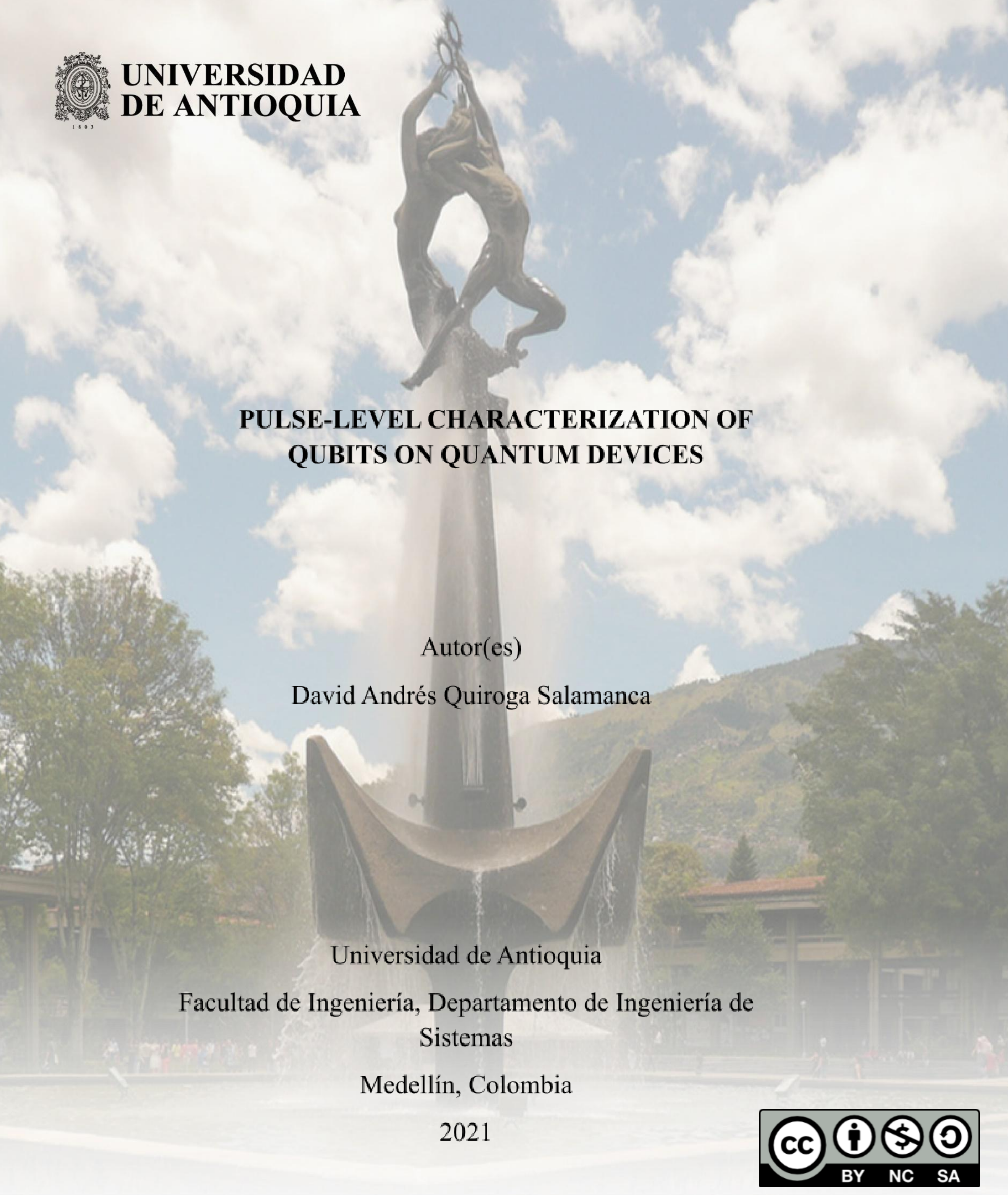


UNIVERSIDAD
DE ANTIOQUIA

PULSE-LEVEL CHARACTERIZATION OF QUBITS ON QUANTUM DEVICES

Autor(es)

David Andrés Quiroga Salamanca

Universidad de Antioquia

Facultad de Ingeniería, Departamento de Ingeniería de Sistemas

Medellín, Colombia

2021

Pulse-level characterization of qubits on quantum devices

David Andrés Quiroga Salamanca

Tesis o trabajo de investigación presentada(o) como requisito parcial para optar al título de:

Ingeniero de sistemas

Asesores (a):

Fredy Alexander Rivera Vélez PhD

Raphael C. Pooser PhD

Línea de Investigación:

Computación Cuántica

Grupo de Investigación:

Sistemas Embebidos e Inteligencia Computacional (SISTEMIC)

Universidad de Antioquia
Facultad de Ingeniería, Departamento de Ingeniería de Sistemas
Medellín, Colombia
2021

DEDICATION

To my mom, my best friend and those who have believed in me.

ACKNOWLEDGMENTS

This work was completed with the collaboration of professors Fredy A. Rivera Velez and Javier F. Botia Valderrama from Universidad de Antioquia and Oak Ridge National Laboratory staff scientists Prasanna Date in the Beyond Moore group and Raphael C. Pooser in the Quantum Information Sciences group.

ABSTRACT OF THE DISSERTATION

PULSE-LEVEL CHARACTERIZATION OF QUBITS ON QUANTUM DEVICES

by

David Andres Quiroga

Universidad de Antioquia, 2021

Medellín, Antioquia

Internal advisor Fredy Rivera, Universidad de Antioquia

External advisor Raphael C. Pooser, Oak Ridge National Laboratory

Recent research has tackled the problem of mitigating noise present in quantum computers in the Noisy Intermediate-Scale Quantum (NISQ) era to enable precise computations and to benefit from the intrinsic properties of quantum mechanics [Pre18]. For this matter, an important task is the characterization of qubits available in quantum devices so as to provide insights on how to reduce noise on the final output of a quantum circuit. Characterization comprises analysis of noise sources and this information can be used to reduce noise with methods such as Cycle Benchmarking [EWP+19], Quantum Error Mitigation [SEFT20], Quantum Error Correction [Sho95] and others [KLR+08] [TBG17]. Here, we study optimization of pulses through Quantum Optimal Control (QOC) to obtain higher gate fidelity [BCR10]. We will explore an algorithm that performs calibration on specific quantum gates by implementing optimized pulse schedules to subsequently use the algorithm for analysis of noise sources. Using calibrated gates as input, several benchmarking protocols, including pulse noise extrapolation, leakage analysis from quantum optimal control, and machine learning based classification of qubit readout, will be tested to extract precise information on how noise influences the analyzed qubits. We will explain and discuss different techniques for obtaining properties of qubits and quantum comput-

ers. We will implement state discrimination with a Machine Learning (ML) focus to analyze readout errors caused by factors such as cross-talk and leakage into higher quantum states. We will perform noise fitting of optimized pulses and evaluate the effectiveness of important quantum algorithms at the pulse level.

Investigaciones recientes han abarcado el problema de mitigar ruido presente en computadores cuánticos en la era NISQ para permitir computaciones precisas y para encontrar ventajas en las propiedades intrínsecas de la mecánica cuántica [Pre18]. Para tal efecto, una tarea importante es la caracterización de qubits disponibles en computadores cuánticos para proveer información sobre cómo reducir ruido en la salida final de un circuito cuántico. La caracterización comprende el análisis de fuentes de ruido y esta información puede ser usada para reducir ruido con métodos como Cycle Benchmarking [EWP^{+}19], Quantum Error Mitigation [SEFT20], Quantum Error Correction [Sho95] y otros [KLR^{+}08] [TBG17]. Aquí estudiamos optimización de pulsos a través de QOC para obtener fidelidades de compuerta más altas [BCR10]. Exploraremos un algoritmo que realiza calibración a compuertas cuánticas específicas implementando pulsos optimizados para consecuentemente utilizar el algoritmo para análisis de fuentes de ruido. Usando compuertas calibradas como entrada, varios protocolos de benchmarking incluyendo extrapolación de ruido, análisis de fuga con control cuántico óptimo y clasificación de datos de salida de qubits basada en machine learning serán probados para extraer información precisa de cómo el ruido influye los qubits analizados. Explicaremos y discutiremos diferentes técnicas para obtener propiedades de qubits y de computadores cuánticos. Implementaremos discriminación de estados con un enfoque en ML para alanizar errores de lectura causados por factores como charla cruzada y fuga hacia estados cuánticos más altos. Realizaremos ajuste del ruido de pulsos optimizados para evaluar la efectividad de algoritmos cuánticos importantes a nivel de pulsos.

TABLE OF CONTENTS

CHAPTER PAGE

LIST OF FIGURES

FIGURE PAGE

CHAPTER 1

INTRODUCTION

Quantum devices are known to be susceptible to noise caused by the environment they are placed in, as qubits tend to provide accurate calculations only when certain conditions on temperature, external variables and computation time are met [RK21]. Such limitations have served as a starting point to design hardware that enables fault-tolerant qubit manipulation up to a certain degree, as well as to design circuits with a small amount of gates to diminish the time required for a complete unitary evolution. Due to this, ideas such as creating superconducting chips for quantum computers [KSB+20] and ways to shorten the amount of required basis gates to obtain a specific operator have risen and are being applied with increasingly better results, as can be seen with transpiling [ZPW19] [WSM20] and a clever selection of basis gates [AMMR13]. Many other high-level approaches that attempt to reduce the effects caused by external conditions on qubits are also advancing at an accelerated pace [NRK+18b] [BIS+18] as universal quantum computers provided by companies like IBM, Microsoft, and Rigetti are accessible for public use with a great amount of experiments being performed on a daily basis, enabling researchers to advance the state of the art.

For the purpose of improving the analysis and possibilities for achieving fault-tolerant quantum computations, characterizing noise acting on qubits becomes a priority as each individual noise source is subject to various conditions and interacts with the system's subprocesses in distinct ways. Subprocesses that interact both with a classical and quantum computer like pulse application and readout tend to exhibit errors due to interaction of the quantum system – both the qubits and their control signals – with the environment, and their sources can range from systematic error to stochastic error to leakage into higher quantum energy states.

Other types of noise sources interact with quantum computers more naturally, as qubits experience coherence errors and decoherence which ultimately steer them towards the ground state, as is represented with $T1$ and $T2$ relaxation times. While metrics are offered to help explain certain errors on quantum gates [RK21], some errors require a deeper analysis to be fully described and therefore benchmarks are used to experimentally identify the variables that explain these types of noise. Metrics used in benchmarks provide information on specific types of errors occurring on a qubit and on the quantum computer as is the case of single and multi-qubit error rates [KLR+08], assignment fidelity [AKE+20] and addressability [GCM+12]. In this way, benchmarks seek to analyze and describe errors to determine how "good" a quantum computer is compared to another. IBMQ offers a brief summary of metrics available for both qubits and quantum computers, and provides reliable error rates obtained through benchmarks. Quantum computers are expected to improve over the years in the amount of available qubits, error rates and connectivity,

Gate-level benchmarks like cycle benchmarking [EWP+19], randomized benchmarking [KLR+08], and tomographic mappings [NC11] have been used to successfully characterize a wide range of quantum computers [WBD+19] [MNW+17]. Despite these benchmarks offering useful information, they act without giving information on how pulse shapes impact the fidelity of their default gates, potentially resulting in pessimistic results if the pulses are not calibrated at the time of performing the benchmark. Benchmarks acting at the pulse level offer insights into information that is otherwise unavailable for benchmarks applicable only to gates, as more flexibility exists for driving a qubit into quantum states and different output levels can be used. Here, pulse optimization can be performed to prepare quantum gates that allow the analysis of error effects resulting from different pulse shapes, driving times and phases. Specific benchmarks we will study and use in this project are the Benewop benchmark (pulse noise extrapolation) [GPD20a], the leakage analysis performed in [WER+20] using QOC, and readout classification using ML techniques [AKE+20] [MGCC15].

CHAPTER 2

OBJECTIVES

2.1 General objective

Apply benchmarks at the pulse level to obtain information about the existing noise on qubits of a quantum device through statistical methods, ML and QOC, characterizing those qubits.

2.2 Specific objectives

- Perform an analysis of the properties of an IBM quantum computer such as $T1$ (spin-lattice) and $T2$ (spin-spin) relaxation times and error rates, and how they impact the presence of noise on qubits.
- Utilize Machine Learning techniques to study the impact of low-level readout errors on quantum devices.
- Implement state-of-the-art benchmarking protocols to characterize noise in a quantum computer.
- Measure the effectiveness of default and random pulses on different qubits of an IBM quantum computer and compare their fidelity with optimized pulses and random pulses.
- Provide conclusions that help explain the results obtained by each approach to noise characterization.

CHAPTER 3

METHODOLOGICAL FRAMEWORK

3.1 Noise and errors

Quantum systems are essentially open systems, where the environment is always interacting with the qubits and quantum gates on quantum devices, even if in a small degree. This is due to the physical settings a quantum device may function on, as even though each setting may be different, they all require specific conditions to function properly. Be it temperature, external magnetic forces or inaccurate pulse shaping, many factors lead to the necessity of considering errors arising from noise in the system. The gate model of quantum computing is a clear example of a model which works best by considering quantum systems as closed systems, since the final result of an ideal quantum evolution is determined by noiseless gates being applied to qubits. That is, given a noiseless unitary quantum gate U to be applied to a state $|\psi_0\rangle$, the final quantum state is described by the equation

$$|\psi_{final}\rangle = U\,|\psi_o\rangle\,.$$

Similarly, when working with density matrices, a state $|\psi_0\rangle$ represented as ρ_0 evolves by

$$\rho_{final} = U\rho_0 U^\dagger.$$

Despite the gate model not inherently describing interaction with the environment, some formulations found in [NC11] and notes by Zlatko Minev utilize the gate model in conjunction with considerations of the way the environment may interact with a quantum system. But first, it is helpful to explain the way classical noise may act on a system through a Markovian process.

Take a classical system with one bit of information initially equivalent to 0. Noise and external factors may eventually flip the value of the bit after a set amount of time with a certain probability p, or leave it unchanged with probability $1-p$. Given a vector $\vec{p} = [p_0, p_1]$ being a vector of probabilities corresponding to the probability a bit is initially 0 (p_0) and 1 (p_1), and the probability a bit is flipped or left unchanged represented by another vector $\vec{q} = [q_0, q_1]$ with q_0 being the probability a bit is 0 and q_1 the probability a bit is 1 after a set time t, the process of bit values changing can be expressed as

$$\begin{bmatrix} q_0 \\ q_1 \end{bmatrix} = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \end{bmatrix}.$$

This corresponds to a representation of classical noise in stochastic processes. A contracted version of the equation is

$$\vec{q} = E\vec{p}$$

where E corresponds to the transition matrix from probabilities $\vec{p}$ to $\vec{q}$. The notation proves to be very convenient when dealing with quantum states since they are simply a normalized vector of numbers that represent the probability of a qubit being in the $|0\rangle$ state, $|1\rangle$ state or a superposition of both, equivalent to $\vec{q}$ and $\vec{p}$ with the transition matrix holding great resemblance to a quantum gate. In fact, noise acting on a qubit can be represented as a quantum gate and the transition matrix E can be treated as a noisy I gate for the scenario previously described when acting on a state $|p\rangle$. Due to this, a bit-flip error in classical computation described by a Markovian process can be translated to quantum computation in a rather seemingless way.

Since noise can affect a quantum system in many different ways, some authors include noise from the environment into a quantum circuit as if it were an additional

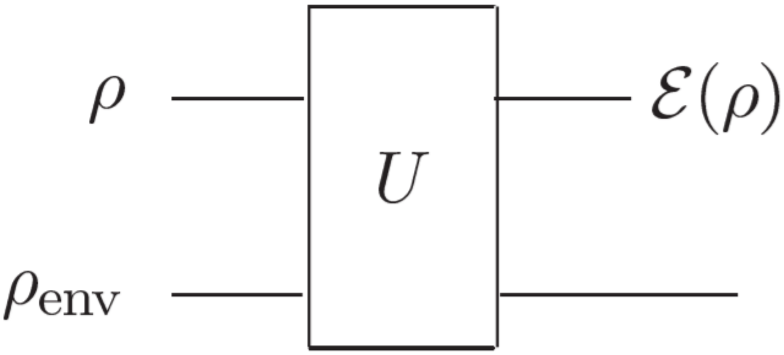


Figure 3.1: A representation of an open quantum system with the principal system ρ and the environment ρ_{env} [NC11].

qubit or system. This way of expressing a noisy quantum system is described with the equation

$$\mathcal{E}(p) = tr_{env}[U(p \otimes p_{env})U^{\dagger}]].$$

where p_{env} is the noisy quantum state acting as a subsystem of the circuit, U is a unitary gate being applied to the quantum states and $\mathcal{E}$ represents the measurement of the final noisy state. This representation can be visualized in Figure 3.1.

The rest of this section will explain the effects of different types of errors stemming from noise present in quantum systems that do not currently allow perfect applications using quantum devices. These errors can be classified as coherent, decoherent and incoherent [PBE^{+}03], or broadly classified as only coherent and incoherent [FWB^{+}16].

3.1.1 Coherent errors

Considering quantum gates being rotations on a Bloch sphere, errors that arise from over-rotations by an angle ϵ on a qubit are determined by coherent errors, and they are attributable to systematic control noise, cross-talk, global external fields and unwanted qubit-qubit interactions [GD17] [KLR+20]. Between coherent and incoherent erors, noise in a realistic quantum device will often be coherent and the different between both errors could be significant [BEKP18] as studies have shown [SWS15] [KLDF16] [WGHF15] [PGH+14] [MPGC13] coherent errors leading to large differences between average-case and worst case fidelity measures. Based on what was stated previously with noisy quantum gates being applicable to the gate model of quantum computation, it is of interest to also detail the way a noisy quantum gate can be used in mathematical expressions. Let $\tilde{U}(\phi)$ be a noisy quantum gate which intends to apply a noiseless $U(\phi)$ gate to a qubit with a desired rotation of ϕ. Take U as equivalent to a rotation gate R_x around the X axis, for simplicity. The effects of noise cause the $\tilde{U}(\phi)$ gate to miss the desired angle by ϵ and therefore introduce an extra operator $U(\epsilon)$ to explain the over-rotation of the qubit. The general effect of the noisy gate is the following

$$\tilde{U}(\phi) = U(\epsilon)U(\phi).$$

It can be shown that $U(\epsilon)U(\phi)$ is equivalent to $U(\phi + \epsilon)$ and therefore $\tilde{U}(\phi) = U(\phi + \epsilon)$

The effects of coherent errors are most visible when applying a specific quantum gate many times in succession, as the target expectation value tends to decrease with respect to its ideal value in a way that quickly leads it to zero. That is, while a single quantum gate $\tilde{U}(\phi)$ applied on a state $|\psi\rangle$ gives

$$|\psi_{target}\rangle = U(\phi + \epsilon) |\psi\rangle$$

and an over-rotation of ϵ, a repetition of d noisy $\tilde{U}(\phi)$ gates resolves to

$$|\psi_{target}\rangle = U(d(\phi + \epsilon)) |\psi\rangle = U(d * \phi + d * \epsilon) |\psi\rangle = U(d * \epsilon)U(d * \phi) |\psi\rangle$$

and therefore an over-rotation of $d * \epsilon$. Actually, if we consider the expectation value Z of a measurement applied to the resulting quantum circuit, we can find the

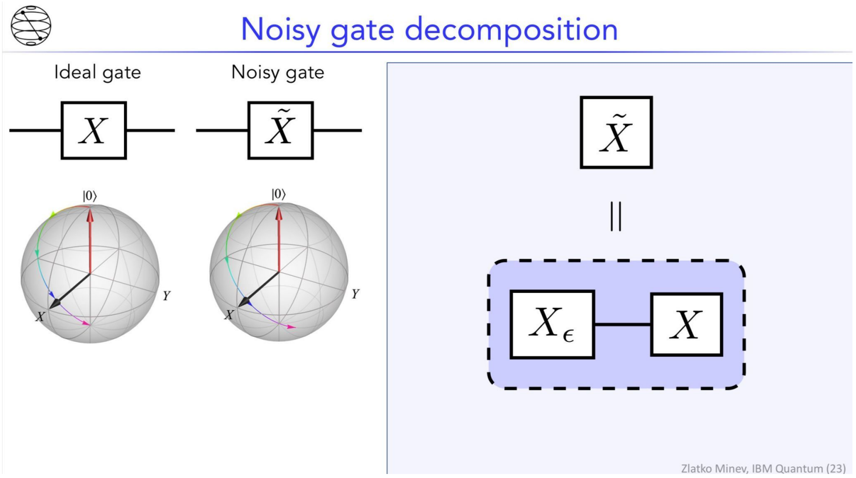


Figure 3.2: The decomposition of a noisy gate into an ideal gate and a gate representing coherent noise (retrieved from Zlatko's lectures).

