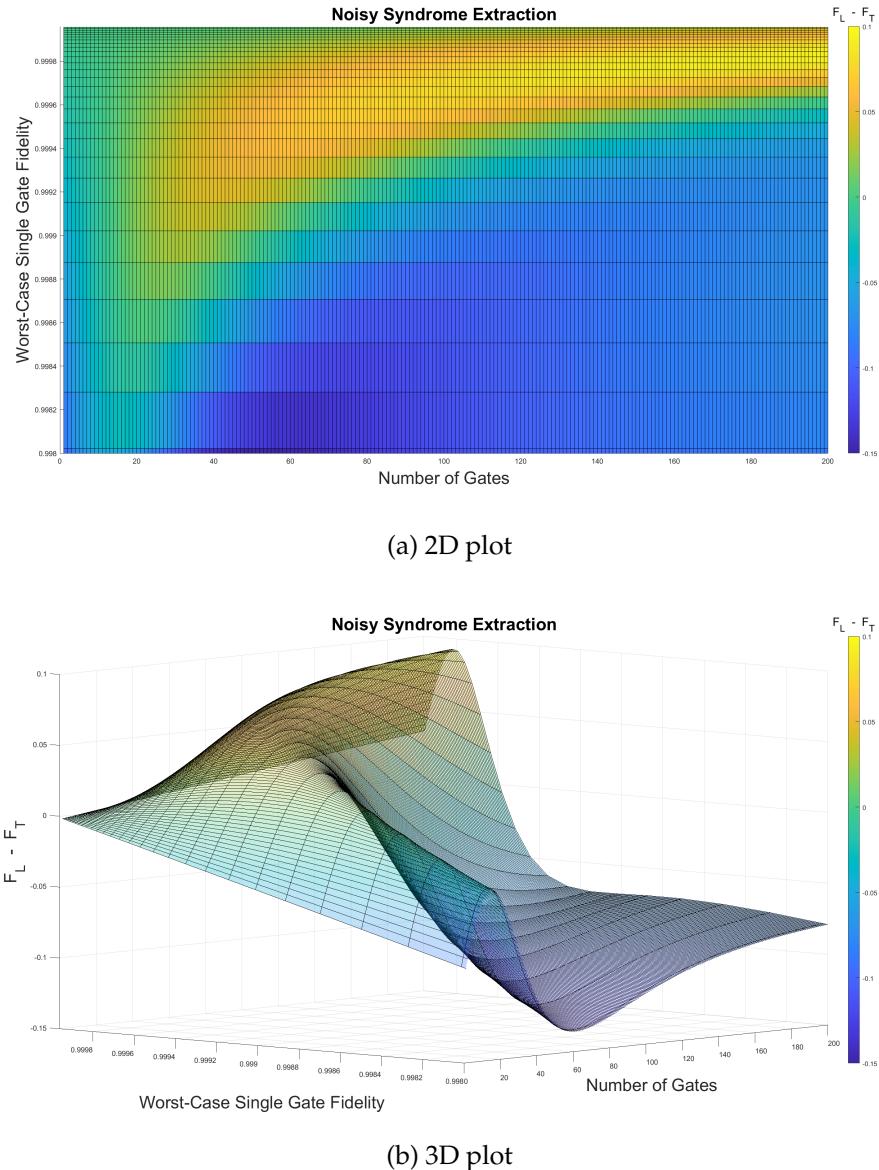


Figure 4.11: Simulation of logical CNOT gate compared to the traditional CNOT gate, for a varying number N_{gates} of CNOT gates and a varying T_2/T_{gate} ratio. This figure shows the last two figures in a three dimensional view. yellow areas show superiority of the logical post selection gadget.

4.4 Logical Post Selection for Quantum Metrology

In the last section we have numerically explored the behaviour of logical control and logical post selection under the change in circuit depth and gate fidelity, taking both the logical qubit (ancilla system) and the physical qubit (sensor) to be noisy. We got an interesting result: for each gate fidelity, there is a range of circuit depth's where introducing logical ancillary system improves the fidelity of our circuit under the presence of noise which causes a systematic error in the system's state.

In this section we demonstrate a version of this result in the case of algorithmic quantum sensing. We seek to find circumstances in which using logical ancilla may improve sensing results, and to determine the situations where it wouldn't. In the following subsection we briefly show that for our cause, of sensing with accuracy, it will always be better to use the non-fault-tolerant implementation of state preparation. Then, in the two subsequent subsections we separate our research to two main scenarios: having access to accelerated Hamiltonians, and not having it. Observing the improved scaling of the error probability for the Kitaev circuit with $K = I$ is made possible due to the natural transversality of the circuit.

4.4.1 Gate Selection

Here I briefly show some of the results on gate selection, deciding whether or not to use flag fault-tolerance. Eventually I have decided to continue with the non fault tolerant implementations, and using the fault tolerant

implementations for QPE remains an interesting question for future work.

Logical Rotation

The tested circuits are those of figure 4.12 (a) in comparison to 4.12 (b). All qubits, including the flag qubits, are subject to decoherence. Due to the enlarged circuit depth, not in all noise regimes the use of flag fault tolerance is of help. Eventually I have decided to work with the non fault tolerant implementation due to the lower percentage of lost information.

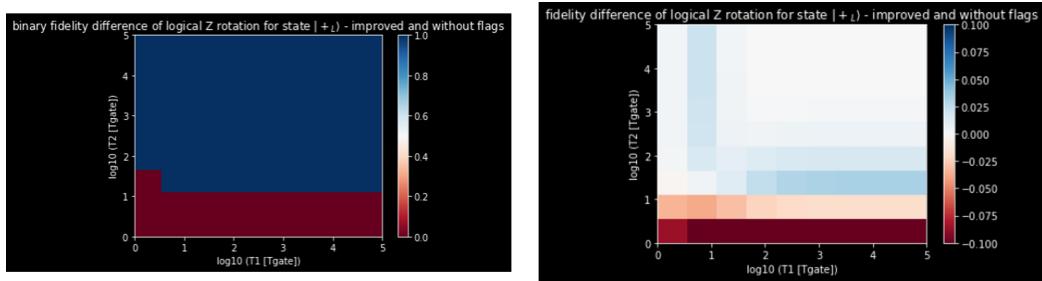


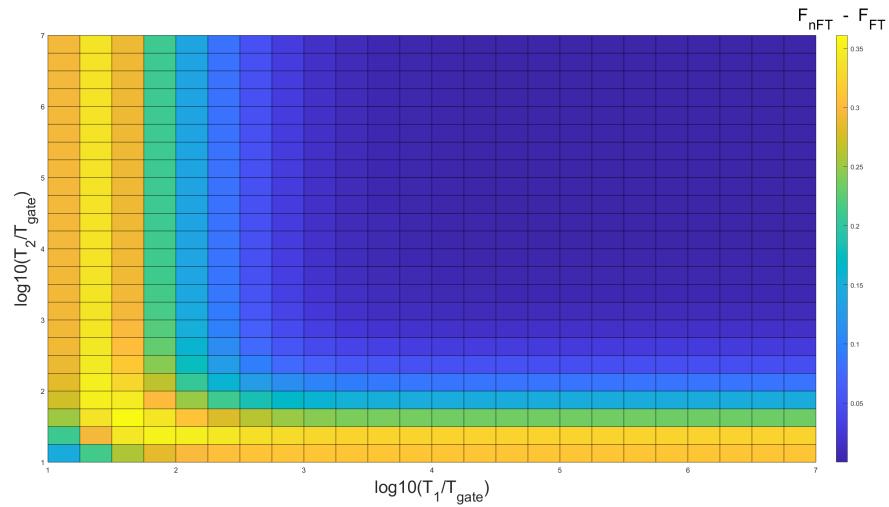
Figure 4.12: Comparing the non fault tolerant and the improved flag fault tolerant implementations. We see that above $T_2 \approx 10^{1.5} T_{\text{gate}} \approx 30 * T_{\text{gate}}$ it is useful to use flag qubits, even if they decohere like the data qubits.

State Preparation

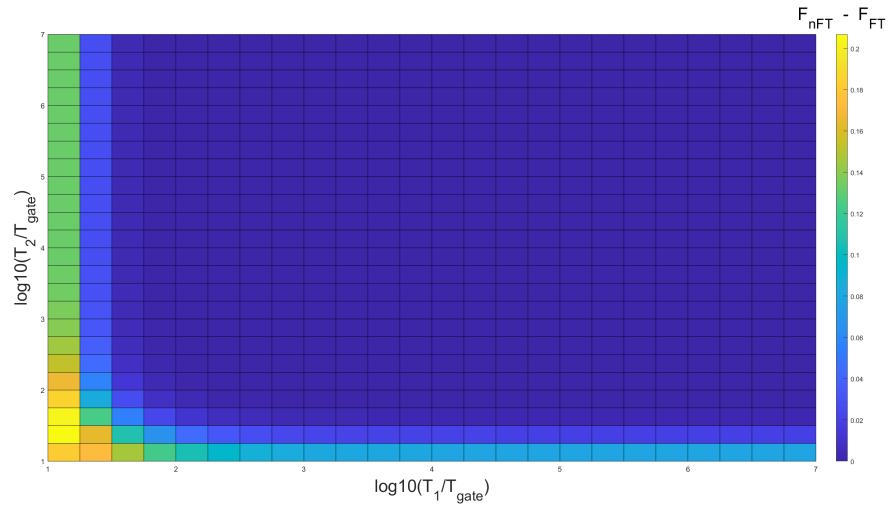
Here we briefly show two simulations of the state preparation procedure - comparing both non fault-tolerant and faulty-tolerant implementations, with and without logical post selection on the result. The simulations are depicted in figure 4.13, and we can easily see that in no circumstances it is better to use the fault-tolerant implementation if we are interested in maximizing the fidelity. The fault-tolerant implementation is of importance only for long computations where it is likely an error occurred before the

entangling CZ gates, to stop the error from propagating, in this case we deal with big circuit depth's in which the depth of state preparation is negligible, thus allowing us to neglect errors occurring at this stage.