accumulation of errors resulting from the noisy gates. In the case presented, the ideal and noisy expectation values are, respectively

$$\langle\psi_{target}|Z|\psi_{target}\rangle = cos(d\pi),$$
$$\langle\tilde{\psi}_{target}|Z|\tilde{\psi}_{target}\rangle = cos(d\pi + d\epsilon).$$

Here, the difference in expectation values for d gate applications results in

$$\begin{aligned}\langle\tilde{\psi}_{target}|Z|\tilde{\psi}_{target}\rangle - \langle\psi_{target}|Z|\psi_{target}\rangle &= cos(d\pi + d\epsilon) - cos(d\pi)\\ &= cos(d\pi)cos(d\epsilon) - sin(d\pi)sin(d\epsilon) - (-1)^d\\ &= (-1)^d(cos(d\epsilon) - 1) \approx -\tfrac{1}{2}(-1)^d d^2\epsilon^2 + \mathcal{O}(\epsilon^3)\end{aligned}$$

and it is proved that coherent error results in a quadratic error. This unwanted increase in rotation makes coherent errors an important source of noise that limit the amount of desirable quantum gates to be applied on a quantum circuit before results become useless. Due to this, pulse shaping, calibration and quantum error correction of coherent errors in general are tasks of great interest in the internal processes of quantum devices for realizing scalable fault-tolerant quantum computing. [GD17]

Crosstalk

Crosstalk is a combination of unwanted interactions between coupled qubits on a quantum chip, as no single noise model captures all aspects of crosstalk [DGL+20]. The importance of this noise source is attributed to it being the most dominant source of gate errors [MSS+19] [MZHH19], demonstrated on architectures that include trapped ion and superconducting systems. [ACLS+17] [KKY+19] [NRK+18a] [OLA+08] Additionally, coherence errors caused by crosstalk can accumulate as an algorithm is run on a circuit [BQP+19], with an increasing impact in multi-qubit calculations. Crosstalk is generated when qubits resonate with each other in a way that given a pair of neighboring qubits, the quantum state stored in one qubit affects the quantum state stored in its neighbor qubit. This may be caused by frequency crowding (qubits being driven at frequencies too close together) or a high interaction strength between the qubits, leading to residual coupling [DGL+20]. Nonetheless, studies are also being performed for finding crosstalk between sets of qubits. Correlations between in-phase and quadrature signals of different pulse settings on neighboring qubits also hint towards crosstalk between them [AKE+20]. These definitions share the same practical effect as errors named addressing errors, and both terms are used interchangeably in some sources. [HHR+05] [HST+20] [NMM+14]

Leakage

Leakage into higher energy states is a common problem in the control of quantum systems, particularly in the case of the transmon architecture [KYG+07], where it can be long-lived and grow to significant levels [CKQ+16]. Leakage can be described as the interaction with a second excited quantum state $|2\rangle$ when the energy of a quantum system exceeds the boundary set for both ground ($|0\rangle$) and excited ($|1\rangle$) energy levels. When leakage is present on a qubit, the quantum state can be expressed as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle + \gamma|2\rangle$ with $\gamma \neq 0$ [DGL+20]. Leakage can be caused by imprecision in quantum control, crosstalk between qubits and poorly configured quantum systems. Contrary to the assumption that a quantum device is set on a two-level qubit subspace, quantum systems such as quantum dots and superconducting qubits have multiple levels [CKQ+16] [GFMG13] [NAB+09] [YN05]. For example, IBMQ quantum device Hamiltonians can be seen to have a three-level version of qubits or "qutrits", and accessing the third level can be done experimentally by tuning the qubit frequencies and calibrating amplitudes to go from the $|0\rangle$ state to the $|1\rangle$ state, and then repeating a similar process to find the $|2\rangle$ state [Tea21a]. Using IBMQ quantum devices and OpenPulse [AKE+20], level one kerneled IQ data can show evidence of a $|2\rangle$ state as a third group of data points as shown in Figure 3.3. Leakage errors have been greatly studied, and consequently reduced in quantum computers on calibrated pulses by using Derivative Reduction by Adiabatic Gate (DRAG) pulse shaping [MW13]. Despite calibration being a standard procedure for suppressing these errors, Quantum Error Correction (QEC) still faces obstacles due to small leakage errors appearing below the threshold for efficiently employing topological QEC [Fow13]. This can be explained by qubits in a leakage state having a randomizing effect on interacting qubits.

3.1.2 Incoherent errors

As with coherent errors, incoherent errors are an important limitation in the precision of coherent control in quantum information processing [BEH+04]. Incoherent errors are caused by static or slowly varying fluctuations in the system's Hamiltonian [PBE+03] which, while not being described by a unitary operator, can be corrected by applying unitary operations that refocus them. Incoherent errors can

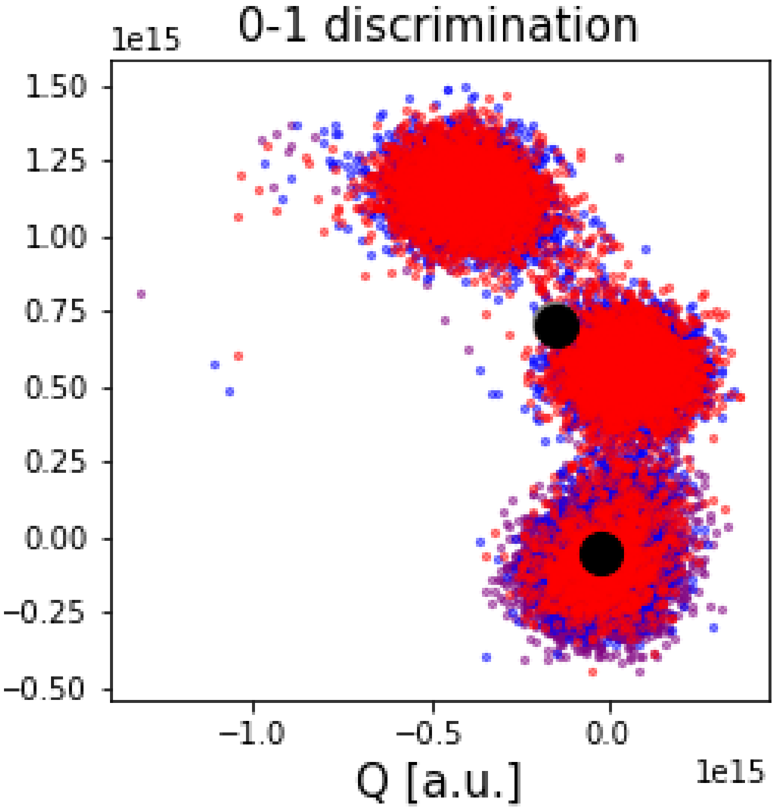


Figure 3.3: Resulting IQ data points from excecuting a miscalibrated pulse on IBMQ's Armonk device, showing the first two quantum states $|0\rangle$ and $|1\rangle$ at $Q \approx 0$ with leakage into a possible $|2\rangle$ state at $Q \approx -0.5$.

also be of many forms such as amplitude damping, phase damping, bit-flip and phase-flip errors [CSA+21] [SZL+19] [ZMM17], and are generally notable when analyzing the expectation values of measured quantum states.

Similar to coherent errors, incoherent errors may also depend on a parameter p that signifies error, this time being the difference in expectation values from a +1 on an operator O when in the presence of noise. In the case of bit-flip errors, p can represent the probability of a $|1\rangle$ state being flipped to become the $|0\rangle$ state, while $1-p$ represents the probability of the state being left unchanged. The density operator of this error representation is the following:

$$\rho = \begin{bmatrix} p & 0 \\ 0 & 1-p \end{bmatrix}$$

Here, measurement on each Pauli operator is described by

$$\langle X\rangle = Tr(\hat{X}\rho) = 0$$

$$\langle Y\rangle = Tr(\hat{Y}\rho) = 0$$

$$\langle Z\rangle = Tr(\hat{Z}\rho) = Tr\left(\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} p & 0 \\ 0 & 1-p \end{bmatrix}\right) = Tr\begin{bmatrix} p & 0 \\ 0 & p-1 \end{bmatrix} = 2p-1.$$

That way, the sum of expectation values of the Pauli operators $\langle X\rangle^2 + \langle Y\rangle^2 + \langle Z\rangle^2 = (2p-1)^2$ in stead of the ideal 1^2. The application of a gate G d times would therefore result in an expectation value $\langle Z\rangle = (2p-1)^d \approx 1-2dp$ where the incoherent error results as linear in p. Expectation values represent measurements of a Bloch sphere on different bases, with the main basis gates of interest being $P \in \{I, X, Y, Z\}$. The Tr or trace operation used in this demonstration is briefly explained in section 3.2.2.

3.1.3 SPAM errors

State preparation and measurement (SPAM) errors are a particular kind of errors that do not relate as much to coherent errors and incoherent errors as they do to the internal processes underwent at the beginning and the end of a circuit. They mostly differ in the fact that SPAM errors do not accumulate with circuit depth [lid13]. Due to this, surface QEC is significantly less sensitive to measurement errors than to gate errors [GS20]. Much can also be said about the disadvantages of possessing SPAM errors, as fault-tolerant quantum computation is limited by these errors. Particularly, because benchmarking protocols like Quantum Process Tomography are sensitive to SPAM errors [GCM$^+$12]. The same also holds true the shorter the circuit is, as the proportion of impact gate errors tend to have on circuits start to reduce when compared to SPAM errors. In a way, SPAM errors provide a bound on gate fidelity measured through benchmarks that rely heavily on individual measurements, creating the need to apply partial benchmarks that use less measurements to try to ignore these errors. For example, experimentally while Quantum Process Tomography may provide errors of 5% for calibrated π pulses on IBM machines, Randomized Benchmarking (a benchmark that is insensitive to SPAM errors) can show errors less than 0.1%. Two specific procedures are performed on measurement processes to allow corrected readout results:

1. Using good discriminators on IQ pulse data to improve assignment fidelity since out of the numerous possibilities for models, not all are simple and effective enough to use as discriminators on a quantum device while maintaining high fidelity.

2. "Let the set of n physical qubits we wish to correct be called the register (the register does not have to include every qubit in the processor). The T matrix SPAM correction technique can be described as follows: Let $x, x' \in \{0, 1\}^n$ be classical

states of the n qubits in the register, and define elements of a $2^n \times 2^n$ matrix T by

$$T(x|x') = T(x_1...x_n|x'_1...x'_n) = Tr[E_{x_1...x_n}\rho_{x'_1...x'_n}].$$

Here E_x is the multiqubit POVM element characterizing the nonideal implementation of the projector $|x\rangle\langle x|$, and $\rho_{x'}$ is the density matrix produced after attempting to prepare classical state $|x'\rangle\langle x'|$. Each column x' of T is the raw probability distribution prob(x), measured immediately after preparing x'. In the absence of any SPAM error,

$$T(x|x') = \delta_{xx'},$$

the $2^n \times 2^n$ identity. The complete implementation of the technique is to measure T and classically apply T^{-1} to subsequently measured probability distributions" [GS20].

To verify SPAM errors appearing in a quantum computation, they can be calculated by the standard measure of single-qubit SPAM error [MGCC15]

$$\epsilon = \frac{T(0|1) + T(1|0)}{2}$$

or by the total SPAM error [HP19]

$$||T_{meas} - I||.$$

3.2 Metrics

3.2.1 T1 and T2 relaxation times

The T_1 time of a qubit is the time it takes for a qubit to decay from the excited state to the ground state [Tea20]. This time is called the relaxation time or spin relaxation time and it is deeply related to incoherent errors as these drive the qubit towards an expectation value of 0 and is impacted more by the time a circuit is being driven than by the amount of gates being applied to the circuit. It is measured by using Inversion Recovery, a common process in other fields as well [BFdS20], where the signal energy of an electromagnetic pulse being applied to a qubit is fitted to an exponential decay by setting different delay times from the moment the pulse was applied. The function used for fitting results to obtain the T_1 time is

$$f : Aexp^{\frac{-t}{T_1}} + C$$

as can be seen in Figure 3.4. T_1 times are indispensable for the solution of chemical problems [BHP80], and many investigations depend on a high accuracy of T_1 [SZ77].

Similarly, the T_2 time is the time required for dephasing of a qubit to ocurr, and can be called coherence time or phase relaxation time [HDSS02] [WZY05]. It is measured by using a process named Hahn echo, a term used in NMR [Tea20]. To set up a Hahn echo experiment, a Ramsey experiment is performed by applying a $\pi/2$ pulse, a π pulse and another $\pi/2$ pulse at the end before measuring the qubit. The decay time resulting from fitting signal results from this experiment shows the T_2 time. The importance of both relaxation and coherence times resides in them needing to be much longer than the gate operation time in order to hold useful results without much deterioration [STHL10].

3.2.2 Single and multi qubit metrics and error rates

Metrics for qubits and quantum gates are a common way of measuring overall fidelity of gates or interactions with the environment as error rates. In a general sense, metrics enable quantitative analysis on the quality of a quantum system [RK21]. An important and very common metric for measuring fidelity of a quantum gate is process fidelity where, given a density matrix ρ, a noisy quantum gate $\tilde{G}$ and the

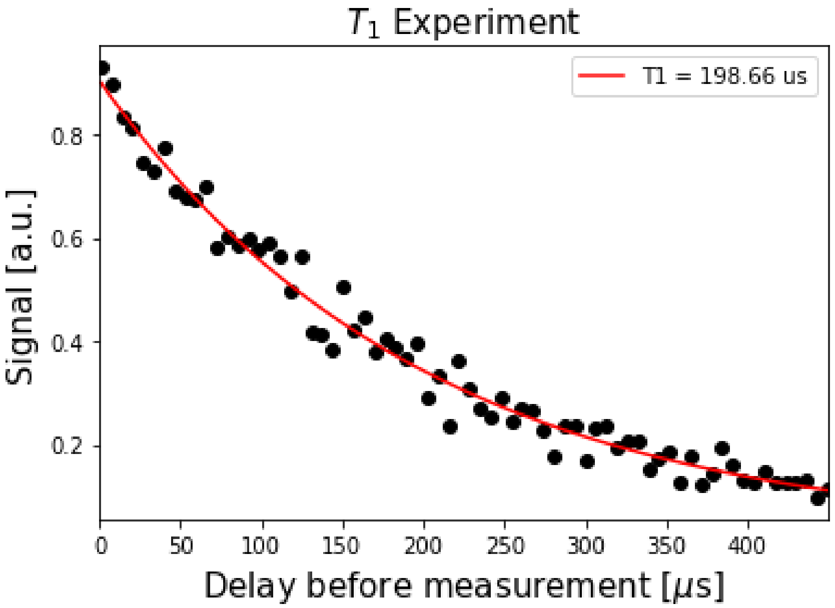


Figure 3.4: Using inversion recovery to find the decay time T_1 from fitting an exponential decay to the pulse signals and delays before measurement.

same noiseless quantum gate G, the process fidelity of the gate is calculated with the equation [EWP+19] [FL11].

$$F(G(\rho), \tilde{G}(\rho)) = Tr[G(\rho)\tilde{G}(\rho)].$$

Tr represents the trace of the matrix and it is equivalent to finding the sum of all its diagonal elements. The trace can also represent similarity between two quantum gates. It is easy to tell that the process fidelity depends directly on the initial state ρ when calculating the process fidelity. This leads to only having partial results on the fidelity of a gate as the whole basis isn't being considered. Such a problem can be solved by utilizing the average gate fidelity [BWG+18] and averaging over applications of the gate on all basis states. Section 3.3.1 describes how the average gate fidelity can be used in that way to benchmark a quantum gate through Quantum Process Tomography. A general although more ambitious way to find the average gate fidelity is to integrate over the surface of the Bloch sphere through the equation [CB07] [BOS+02]

$$\langle F \rangle = \frac{1}{4\pi} \int Tr[G(\rho)\tilde{G}(\rho)] d\Omega.$$

Also using the trace of a matrix, the trace distance measures the distinguishability of two quantum states through [Rus94]

$$D(G(\rho), \tilde{G}(\rho)) = \frac{1}{2} Tr[\sqrt{(G(\rho) - \tilde{G}(\rho))^{\dagger}(G(\rho) - \tilde{G}(\rho))}].$$

Conveniently, the trace distance relates to the process fidelity enabling one to be found through the other. This is expressed by the inequality

$$1 - \sqrt{F(G(\rho), \tilde{G}(\rho))} \leq D(G(\rho), \tilde{G}(\rho)) \leq \sqrt{1 - F(G(\rho), \tilde{G}(\rho))}$$

Another distance metric between noisy and ideal gates that measures distuinguishability is given by [FvdG99] [Blu17]

$$||G - \tilde{G}||_d = max_{\rho}(||(G \otimes I)[\rho] - (\tilde{G} \otimes I)[\rho]||_1).$$

An essentially different matrix that focusses on finding the difference between two probability distributions is the Hellinger distance Given two probability distributions p and q with individual samples p_i and q_i, the Hellinger distance is

$$\frac{1}{\sqrt{2}} \sqrt{\sum_i (\sqrt{p}_i - \sqrt{q}_i)^2}.$$

The way the Qiskit library returns measurements greatly benefits from the Hellinger fidelity, which is the same as $1 - Hellinger\ distance$.

Relevant error rates adopted by IBM for single and multiple qubits on their quantum devices are found using Randomized Benchmarking, a benchmark that essentially ignores SPAM errors by implementing different lengths of gate repetitions, explaining the really low error rates obtainable. In reality, error rates found through

Randomized Benchmarking are only useful when they are low enough to be fitted onto an exponential function, whereas high error on the gates could return meaningless error rates. Single qubit Clifford error rates can be found using the basic procedure while two-qubit Clifford error rates are available when applying simultaneous Randomized Benchmarking. Alternatively, error rates of particular gates can also be found. A description of Randomized Benchmarking can be found in section 3.3.1 under Randomized Benchmarking.

3.2.3 Addressability

The addressability metric can be defined as the additional errors induced on subsystem or qubit k from controlling k' and is calculated through the following equation:

$$\delta r_{k|k'} = |r_k - r_{k|k'}|$$

The addressability metric combines the errors from classical cross-talk leakage errors and quantum coupling between the systems. This also quantifies the increase in error rate from simultaneous control on different qubits. Therefore, addressability relates to the differences in the measured average gate fidelities in the experiments performed on two qubits [GCM$^+$12]. Also, addressability error can have an inverse proportion to the amount of detuning between qubits. A procedure using Randomized Benchmarking to calculate this metric can be found in section 3.3.1.

3.3 Benchmarks

3.3.1 Gate Model Benchmarks

Quantum State Tomography

Quantum State Tomography [VR89] (or QST for short) is, as its name states, a way to determine the resulting state from applying a quantum gate G. It stands as a standard for verification and benchmarking of quantum devices [CPF$^+$10]. This is done by performing measurements on a set of observables equivalent to the Pauli basis, and using the expectation values to fully determine the quantum state [SBRF93] [HHR$^+$05] [LKS$^+$05] [JKMW01] [DWM95] [LR09]. To that effect, the Pauli basis gates $P \in \{I, X, Y, Z\}$ are applied to the qubits before measurement is performed in order to obtain expectation values on those bases in the following manner:

$$\langle X \rangle = Tr(\hat{X}\rho) = Tr\left(\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = Tr\begin{bmatrix} c & d \\ a & b \end{bmatrix} = c + b,$$

$$\langle Y \rangle = Tr(\hat{Y}\rho) = Tr\left(\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = Tr\begin{bmatrix} -ci & -di \\ ai & bi \end{bmatrix} = (b - c)i,$$

$$\langle Z \rangle = Tr(\hat{Z}\rho) = Tr\left(\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = Tr\begin{bmatrix} a & b \\ -c & -d \end{bmatrix} = a - d,$$

$$\langle I \rangle = Tr(\hat{I}\rho) = Tr\left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = Tr\begin{bmatrix} a & b \\ c & d \end{bmatrix} = a + d.$$

The way to find the resulting density matrix that represents the quantum state is by mapping the expectation values as a linear sum of Pauli matrices [PvJ11]

$$\rho = \frac{1}{2}(\langle I \rangle I + \langle X \rangle X + \langle Y \rangle Y + \langle Z \rangle Z).$$

QST is especially susceptible to SPAM noise, since a very small circuit is constructed with only one quantum gate besides measurement and the gate to be analyzed. Due to this, QST may not provide accurate fidelity measurements as it always presents a deviation from the intended state. It may even become infeasible for larger systems due to the amount of measurements required which is 4^n for a number of qubits n considering a combination of each gate of the Pauli basis [CPF+10].

Quantum Process Tomography

Quantum Process Tomography (QPT) works on a similar procedure as QST, with the main difference that a more complete characterization of the qubit is performed. Instead of a quantum state, a quantum operation or process is identified and thus more information can be made available [RK21] [CN97] [PCZ97]. The implication of QST only being able to identify a quantum state is that a gate, for example, cannot be reconstructed by using the expectation values found in QST and therefore would make many of the applications thought of today unfeasible. The way QPT makes this possible is by once again performing measurements in all the basis gates and also preparing a circuit in all the basis states too. The difference being that QPT focuses on an average gate fidelity by performing the same measurements done in QST from different starting points on the Bloch sphere. For the single qubit case, each Pauli gate $P \in \{I, X, Y, Z\}$ is combined with the same set P to find transitions from and to each Pauli gate with a set describing those combinations being $Q \in \{\{I, I\}, \{I, X\}, \{I, Y\}, \{I, Z\}, \{X, I\}, \{X, X\}, \{X, Y\}, \{X, Z\}, \{Y, I\}, \{Y, X\}, \{Y, Y\}, \{Y, Z\}, \{Z, I\}, \{Z, X\}, \{Z, Y\}, \{Z, Z\}\}$. The combination of Pauli gates and their effect on all measurements performed can be evidenced by taking each ith couple from Q and finding its expectation value by measuring the trace of

$$Tr(\hat{Q_{i,0}} \rho \hat{Q_{i,1}}).$$

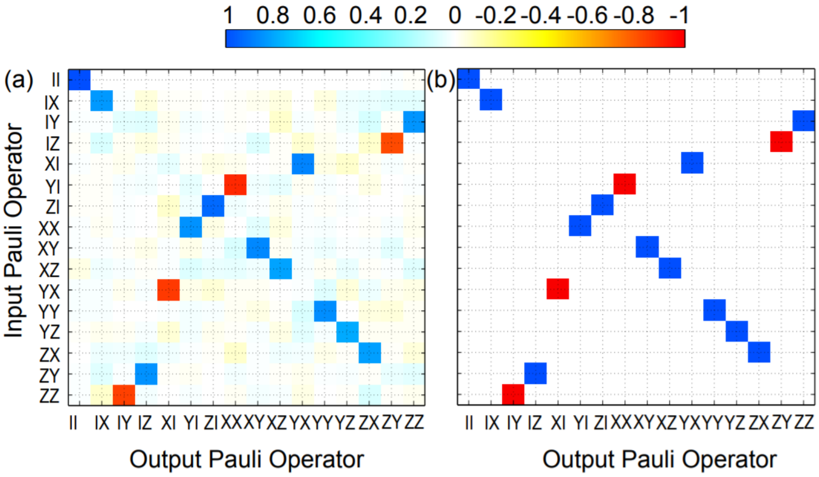


Figure 3.5: "(color online) Quantum process tomography of the $ZX_{-\pi/2}$ gate with $T_2 = 178ns$. (a) Experimentally extracted Pauli transfer matrix. The gate fidelity is $F_g = 0.8830$ raw and $F_{mle} = 0.8799$ after applying a maximum likelihood algorithm. (b) Ideal Pauli transfer matrix representation of the $ZX_{-\pi/2}$ gate." [CGC+13].

Choi matrices [Cho75] are a useful tool that not only represent expectation values being prepared in one Pauli basis against another Pauli basis measurement, but enables the calculation of a complete gate fidelity very similar to average gate fidelity. These Choi matrices are reconstructed through expectation values in the same way as QST, and are also a standard for representing a quantum gate. An example of a Choi matrix resulting after applying QPT to a $ZX_{-\pi/2}$ gate is shown in figure 3.5. SPAM errors have a slightly deeper effect in QPT as an additional gate is used to prepare the quantum state which will be measured and therefore both SPAM and coherence errors will slightly worsen fidelity values found from using this benchmark.

Randomized Benchmarking

Randomized Benchmarking (RB) is a benchmark that is based on repetitions of gates belonging to a unitary group $\mathcal{G} \subseteq U(d)$ (where $d = 2^n$ with n being the number of qubits the gates act on) to obtain single and multi-qubit error rates, as well as error rates for the individual gates that belong to said group. A common group of interest when performing RB is the Clifford group, which are gates that can generate the Pauli group, along with the Pauli group itself. The gates that generate the Pauli group are the Hadamard and the stabilizer S for the single qubit case with the addition of the CNOT gate for the two qubit case [Got98b]. Some authors also use the term Clifford quotient group [KS14] [Tol18]. Clifford gates are specially useful for RB because each of the gates can be realized efficiently on a quantum processor [MGE11], thus reducing unwanted noise to some degree. The general outline of the procedure done in RB is the following:

1. Generate a sequence of $m+1$ quantum gates where the first m gates are chosen randomly from the Clifford group in a uniform manner. The last gate is the inverse of the first gates so that the resulting circuit operator is the identity gate, given the property of unitary matrices that states a unitary matrix O multiplied by its transposed conjugate $O^\dagger$ results in the I matrix. The benchmark uses twirling over the chosen sequence in order to obtain Clifford gates at each step of the sequence, multiplied by 1 or -1. That is: given operators $\{P, O\} \in \mathcal{G}$, $O_{twirl} = OPO^\dagger$ where $-1^n * O_{twirl} \in \mathcal{G}$ with $n \in \{0, 1\}$. The final inverse gate is therefore the inverse of a O_{twirl} being uniquely determined by the mth gate, further simplifying the operations required in the benchmark.

2. Measure the survival probability $Tr[E_\psi S_{i_m}(\rho_\psi)]$ for a set of generated sequences with ρ_ψ being the initial state, E_ψ the POVM element towards which the sequence will be measured and S_{i_m} being the final noisy sequence operator belonging to the ith sequence. ρ_ψ includes preparation errors while E_ψ represents measurement errors also.

3. Calculate the average sequence fidelity over random sequence runs, where the

average sequence S_m is used in

$$F_{seq}(m, \psi) = Tr[E_\psi S_m(\rho_\psi)].$$

4. " Fit the results for the averaged sequence fidelity [Eq. (2)] to the model

$$F^{(1)}_{seq}(m, \psi) = A_1 p^m + B_1 + C_1(m-1)(q-p^2)p^{m-2}$$

derived below. The coefficients A_1, B_1, and C_1 absorb the state preparation and measurement errors as well as an edge effect from the error on the final gate. The difference $q - p^2$ is a measure of the degree of gate-dependence in the errors, and p determines the average error-rate r according to the relation

$$r = 1 - p - (1 - p)/d.$$

In the case of gate-independent and time-independent errors the results will fit the simpler model

$$F^{(0)}_{seq}(m, \psi) = A_0 p^m + B_0$$

derived below, where A_0 and B_0 absorb state preparation and measurement errors as well as an edge effect from the error on the final gate." [MGE11]

An important metric resulting from the benchmark is the average error per Clifford sequence formulated as

$$EPC = \frac{2^n - 1}{2^n}(1 - p)$$

in the case of gate-independent and time-independent errors, compensating for state preparation and measurement (SPAM) errors by considering only the exponential decay of sequences of random gates [GCM+12]. By using the error per Clifford and the amount of gates per Clifford, an error per gate value can be determined. The inverse of the Clifford sequence is efficiently calculated by the Gottesman-Knill theorem in [Got98a]. For further understanding of the procedure, an in-depth mathematical explanation of the RB procedure can be found in [MGE12].

RB has been widely used to characterize gates in NMR systems [RLL09] [CCL01] [WHE$^+$04], atomic ions for different types of traps [KLR$^+$08] [BUV$^+$09] [BWC$^+$11] [OCNP10] [GTL$^+$16] [BHL$^+$16] [GMT$^+$12], solid-state spin qubits [VHY$^+$14] and superconducting qubits [CGT$^+$09] [CDG$^+$10] [BKM$^+$14] [MWS$^+$17] [CGC$^+$13]. Despite the limitation of only obtaining information for the gates included in the Clifford group, several alternative uses have been devised using RB such as estimating the error of a particular gate through interleaving [MGJ$^+$12], purity [MFM$^+$16] [WGHF15] and leakage analysis [WG18] [WBE16]. This method accounts for the drawbacks presented with QPT as fewer measurements are necessary to characterize gates on qubits and it does so in a way that is insensitive to errors arising from SPAM with the limitation that the total gate duration of the sequence needs to be lower than the coherence time of the qubit [CGC$^+$13]. Another important limitation is that for the noise fitting of RB to work, low error rates are needed as very high error rates may deviate from fitting to an exponential decay to rather fitting to a different model, possibly rendering results useless. The benchmark has been used to characterize single-qubit, two-qubit [CGC$^+$13] and three-qubit [MSS$^+$19] error rates and in fact, the single qubit gate error rates and CNOT error rates presented in IBM quantum computer properties are calculated using RB for particular gates.

For the case of characterizing specific gates and qubits in a quantum computer, RB acquires additional uses as both finding the error rates for specific gates and quantifying addressability in a pair of neighboring qubits are tasks of interest. One of the most common ways to find the error rates on specific gates belonging to the Clifford group is interleaving, where a variation is made in the preparation of circuits to be measured. In interleaved RB, the target gate is applied between every other Clifford gate in an attempt to make the errors dependent on the performance of the target gate through randomization. Figure 3.6 illustrates the way the circuits are prepared. The procedure explained in detail found in [MGJ$^+$12] is done the following way:

1. Implement the standard randomized benchmarking previously described.

2. "Choose K sequences of Clifford elements where the first Clifford $\mathcal{C}_{i_1}$ in each sequence is chosen uniformly at random from Clif_n, the second is always chosen to be $\mathcal{C}$, and alternate between uniformly random Clifford elements and deterministic $\mathcal{C}$ up to the mth random gate [see Fig. 1(b)]. The $(m+1)$th gate is chosen to be the inverse of the composition of the first m random gates and m interlaced $\mathcal{C}$ gates (we adopt the convention of defining the length of a sequence by the number of random gates). The superoperator representing the sequence is

$$\mathcal{V}_{i_m} = \mathcal{A}_{i_{m+1}} \circ \mathcal{C}_{i_{m+1}} \circ (\bigcirc_{j=1}^{m}[\mathcal{C} \circ \mathcal{A}_{\mathcal{C}} \circ \mathcal{A}_{i_j} \circ \mathcal{C}_{i_j}]).$$

For each of the K sequences, measure the survival probability $Tr[E_\psi \mathcal{V}_{i_m}(\rho_\psi)]$ and average over the K random sequences to find the new sequence fidelity $F_{seq}(m, \psi)$. Fit $F_{seq}(m, \psi)$ to one of the new zeroth or first-order models to obtain the depolarizing parameter $p_{\mathcal{C}}$. The expressions for these models are given by Eq. (2) where p is replaced by the new depolarizing parameter $p_{\mathcal{C}}$." [MGJ$^+$12]

3. "From the values obtained for p (Step 1) and $p_{\mathcal{C}}$ (Step 2), the gate error of $\mathcal{A}_{\mathcal{C}}$ (which is exactly given by $r_{\mathcal{C}} = 1-$ average gate fidelity of $\mathcal{A}_{\mathcal{C}}$) is estimated by

$$r_{\mathcal{C}}^{est} = \frac{(d-1)(1 - p_{\mathcal{C}}/p)}{d},$$

and must lie in the range $[r_{\mathcal{C}}^{est} - E, r_{\mathcal{C}}^{est} + E]$ where

$$E = min \begin{cases} \frac{(d-1)[|p - p_{\mathcal{C}}/p| + (1-p)]}{d} \\ \frac{2(d^2-1)(1-p)}{pd^2} + \frac{4\sqrt{1-p}\sqrt{d^2-1}}{p} \end{cases}$$

" [MGJ$^+$12]

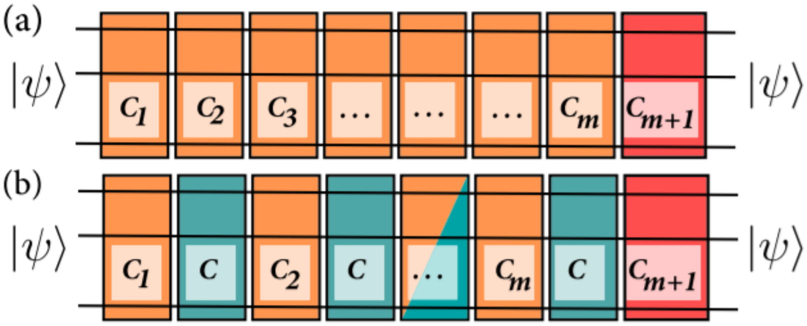


Figure 3.6: (color online). Randomized benchmarking protocols. (a)–(b) Schemes for the standard and interleaved benchmarking protocols. The target gate, $\mathcal{C}$ (green) is interleaved with random gates $\mathcal{C}_i$ (orange) chosen from Clif$_n$. A final gate $\mathcal{C}_{m+1}$ (red) is performed to make the total sequence equal to the identity operation [MGJ+12].

Simultaneous RB is specially useful for characterizing errors on different qubits at the same time, along with characterizing errors for different subsets of qubits. An example of using simultaneous RB for characterizing three-qubit errors is shown in figure 3.7, where different configurations of Clifford gates were used to obtain error rates on three connected qubits.

The addressability protocol implements simultaneous RB using circuits configured as shown in figure 3.8, where gates are applied to pairs of neighboring qubits to find information on cross-talk errors.

"In short, the addressability protocol is the following three experiments:

Experiment 1: Implement RB on the first subsystem, i.e. twirling with the group $C \otimes I$ (see Fig. 1b). Fit the average decay of subsystem 1's initial state to obtain α_1 and thus the error r_1.

Experiment 2: Perform the same experiment on subsystem 2 (see Fig. 1c) yielding α_2 and r_2.

Experiment 3: Implement RB on both subsystems simultaneously, i.e. twirl with $C \otimes C$, (see Fig. 1d). Fit the decays of the single subsystems (e.g. ZI and IZ) and

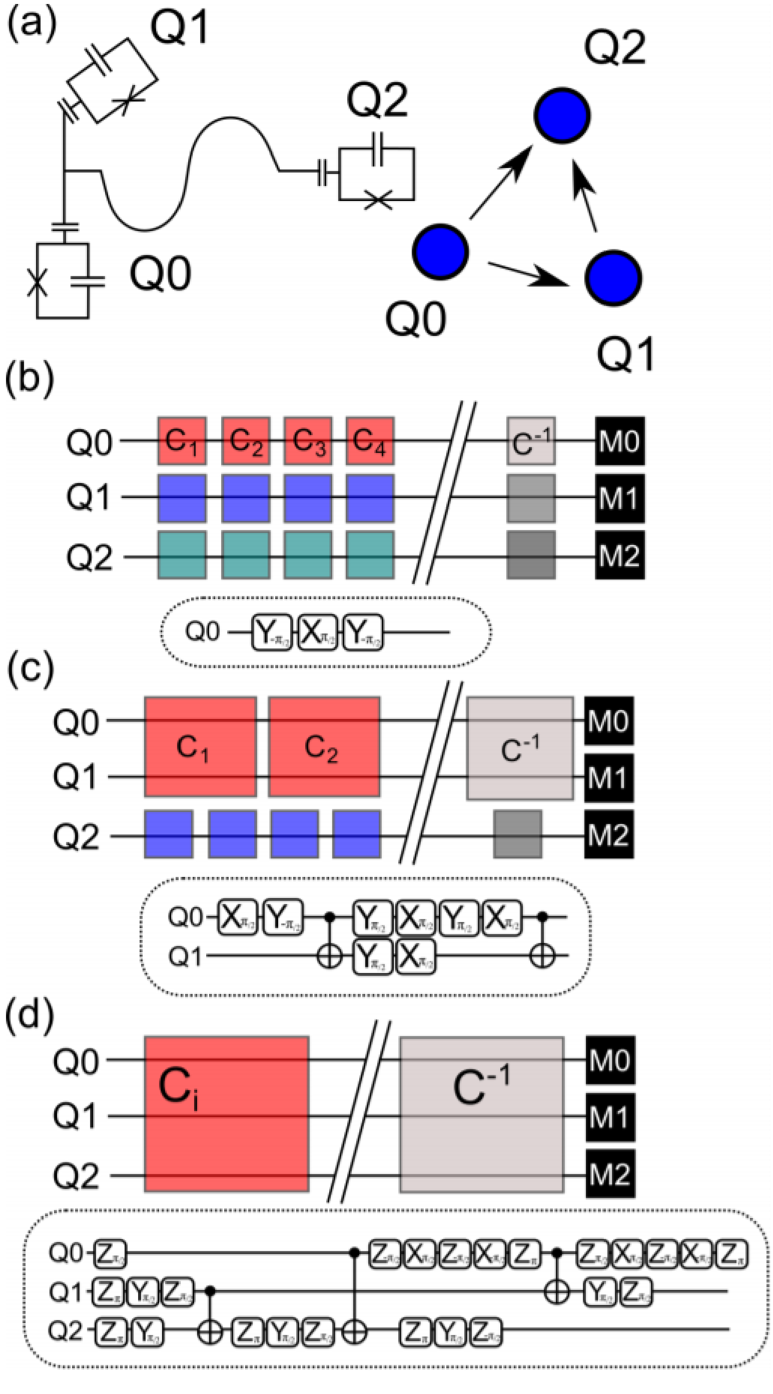


Figure 3.7: (Color Online) (a) Schematic of the experimental setup and connectivity of the CNOT 2Q gates (control→target). (b) 1Q simultaneous RB {[0], [1], [2]}, (c) 2Q-1Q simultaneous RB {[0, 1], [2]} and (c) 3Q RB {[0, 1, 2]}. Under each is a sample (b) 1Q (c) 2Q and (d) 3Q Clifford gate [MSS+19].

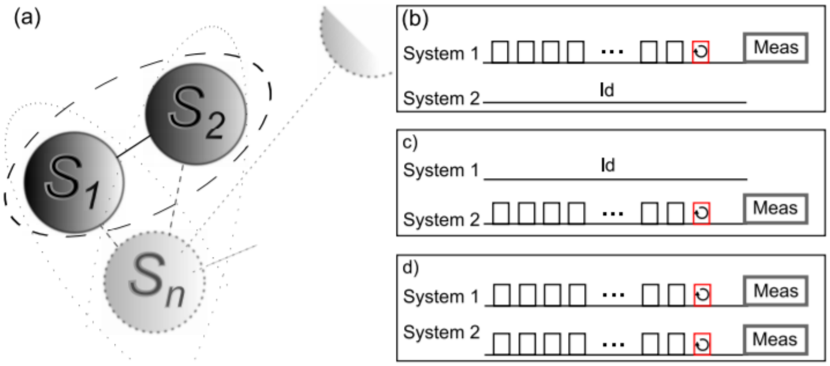


Figure 3.8: (color online). (a) General approach to characterize the effect of cross-talk and unwanted quantum interactions in a processor with multiple subsystems. The experiments are performed pairwise. For each pair, three experiments are performed: RB is performed on each subsystem while leaving the other subsystem unperturbed (b and c); then RB is performed simultaneously (d) [GCM+12].

two-qubit correlations (e.g. ZZ) to obtain $\alpha_{1|2}$, $\alpha_{2|1}$ and α_{12}. The addressability is then quantified from Eq. 13, and potential correlations in the errors are flagged by examining $\delta\alpha = \alpha_{12} - \alpha_{1|2}\alpha_{2|1}$." [GCM+12]

3.3.2 Discriminating Quantum States Using Machine Learning and Quantum Machine Learning

Discriminating quantum states

Machine learning (ML) techniques have been used for the discrimination of quantum measurement data for some time now as an approach to retrieve amplitudes resulting from execution of a quantum circuit. One application of these techniques on quantum measurement trajectories can be seen in [MGCC15], where SVM, KMeans, LDA, and the RUSBoost algorithms are used to discriminate data belonging to the $|0\rangle$ and the $|1\rangle$ states. While these discriminators are not offered out-of-the-box, Qiskit Pulse enables the use of different ML models to work as readout discriminators and also offers a Linear Discriminant Analysis module to use as a custom discriminator [AKE+20]. The main difference in the studies performed on quantum measurement trajectories with the way quantum states are being discriminated in Qiskit Pulse is the form of the data, given that in-phase and quadrature (IQ) signal data is returned from computations on IBMQ devices that recieve schedules as inputs.