Note that we simulate both for only flag measurements and flag measurements along with logical post selection. Both scenarios point us towards the non fault-tolerant implementation.



(a) Without Syndrome Extraction



(b) With Syndrome Extraction

Figure 4.13: Mapping the fidelity difference between the the non fault tolerant and the fault tolerant implementations of state preparation, 3.6

4.4.2 Kitaev's Approach - with Accelerated Hamiltonians

As mentioned in section 3.6.2, algorithmic quantum sensing is especially powerful under the assumption of accelerated Hamiltonians. We state again our assumptions for this section:

- All measurements are perfect
- There is no energy relaxation, only dephasing: $T_1 \approx \infty$
- Sensor post selection is being used
- We have access to accelerated Hamiltonians
- We assume no good qubits - all qubits decohere

Although we assume no good qubits, we measure here the operator $R_z(\theta)$ and we put the sensor qubit in its ground state, $|0\rangle$. Dephasing does not affect this state, and we assumed no energy relaxation, so in practice we assume a perfect sensor and noisy ancilla. Thus, intuitively, we expect no improvement whatsoever. Our only hope is the non intuitive nature of quantum mechanics.

Data collection has been made as described in section 3.6.2 for Kitaev's approach. As in section 4.2 we separate our results to two scenarios, caring of accuracy for the following part or efficiency for the subsequent one.

Target: Accuracy

Two interesting phenomena occur in figure 4.14:

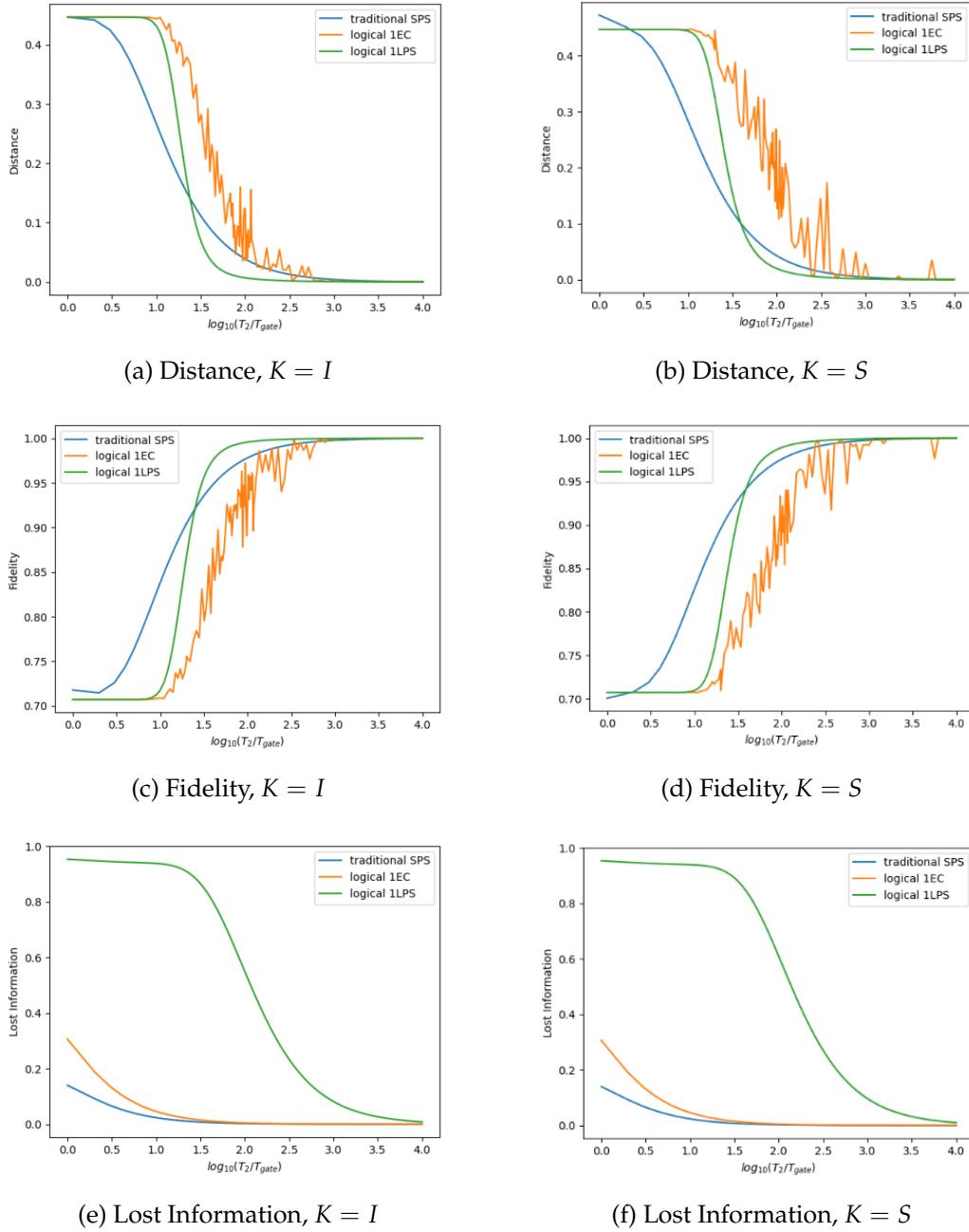


Figure 4.14: Simulation of the logical and traditional Kitaev circuits, averaged over 10 angles evenly distributed in the range $[0, 2\pi)$. Simulation is done for decay time of $T_1 = \infty$, and eigenvector $|\psi\rangle = |0\rangle$. It is evident that the error corrected algorithm is stochastic, and that there is a coressover of noise where it is better to use LPS for accurate sensing even if the ancillas are noisy, at the cost of a high percentage of lost information.

First, counter intuitively, we get a threshold for which logical control is better than physical control. This is counter intuitive because only the ancillas are noisy. Apparently, there are circumstances in which it is better to use five noisy ancillas than only one noisy ancilla. The thresholds are summarised in table 4.1

Fidelity	T_2/T_{gate}	Worst-Case Single Gate Fidelity	Worst-Case Entangling Gate Fidelity
$K = I$	24	0.99	0.98
$K = S$	40	0.994	0.987

Table 4.1: Thresholds for Accelerated Hamiltonian from the data of figure 4.14

Surprisingly, these thresholds are well below [23] the achievements of today's state of the art technology! Thus, for the purpose of accurate sensing, with infinite sensing time, using logical post selection is the better approach for the NISQ era.

The second interesting result we can see in these graphs is that the error-corrected algorithm is stochastic. Earlier we explained it is the result of a faulty error detection. This random faulty detection projects the system's state to one with 100% error, and then applies a faulty 'correction operator', which might make the state even further than the ideal. This behaviour is not seen when using logical post selection, because in this scenario we project to a state which includes no error at all.

Figure 4.15 shows the final results of graph 4.14 (c) as after some signal-processing to smooth the stochastic behaviour, and the same graph but in

log scale. We witness improvement of up to one order of magnitude in the resultant Infidelity.

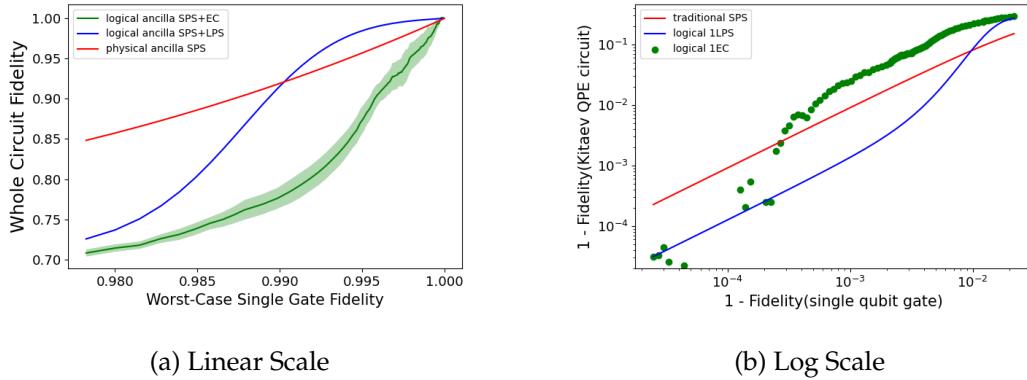


Figure 4.15: Simulation of the logical and traditional Kitaev circuits, averaged over 10 angles evenly distributed in the range $[0, 2\pi)$. Simulation is done for decay time of $T_1 = \infty$, and eigenvector $|\psi\rangle = |0\rangle$. we witness up to an order of magnitude improvement in infidelity.

Resource: Minimal Number of Trials

Here we use the same procedure as in section 4.2.2, plotting the minimal number of trials needed for the algorithm to work correctly. Two equations are of great importance here: equation 3.8 giving the T_2 threshold for which the algorithm fails entirely, and equation 3.9 giving the minimal number of trials required for the algorithm to work, given T_2 is below the threshold determined by equation 3.8.