On IBM quantum computers, the task of state discrimination corresponds to the first and second level of their quantum hardware's obtainable results through OpenPulse [MAB+18], mapping the phase (IQ) signals of their first levels onto qubit states and counts. While state discrimination can be performed both on level 0 and on level 1, they solve the problem through a different type of data input. Level 0 focuses on classifying shot trajectories through ML by using filter functions for each trajectory [CVC+13] and algorithms such as Quadratic Discriminant Analysis (QDA) and Support Vector Machines (SVM). ML algorithms for discrimination may be specified on Qiskit if they are formatted as a scikit-learn classifier. Despite Quantum Machine Learning algorithms not having been used to perform discrimination at readout, it is possible to achieve this by specifying Quantum Machine Learning algorithms as scikit-learn classifiers.

Quantum Machine Learning

Quantum Machine Learning (QML) focuses on the use of quantum algorithms, which harness the capabilities of quantum mechanics, to solve common ML problems and to improve existing algorithms [SSP14] [AH15]. Many algorithms implement quantum basic linear algebra subroutines (qBLAS) that have shown exponential quantum speedups over the most optimal methods available in classical computation [BWP+17]. The use of amplitude amplification [BHMT02] and the

HHL [HHL09] algorithm in QML algorithms result in speedups in computational and query complexity over their classical counterparts. Methods such as Bayesian Interference [LYC14], PCA [LMR14], Support Vector Machines [RML14] and many others [WKS16] [WBL12] [DTB16] have adapted implementations of qBLAS to achieve speedups of up to $O(\sqrt{N})$ and $O(logN)$. Adiabatic quantum computers can also provide adequate solutions for ML problems provided that models can be represented as quadratic unconstrained binary optimization (QUBO) problems as is theorized in [DAPN21], posing better time and space complexities than their classical counterparts. The balanced k-means clustering algorithm implemented on adiabatic quantum computers demonstrates performance similar to the best classical approaches [AD20]. There exist several limitations on QML algorithms as only small-scale quantum computers and special-purpose quantum simulators are currently available to implement them. Particularly, feeding input by encoding classical data into quantum computers and obtaining the output from calculations performed constitutes some overhead [Aar15], which can be alleviated to some extent using qRAM and operating directly on quantum data in a manner similar to variational methods [PMS+14].

3.3.3 Pulse-level Benchmarks

3.3.4 Benewop Benchmark

The Benewop Benchmark [GPD20b] focuses on performing noise extrapolation at the pulse level by stretching a pulse for a certain duration to drive a qubit from the $|0\rangle$ state to the $|1\rangle$ state. It first utilizes Rabi Oscillations to obtain the parameters that generate a calibrated π pulse by setting different drive amplitudes and σ values to a gaussian pulse, a procedure known to be used in IBM quantum devices for default gate pulse calibration. After this procedure, the calibrated pulse is applied with a constant drive amplitude in the middle for a certain drive time to analyze the relation between drive amplitudes applied to qubits and decay rates for a given Rabi period. Using this information and after fitting results from different amounts Rabi cycles, noise factors for the different pulses can be obtained as a mathematically representative error metric for qubits.

3.4 Quantum Gate Calibration

Quantum gate calibration is a process underwent in the lab immediately after a quantum device has been fabricated and installed into a system [Tea20], and is constantly done after a time period to ensure accurate default gate pulse application up to a certain degree, removing as much accumulated noise as possible.

3.4.1 Frequency Sweep

The first subprocess is performing a Frequency Sweep to search for the qubit frequency. The qubit frequency is the difference in energy between the ground and excited states [Tea20], and is of upmost importance as a slight difference in frequency GHz could extinguish the chance to reach an excited state in the qubit. We use Armonk device's estimated qubit frequency as the center frequency for the sweep and span a short range of frequencies with small increments to find the frequency that returns the highest measured signal, as appears in Figure 3.9. The highest measured signal is an indicator of driving a qubit off-resonance, important for effectively making a change in its state.

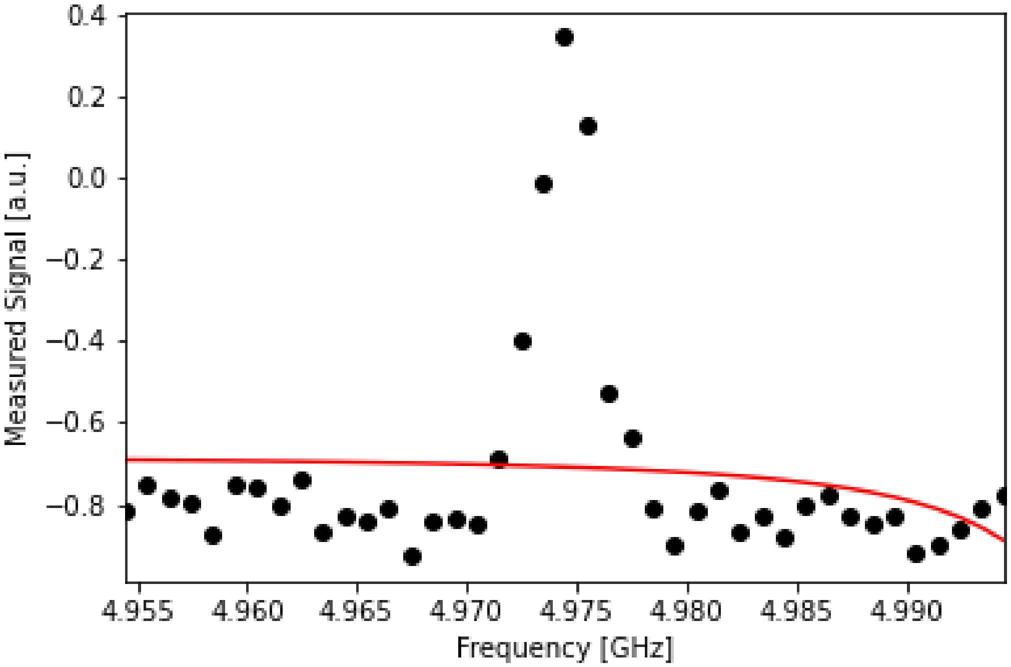


Figure 3.9: Frequency Sweep shot measurement results as black dots fitted to a Lorentzian shape on the red line.

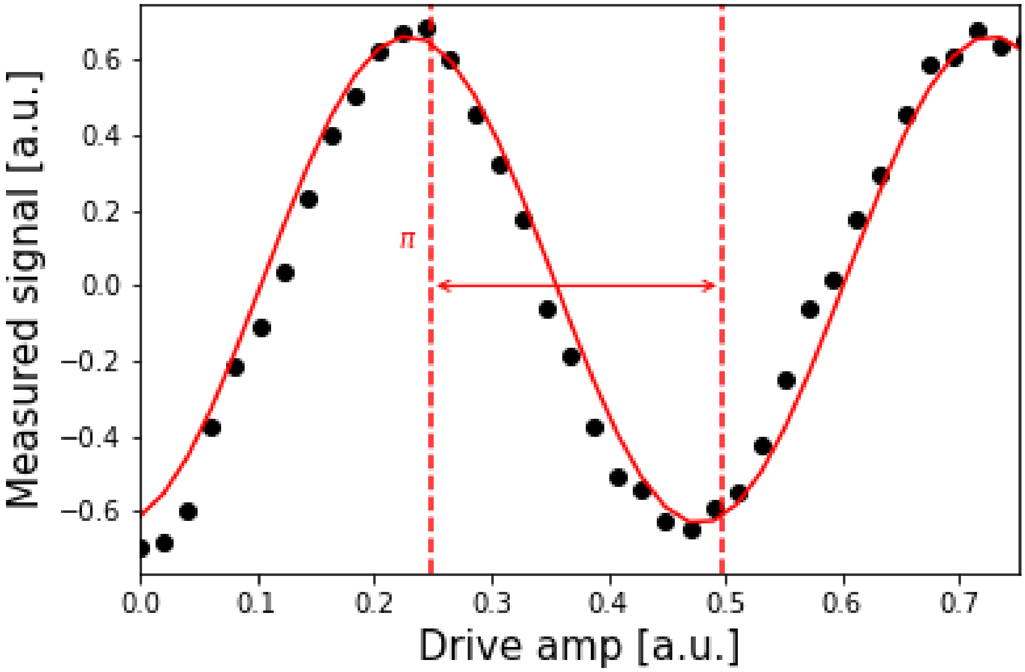


Figure 3.10: Rabi Oscillation measurement signals corresponding to shots as black dots on different drive amplitudes, fitted to a sinusoidal curve on the red line. The π pulse appears as the amplitude difference between the largest difference in measurement signals.

3.4.2 Rabi Oscillations

After knowing the correct calibrated frequency to drive a transition in the qubit's state, we can continue with testing the amplitude of gaussian pulses that represent a complete transition from $|0\rangle$ to $|1\rangle$ state. Rabi Oscillations are sinusoidal oscillations [Tea20] which describe the fraction of the shots that successfully transitioned to the $|1\rangle$ state. In the case of a X gate pulse (π pulse) we expect to observe the maximum difference in measured signals to represent a complete transition. The drive amplitude difference corresponding to the maximum measured signal difference represents the amplitude we will use for the calibrated gaussian pulse, which can be seen on Figure 3.10.

3.4.3 Discriminating Quantum States

A posterior process for obtaining the quantum state counts from the result of a pulse-level execution is discriminating the quantum state signals present at readout in an in-phase and quadrature (IQ) plane like the one in figure 3.3. This is an important task as high assignment fidelities allow accurate measurements to be performed on quantum devices and therefore obtaining more useful results. It is currently done using ML models to classify trajectory [MGCC15] and IQ data [Tea20] with an experimental focus. While providers offer quantum state counts by applying their own readout scheme, users can apply models to discriminate states in a custom manner to not only obtain higher assignment fidelity but also to infer properties of a quantum device and benchmark it [AKE+20]. While this experiment can be performed using any pulse, [Tea20] and [AKE+20] utilize a π pulse to ensure clear differences in both sets of signals ($|0\rangle$ and $|1\rangle$ state signals) for further analysis.

3.5 Quantum Optimal Control

One of the biggest challenges for near future quantum computing implementations is the reduction of noise through strategies implementable with a moderate number of qubits, while still obtaining high fidelity. Quantum error mitigation schemes [GPD20a] [SEFT20] [TBG17] have recently been appearing for the task of noise reduction in quantum computers, and attempts to implement them have been successful. Among these schemes, Quantum Optimal Control (QOC) has shown great relevance throughout the history of quantum computers as its objective has been to stimulate quantum evolution to a desired state through the use of control laser fields, allowing the design of laser pulse shapes to do so [BCR10]. One important use case of QOC is nuclear magnetic resonance spectroscopy as it is focused on designing pulse sequences that maximize a certain performance criterion, consequently solving problems like minimizing the time required to produce a given unitary propagator [KRK$^+$05]. This is done through the optimization of pulses using machine learning elements such as loss functions and gradient ascent algorithms, so as to reach maximum overlap with a certain target unitary or quantum gate and in doing so, greatly improving fidelity [WER$^+$20] and performance indicators for more precise qubit state transitions. QOC algorithms can focus on different types of pulses as initial guesses that are then optimized depending on the desired target pulse shape and iteration update strategies. These can be pulse envelopes such as piecewise constant pulses, gaussian pulses, square pulses, among others.

3.5.1 GRAPE

Gradient Ascent Pulse Engineering is a gradient ascent-based QOC algorithm which focuses on piecewise constant pulses being applied to a qubit on a quantum device, optimizing the pulse by concurrently updating the control amplitudes of the pulse. One of the GRAPE approach's biggest advantages is the fact that it only requires two full time evolutions on the application of a pulse to correctly update all the control terms of a pulse, updating each one at each step of the quantum system's evolution [KRK$^+$05]. The first full time evolution is determined by the application of unitary propagators to an initial unitary (or density matrix) for each step j. The equation that represents this at the end of the first evolution is

$$U(T) = U(t_n)U(t_{n-1})...U(t_0)$$

where $U(t_n) = \exp[-i\mathcal{H}(t_n)\nabla t]$ is the unitary propagator resulting from the time-dependent Hamiltonian expressed by $\mathcal{H}(t_n) = \mathcal{H}_0 + \sum_{j=0}^{m} u_j(t_n)\mathcal{H}_j$, with $\mathcal{H}_0$ being the independent term and $\mathcal{H}_j$ being the pulse-dependent term. Note that the expressed Hamiltonian formula is valid for any amount of pulse-dependent terms, whereas if only one is supported, $\sum_{j=0}^{m} u_j(t_n)\mathcal{H}_j$ becomes $u(t_n)\mathcal{H}$. The forward-propagator at step j is determined by

$$\rho_j = U_j U_{j-1} ... U_1$$

For the second full time evolution, a back-propagator is calculated with the formula

$$\lambda_j = U_{j+1}^{\dagger} U_{j+2}^{\dagger} ... U_N^{\dagger}$$

These propagators are useful for evaluating the performance function after a full time evolution, and for calculating the gradient that will modify the control amplitudes of the pulse. The equations for this are, respectively:

$$\epsilon > 1 - \phi = \langle U_{target} | U(T) \rangle$$

$$\frac{\delta\phi}{\delta u_k(j)} = -Re \left\langle \lambda_j U_{target} | \rho_j \right\rangle .$$

A flow chart of the GRAPE algorithm steps is presented in Figure 3.11.

3.6 Quantum Programming Stacks

We conduct a survey and research on available quantum programming stacks for connotation of their main assets and features to find the best choice for designing, implementing and testing QOC algorithms.

3.6.1 Q#

“Q# is Microsoft’s open-source programming language for developing and running quantum algorithms. It’s part of the Quantum Development Kit (QDK), which

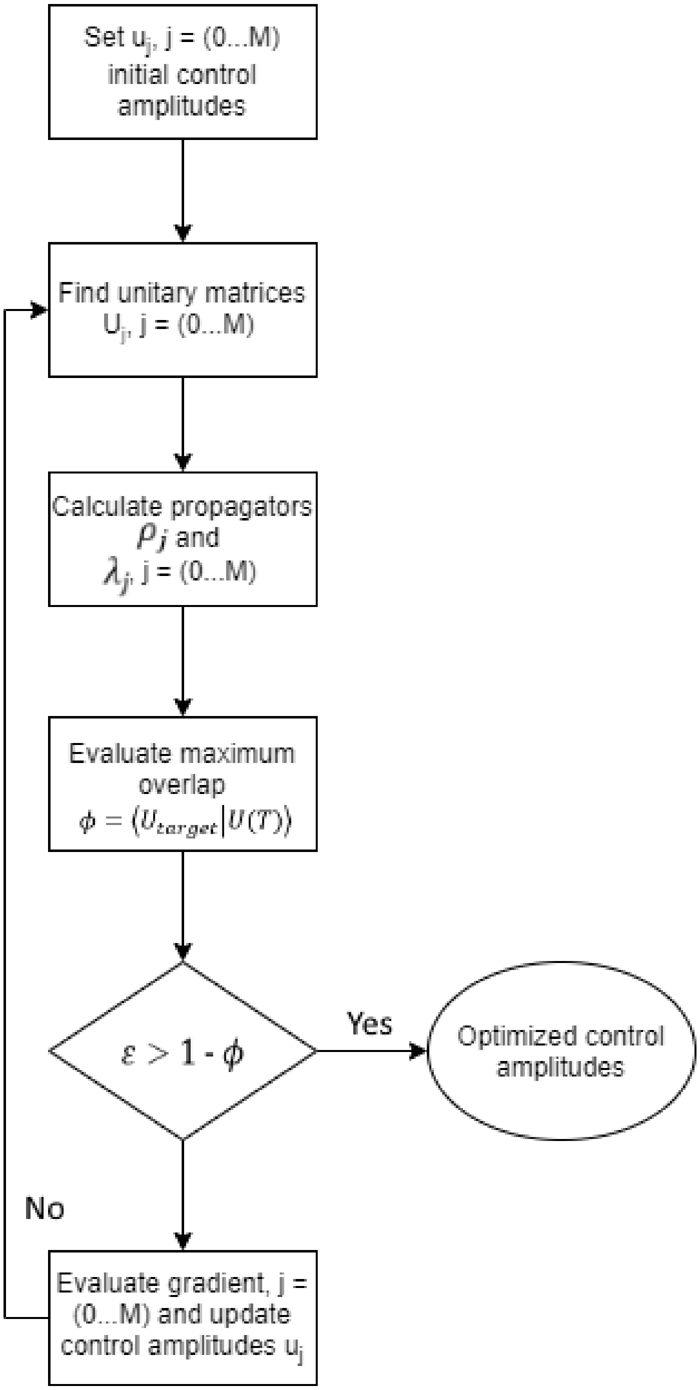

Set u_j, j = (0...M) initial control amplitudes
Find unitary matrices U_j, j = (0...M)
Calculate propagators ρ_j and λ_j, j = (0...M)
Evaluate maximum overlap $\phi = \langle U_{target} | U(T) \rangle$
$\varepsilon > 1 - \phi$
Yes
Optimized control amplitudes
No
Evaluate gradient, j = (0...M) and update control amplitudes u_j

Figure 3.11: GRAPE algorithm workflow steps.

includes Q libraries, quantum simulators, extensions for other programming environments, and API documentation. In addition to the Standard Q library, the QDK includes Chemistry, Machine Learning, and Numeric libraries." [Bra] This programming language has no current capability to connect to a real quantum computer [LaR19] and, combined with not offering any built-in QOC algorithms, makes it not an ideal programming language for the implementation of QOC algorithms.

3.6.2 CirQ

"Cirq is a software library for writing, manipulating, and optimizing quantum circuits and then running them against quantum computers and simulators. Cirq attempts to expose the details of hardware, instead of abstracting them away, because, in the Noisy Intermediate-Scale Quantum (NISQ) regime, these details determine whether or not it is possible to execute a circuit at all." [Dev] Cirq allows programmers to define Schedules and Devices to work at the lowest level of algorithm execution, for example specifying the duration of pulses and gates [LaR19]. Unfortunately, Cirq does not currently provide cloud access to Google's quantum computers for general users and is therefore not an ideal option.

3.6.3 Strawberry Fields

"Strawberry Fields is an open-source quantum programming architecture for light-based quantum computers. Built in Python, Strawberry Fields is a full-stack library for design, simulation, optimization, and quantum machine learning of continuous-variable circuits." [KIQ$^+$19] Although only quantum simulators are available, built-in optimization algorithms and various machine learning tools are provided with the TensorFlow simulator it uses. [KIQ$^+$19]

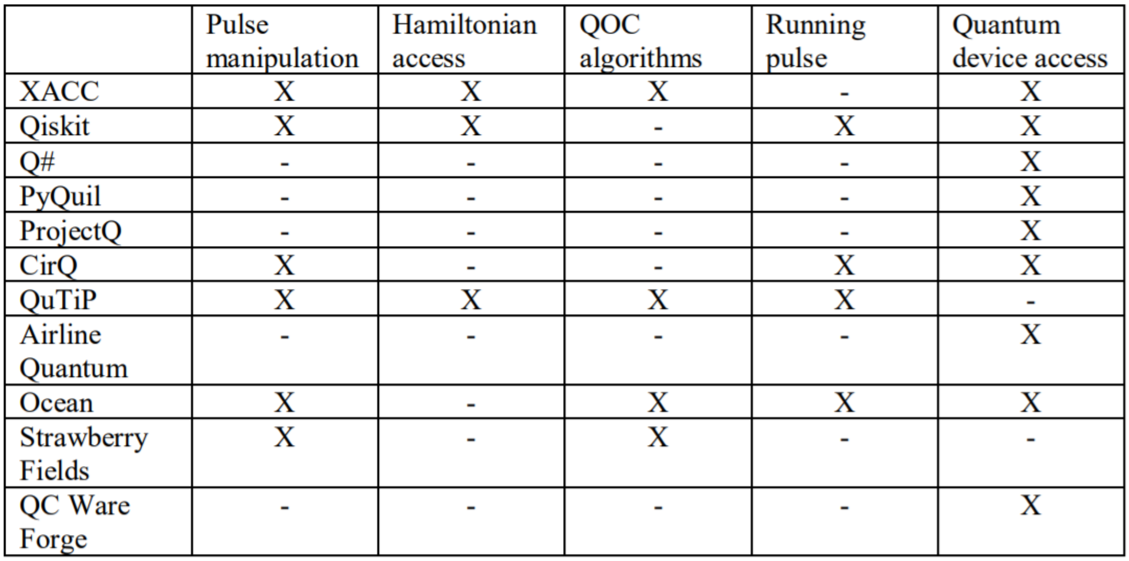
3.6.4 XACC

“XACC is an extensible compilation framework for hybrid quantum-classical computing architectures. It enables the execution of quantum kernels on IBM, Rigetti, IonQ, and D-Wave QPUs and simulators. XACC is the only quantum-classical programming framework that provides a system-level software infrastructure permitting closer CPU-QPU integration. XACC further differentiates itself by promoting device interoperability as a core feature and design goal. It is also able to target different quantum computing models: gate-based and annealing.” [McC] In this way, XACC can implement QOC capabilities and already does so with its built-in GRAPE, GOAT and Krotov algorithms.

3.6.5 Qiskit

“Qiskit is an open-source framework for working with quantum computers at the level of circuits, pulses, and algorithms.” [qis] Also, Qiskit is composed of various use cases allowing its comprehensive use in low level to real world applications. For the purpose of implementing QOC algorithms, it allows pulse-level control of a general quantum device using OpenPulse [CBSG17]. Despite having its own limitations in the kind of results available after each run of a pulse on a schedule, it provides many useful insights on the state of a qubit after a pulse has been applied, such as counts on each state (between the ground and excited states), and signal measurements.

To select the ideal quantum programming stack, the differentiating factor is the ability to manipulate pulses as well as providing a way to access a quantum device’s Hamiltonian. Other important factors to consider are pre-existing QOC algorithms the stack has and the possibility of running an optimized pulse on a quantum device. The stacks mentioned in this report along with other less relevant stacks for QOC



	Pulse manipulation	Hamiltonian access	QOC algorithms	Running pulse	Quantum device access
XACC	X	X	X	-	X
Qiskit	X	X	-	X	X
Q#	-	-	-	-	X
PyQuil	-	-	-	-	X
ProjectQ	-	-	-	-	X
CirQ	X	-	-	X	X
QuTiP	X	X	X	X	-
Airline Quantum	-	-	-	-	X
Ocean	X	-	X	X	X
Strawberry Fields	X	-	X	-	-
QC Ware Forge	-	-	-	-	X

Figure 3.12: Comparison between different quantum programming stacks on features that allow QOC algorithm implementation, as well as testing the pulses optimized with these algorithms on physical quantum devices.

are compared in Figure 3.12 for finding the ideal option to base QOC algorithms on.

After comparing various quantum programming stacks, Qiskit proves to be the ideal quantum programming stack for general QOC algorithm implementation purposes, as it allows to effectively calculate the real fidelity of a pulse through one of its quantum devices. Although no pre-existing QOC algorithms exist in Qiskit, Qiskit's OnePulse allows deeper insight into pulses and quantum device Hamiltonian configuration which, given the correct values for all variables in the Hamiltonian, can enable a precise implementation of GRAPE as can be done with their pulse simulator. Other fundamental stacks to be considered are XACC and QuTiP, as XACC already implements GRAPE, GOAT and Krotov algorithms, and can access many different quantum devices to test quantum circuits with an efficient compilation step. QuTiP implements GRAPE and CRAB given a Hamiltonian with all its variables and values, but can't test pulses on quantum devices. These two stacks help to obtain useful insight on the correct implementation of those algorithms, and allow to observe fidelity changes and differences.

CHAPTER 4

METHODOLOGY

4.1 Closed-loop optimization algorithm

A novel QOC hybrid algorithm loosely inspired by GRAPE is proposed and tested, taking into consideration a few of the limitations present in the implementation of the GRAPE algorithm on Qiskit, such as not having the values of every variable in the Hamiltonian [Tea21a], and being unable to otherwise calculate the resulting propagator for each time step, making it difficult to precisely obtain the gradient ascent increment for each step. The main difference in this implementation is reinforcement learning taking the place of propagator calculation. With OpenPulse [MAB$^+$18], it is possible to test multiple pulses concurrently on a quantum device given that each pulse is independent from one another, meaning that each pulse will run independently and return different results.

The Closed-loop optimization algorithm optimizes a piece-wise pulse by performing iterative runs on a quantum computer, updating a set of amplitudes by a small offset on each run to find the amplitudes that hold the highest fidelity towards a target quantum circuit or unitary. The algorithm implements the divide-and-conquer programming paradigm to select the set of amplitudes by creating partitions in the pulse as is shown in Figure 4.2a to then drastically reduce execution time and the total amount of runs required to perform the optimization. Using this to our advantage, we can obtain the fidelity from each amplitude update as a replacement for gradient calculation. To use the algorithm, we first specify an initial schedule constructed with pulses like random pulses or calibrated pulses, along with other instructions such as phase shifts. We also set a number p of different pulses partitions with a small offset of $\alpha(i - \frac{p}{2})$ on each, where α is a small step, and i is the current partition index. Therefore, for a control amplitude at step j, p partitions with control amplitude

$$U_{i,j} = u_j + \alpha(i - \frac{p}{2}) \text{ for } i = 0, 1, ..., p-1$$

are constructed and simultaneously executed on a quantum device to obtain their counts and update the control pulse with the partition that obtained the highest fidelity, finally returning a schedule like the one in Figure 4.2b.

As with the GRAPE algorithm, a potential problem is the amount of control amplitudes to be optimized, and thus the amount of runs required on a quantum device to correctly update each control amplitude. Actually, updating all the control amplitudes on this algorithm may be more time-consuming than only utilizing the propagators on a simulated case, since with IBM's Armonk device, each batch of pulses are attached to a queue and are executed on a FIFO basis (except when reserved for dedicated use, an ideal use case for this algorithm). Despite this, our solution provides a great time reduction in comparison to a straight-forward approach of running the modified pulse for each amplitude as can be done with current solutions [QC], given the constraint of the shortest pulse length allowed on Qiskit being 64 amplitudes, with most single qubit default pulses consisting of 640 amplitudes. Our way to solve this problem is by discretizing a pulse onto divisions with less granularity than the constant piece-wise pulse, dividing the full pulse into d parts to be updated in the same way as would be done with the highest granularity, except that only d executions would be necessary to update all the control amplitudes. In this way, even though not every change in a control amplitude would reach maximum performance, great fidelity improvements would be made given enough partitions p. Figure 4.1 illustrates the general outline of this algorithm.

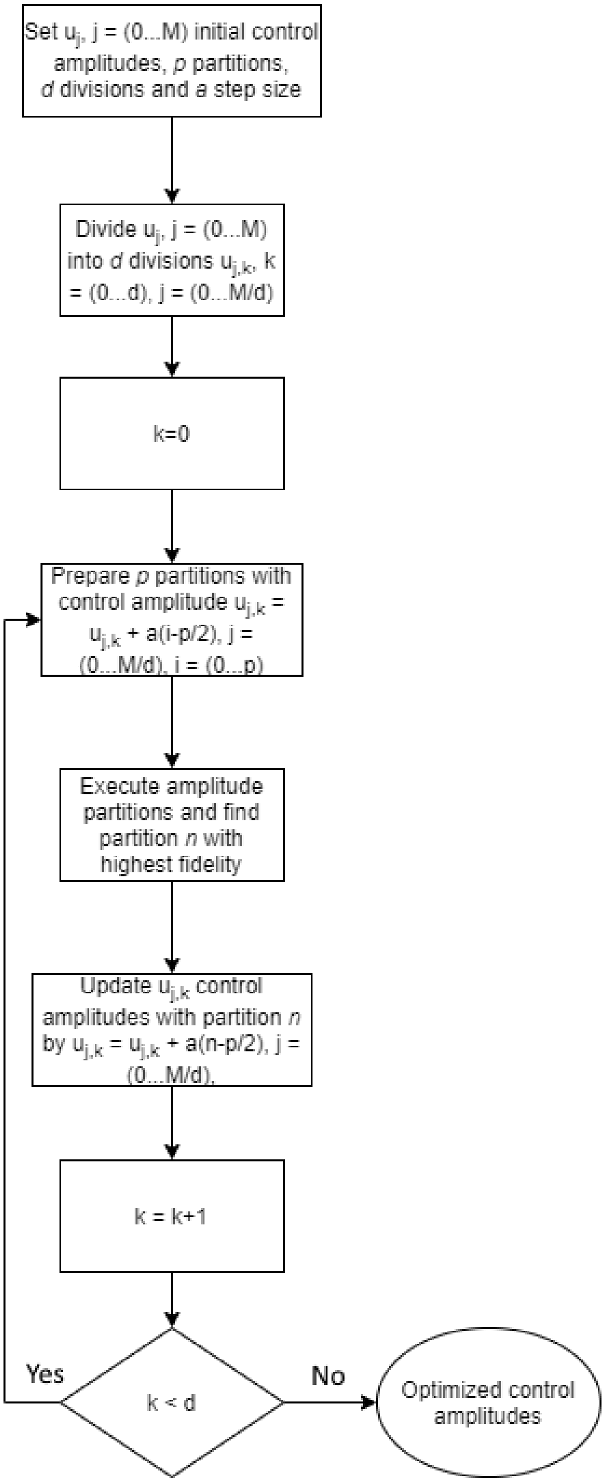
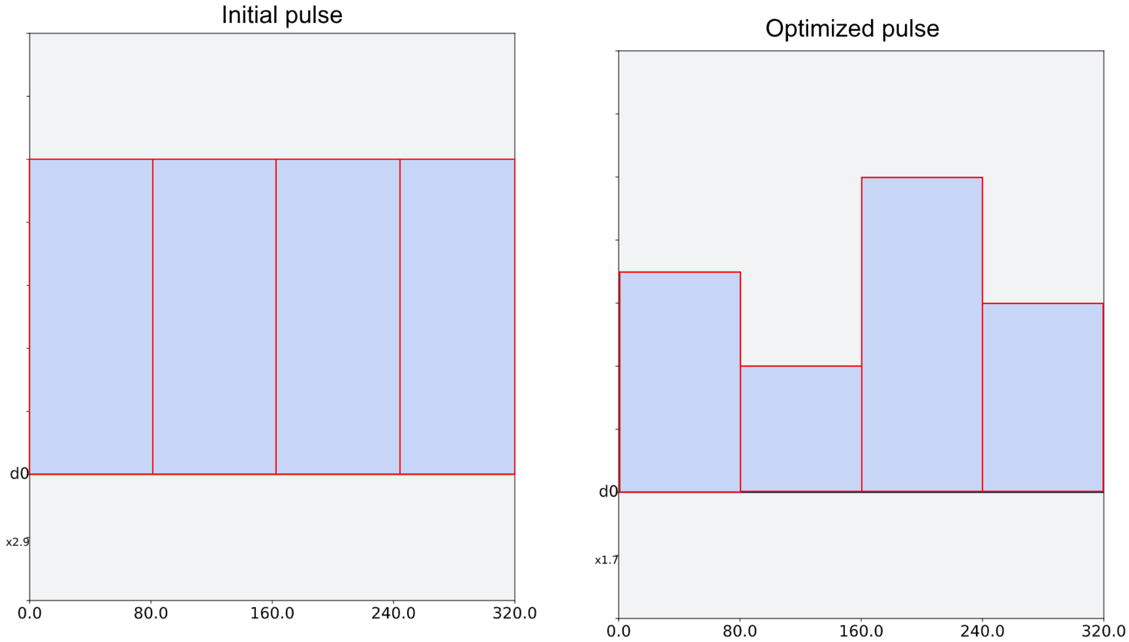

Set u_j, j = (0...M) initial control amplitudes, *p* partitions, *d* divisions and *a* step size
Divide u_j, j = (0...M) into *d* divisions $u_{j,k}$, k = (0...d), j = (0...M/d)
k=0
Prepare *p* partitions with control amplitude $u_{j,k}$ = $u_{j,k}$ + a(i-p/2), j = (0...M/d), i = (0...p)
Execute amplitude partitions and find partition *n* with highest fidelity
Update $u_{j,k}$ control amplitudes with partition *n* by $u_{j,k}$ = $u_{j,k}$ + a(n-p/2), j = (0...M/d),
k = k+1
k < d
Yes
No
Optimized control amplitudes

Figure 4.1: Closed-loop optimization algorithm workflow steps.



(a) Initial constant pulse separated into 4 divisions.

(b) Optimized π pulse with 4 divisions.

Figure 4.2: Illustration of pulse divisions performed on a constant pulse to optimize into a π pulse using Quantum Process Tomography.

4.1.1 Closed-loop optimizer features and constraints

The amplitude updates that hardware-based pulse optimization relies on [QC] can be a great speed constraint if any fine-tuning were to be performed on an initial pulse. Because of this, one of the most important features for the use of the algorithm is the ability to queue multiple pulse schedules at the same time on a quantum device, reducing the amount of jobs required to fully optimize the pulse. Final pulse fidelity is determined by the granularity of a pulse optimization, with decent results being obtainable with as little as two divisions for any amount of p partitions below the maximum number of experiments C (around 900 as can be seen in the IBM Quantum Experience main page [Tea]) allowed on a single job of a quantum device.

Readout correction is implemented at result retrieval to reduce effects of noise appearing on the IQ signals of pulse results. Qiskit offers this functionality by running a set of measurement calibration circuits before running the actual pulse. This allows the use of a measurement fitter to obtain accurate readout results at the cost of an extra job run per division d of the optimization routine. While notable noise mitigation can be obtained with this functionality [Tea21b], the user can decide to disable it if a longer optimization process is to be avoided.

The algorithm enables the use of several error modes to specify the way fidelity is calculated between a custom pulse or its counts and a target unitary. Among the error modes are Quantum State Tomography [NC11] for a comprehensive measurement of the desired state and QPT [NC11] to analyze the whole spectrum of basis gates for a given amount of qubits. QPT is mainly implemented as a way to obtain the fidelity of single-qubit and two-qubit gate pulses, as a total of 144 combinations of basis gate transitions can be observed. Due to this, two-qubit gate pulses only permit 6 partitions per job execution on most IBM quantum devices, greatly slowing down the optimization process. A workaround for this problem is utilizing the algorithm on several full pulse optimization iterations with a smaller search space on each consecutive iteration by reducing the α parameter of the algorithm. RB can also be used to solve slow optimization times on two-qubit gates.

4.1.2 Initial optimizer configuration

For the configuration of the closed-loop optimization algorithm we specify a quantum device on which we perform the optimization, an initial pulse schedule we wish to optimize and the target unitary. The initial pulse can be given in several ways supported by Qiskit's pulse library, such as a custom pulse, a random pulse with a certain duration, and a pulse with a Gaussian, Square Gaussian, DRAG or a constant envelope. In addition to the available options for pulse generation, the algorithm can also receive a quantum circuit as input to obtain the corresponding calibrated pulses for each basis gate after a transpilation step. This enables us to specify any quantum circuit for which we wish to optimize a default calibrated pulse based on the schedules stored on a specific quantum device. We can also select an amount of d divisions which represent the amount of different pulses we test in a single run and the p pulse partitions we wish to utilize, or the amount of runs we will execute for a given pulse. It is recommended to use the maximum odd amount of divisions a quantum device can support, equivalent to the amount of pulses a quantum device can execute in a single run rounded down to the nearest odd number.

For comparing pulses offered by default on IBMQ quantum devices with optimized pulses from the closed-loop pulse engineering algorithm, we optimize a random X gate pulse applied on the ibmq_bogota device by setting a random initial constant amplitude between 0.25 and 0.5 with a width of 160 amplitudes for a total gate duration of $35.552ns$. The initial pulse can be seen in Figure 4.3. Also, a default DRAG pulse is retrieved from the ibmq_bogota device and is shown in Figure 4.4. This DRAG pulse represents a π rotation as a X gate pulse calibrated by IBM before retrieval. The optimization procedure was performed using QPT on each optimization step and error bars were retrieved by running the initial random pulse, the default pulse and the random optimized pulse 20 times.

4.1.3 Optimizer post-execution result retrieval

After executing the algorithm, we can obtain the best optimized pulse schedule for our target unitary, along with the pulse fidelity and the counts from its execution. We can also obtain a complete log of the counts, loss rates and modified schedules used in the optimization process, so as to gain deeper insight into the pulse amplitude updates. To obtain the difference in fidelity between the optimized pulse and the initial pulse, the difference in loss rates between the first loss rate and the final loss rate can be calculated. Particularly, the last element of each returned list contains the information corresponding to the optimized pulse.

4.2 Quantum KMeans

The Quantum KMeans algorithm is an implementation of the KMeans in which the distances between data points and cluster centroids are calculated using destructive interference on a quantum circuit. To achieve this, classical data must be mapped onto a quantum circuit, while a quantum algorithm for distance computation must be implemented. Our algorithm uses two data encoding strategies and sends jobs in batches to IBMQ devices. The algorithm was written in Python using the Qiskit library and several data preprocessing functions available through the Scikit-learn library.

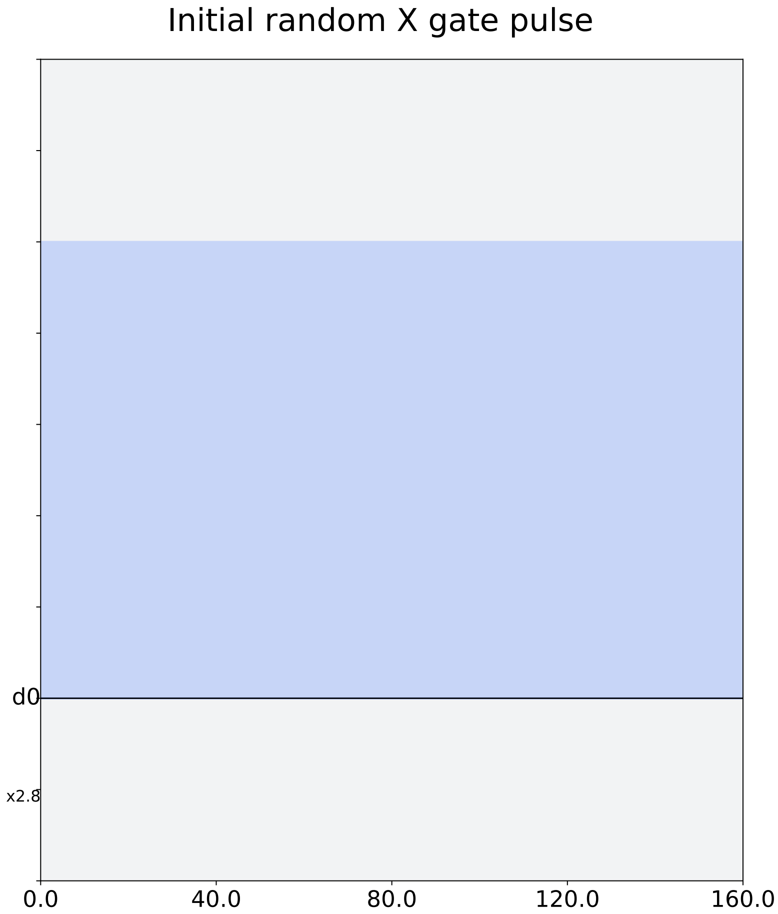


Figure 4.3: Initial constant random pulse with an amplitude of 0.35 applied to the ibmq_bogota device with the independent axis being the amount of individual pulse amplitudes and the dependent axis being the amplitude.