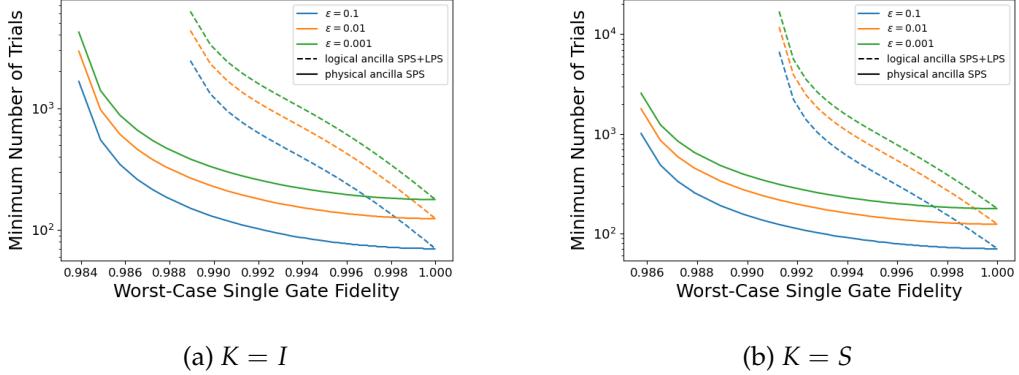


Figure 4.16: Minimal number of trials required for Kitaev’s algorithm to work under T_2 noise and R_z measurement. Algorithm fails according to equation 3.8 below the plotted fidelities. We witness that for accelerated Hamiltonians there is no scenario in which LPS is better physical ancilla if we are interested in sensing with little use of quantum resources.

As in the sensor post selection case, logical post selection here only spoils the results due to the high percentage of information lost. The following part of our study will focus on finding the circumstances in which logical post selection is helpful even in the resource limited scenarios.

Confirming New Error Probability Scaling

Going back a few steps to section 3.5.1, Kitaev’s phase estimation circuit with $K = I$ is almost fault-tolerant - the only non fault tolerant part of it is the first one, the state preparation. Thus we can use this circuit to approximate a fit to the scaling of the error probability after logical post selection. We do logical post selection after the whole iteration, and thus we need to consider the error probability as the probability a single error occurred in a single qubit throughout the whole iteration. Due to the circuit’s transver-

sality this error will not propagate to other qubits. We extract the error probability according to

$$P_{error} = 1 - F^2 \quad (4.4)$$

With F being the calculated fidelity of figure 4.14. We see in figure 4.17 that indeed the best approximation to the scaling is not a forth or a second degree polinomial, but a third degree polinomial - as predicted by our calculations.

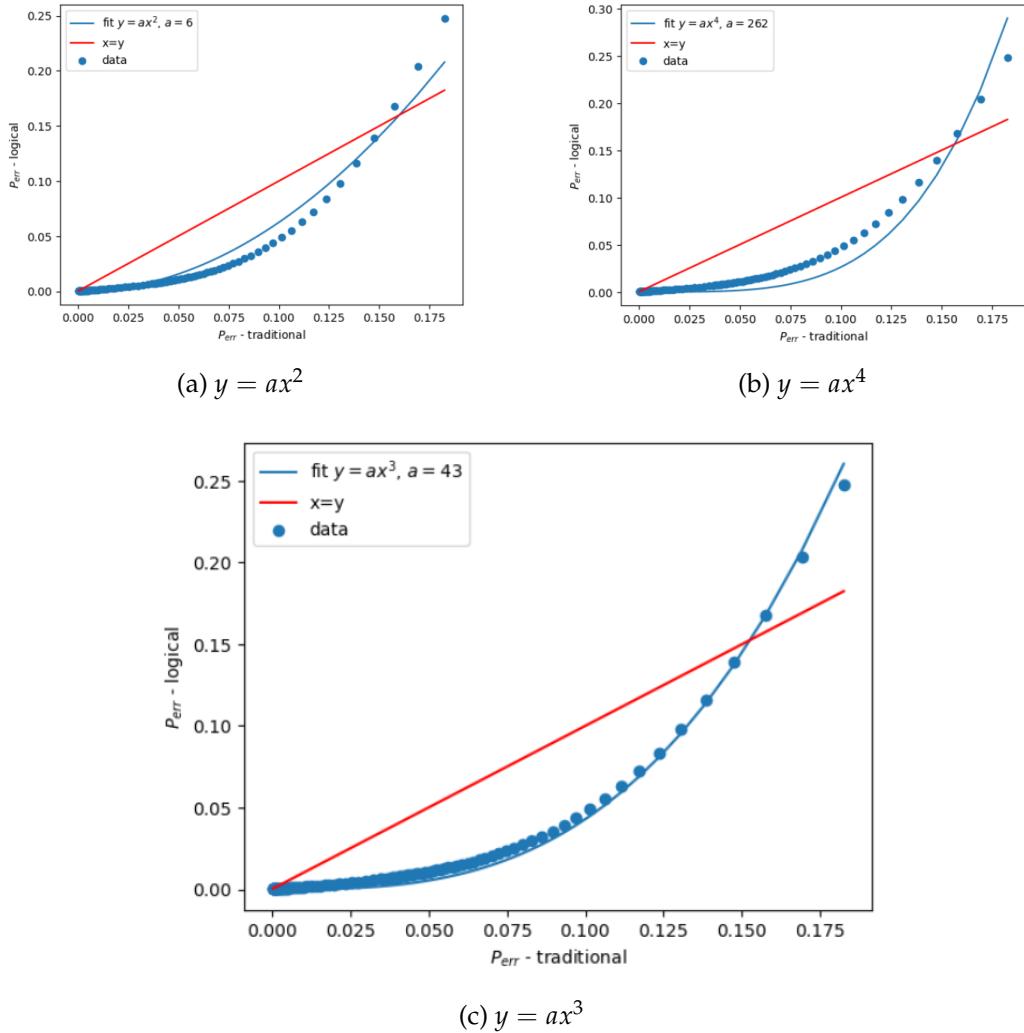


Figure 4.17: Confirming the error probability as predicted by equation 3.5. Error probability extracted from fidelity according to equation 4.4.

4.4.3 IPEA - without Accelerated Hamiltonians

Up until now we had a strong assumption on the rules of the game - we had access to accelerated Hamiltonians. From this point forward, we give up on that assumption. Here we choose to work with the Iterative Phase

Estimation Algorithm (IPEA), and use it to extract probability distributions over all possible results, and measure the standard deviation and mean of those histograms. the IPEA will output two numbers - the estimation for the phase, and the standard deviation (the error) of that estimation. We note that one iteration of IPEA is not that different then one iteration of Kitaev's algorithm - the only difference is the location of the phase kick within the circuit.

Estimating whether or not logical post selection is better then the traditional approach using this algorithm is optimal since here we consider all required iterations to estimate the phase up to some precision, in contrast to the previous work that compared only one iteration. This was possible due to the accelerated Hamiltonian, making all iterations look the same after averaging on different phases.

Exploring the Corrected Statistical Error for Various Case Studies

As stated earlier, the histogram of results has an underlying STD. this standard deviation can be corrected to be an approximation (in the large number of experiments limit) of the experimental statistical error. In this subsection we focus on this error, which takes into account the amount of lost information, and plot it's behaviour with decoherence rates and desired digital precision.

We simulate according to section 3.6.2, and state again our assumptions in table 4.2 on this simulation according to the results in figures 4.18, 4.19, 4.20, 4.21.

We see in figure 4.18 that giving up the assumption of accelerated Hamiltonians is not enough, and we need to further assume access to 'Good'

qubits - qubits with long coherence and relaxation times. We do that and the encouraging results are plotted in figure 4.19, showing an absolute win for the logical method for circuit depths greater than about 50 gates, and for some fidelity threshold. We check the same situation for the more vulnerable case (figure 4.20) with T_1 from initial sensor state $|+\rangle$ which exhibits both energy relaxation and dephasing processes. In this scenario as well we see a win for the logical algorithm, for a certain range of gate fidelities and circuit depths. A similar scenario, where the sensor is put in the excited state and is susceptible to T_1 noise, with perfect ancillas, measuring $R_z(\frac{2\pi}{\sqrt{3}})$ is depicted in figure 4.21. In both scenarios we see a threshold of approximately 0.998 worst case single gate fidelity.

Figure 4.18	Figure 4.19	Figure 4.20	Figure 4.21
Perfect Measurements	Perfect Measurements	Perfect Measurements	Perfect Measurements
$T_1 \approx \infty$	$T_1^{sensor} \approx \infty$, Perfect Ancillas	$T_2^{sensor} \approx \infty$, Perfect Ancillas	$T_2^{sensor} \approx \infty$, Perfect Ancillas
No SPS	No SPS	No SPS	No SPS
No Accelerated Hamiltonian	No Accelerated Hamiltonian	No Accelerated Hamiltonian	No Accelerated Hamiltonian
$\rho_0 = 1\rangle$	$\rho_0 = +\rangle$	$\rho_0 = +\rangle$	$\rho_0 = 1\rangle$

Table 4.2: Assumptions for the simulations of this section

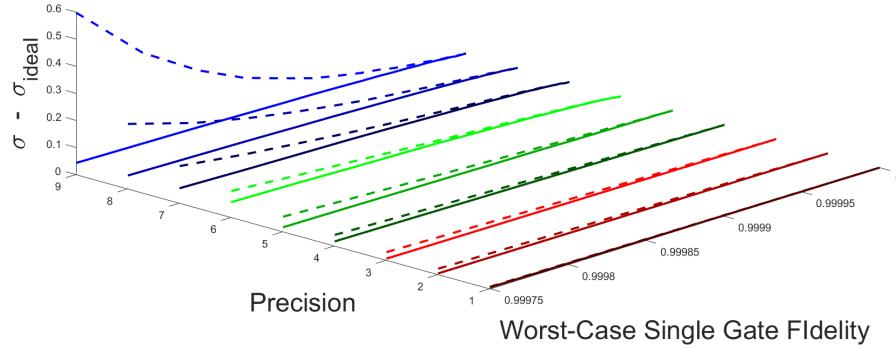
Good Sensor & Noisy Ancilla

Figure 4.18: Measuring $R_z(\frac{2\pi}{\sqrt{3}})$ with $T_1 \approx \infty$ and noisy ancilla, with the Iterative Phase Estimation Algorithm. Dotted line is the logical control, and continuous line is traditional control. σ is calculated according to 3.10 up to the factor \sqrt{N} .