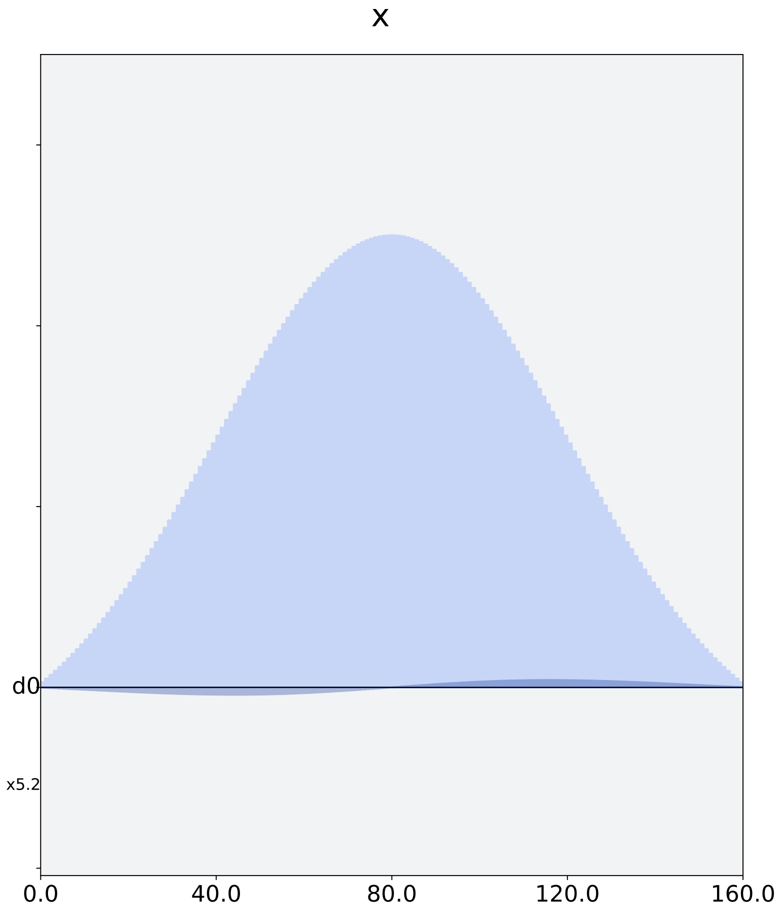


Figure 4.4: Default DRAG π pulse as a calibrated X gate pulse with an amplitude of ≈ 0.19, $\sigma \approx 40$ and $\beta \approx -1.16$ applied to the ibmq_bogota device with the independent axis being the amount of individual pulse amplitudes and the dependent axis being the amplitude.

Out of the two encoding strategies, amplitude encoding of the input dataset onto qubits provides a convenient way to map n-dimensional input vectors, given the wide range of different values that can be obtained. Since qubit amplitudes must have a norm of 1, normalization of input vectors must be done in order to encode amplitudes onto quantum states. That is, for an n-dimensional input vector

$$\mathbf{a} = \begin{pmatrix} a_0 & a_1 & ... & a_n \end{pmatrix}, \mathbf{a}' = \frac{\mathbf{a}}{\|\mathbf{a}\|} = \begin{pmatrix} a'_0 & a'_1 & ... & a'_n \end{pmatrix}$$

will be encoded as

$$|\psi\rangle = a'_0 |0\rangle + a'_1 |1\rangle + ... + a'_n |n\rangle .$$

Another way of encoding data implemented by the Quantum KMeans algorithm is establishing a rotation angle θ in which qubits are set with an Ry gate. As a specific case, given a 2-dimensional input vector

$$\mathbf{a} = \begin{pmatrix} a_0 & a_1 \end{pmatrix}, \theta = \arctan \frac{a_1}{a_0}$$

is the angle that will be applied as an $Ry(\theta)$ gate to a specific qubit.

The next important aspect is calculating distances using a quantum circuit. For this, a Swap test circuit is applied as shown in Figure 4.5 to find how much two quantum states differ and therefore act as a distance measurement in the form

$$D(|x\rangle, |y\rangle) = \sqrt{2 - 2|\langle x|y\rangle|}.$$

Despite one-on-one distance calculations being a slow approach to a KMeans algorithm alternative, our implementation partially makes up for the speed difference by batching the circuits used for distance calculations on IBMQ's quantum devices, since as much as 900 circuits can currently be sent in a single job. This reduces the

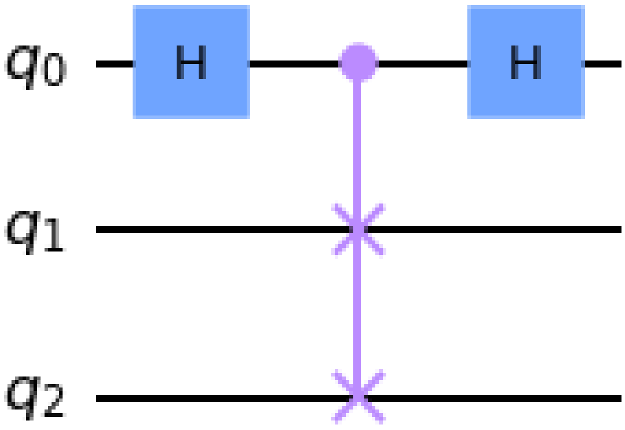


Figure 4.5: SwapTest circuit for measuring distances between qubit 1 (second line) and qubit 2 (third line).

amount of separate jobs a classical computer has to send to a quantum computer as given $c = 900$ circuits, only $\frac{nk}{c}$ jobs are sent for each iteration of the algorithm in contrast to the nk jobs required for each local simulated iteration. Convergence speed is also accounted for by including QKMeans++, a quantum version of the KMeans++ initial cluster center guessing strategy in which initial cluster centers are assigned to a data point based on the greatest distance it has to its closest cluster center [AV07]. The general outline of the algorithm is described in Algorithm 1.

4.3 Characterizating crosstalk on a quantum computer

For the task of characterizing crosstalk on a quantum computer using ML and QML, we start by obtaining IQ signal data from all qubits on a quantum device when applying a ground and a π pulse to obtain both $|0\rangle$ and $|1\rangle$ state data. We then utilize the QML algorithm to group clusters together in a similar fashion as would

Algorithm 1: Quantum K-Means algorithm in pseudocode

```
Input: $X, backend, init, n\_clusters, max\_iter, tol$
Output: $qkmeans$: A trained model
$X \longleftarrow$ preprocess($X$)
$finished \longleftarrow false$
$iteration \longleftarrow 0$
if $init ==$'qk-means++' then
    $cluster\_centers \longleftarrow$ qkmeans_plusplus($X, cluster\_centers, backend$)
else
    $cluster\_centers \longleftarrow$ sample $n\_clusters$ elements from $X$
end
while $not\ finished\ and\ iteration < max\_iter$ do
    $distances \longleftarrow$ CSWAP($X, cluster\_centers, backend$)
    $labels \longleftarrow$ indices of minimum $distances$
    $new\_centroids \longleftarrow$ group $X$ by $labels$ and average
    $inertia \longleftarrow \sum\sum |new\_centroids - cluster\_centers|$
    if $inertia < tol$ then
        $finished \longleftarrow true$
    end
    $cluster\_centers \longleftarrow new\_centroids$
    $iteration \longleftarrow iteration + 1$
end
```

be done with classical ML clustering algorithms, while also performing clustering using classical ML algorithms. The results would be assignment fidelities for each qubit that enable the construction of learning curves with information describing the mean, CI (confidence interval) and convergence rate. We then compare the resulting scores from the classical KMeans and the QKMeans algorithm on every qubit in order to make conclusions on the performance of both algorithms when discriminating quantum states.

The next set of experiments is oriented towards analyzing the data for information on cross-talk using correlation coefficients and the assignment fidelities as well as the CI of each schedule in a set of schedules. This is done by preparing schedules for the $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$ states using two neighboring qubits and fitting two

ML models on a dataset comprised of the $|0\rangle$ and $|1\rangle$ state for a specific qubit ($|01\rangle$ and $|00\rangle$ with the least significant bit corresponding to the qubit of choice), and states with all four schedules, respectively. Information on assignment fidelities is then compared to find any significant cross-talk happening on a qubit coupling. We also analyze Pearson coefficients between arrays of signal data retrieved in the form $r_j(ES_{i,X}, GS_{i,Y})$ where $X, Y \in \{I, Q\}$ and qubit i is in state $j \in \{0, 1\}$ when the other qubit is in the excited state (ES) and ground state (GS). Results from the correlation analysis are contrasted to verify cross-talk on neighboring qubit couples.

4.4 Leakage analysis experiment configuration

Based on crosstalk effects found in qubit couples (1, 2) and (2, 3), we perform a leakage analysis on the pulse data retrieved from the correlation experiments done in the ibmq_bogota device to find evidence of leakage due to crosstalk. With several studies confirming that leakage may appear due to crosstalk, our task is to find leakage on the quantum device of interest through the IQ pulse signals from schedules with configurations set for crosstalk, applying a ML approach. We use the KMeans clustering algorithm to analyze the effects of clustering with different amounts of clusters and the assignment fidelity values they provide. To that effect, we use two and three clusters representing the $|0\rangle$ and $|1\rangle$ states, and $|0\rangle$, $|1\rangle$ and $|2\rangle$ states, respectively. We perform the clustering on all qubit couple data retrieved from preparing the $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$ states on each couple to be able to compare possible leakage into the $|2\rangle$ state on all qubit couples. The criteria we use to determine presence of leakage is determined by both a visual analysis and an analysis on the assignment fidelity resulting from clustering with two and three clusters. That is, if we can visually determine a third cluster, more often than not we can calculate the amount of leakage contributed by that third cluster. A similar but more numerical strategy is to compare the assignment fidelity from both cases of clustering, where if the difference between the scores is small enough, a third state can be identified as the cluster with the least amount of data points. If the data points corresponding to the third cluster represent a small proportion compared to the amount of data points in the other clusters, we can also conclude the presence of leakage into the $|2\rangle$ state. When concluding leakage on a qubit couple, we intend to mitigate the effects of leakage by erasing leaked data points as if they were atypical data points, effectively removing effects of leakage at the cost of a small portion of information. In fact, an additional reason why the proportion of leaked data points should be taken into consideration is the amount of data points to be potentially erased.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Gate optimization

After optimizing the initial random gate on imbq_bogota with the closed-loop optimization algorithm using 8 divisions and 60 partitions, the resulting pulse rapidly gained a fidelity of 94.3% in the third job run with a maximum fidelity of 95.9%. The convergence curve is showed in Figure 5.1. The final optimized pulse is shown in Figure 5.2. The initial random X gate process fidelity was 0.01332 ± 0.00358, the optimized X gate process fidelity was 0.93063 ± 0.00632 and the default X gate process fidelity was 0.95560 ± 0.00839. This shows that the gate optimization using process fidelity successfully improves that metric over random initial pulses, while being close but not showing improved process fidelity over the default pulse. RB was also used to test the final single-qubit error rate of the same optimized pulse and the results were: the random X gate had a fidelity of 0.82939 ± 0.20232, the optimized X gate fidelity of 0.75382 ± 0.18798 and the default X gate fidelity of 0.99777 ± 0.00058. Considering that QPT is a more complete benchmark than RB as measurements over all basis gates are performed, this difference in error rates may be explained due to a pulse with a very high error showing obsolete results on error rates calculated using RB, as is confirmed in several studies. Nonetheless, we can observe that the optimized pulse also holds a decent fidelity when being measured with RB.

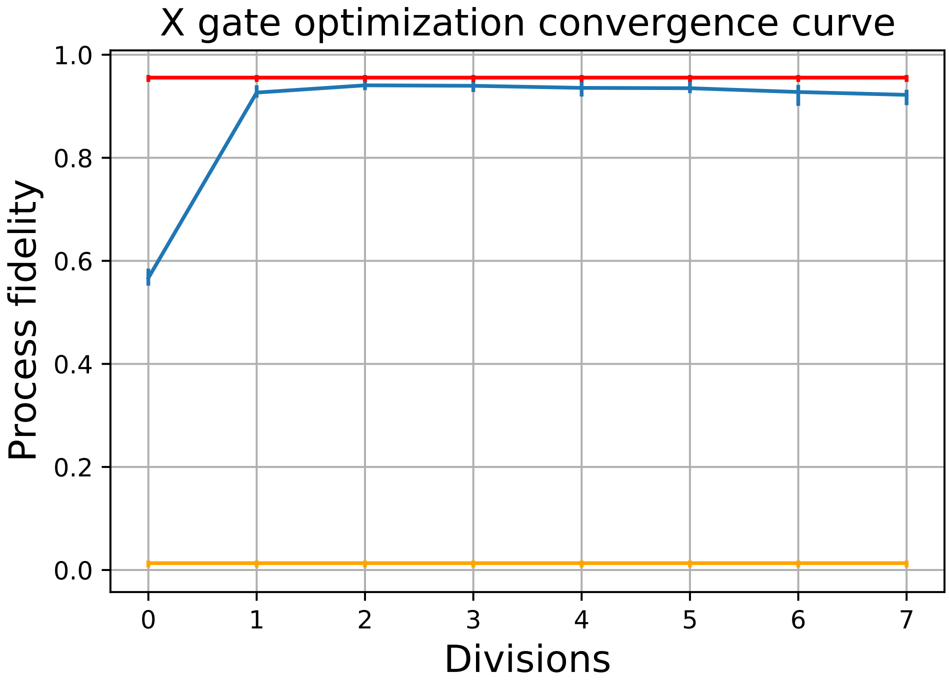


Figure 5.1: Resulting X gate pulse convergence curve from optimizing the initial random constant pulse on ibmq_bogota with error bars. The blue line shows the process fidelity of the optimized pulse at each division (or job run), the yellow line shows the process fidelity of the initial initial random constant pulse and the red line shows the process fidelity of the default π pulse.

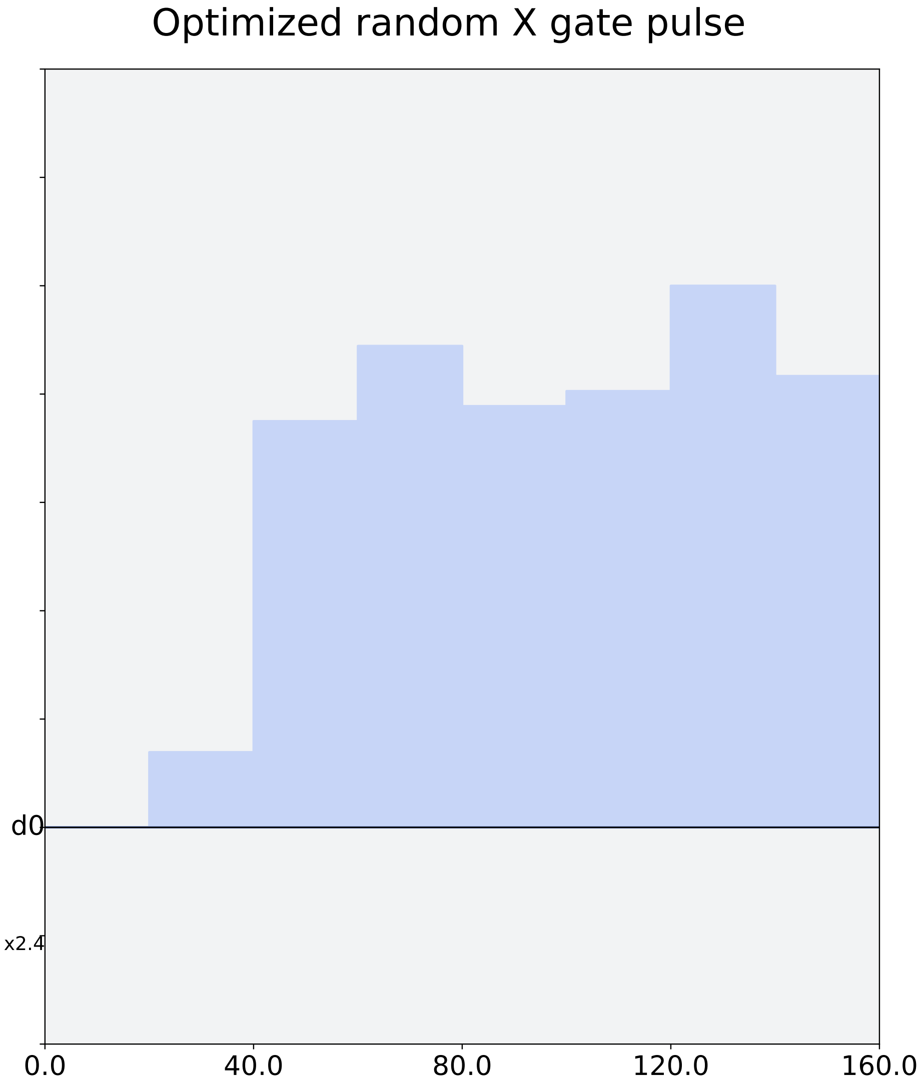


Figure 5.2: X gate pulse after optimizing the initial random constant pulse on ibmq_bogota.

	Single	Both
Q0	0.753 ±0.2218	0.793 ±0.2182
Q1	0.752 ±0.1986	0.714 ±0.1948
Q1	0.666 ±0.1842	0.714 ±0.1947
Q2	0.704 ±0.2211	0.750 ±0.2270
Q2	0.845 ±0.2107	0.702 ±0.2282
Q3	0.798 ±0.2370	0.797 ±0.2346
Q3	0.701 ±0.2357	0.750 ±0.2379
Q4	0.843 ±0.2167	0.750 ±0.2384

Table 5.1: Assignment fidelity scores for the test dataset on all neighboring qubit couples with $|0\rangle$ and $|1\rangle$ state schedules for single and $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$ state schedules for both using the classical KMeans approach.

	Single	Both
Q0	0.881 ±0.0173	0.889 ±0.0175
Q1	0.825 ±0.0278	0.828 ±0.0255
Q1	0.832 ±0.0435	0.821 ±0.0200
Q2	0.914 ±0.0294	0.911 ±0.0180
Q2	0.926 ±0.0224	0.936 ±0.0158
Q3	0.969 ±0.0174	0.963 ±0.0121
Q3	0.951 ±0.0224	0.953 ±0.0133
Q4	0.951 ±0.0189	0.952 ±0.0134

Table 5.2: Fowlkes Mallows scores for the test dataset on all neighboring qubit couples with $|0\rangle$ and $|1\rangle$ state schedules for single and $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$ state schedules for both using the classical KMeans approach.

5.2 Classical Machine Learning

When using the KMeans clustering algorithm on the $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$ state schedules for all qubit couplings, we obtained test scores for qubits when its qubit couple is in the ground and excited state, respectively. Tables 5.1 and 5.2 show these scores using assignment fidelity and the Fowlkes Mallows score means with confidence intervals. Both scores for inidividual qubits without effects from neighboring qubits are shown in the first column of tables 5.1 and 5.2.

	Single	Both
Q0	0.790 ±0.1489	0.708 ±0.1858
Q1	0.714 ±0.1708	0.751 ±0.0074
Q1	0.793 ±0.1778	0.749 ±0.1656
Q2	0.741 ±0.2186	0.754 ±0.0055
Q2	0.765 ±0.2120	0.781 ±0.2216
Q3	0.795 ±0.2232	0.752 ±0.0064
Q3	0.759 ±0.2200	0.856 ±0.1955
Q4	0.722 ±0.1919	0.751 ±0.0046

Table 5.3: Assignment fidelity scores for the test dataset on all neighboring qubit couples with $|0\rangle$ and $|1\rangle$ state schedules for single and $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$ state schedules for both using the quantum KMeans approach.

	Single	Both
Q0	0.734 ±0.1624	0.790 ±0.1438
Q1	0.743 ±0.0412	0.740 ±0.0893
Q1	0.756 ±0.0211	0.753 ±0.0263
Q2	0.906 ±0.0029	0.905 ±0.0090
Q2	0.907 ±0.0048	0.900 ±0.0939
Q3	0.915 ±0.0036	0.951 ±0.0093
Q3	0.937 ±0.0098	0.837 ±0.1470
Q4	0.896 ±0.1075	0.889 ±0.1348

Table 5.4: Fowlkes Mallows scores for the test dataset on all neighboring qubit couples with $|0\rangle$ and $|1\rangle$ state schedules for single and $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$ state schedules for both using the quantum KMeans approach.

	(0, 1)	(1, 2)	(2, 3)	(3, 4)
$r_0(ES_{i,X}, GS_{i,Y})$	-0.0236	0.0070	-0.0265	-0.0058
$r_0(ES_{i,Y}, GS_{i,X})$	-0.0161	0.0063	-0.0057	0.0042
$r_1(ES_{i,X}, GS_{i,Y})$	0.0017	0.0011	0.1814	-0.0434
$r_1(ES_{i,Y}, GS_{i,X})$	-0.0211	0.0214	0.2147	0.0166
$r_0(ES_{i+1,X}, GS_{i+1,Y})$	0.0311	0.0869	0.0224	0.0210
$r_0(ES_{i+1,Y}, GS_{i+1,X})$	-0.0394	0.1927	0.0369	-0.0102
$r_1(ES_{i+1,X}, GS_{i+1,Y})$	0.02562	0.1923	0.0227	-0.0086
$r_1(ES_{i+1,Y}, GS_{i+1,X})$	-0.0012	0.1111	-0.0111	0.0033

Table 5.5: Pearson correlation coefficients in the form $r_j(ES_{i,X}, GS_{i,Y})$ where $X, Y \in \{I, Q\}$ and qubit i is in state $j \in \{0, 1\}$ when the other qubit is in the excited state and ground state and i is the first qubit of the couple, for each qubit coupling on the device.

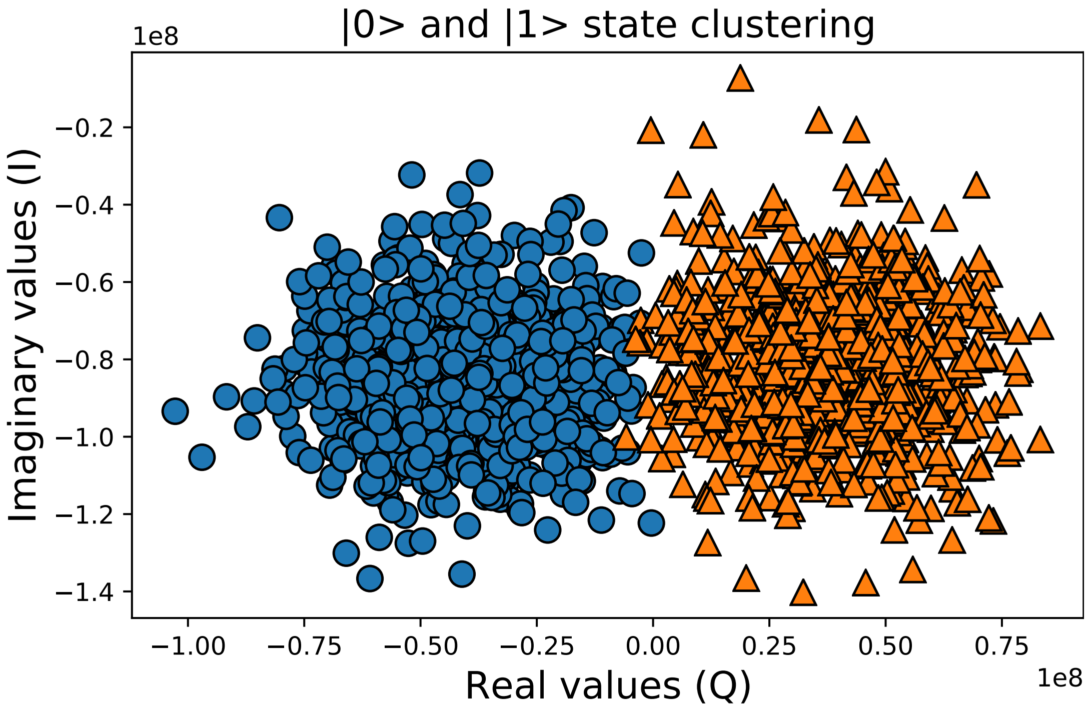


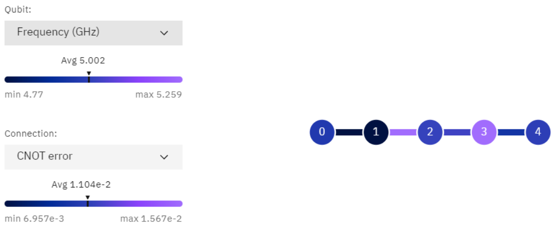
Figure 5.3: Result of running the QKMeans clustering algorithm on the ibmq_qasm simulator for signal data retrieved from qubit 0 of ibmq_bogota when applying schedules $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$.

5.3 Quantum Machine Learning

Results for the QML approach are obtained in a similar manner as the classical ML results but with the QKMeans as the selected clustering algorithm. Results for neighboring qubit couples are shown in tables 5.3 and 5.4 while results for inidividual qubits are shown in the first column of tables 5.3 and 5.4. Clustering of signal data on qubit 0 is shown in Figure 5.3.

5.4 Pearson correlation coefficients

With the IQ signal data retrieved on all four schedule variations, we extract arrays of each feature for a total of 8 arrays. These arrays are used as input to analyze the Pearson correlation coefficients between each pair of arrays. This procedure is

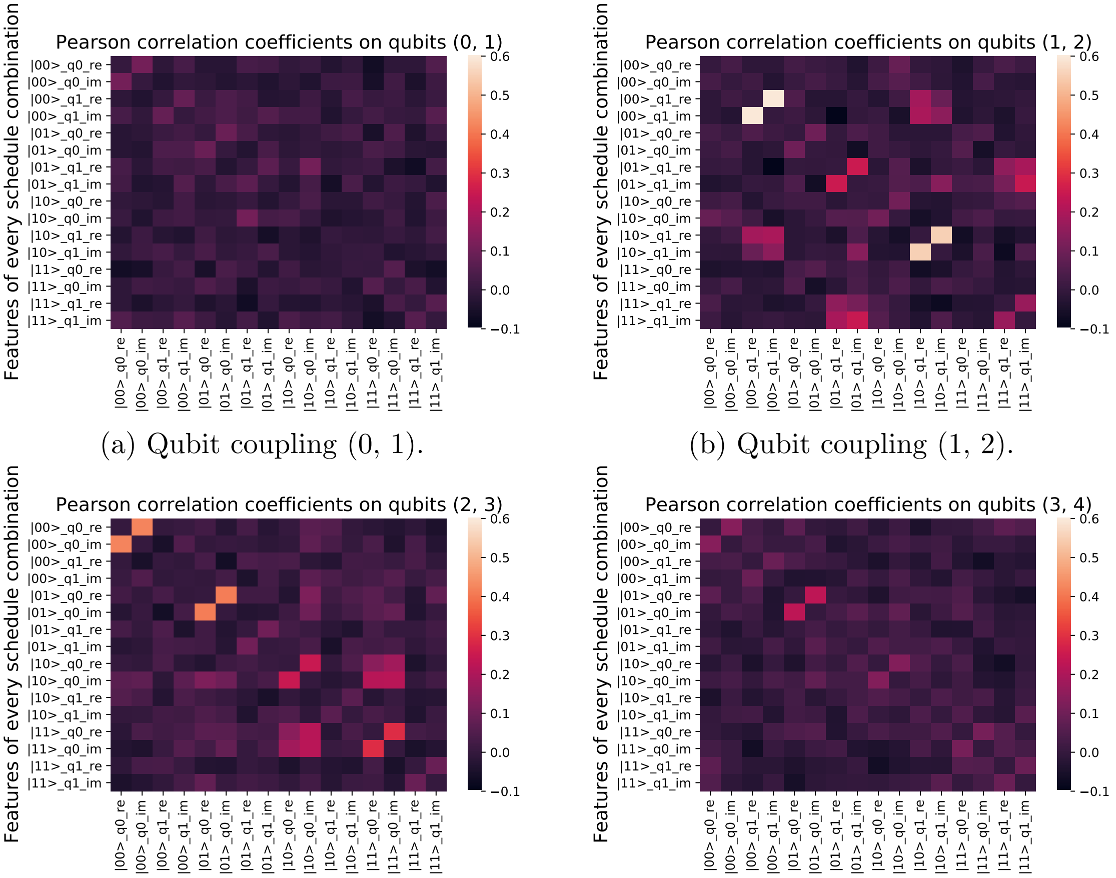

Qubit: Frequency (GHz)

Connection: CNOT error

Figure 5.4: The qubit coupling map of the ibmq_bogota device at the time of signal data retrieval. The map also shows the frequency values of each qubit and the CNOT errors between each pair of connected qubits,

implemented for each qubit coupling on the ibmq_bogota device with the mapping shown in Figure 5.4. A heatmap that describes the full spectrum of correlations between neighboring qubits is illustrated in Figure 5.5. The Pearson correlation coefficients of interest in the form $r_j(ES_{i,X}, GS_{i,Y})$ for each qubit coupling are shown in table 5.5. The further the coefficients are from zero, the higher the correlation.

Similar test and training scores were observed when using the KMeans and the QKMeans algorithms on single qubits when no other qubit was in the excited state. The assignment fidelity of the QKMeans was slightly better than that of the KMeans by 0.2%, while the KMeans held better Fowlkes Mallows scores by 5.6%. This result describes how well both algorithms cluster the presented data, as only a small difference in assignment fidelity and Fowlkes Mallows scores can be found. When observing the Pearson correlation heatmaps, the most significant correlation can be seen between qubits 1 and 2 as off-diagonal coefficients are the highest, followed by qubit couples (2, 3), (3, 4), and (0, 1) in descending order, with qubit couple (0, 1) presenting the lowest scale in Pearson correlation coefficients. This is also verified with the specific correlations in the form $r_j(ES_{i,X}, GS_{i,Y})$, where weak correlation can be found in qubit couples (1, 2) and (2, 3). We use the classical KMeans Fowlkes Mallows score to perform the correlation analysis due to it presenting the most stable results out of both clustering algorithm and scores. Weak correlation in



(a) Qubit coupling (0, 1).

(b) Qubit coupling (1, 2).

(c) Qubit coupling (2, 3).

(d) Qubit coupling (3, 4).

Figure 5.5: Heatmaps showing the Pearson correlation coefficients between each array belonging to the signal data of the four prepared schedules acting on each qubit coupling in the ibmq_bogota device. The axes labels follow the following format for the signal data received: {state}_{couple qubit}_{real or imaginary part of the complex number}.

qubit couples (1, 2) and (2, 3) is corroborated with the ML scores for cases "single" and "both", showing scores that are further apart than the same cases on the rest of the qubit couplings. The greatest difference between scores on "single" and "both" cases on qubit couples (1, 2) and (2, 3) are 11% on qubit 1 and 10% on qubit 2, respectively.

5.5 Leakage analysis results

When applying the leakage analysis strategy explained in section 4.4 with pulse data retrieved from the ibmq_bogota device, we find that most qubit couple results we perform clustering with three clusters result in an erroneous partitioning of the data points. Visually, we determine the appearance of leakage on IQ data points when a third cluster like the one in Figure 3.3 or similar can be identified. That figure was the result of an experiment to find leakage with an uncalibrated X gate on the ibmq_armonk device, where we can find a clear and great amount of leakage. Out of all qubit couples, couples (1, 2) and (2, 3) show a slight leakage on the IQ data plane where after applying the KMeans algorithm, the results of clustering can be shown in Figure 5.6 and Figure 5.7 while the rest show no leakage, as expected because of the almost non-existent correlation on the other qubit couples. Despite visual evidence of leakage appearing, errors in the clustering process can also be observed with qubit couple (2, 3) as almost $\frac{1}{4}$ of the data is clustered in the $|2\rangle$ state with some data points belonging to the $|0\rangle$ and $|1\rangle$ state being classified as leaked points. With our criteria of taking into account the proportion of possible data points corresponding to the $|2\rangle$ state to determine leakage, we confirm that while leakage does exist, the clustering process was faulty and leakage was not completely identified in that case. This may be due to the amount of real leaked data points being too small,

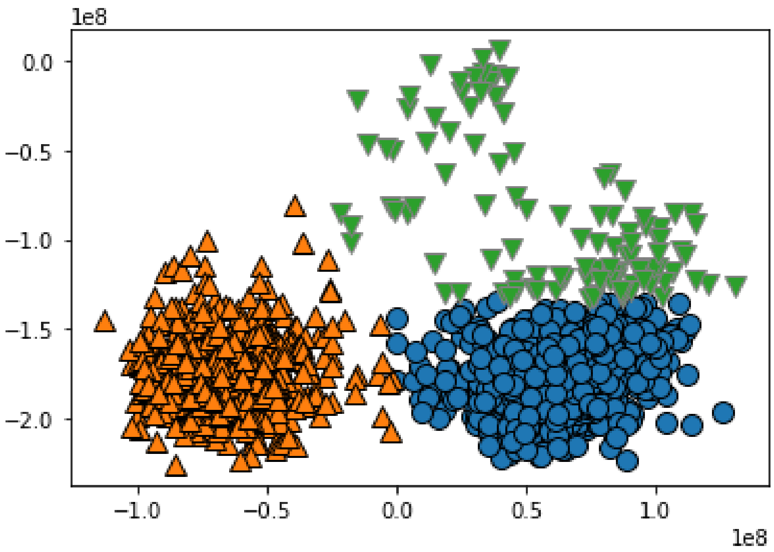


Figure 5.6: Application of the KMeans algorithm to cluster three clusters corresponding to states $|0\rangle$, $|1\rangle$ and $|2\rangle$ on qubit couple (1, 2).

potentially causing a class imbalance if clustered correctly. However, qubit couple (1, 2) where we noticed the most amount of correlation provided us with IQ data points that were compliant with both visual and numerical strategies we use to determine leakage. This applied to both single and both state combinations. The amount of leaked data points was ≈ 4.15%, giving a small enough proportion to identify the third cluster as a leaked state. After applying the error mitigation scheme for reducing leakage effects, we find a new assignment fidelity of 0.988 ± 0.0007 on test data compared to the previous assignment fidelity of 0.978 ± 0.0005 for the single case, with corresponding values 0.987 ± 0.0003 and 0.980 ± 0.0079 for the both case, having an improvement of ≈ 1% and ≈ 0.7%, respectively.

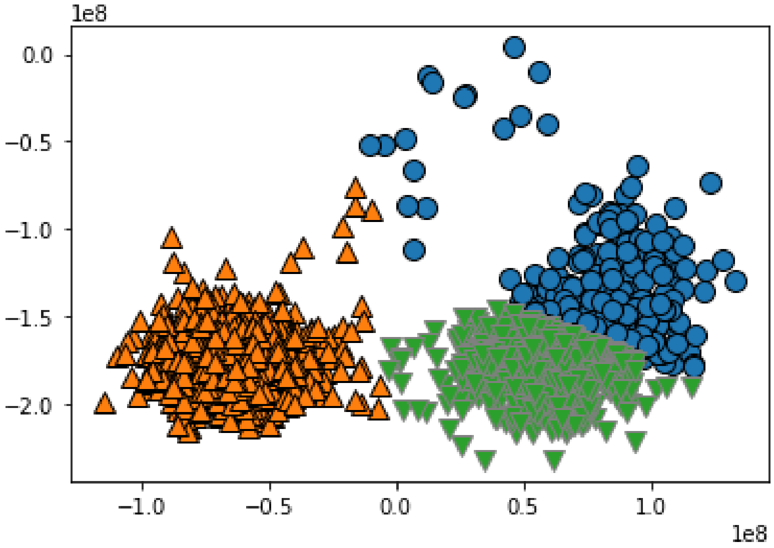


Figure 5.7: Application of the KMeans algorithm to cluster three clusters corresponding to states $|0\rangle$, $|1\rangle$ and $|2\rangle$ on qubit couple (2, 3),

CHAPTER 6

CONCLUSIONS

We observed that the closed-loop optimization algorithm managed to efficiently optimize a gate on the criteria of process fidelity using the QPT benchmark, while the RB gate fidelity proved a decent rate using interleaved RB, although lower than the default pulses' on both occasions. We can conclude that the different metrics taken into account by the benchmarks may differ very greatly when the process fidelity is low enough, rendering those results useless. Finally, closed-loop optimization algorithms enable the task of quickly optimizing gates up to a slightly lower process fidelity than default pulses.

We implemented a Quantum KMeans variant with the Qiskit library compatible with functions from the scikit-learn library. The training and testing scores of this algorithm were at par with the KMeans algorithm except when clusters weren't visually separable, in which case only a small difference in scores was observed. We demonstrated the use of a cross-talk pulse-level benchmarking scheme on IBM's ibmq_bogota quantum computer through the use of ML, QML and Pearson correlation coefficients, finding enough evidence to determine weak correlation in qubit couples (1, 2) and (2, 3). After analyzing the training scores, we also conclude that the 1 qubit has the worst performance at readout evidenced by the signal data not being visually separable and the low scores obtained on both clustering algorithms compared to the other qubits. Its poor performance is further verified by the calibration data showing a high readout error of 8,4%. Training times for QKMeans were similar to the training times on a KMeans implementation that was not optimized, whereas the KMeans algorithm available on the scikit-learn library was faster. Future work for this project is an optimization of the QKMeans algorithm for faster training times given the current constraints of universal quantum computers like the one used for this analysis. Despite there being the possibility to batch several circuits at a single time on a quantum computer for the QKMeans, the main difference in speed the scikit-learn implementation of KMeans has against the QKMeans comes from the replacement of one-on-one distance measurements for distance measurements using a small amount of dot products, greatly reducing its training time on a classical computer.

Also, we performed a leakage benchmark and error mitigation using the KMeans algorithm for two and three clusters on ibmq_bogota for the same pulse data as the correlation analysis and found evidence of leakage in qubit couples (1, 2) and (2, 3) with a correct leakage estimate of $\approx 4.15\%$ on qubit couple (1, 2) through a visual and numerical leakage identification strategy, while the rest of the qubits had no observable leakage. We mitigated leakage on qubit couple (1, 2) to improve the assignment fidelity from 0.978 ± 0.0005 to 0.988 ± 0.0007 on the single case and 0.980 ± 0.0079 to 0.987 ± 0.0003 on the both case using cross-validation, showing an improvement of $\approx 1\%$ and $\approx 0.7\%$, respectively. We therefore confirm the practical use of a clustering ML algorithm for discriminating quantum states at readout by providing an experimental process to suppress errors caused by leakage into higher quantum energy states, which works at the small cost of removing data points belonging to the $|2\rangle$ state as outlier data points.

Future work for this research is performing a full crosstalk, leakage, ML, QML, QPT and RB analysis based on pulses optimized using the closed-loop optimization algorithm, and compare a wider range of metrics to default pulses offered by IBM quantum devices. Specifically, the application of different ML and QML algorithms on readout using the leakage error mitigation scheme and the use of different pulse-level benchmarks, along with applications of optics related algorithms for error mitigation. Future work on other fields is the application of the closed-loop optimization algorithm on nuclear magnetic resonance machines and the use of QK-means for demodulation on optic fiber.

BIBLIOGRAPHY

[Aar15] Scott Aaronson. Read the fine print. *Nature Physics*, 11(4):291–293, Apr 2015.

[ACLS$^+$17] D. P. L. Aude Craik, N. M. Linke, M. A. Sepiol, T. P. Harty, J. F. Goodwin, C. J. Ballance, D. N. Stacey, A. M. Steane, D. M. Lucas, and D. T. C. Allcock. High-fidelity spatial and polarization addressing of ca+43 qubits using near-field microwave control. *Physical Review A*, 95(2), Feb 2017.

[AD20] Davis Arthur and Prasanna Date. Balanced k-means clustering on an adiabatic quantum computer, 2020.

[AH15] Steven H. Adachi and Maxwell P. Henderson. Application of quantum annealing to training of deep neural networks, 2015.

[AKE$^+$20] Thomas Alexander, Naoki Kanazawa, Daniel J Egger, Lauren Capelluto, Christopher J Wood, Ali Javadi-Abhari, and David C McKay. Qiskit pulse: programming quantum computers through the cloud with pulses. *Quantum Science and Technology*, 5(4):044006, Aug 2020.

[AMMR13] M. Amy, D. Maslov, M. Mosca, and M. Roetteler. A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 32(6):818–830, Jun 2013.

[AV07] David Arthur and Sergei Vassilvitskii. K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, page 1027–1035, USA, 2007. Society for Industrial and Applied Mathematics.

[BCR10] Constantin Brif, Raj Chakrabarti, and Herschel Rabitz. Control of quantum phenomena: past, present and future. *New Journal of Physics*, 12(7):075008, Jul 2010.

[BEH$^+$04] N. Boulant, J. Emerson, T. F. Havel, D. G. Cory, and S. Furuta. Incoherent noise and quantum information processing. *The Journal of Chemical Physics*, 121(7):2955–2961, Aug 2004.

[BEKP18] Sergey Bravyi, Matthias Englbrecht, Robert König, and Nolan Peard. Correcting coherent errors with surface codes. *npj Quantum Information*, 4(1), Oct 2018.

[BFdS20] M. Belli, M. Fanciulli, and R. de Sousa. Probing two-level systems with electron spin inversion recovery of defects at the si/sio2 interface. *Physical Review Research*, 2(3), Sep 2020.

[BHL$^+$16] C. J. Ballance, T. P. Harty, N. M. Linke, M. A. Sepiol, and D. M. Lucas. High-fidelity quantum logic gates using trapped-ion hyperfine qubits. *Phys. Rev. Lett.*, 117:060504, Aug 2016.

[BHMT02] Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Quantum Computation and Information*, page 53–74, 2002.

[BHP80] Klaus Bock, Laurance D Hall, and Christian Pedersen. An experimental evaluation of nonselective proton spin–lattice relaxation rates: analyses of data for the eight isomeric 2, 3, 4-tri-o-acetyl-1, 6-anhydro-β-d-hexopyranoses. *Canadian Journal of Chemistry*, 58(18):1916–1922, 1980.