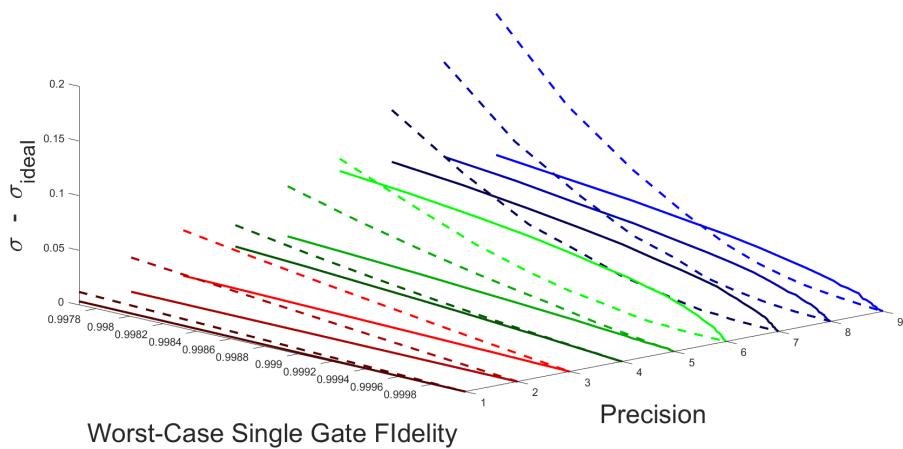
Noisy Sensor & Good Ancilla & T_2 

Figure 4.19: Measuring $R_x(\frac{2\pi}{\sqrt{3}})$ with dephasing sensor and perfect ancilla, with the Iterative Phase Estimation Algorithm. Dotted line is the logical control, and continuous line is traditional control. σ is calculated according to 3.10 up to the factor \sqrt{N} .

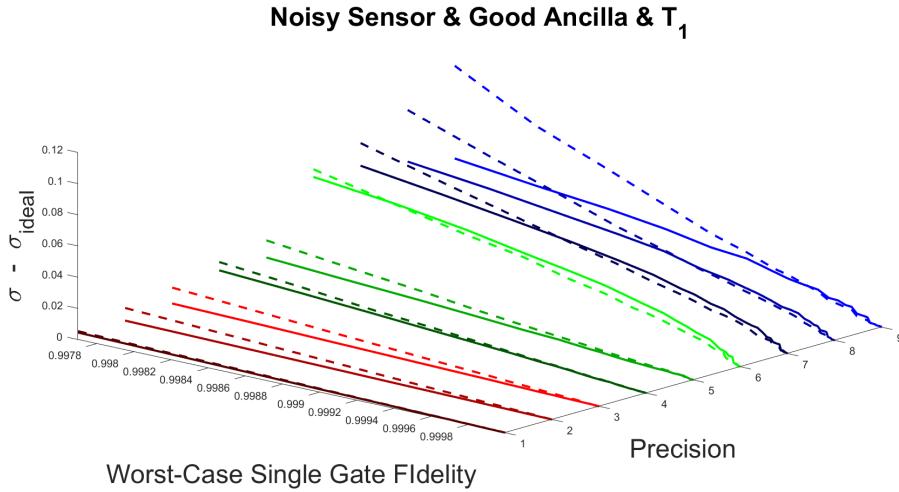


Figure 4.20: Measuring $R_x(\frac{2\pi}{\sqrt{3}})$ with relaxing sensor (T_1 noise according to section) and perfect ancilla, with the Iterative Phase Estimation Algorithm. Dotted line is the logical control, and continues line is traditional control. σ is calculated according to 3.10 up to the factor \sqrt{N} .

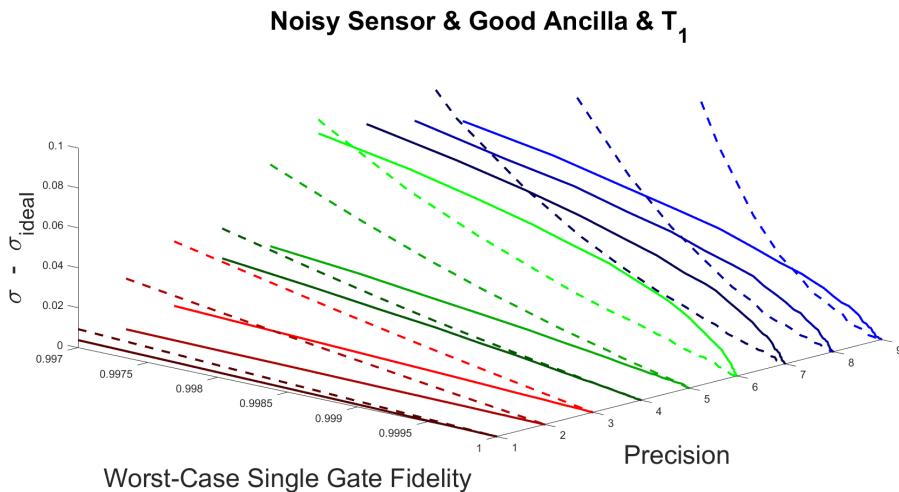


Figure 4.21: Measuring $R_z(\frac{2\pi}{\sqrt{3}})$ with relaxing sensor (T_1 noise according to section) and perfect ancilla, with the Iterative Phase Estimation Algorithm. Dotted line is the logical control, and continues line is traditional control. σ is calculated according to 3.10 up to the factor \sqrt{N} .

Quantifying Success Probability of the Whole Algorithm

As mentioned earlier, it is sometimes helpful to measure each digit a number of times and take a majority vote. We work under the assumption that in NISQ era we will always use the deepest possible circuit to best measure the phase, i.e. at most two possible results contribute to the algorithm's success probability. It is evident that by acting as a post selecting oracle the LPS gadget increased the probability of success after a single run for a wide range of parameters. The histogram is calculated by Eq.3.2 and the success probability is calculated by 3.1 under the assumption that the interval contains only two bins. It is apparent that increasing n leads to some flattening of the probability difference and to a larger proximity of the physical and logical control methods. Figure 4.23 shows a two dimensional map of the success probability difference between physical and logical control methods for different desired precision. the X-axis is the worst case single qubit gate fidelity and the Y-axis is the number of post selected trials for each digit. It is apparent that from a circuit depth of around a few dozens of gates, there is a threshold in approximately 0.98-0.985 worst case gate fidelity. These results are well within the capabilities of today's hardware.

A Closer Look at the Histogram of Results

So why does all that magic happen? The answer to that question takes its clearest form by looking at the resultant probability distribution that is presented in figure 4.24 - showing that LPS brings the histogram of results closer to the ideal histogram.

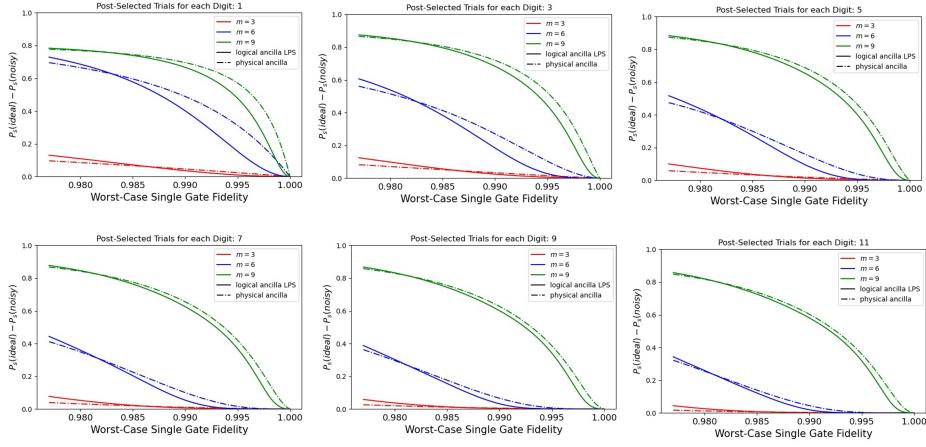


Figure 4.22: Scaling of the Success probability as a function of noise for different n 's. For each sub-figure we see a cross-over where the probability of success is higher when using LPS.

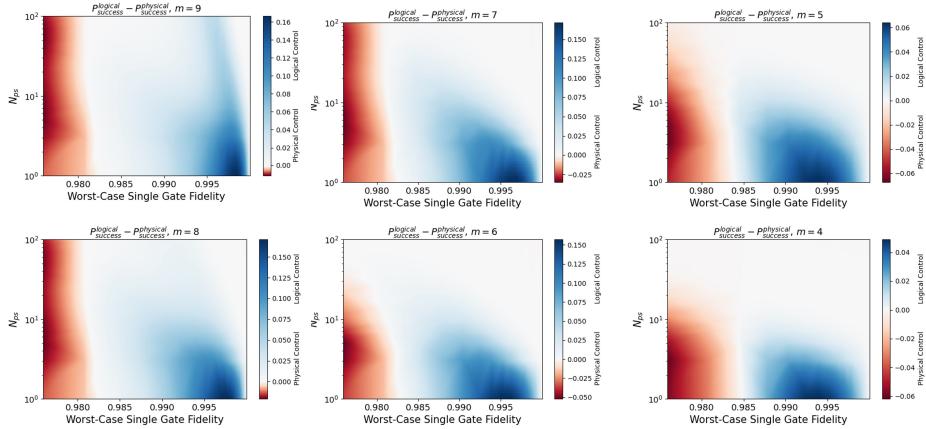


Figure 4.23: A two dimensional map of the success probability difference between physical and logical control methods for different desired precision. It is evident that for gate fidelities higher than approximately 0.985 for each desired precision, the probability of success is higher when using LPS, and gets closer to the traditional approach as the number of post selected trials for each digits is larger.

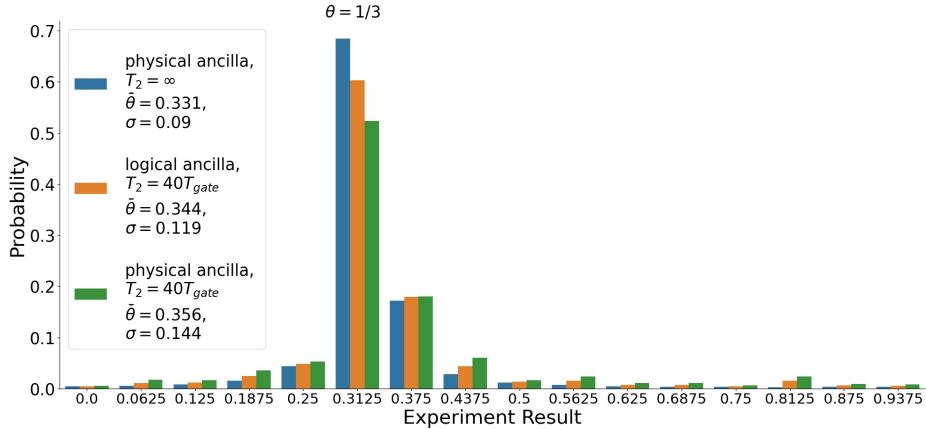


Figure 4.24: A showcase of the probability distribution of all possible results of IPEA, measuring $R_x(2\pi/3)$ up to 4 binary digits, in the presence of decoherence. It is apparent that the histogram of results is less biased after LPS.

Focusing on the second case, the data of figure 4.19, we can plot 4.25 the mean of the circular probability distribution both in linear scale and in logarithmic scale. It is easy to see that there exists parameter regimes of deep circuits and approximately a minimum of 0.986 worst case single gate fidelity such that the logical control gives a better estimate of the mean. The shaded regions indicate on parameter regimes where it is beneficial to use LPS accuracy-wise. We see improvement of up to one order of magnitude in the distance of the mean from the ideal mean, before reaching the desired digital accuracy.

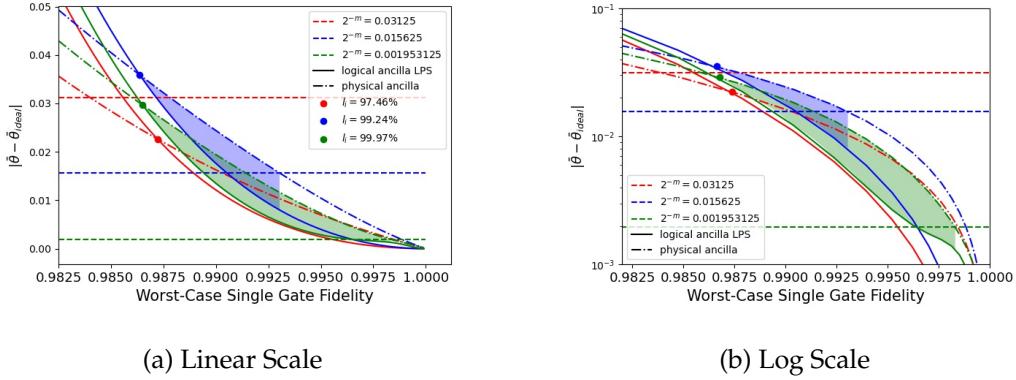


Figure 4.25: Measuring the mean of the circular probability distribution of results for a number of desired precisions m . The shaded regions indicate parameter regimes where it is beneficial to use LPS accuracy-wise. It is easy to see that there exists parameter regimes of deep circuits and approximately a minimum of 0.986 worst case single gate fidelity such that the logical control gives a better estimate of the mean.

Further results focus on the statistical error of the measurements by equation 3.10 and on the minimum number of trials required to obtain digital accuracy by equation 3.11. It is evident from 4.26(c,d) that up from approximately 0.997 worst case single gate fidelity we start observing an improvement of results in the sense of a lower sensing time, and from 4.26(a) and 4.25 that up from approximately 0.986 worst case single gate fidelity we start observing an improvement of results in the sense of accurate measurement.

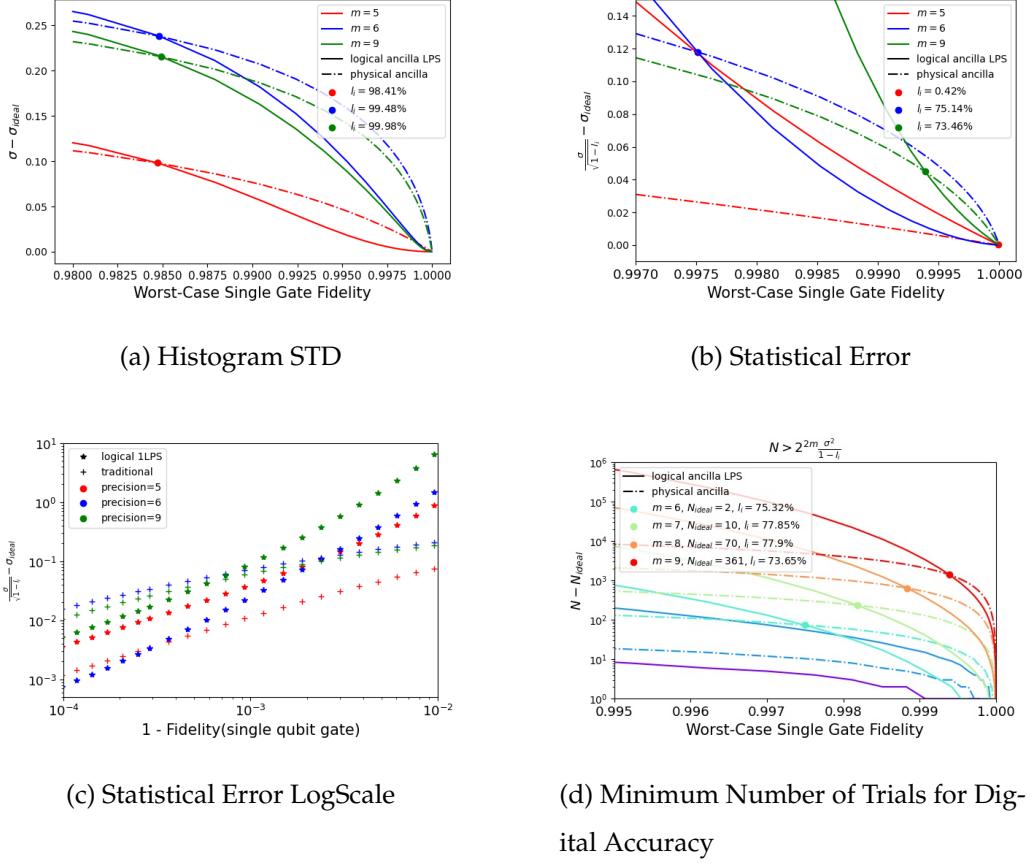


Figure 4.26: (a) Base standard deviation of the histogram. (b) Statistical error in estimating the phase. (c) Statistical error in estimating the phase, in Log Scale. (d) The assessed minimum number of trials needed to reach digital accuracy by Eq.3.11. It is evident from 4.26(c,d) that up from approximately 0.997 worst case single gate fidelity we start observing an improvement of results in the sense of a lower sensing time, and from 4.26(a) and 4.25 that up from approximately 0.986 worst case single gate fidelity we start observing an improvement of results in the sense of accurate measurement.

5 Conclusion & Outlook

We have introduced the concept of logical-physical qubit interaction and found a unique parameter regime where it is beneficial to use this kind of interaction, as a function of the circuit depth and the worst case single gate fidelity, under the presence of dephasing being the most significant cause of error [11, 46]. We have defined a number of settings describing real-world sensing scenarios and we have used our idea to show an improvement of up to an order of magnitude for algorithmic quantum sensing, by encoding the noise into a bigger Hilbert space and post selecting the purified state. Although the gate fidelities needed for the method to contribute to Heisenberg-limit scaling are far beyond the reach of today's state-of-the-art technology, we have witnessed situations in which using logical post selection leads to more precise measurements - where the ancilla are much more resiliant to noise then the sensor, or where Hamiltonian fast-forwarding is possible. The concept of hybrid logical-physical interaction has a considerable depth for further research. We think the concepts of logical control and logical post selection can be efficient in a big number of algorithms and purposes, especially ones that require long-lived ancilla qubits like state distillation, error mitigation and algorithmic sensing [Czarnik et al., Piveteau et al.]. The most significant drawback of

the method is the lost information due to post selection, and recent studies show promise to discover a method of simulating without the need for post selection [21]. This idea opens a wide set of frontiers to be searched, like finding a way to implement a an entangling gate between a logical qubit and a physical qubit with minimum error or error propagation, possibly by using flag fault-tolerance [9, 10, 36, Tansuwanont et al.]. We have defined an attribute of quantum error correction codes that enables this kind of interaction. What other attributes are there, and how do they project on the topology of the interaction or on the general structure of these QECC? Can it be implemented with today's most promising codes, the surface codes [17]? These are examples of questions to be asked around the concept of logical-physical qubit interaction.

A Code Guide

A.1 About

The simulation of quantum circuit comprising of individual qubits is a subject under study. A tool for simulating quantum circuits with decoherence (energy relaxation and dephasing) is hard to find, especially with precise control over the qubit's decoherence times T_1 , related to energy relaxation, and T_2 , related to dephasing. This is a documentation on the basic tool written for my thesis research, with a few examples on how to use it. This package includes the basic simulation tool for noisy or ideal circuits, and notebooks that re-create the results of my research. The documentations includes some basic examples of how to use the tool.

A.2 Installation

A.2.1 required packages

- Numpy
- QisKit (will not be required in future work)

- QuTiP
- SciPy

A.2.2 Download

All relevant code is open for use in the following url:

https://github.com/nadavcarmel40/paper_recalc.

To install and use the package, just install

<https://qutip.org/docs/latest/installation.html>

by following the instructions on the above web page.

The basic tool enabling the simulations can be found under the ‘simulators’ folder in the attached GitHub repository. ‘BigStepSimulator’ is a state-vector simulator and ‘SmallStepSimulator’ is a density matrix simulator. The state vector simulator is fast and can simulate quantum circuits without noise, and the density matrix simulator is slower and can simulate noisy circuits.

A.3 The Simulators

The basic tools created for my project are the simulators. They are constructed with the same user interface, so this documentation will show examples with the density matrix simulator but is relevant for both.

A.3.1 Creating a Quantum Register

The first and most basic step of every simulation is the creation of the underlying quantum register. This subsection is a walk-through of how to create this register.

Here we will introduce the set of parameters that define the quantum register:

- The register is built from N qubits, that can interact and be entangled with each other.
- The register starts from some state (which is a density matrix - a positive semi-definite hermitian matrix with trace one) in the N -qubit Hilbert-Space, $\rho_0 \in M_{2^N, 2^N}(\mathbb{C})$.
- Each qubit q has the amplitude-damping time T_1^q . If all qubits have the same T_1 , we can just pass the parameter $T1$. Else, we set $T1$ to some unimportant constant and pass the Python list of amplitude damping times, ordered as are the qubits, as the parameter $T1s = [T_1^0, \dots, T_1^N]$.
- Each qubit q has the pure dephasing time T_2^q . If all qubits have the same T_2 , we can just pass the parameter $T2$. Else, we set $T2$ to some unimportant constant and pass the Python list of pure dephasing times, ordered as are the qubits, as the parameter $T2s = [T_2^0, \dots, T_2^N]$.
- The simulation works with finite size time steps, each of length dt . Default is $10^{-4}T_1$ assuming all qubits have the same T_1 .
- Another important parameter is the time it takes to apply a single

gate, T_{gate} . We usually take this time to be '1' in arbitrary units, and work in the units of this time. default is $20 * dt$.

- The last parameter required to define a quantum register is a list of all of the qubit's frequencies. Default is 6[GHZ] for all qubits, and this essentially has no impact on the simulation results. The difference is whether or not to work in the qubit's rotating frame.

So, if we want to create a quantum register, we first need to import the relevant classes:

```
from qutip import *
from simulators.BigStepSimulation import \
    EfficientQuantumRegister
from simulators.SmallStepSimulation import \
    InCoherentQuantumRegister
```

And to create a register comprised of $N = 2$ qubits starting from the state $\rho_0 = | + 1 \rangle$ with $T_{gate} = 1$, $T_1/T_{gate} = 10^3$ and $T_2/T_{gate} = 10^4$ for both qubits, with $dt = T_{gate}/20$, we can run the following code block:

```
plus = 1/np.sqrt(2)*(fock_dm(2,0) + fock_dm(2,1))
rho0 = tensor([plus, fock_dm(2,1)])
register = InCoherentQuantumRegister(2, rho0, 1e3, 1e4, \
    Tgate=1, dt=1/20)
```

Or if we want to initialize the register such that qubit 2 has different lifetimes from qubit 1, we can initialize it like so:

```
T1s = [1e3,2e3]
T2s = [1e4,3e4]
qr = InCoherentQuantumRegister(2,rho0,None,None,T1s=T1s,\ 
                                T2s=T2s,Tgate=1,dt=1/20)
```

We can also control all decoherence related parameters of the register via the function 'setError(self, dephase=True, amplitude_damp=True, T1s=None, T2s=None)'. It enables us to switch T_1, T_2 processes on and off, and updates the qubit's lifetimes. For example, if we want to switch amplitude damping off, we can just run:

```
qr.setError(amplitude_damp=False)
```

Some important and useful attributes of the quantum register are:

- self.state - The state of the register. Can be accessed any time.
- self.qI - The identity matrix of the register
- self.dt, self.Tgate - The times defining how big are the time steps of the simulation. Can be changed, for example, before and after the activation of some gates if one wants to make them last a different amount of time, or have better or worse trotterization.
- self.dephase, self.amplitude_damp - Boolean values that can be changed if we want to switch T_1, T_2 processes on and off.
- self.Sx, self.Sy, self.Sz - Lists of the Pauli operators acting on the register's qubits. For example, self.Sx[q] is the Pauli-X operator acting on the qubit 'q'.

A.3.2 Simulating a Quantum Circuit

Now, we can run a simple quantum circuit on the quantum register we have just created. All quantum circuits are passed to the register as a list of commands, each command is a physical command acting on a physical qubit. The list of commands is built in a logical way and can be understood easily by looking at figure 3.3, describing a general simulation of gates.

The commands is a list containing all big time steps T, in each one a number of different gates can be acted upon the register. Each of these lists is a list itself, containing all gates acting at that moment. Each gate can be represented as a tuple object, in the form ('c',q₁,q₂,operator).

'c' is the command. It can be one of the Strings:

$$C \in \{i, X, Y, Z, H, CNOT, CZ, Rx, Ry, Rz, SingleQubitOperator, m\}$$

Note that every controlled operator can be constructed as a series of CNOTs and single qubit gates be Nielsen and Chaung.

'q₁' is the qubit the operator is acted upon.

'q₂' is the control qubit in a two-qubit gate (Else, None).

'operator' is some extra information needed for the gate. It can be:

- None for defined gates (H,X,Y,Z,CNOT,CZ).
- Angle for Rotations (Rx,Ry,Rz) in Radians.
- Number of gates to wait (int) for the identity gate 'c'='i'.
- A 2x2 Numpy or QuTiP matrix for general single qubit operator.

For example, creating a quantum register of two qubits starting in the |00⟩ state, and creating the bell state A.1, can be simulated using the simple code block:

```

T1s = [1e3,2e3]
T2s = [1e4,3e4]
rho0 = tensor([fock_dm(2,0),fock_dm(2,0)])
qr = InCoherentQuantumRegister(2,rho0,None,None,T1s=T1s \
, T2s=T2s, Tgate=1, dt=1/20)
qr.run([(H,0,None,None)],[(CNOT,1,0,None)])

```

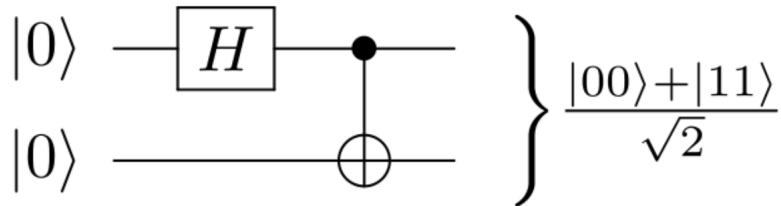


Figure A.1: The Bell State

A.3.3 Visualizing The State History of a Qubit

Here we will show two sanity checks for the simulation, confirming that the decoherence is correct and that all defined gates act as we expect them to. For this purpose we will simulate a noisy register of only one qubit, and we will introduce some new attributes of the quantum register.

Confirming Decoherence

Here we do T_2 relaxation on the state $|+\rangle\langle+| = \frac{1}{2}(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix})$, and expect to see that the state evolves as $\rho(t) = \frac{1}{2}(\begin{smallmatrix} 1 & e^{-t/T_2} \\ e^{-t/T_2} & 1 \end{smallmatrix})$. we plot (fig.A.2 (a)) the coherence and see how close is the graph of $\rho_{10}(t)$ to e^{-t/T_2} .

The simulation is done by executing the code block below. A similar code, setting 'dephase'=False and 'amplitude_damp'=True can simulate energy relaxation on the state $|+\rangle\langle+| = \frac{1}{2}(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix})$, and we expect to see that the state evolves as $\rho(t) = \frac{1}{2}(\begin{smallmatrix} 2-e^{-t/T_1} & e^{-t/2T_1} \\ e^{-t/2T_2} & e^{-t/T_1} \end{smallmatrix})$. We plot (fig.A.2 (b,c,d)) all elements of the density matrix to see that they are well-behaved. Overall the decoherence simulation is as expected and plotted in figure A.2

```
def get_rho_from_bloch(x,y,z):
    """
    (x,y,z) is point in bloch sphere.
    returns density matrix
    """
    return 0.5*qeye(2) + x/2*sigmax() + y/2*sigmay() \
        + z/2 * sigmaz()

import matplotlib.pyplot as plt
from scipy.optimize import curve_fit

plus = 1/np.sqrt(2)*(basis(2,0)+basis(2,1))
T2 = 1000
qubit = InCoherentQuantumRegister(1,plus*plus.dag())\
    , T1=1, T2 = T2, Tgate=1, dt = 1/20)
qubit.setError(dephase = True, amplitude_damp=False)
qubit.setCollectData(data = True, bloch=True)
qubit.run([[('i', 0, None, 3000)]])
xs, ys, zs = qubit.history[0][0], qubit.history[0][1], \
```

```

    qubit.history[0][2]
time = qubit.times
print('constructing coherence')
coherence = []
for i in range(len(xs)):
    rho = get_rho_from_bloch(xs[i], ys[i], zs[i])
    coherence.append(rho[0,1])

def func(x,a):
    return 1/2*np.exp(-x/a)
popt, pcov = curve_fit(func, time, coherence)
print(popt)
plt.plot(np.array(time), func(np.array(time),*popt),\
         label = 'fit $y=0.5e^{-x/a}$, $a=' + str(popt[0]))
plt.style.use('default')
plt.title('pure dephasing of $|+\rangle$ with \
           $T_2/T_{gate}=10^3$')
plt.xlabel('$t/T_{gate}$')
plt.ylabel('$|\rho_{01}|$')
plt.plot(time, coherence, label='simulation')
plt.legend()
plt.show()

```

So, what's new in that code?

- self.setCollectData() - Gets the two boolean parameters 'data' and 'bloch' (with default values False). 'data' controls the collection 'self.history' and 'self.purities'. 'bloch' controlles the

collection of bloch sphere figures for each qubit, 'self.blochs' and 'self.bloch3ds'.

- self.history - List of lists. Each list is of the form [xs,ys,zs] such that (xs[i],ys[i],zs[i]) is the position of the qubit's state on bloch sphere in time $i * dt$.
 - self.purities - If the register has N qubits, then this is a list of length $N + 1$. the first N elements are lists such as the i 'th list contains the purity as a function of time dt of the i 'th qubit, and the last list (in place $N + 1$) is the purity of the entire register as a function of time. Purity is calculated as $Tr(\rho^2)$.
 - self.blochs - A list of QuTiP's bloch spheres, one for each qubit, with the qubit's trajectory.
 - self.bloch3ds - A three dimentional version of the above.
- self.times - a list of dt times the qubit had lived through.

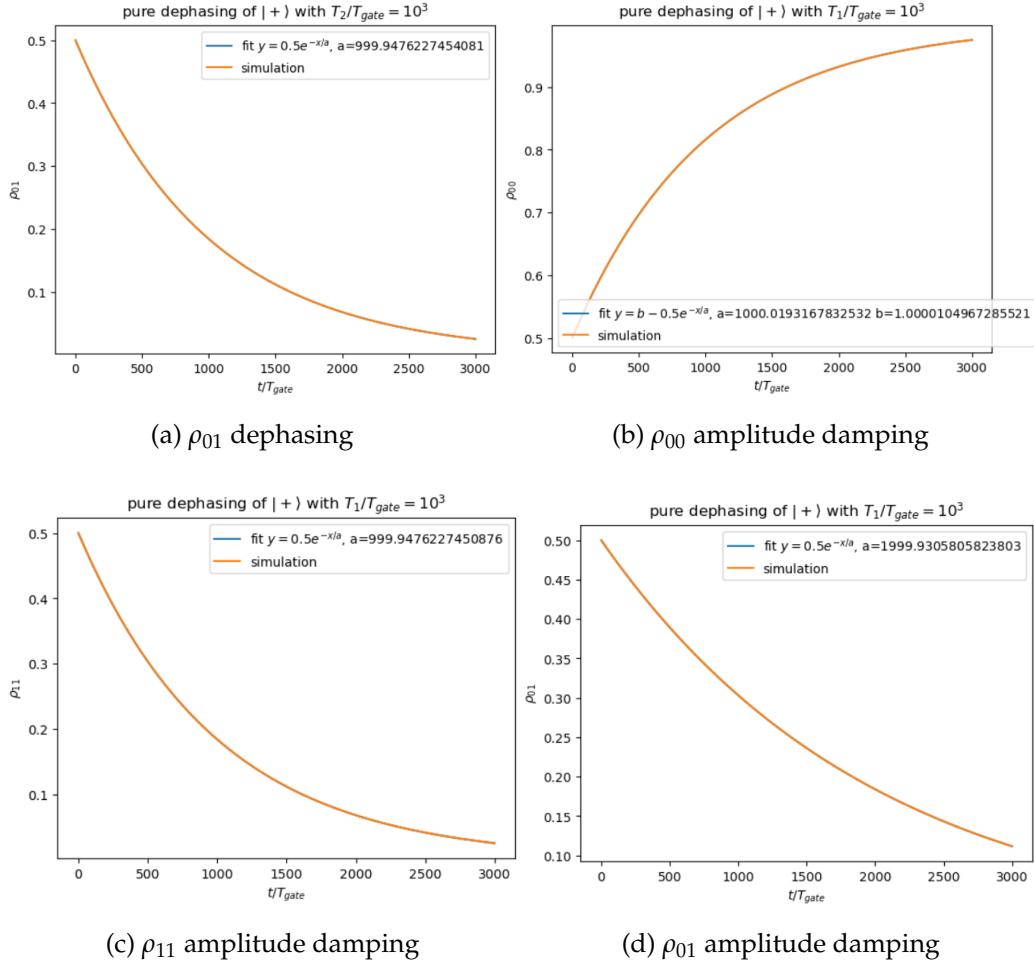


Figure A.2: Relaxation for three decoherence times. Fit and simulation are the same.

Confirming Defined Gates

Here we show the action of all previously defined gates on one (or two) qubits, on the Bloch Sphere. One can reproduce these figures by running the code blocks similar to the one below:

```
qubit.setCollectData(data = True, bloch=False)
qubit.run([(('Rz', 0, None, np.pi / 4)])]
bloch = Bloch()
bloch.add_points([qubit.history[0][0], \
                  qubit.history[0][1], qubit.history[0][2]])
bloch.show()
```

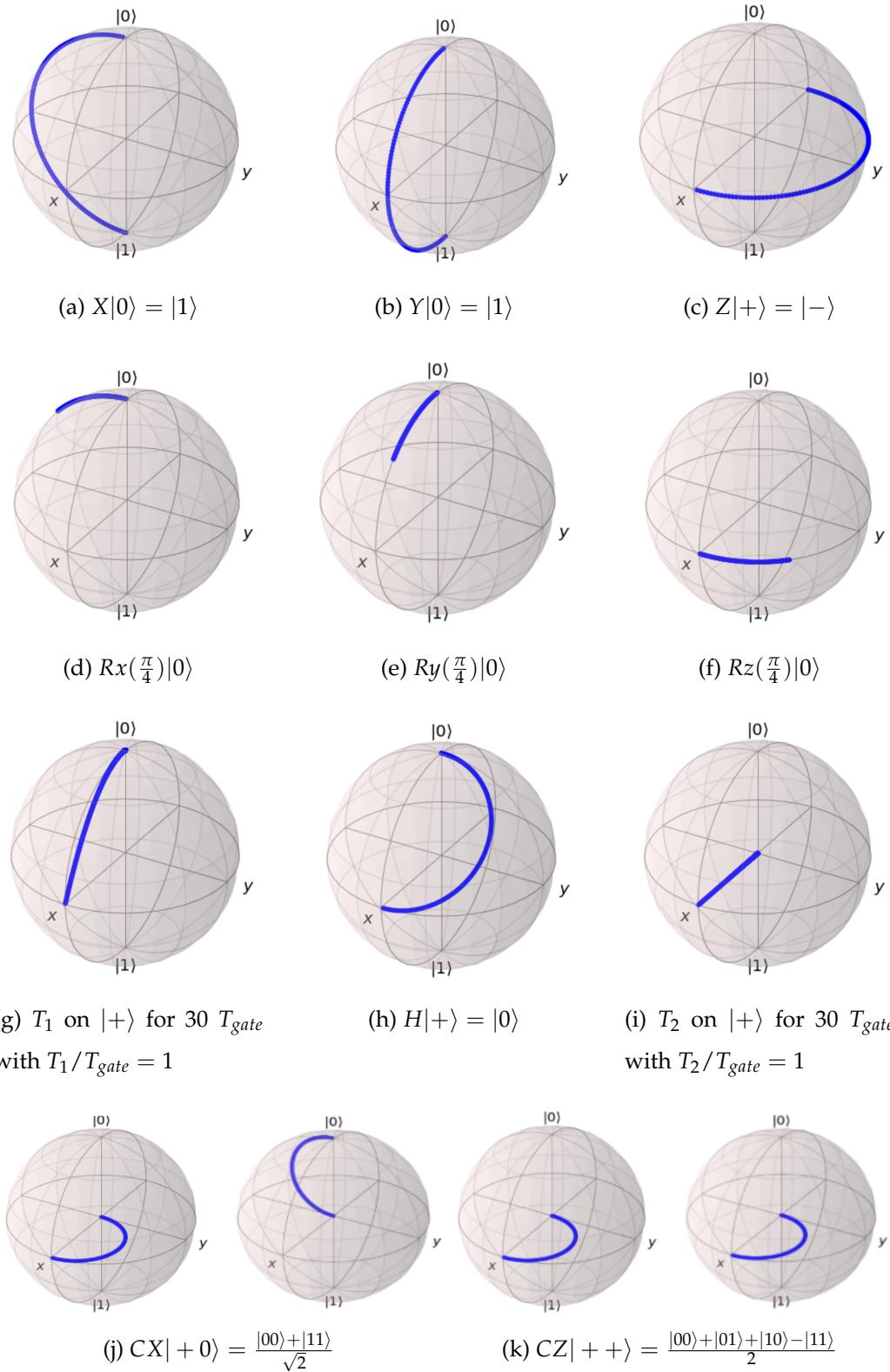


Figure A.3: All Defined Gates

Bibliography

- [Aharonov] Aharonov, D. Fault-Tolerant Quantum Computation With Constant Error. pages 1–18.
- [2] Aharonov, Y. and Popescu, S. (2002). Measuring Energy, Estimating Hamiltonians, and the Time-Energy Uncertainty Relation. Technical report.
- [3] Ahmadi, H. and Chiang, C.-F. (2010). Quantum Phase Estimation with Arbitrary Constant-precision Phase Shift Operators.
- [4] Arrad, G., Vinkler, Y., Aharonov, D., and Retzker, A. (2013). Increasing sensing resolution with error correction.
- [5] Arvidsson-Shukur, D. R., Yunger Halpern, N., Lepage, H. V., Lasek, A. A., Barnes, C. H., and Lloyd, S. (2020). Quantum advantage in postselected metrology. *Nature Communications*, 11(1):1–7.
- [6] Atia, Y. and Aharonov, D. (2017). Fast-forwarding of Hamiltonians and exponentially precise measurements. *Nature Communications*, 8(1).
- [7] Butler, K. and Stephens, M. (2016). The Distribution of a Sum of Binomial Random Variables. (February).

- [8] Cappellaro, P., Layden, D., Jiang, L., Zhou, S., Zhang, M., and Preskill, J. (2019). Error-corrected quantum sensing. page 51.
- [9] Chao, R. and Reichardt, B. W. (2017). Fault-tolerant quantum computation with few qubits.
- [10] Chao, R. and Reichardt, B. W. (2019). Flag fault-tolerant error correction for any stabilizer code.
- [11] Chapeau-Blondeau, F. and Belin, E. (2020). Fourier-transform quantum phase estimation with quantum phase noise. *Signal Processing*, 170(March):1–24.
- [12] Cheng, S., Cao, C., Zhang, C., Liu, Y., Hou, S.-Y., Xu, P., and Zeng, B. (2020). Simulating Noisy Quantum Circuits with Matrix Product Density Operators.
- [13] Cîrstoiu, C., Holmes, Z., Iosue, J., Cincio, L., Coles, P. J., and Sornborger, A. (2020). Variational fast forwarding for quantum simulation beyond the coherence time. *npj Quantum Information*, 6(1).
- [14] Cruz, P. M. Q., Catarina, G., and Gautier, R. (2020). Optimizing quantum phase estimation for the simulation of Hamiltonian eigenstates. pages 1–16.
[Czarnik et al.] Czarnik, P., Arrasmith, A., Cincio, L., and Coles, P. J. Qubit-efficient exponential suppression of errors.
- [16] Degen, C. L., Reinhard, F., and Cappellaro, P. (2016). Quantum sensing.

- [17] Dennis, E., Kitaev, A., Landahl, A., and Preskill, J. (2002). Topological quantum memory. *Journal of Mathematical Physics*, 43(9):4452–4505.
- [18] Dobšíček, M., Johansson, G., Shumeiko, V., and Wendin, G. (2007). Arbitrary accuracy iterative quantum phase estimation algorithm using a single ancillary qubit: A two-qubit benchmark. *Physical Review A - Atomic, Molecular, and Optical Physics*, 76(3).
- [19] Garc, I. and Shepelyansky, D. L. (2007). Quantum phase estimation algorithm in presence of static imperfections. (Section 4).
- [20] Herrera-Martí, D. A., Gefen, T., Aharonov, D., Katz, N., and Retzker, A. (2014). Quantum Error-Correction-Enhanced Magnetometer Overcoming the Limit Imposed by Relaxation.
- [21] Ippoliti, M. and Khemani, V. (2021). Postselection-Free Entanglement Dynamics via Spacetime Duality. *Physical Review Letters*, 126(6):60501.
- [Kais] Kais, S. A Universal Quantum Circuit Scheme For Finding Complex Eigenvalues.
- [23] Kandala, A., Wei, K. X., Srinivasan, S., Magesan, E., Carnevale, S., Keefe, G. A., Klaus, D., Dial, O., and McKay, D. C. (2020). Demonstration of a High-Fidelity CNOT for Fixed-Frequency Transmons with Engineered ZZ Suppression.
- [24] Kapourniotis, T. and Datta, A. (2019). Fault-tolerant quantum metrology. *Physical Review A*, 100(2).

- [25] Khatri, S., LaRose, R., Poremba, A., Cincio, L., Sornborger, A. T., and Coles, P. J. (2018). Quantum-assisted quantum compiling.
- [26] Kitaev, A. Y. (1995). Quantum measurements and the Abelian Stabilizer Problem. pages 1–22.
- [27] Koch, J., Yu, T. M., Gambetta, J., Houck, A. A., Schuster, D. I., Majer, J., Blais, A., Devoret, M. H., Girvin, S. M., and Schoelkopf, R. J. (2007). Charge-insensitive qubit design derived from the Cooper pair box. *Physical Review A - Atomic, Molecular, and Optical Physics*, 76(4):1–21.
- [28] Krantz, P., Kjaergaard, M., Yan, F., Orlando, T. P., Gustavsson, S., and Oliver, W. D. (2019). A quantum engineer’s guide to superconducting qubits. *Applied Physics Reviews*, 6(2):1–66.
- [29] Layden, D. and Cappellaro, P. (2018). Spatial noise filtering through error correction for quantum sensing. *npj Quantum Information*, 4(1).
- [30] Ma, Z., Gokhale, P., Zheng, T.-X., Zhou, S., Yu, X., Jiang, L., Maurer, P., and Chong, F. T. (2020). Adaptive Circuit Learning for Quantum Metrology.
- [31] Meyer, J. J. (2021). Fisher Information in Noisy Intermediate-Scale Quantum Applications.
- [32] Nielsen, M. A. and Chuang, I. L. (2010). *Quantum computation and quantum information*. Cambridge University Press.
- [33] O’Brien, T. E., Tarasinski, B., and Terhal, B. M. (2018). Quantum

phase estimation of multiple eigenvalues for small-scale (noisy) experiments.

- [Piveteau et al.] Piveteau, C., Sutter, D., Bravyi, S., Gambetta, J. M., and Temme, K. Error mitigation for universal gates on encoded qubits. pages 1–15.
- [35] Preskill, J. (2018). Lecture Notes for Ph219 / CS219 : Quantum Information Chapter 3. (October).
- [36] Reichardt, B. W. (2018). Fault-tolerant quantum error correction for Steane’s seven-qubit color code with few or no extra qubits.
- [37] Reiter, F., Sørensen, A. S., Zoller, P., and Muschik, C. A. (2017). Dissipative quantum error correction and application to quantum sensing with trapped ions. *Nature Communications*, 8(1).
- [Santagati et al.] Santagati, R., Wang, J., Gentile, A. A., Paesani, S., Wiebe, N., McClean, J. R., Shadbolt, P. J., Bonneau, D., Silverstone, J. W., Tew, D. P., Zhou, X., and Thompson, M. G. Witnessing eigenstates for quantum simulation of Hamiltonian spectra.
- [39] Steck, D. A. (2012). Quantum and Atom Optics, Lecture Notes. page 843.
- [40] Takita, M., Inoue, K., Lekuch, S., Minev, Z. K., Chow, J. M., and Gambetta, J. M. (2021). Exploiting dynamic quantum circuits in a quantum algorithm with superconducting qubits.
- [Tansuwannont et al.] Tansuwannont, T., Chamberland, C., and Leung, D. Flag fault-tolerant error correction, measurement, and quantum computation for cyclic CSS codes. Technical report.

- [42] Temme, K., Bravyi, S., and Gambetta, J. M. (2017). Error Mitigation for Short-Depth Quantum Circuits. *Physical Review Letters*, 119(18):1–5.
- [Unden et al.] Unden, T., Balasubramanian, P., Louzon, D., Vinkler, Y., Plenio, M. B., Markham, M., Twitchen, D., Lovchinsky, I., Sushkov, A. O., Lukin, M. D., Retzker, A., Naydenov, B., McGuinness, L. P., and Jelezko, F. Quantum metrology enhanced by repetitive quantum error correction. Technical report.
- [44] Vool, U. and Devoret, M. (2017). Introduction to quantum electromagnetic circuits. *International Journal of Circuit Theory and Applications*, 45(7):897–934.
- [45] Yoder, T. J., Takagi, R., and Chuang, I. L. (2016). Universal fault-tolerant gates on concatenated stabilizer codes. *Physical Review X*, 6(3).
- [46] Yu, A. and Chernyavskiy, A. Y. (2021). Proceedings of spie. (March 2019).
- [47] Zhou, S., Zhang, M., Preskill, J., and Jiang, L. (2018). Achieving the Heisenberg limit in quantum metrology using quantum error correction.
- [48] Zhou, S., Zou, C. L., and Jiang, L. (2020). Saturating the quantum Cramér-Rao bound using LOCC. *Quantum Science and Technology*, 5(2):36–38.