[BIS$^+$18] Sergio Boixo, Sergei V. Isakov, Vadim N. Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, Michael J. Bremner, John M. Martinis, and Hartmut Neven. Characterizing quantum supremacy in near-term devices. *Nature Physics*, 14(6):595–600, Jun 2018.

[BKM$^+$14] R. Barends, J. Kelly, A. Megrant, A. Veitia, D. Sank, E. Jeffrey, T. C. White, J. Mutus, A. G. Fowler, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, C. Neill, P. O'Malley, P. Roushan, A. Vainsencher, J. Wenner, A. N. Korotkov, A. N. Cleland, and John M. Martinis. Superconducting quantum circuits at the surface code threshold for fault tolerance. *Nature*, 508(7497):500–503, Apr 2014.

[Blu17] Robin Blume-Kohout. How distinguishable are two quantum processes? a.k.a. What is the error rate of a quantum gate?9D. In *APS March Meeting Abstracts*, volume 2017 of *APS Meeting Abstracts*, page B52.006, March 2017.

[BOS$^+$02] Mark D. Bowdrey, Daniel K.L. Oi, Anthony J. Short, Konrad Banaszek, and Jonathan A. Jones. Fidelity of single qubit maps. *Physics Letters A*, 294(5-6):258–260, 2002.

[BQP$^+$19] R. Barends, C.M. Quintana, A.G. Petukhov, Yu Chen, D. Kafri, K. Kechedzhi, R. Collins, O. Naaman, S. Boixo, F. Arute, and et al. Diabatic gates for frequency-tunable superconducting qubits. *Physical Review Letters*, 123(21), Nov 2019.

[Bra] Bradben. What are the q programming language & qdk? - azure quantum.

[BUV$^+$09] M.J. Biercuk, H. Uys, A.P. VanDevender, N. Shiga, W.M. Itano, and J.J. Bollinger. High-fidelity quantum control using ion crystals in a penning trap. *Quantum Information and Computation*, 9(11&12):920–949, 2009.

[BWC$^+$11] K. R. Brown, A. C. Wilson, Y. Colombe, C. Ospelkaus, A. M. Meier, E. Knill, D. Leibfried, and D. J. Wineland. Single-qubit-gate error below $\mathbf{10^{-4}}$ in a trapped ion. *Phys. Rev. A*, 84:030303, Sep 2011.

[BWG$^+$18] Stefanie J. Beale, Joel J. Wallman, Mauricio Gutiérrez, Kenneth R. Brown, and Raymond Laflamme. Quantum error correction decoheres noise. *Phys. Rev. Lett.*, 121:190501, Nov 2018.

[BWP$^+$17] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, Sep 2017.

[CB07] R. Cabrera and W.E. Baylis. Average fidelity in n-qubit systems. *Physics Letters A*, 368(1):25–28, 2007.

[CBSG17] Andrew W. Cross, Lev S. Bishop, John A. Smolin, and Jay M. Gambetta. Open quantum assembly language, 2017.

[CCL01] Andrew M. Childs, Isaac L. Chuang, and Debbie W. Leung. Realization of quantum process tomography in nmr. *Phys. Rev. A*, 64:012314, Jun 2001.

[CDG$^+$10] J. M. Chow, L. DiCarlo, J. M. Gambetta, F. Motzoi, L. Frunzio, S. M. Girvin, and R. J. Schoelkopf. Optimized driving of superconducting artificial atoms for improved single-qubit gates. *Physical Review A*, 82(4), Oct 2010.

[CGC$^+$13] A. D. Córcoles, Jay M. Gambetta, Jerry M. Chow, John A. Smolin, Matthew Ware, Joel Strand, B. L. T. Plourde, and M. Steffen. Process verification of two-qubit quantum gates by randomized benchmarking. *Physical Review A*, 87(3), Mar 2013.

[CGT+09] J. M. Chow, J. M. Gambetta, L. Tornberg, Jens Koch, Lev S. Bishop, A. A. Houck, B. R. Johnson, L. Frunzio, S. M. Girvin, and R. J. Schoelkopf. Randomized benchmarking and process tomography for gate errors in a solid-state qubit. *Phys. Rev. Lett.*, 102:090502, Mar 2009.

[Cho75] Man-Duen Choi. Completely positive linear maps on complex matrices. *Linear Algebra and its Applications*, 10(3):285–290, 1975.

[CKQ+16] Zijun Chen, Julian Kelly, Chris Quintana, R. Barends, B. Campbell, Yu Chen, B. Chiaro, A. Dunsworth, A. G. Fowler, E. Lucero, E. Jeffrey, A. Megrant, J. Mutus, M. Neeley, C. Neill, P. J. J. O'Malley, P. Roushan, D. Sank, A. Vainsencher, J. Wenner, T. C. White, A. N. Korotkov, and John M. Martinis. Measuring and suppressing quantum state leakage in a superconducting qubit. *Phys. Rev. Lett.*, 116:020501, Jan 2016.

[CN97] Isaac L. Chuang and M. A. Nielsen. Prescription for experimental determination of the dynamics of a quantum black box. *Journal of Modern Optics*, 44(11-12):2455–2467, Nov 1997.

[CPF+10] Marcus Cramer, Martin B. Plenio, Steven T. Flammia, Rolando Somma, David Gross, Stephen D. Bartlett, Olivier Landon-Cardinal, David Poulin, and Yi-Kai Liu. Efficient quantum state tomography. *Nature Communications*, 1(1), Dec 2010.

[CSA+21] Zijun Chen, Kevin J. Satzinger, Juan Atalaya, Alexander N. Korotkov, Andrew Dunsworth, Daniel Sank, Chris Quintana, Matt McEwen, Rami Barends, Paul V. Klimov, Sabrina Hong, Cody Jones, Andre Petukhov, Dvir Kafri, Sean Demura, Brian Burkett, Craig Gidney, Austin G. Fowler, Harald Putterman, Igor Aleiner, Frank Arute, Kunal Arya, Ryan Babbush, Joseph C. Bardin, Andreas Bengtsson, Alexandre Bourassa, Michael Broughton, Bob B. Buckley, David A. Buell, Nicholas Bushnell, Benjamin Chiaro, Roberto Collins, William Courtney, Alan R. Derk, Daniel Eppens, Catherine Erickson, Edward Farhi, Brooks Foxen, Marissa Giustina, Jonathan A. Gross, Matthew P. Harrigan, Sean D. Harrington, Jeremy Hilton, Alan Ho, Trent Huang, William J. Huggins, L. B. Ioffe, Sergei V. Isakov, Evan Jeffrey, Zhang

Jiang, Kostyantyn Kechedzhi, Seon Kim, Fedor Kostritsa, David Landhuis, Pavel Laptev, Erik Lucero, Orion Martin, Jarrod R. McClean, Trevor McCourt, Xiao Mi, Kevin C. Miao, Masoud Mohseni, Wojciech Mruczkiewicz, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Michael Newman, Murphy Yuezhen Niu, Thomas E. O'Brien, Alex Opremcak, Eric Ostby, Bálint Pató, Nicholas Redd, Pedram Roushan, Nicholas C. Rubin, Vladimir Shvarts, Doug Strain, Marco Szalay, Matthew D. Trevithick, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, Sergio Boixo, Vadim Smelyanskiy, Yu Chen, Anthony Megrant, and Julian Kelly. Exponential suppression of bit or phase flip errors with repetitive error correction, 2021.

[CVC$^+$13] Josephine B. Chang, Michael R. Vissers, Antonio D. Córcoles, Martin Sandberg, Jiansong Gao, David W. Abraham, Jerry M. Chow, Jay M. Gambetta, Mary Beth Rothwell, George A. Keefe, Matthias Steffen, and David P. Pappas. Improved superconducting qubit coherence using titanium nitride. *Applied Physics Letters*, 103(1):012602, 2013.

[DAPN21] Prasanna Date, Davis Arthur, and Lauren Pusey-Nazzaro. Qubo formulations for training machine learning models. *Scientific Reports*, 11(1):10029, May 2021.

[Dev] The Cirq Developers. Cirq.

[DGL$^+$20] Yongshan Ding, Pranav Gokhale, Sophia Fuhui Lin, Richard Rines, Thomas Propson, and Frederic T. Chong. Systematic crosstalk mitigation for superconducting qubits via frequency-aware compilation. *2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, Oct 2020.

[DTB16] Vedran Dunjko, Jacob M. Taylor, and Hans J. Briegel. Quantum-enhanced machine learning. *Phys. Rev. Lett.*, 117:130501, Sep 2016.

[DWM95] T. J. Dunn, I. A. Walmsley, and S. Mukamel. Experimental determination of the quantum-mechanical state of a molecular vibrational mode using fluorescence tomography. *Phys. Rev. Lett.*, 74:884–887, Feb 1995.

[EWP$^+$19] Alexander Erhard, Joel J. Wallman, Lukas Postler, Michael Meth, Roman Stricker, Esteban A. Martinez, Philipp Schindler, Thomas Monz, Joseph Emerson, and Rainer Blatt. Characterizing large-scale quantum computers via cycle benchmarking. *Nature Communications*, 10(1), Nov 2019.

[FL11] Steven T. Flammia and Yi-Kai Liu. Direct fidelity estimation from few pauli measurements. *Phys. Rev. Lett.*, 106:230501, Jun 2011.

[Fow13] Austin G. Fowler. Coping with qubit leakage in topological codes. *Physical Review A*, 88(4), Oct 2013.

[FvdG99] C.A. Fuchs and J. van de Graaf. Cryptographic distinguishability measures for quantum-mechanical states. *IEEE Transactions on Information Theory*, 45(4):1216–1227, May 1999.

[FWB$^+$16] Guanru Feng, Joel J. Wallman, Brandon Buonacorsi, Franklin H. Cho, Daniel K. Park, Tao Xin, Dawei Lu, Jonathan Baugh, and Raymond Laflamme. Estimating the coherence of noise in quantum control of a solid-state qubit. *Physical Review Letters*, 117(26), Dec 2016.

[GCM$^+$12] Jay M. Gambetta, A. D. Córcoles, S. T. Merkel, B. R. Johnson, John A. Smolin, Jerry M. Chow, Colm A. Ryan, Chad Rigetti, S. Poletto, Thomas A. Ohki, Mark B. Ketchen, and M. Steffen. Characterization of addressability by simultaneous randomized benchmarking. *Phys. Rev. Lett.*, 109:240504, Dec 2012.

[GD17] Daniel Greenbaum and Zachary Dutton. Modeling coherent errors in quantum error correction. *Quantum Science and Technology*, 3(1):015007, Dec 2017.

[GFMG13] Joydip Ghosh, Austin G. Fowler, John M. Martinis, and Michael R. Geller. Understanding the effects of leakage in superconducting quantum-error-detection circuits. *Physical Review A*, 88(6), Dec 2013.

[GMT$^+$12] J. P. Gaebler, A. M. Meier, T. R. Tan, R. Bowler, Y. Lin, D. Hanneke, J. D. Jost, J. P. Home, E. Knill, D. Leibfried, and D. J. Wineland. Randomized benchmarking of multiqubit gates. *Phys. Rev. Lett.*, 108:260503, Jun 2012.

[Got98a] D Gottesman. The heisenberg representation of quantum computers. 6 1998.

[Got98b] Daniel Gottesman. Theory of fault-tolerant quantum computation. *Physical Review A*, 57(1):127–137, Jan 1998.

[GPD20a] J. W. O. Garmon, R. C. Pooser, and E. F. Dumitrescu. Benchmarking noise extrapolation with the openpulse control framework. *Phys. Rev. A*, 101:042308, Apr 2020.

[GPD20b] J. W. O. Garmon, Raphael Pooser, and Eugene Dumitrescu. Benchmarking noise extrapolation with the openpulse control framework. *Physical Review A*, 101(4), 4 2020.

[GS20] Michael R. Geller and Mingyu Sun. Efficient correction of multiqubit measurement errors, 2020.

[GTL+16] J. P. Gaebler, T. R. Tan, Y. Lin, Y. Wan, R. Bowler, A. C. Keith, S. Glancy, K. Coakley, E. Knill, D. Leibfried, and D. J. Wineland. High-fidelity universal gate set for $^9\mathrm{Be}^+$ ion qubits. *Phys. Rev. Lett.*, 117:060505, Aug 2016.

[HDSS02] XUEDONG HU, ROGERIO DE SOUSA, and S. DAS SARMA. Decoherence and dephasing in spin-based solid state quantum computers. *Foundations of Quantum Mechanics in the Light of New Technology*, Oct 2002.

[HHL09] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009.

[HHR+05] H. Häffner, W. Hänsel, C. F. Roos, J. Benhelm, D. Chek-al kar, M. Chwalla, T. Körber, U. D. Rapol, M. Riebe, P. O. Schmidt, and et al. Scalable multiparticle entanglement of trapped ions. *Nature*, 438(7068):643–646, Dec 2005.

[HP19] Kathleen E. Hamilton and Raphael C. Pooser. Error-mitigated data-driven circuit learning on noisy quantum hardware, 2019.

[HST+20] A.C. Hughes, V.M. Schäfer, K. Thirumalai, D.P. Nadlinger, S.R. Woodrow, D.M. Lucas, and C.J. Ballance. Benchmarking a high-fidelity mixed-species entangling gate. *Physical Review Letters*, 125(8), Aug 2020.

[JKMW01] Daniel F. V. James, Paul G. Kwiat, William J. Munro, and Andrew G. White. Measurement of qubits. *Phys. Rev. A*, 64:052312, Oct 2001.

[KIQ$^+$19] Nathan Killoran, Josh Izaac, Nicolás Quesada, Ville Bergholm, Matthew Amy, and Christian Weedbrook. Strawberry fields: A software platform for photonic quantum computing. *Quantum*, 3:129, Mar 2019.

[KKY$^+$19] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver. A quantum engineer's guide to superconducting qubits. *Applied Physics Reviews*, 6(2):021318, Jun 2019.

[KLDF16] Richard Kueng, David M. Long, Andrew C. Doherty, and Steven T. Flammia. Comparing experiments to the fault-tolerance threshold. *Phys. Rev. Lett.*, 117:170502, Oct 2016.

[KLR$^+$08] E. Knill, D. Leibfried, R. Reichle, J. Britton, R. B. Blakestad, J. D. Jost, C. Langer, R. Ozeri, S. Seidelin, and D. J. Wineland. Randomized benchmarking of quantum gates. *Physical Review A*, 77(1), Jan 2008.

[KLR$^+$20] S. Krinner, S. Lazar, A. Remm, C.K. Andersen, N. Lacroix, G.J. Norris, C. Hellings, M. Gabureac, C. Eichler, and A. Wallraff. Benchmarking coherent errors in controlled-phase gates due to spectator qubits. *Physical Review Applied*, 14(2), Aug 2020.

[KRK$^+$05] Navin Khaneja, Timo Reiss, Cindie Kehlet, Thomas Schulte-Herbrüggen, and Steffen J. Glaser. Optimal control of coupled spin dynamics: design of nmr pulse sequences by gradient ascent algorithms. *Journal of Magnetic Resonance*, 172(2):296–305, 2005.

[KS14] Robert Koenig and John A. Smolin. How to efficiently select an arbitrary clifford group element. *Journal of Mathematical Physics*, 55(12):122202, Dec 2014.

[KSB$^+$20] Morten Kjaergaard, Mollie E. Schwartz, Jochen Braumüller, Philip Krantz, Joel I.-J. Wang, Simon Gustavsson, and William D. Oliver. Superconducting qubits: Current state of play. *Annual Review of Condensed Matter Physics*, 11(1):369–395, Mar 2020.

[KYG$^+$07] Jens Koch, Terri M. Yu, Jay Gambetta, A. A. Houck, D. I. Schuster, J. Majer, Alexandre Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf. Charge-insensitive qubit design derived from the cooper pair box. *Phys. Rev. A*, 76:042319, Oct 2007.

[LaR19] Ryan LaRose. Overview and comparison of gate level quantum software platforms. *Quantum*, 3:130, Mar 2019.

[lid13] *Quantum Error Correction.* Cambridge University Press, 2013.

[LKS$^+$05] D. Leibfried, E. Knill, S. Seidelin, J. Britton, R. B. Blakestad, J. Chiaverini, D. B. Hume, W. M. Itano, J. D. Jost, C. Langer, R. Ozeri, R. Reichle, and D. J. Wineland. Creation of a six-atom 'schrödinger cat' state. *Nature*, 438(7068):639–642, Dec 2005.

[LMR14] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. Quantum principal component analysis. *Nature Physics*, 10(9):631–633, Sep 2014.

[LR09] A. I. Lvovsky and M. G. Raymer. Continuous-variable optical quantum-state tomography. *Rev. Mod. Phys.*, 81:299–332, Mar 2009.

[LYC14] Guang Hao Low, Theodore J. Yoder, and Isaac L. Chuang. Quantum inference on bayesian networks. *Phys. Rev. A*, 89:062315, Jun 2014.

[MAB$^+$18] David C. McKay, Thomas Alexander, Luciano Bello, Michael J. Biercuk, Lev Bishop, Jiayin Chen, Jerry M. Chow, Antonio D. Córcoles, Daniel Egger, Stefan Filipp, Juan Gomez, Michael Hush, Ali Javadi-Abhari, Diego Moreda, Paul Nation, Brent Paulovicks, Erick Winston, Christopher J. Wood, James Wootton, and Jay M. Gambetta. Qiskit backend specifications for openqasm and openpulse experiments, 2018.

[McC] Alex McCaskey. Xacc documentation.

[MFM$^+$16] David C. McKay, Stefan Filipp, Antonio Mezzacapo, Easwar Magesan, Jerry M. Chow, and Jay M. Gambetta. Universal gate for fixed-frequency qubits via a tunable bus. *Phys. Rev. Applied*, 6:064007, Dec 2016.

[MGCC15] Easwar Magesan, Jay M. Gambetta, A.D. C'orcoles, and Jerry M. Chow. Machine learning for discriminating quantum measurement trajectories and improving readout. *Physical Review Letters*, 114(20), May 2015.

[MGE11] Easwar Magesan, J. M. Gambetta, and Joseph Emerson. Scalable and robust randomized benchmarking of quantum processes. *Physical Review Letters*, 106(18), May 2011.

[MGE12] Easwar Magesan, Jay M. Gambetta, and Joseph Emerson. Characterizing quantum gates via randomized benchmarking. *Physical Review A*, 85(4), Apr 2012.

[MGJ$^+$12] Easwar Magesan, Jay M. Gambetta, B. R. Johnson, Colm A. Ryan, Jerry M. Chow, Seth T. Merkel, Marcus P. da Silva, George A. Keefe, Mary B. Rothwell, Thomas A. Ohki, Mark B. Ketchen, and M. Steffen. Efficient measurement of quantum gate error by interleaved randomized benchmarking. *Phys. Rev. Lett.*, 109:080505, Aug 2012.

[MNW$^+$17] Kristel Michielsen, Madita Nocon, Dennis Willsch, Fengping Jin, Thomas Lippert, and Hans De Raedt. Benchmarking gate-based quantum computers. *Computer Physics Communications*, 220:44–55, 2017.

[MPGC13] Easwar Magesan, Daniel Puzzuoli, Christopher E. Granade, and David G. Cory. Modeling quantum noise for efficient testing of fault-tolerant circuits. *Phys. Rev. A*, 87:012324, Jan 2013.

[MSS$^+$19] David C. McKay, Sarah Sheldon, John A. Smolin, Jerry M. Chow, and Jay M. Gambetta. Three-qubit randomized benchmarking. *Physical Review Letters*, 122(20), May 2019.

[MW13] F. Motzoi and F. K. Wilhelm. Improving frequency selection of driven pulses using derivative-based transition suppression. *Physical Review A*, 88(6), Dec 2013.

[MWS$^+$17] David C. McKay, Christopher J. Wood, Sarah Sheldon, Jerry M. Chow, and Jay M. Gambetta. Efficient z gates for quantum computing. *Phys. Rev. A*, 96:022330, Aug 2017.

[MZHH19] Pranav Mundada, Gengyan Zhang, Thomas Hazard, and Andrew Houck. Suppression of qubit crosstalk in a tunable coupling superconducting circuit. *Phys. Rev. Applied*, 12:054023, Nov 2019.

[NAB$^+$09] Matthew Neeley, Markus Ansmann, Radoslaw C. Bialczak, Max Hofheinz, Erik Lucero, Aaron D. O'Connell, Daniel Sank, Haohua Wang, James Wenner, Andrew N. Cleland, Michael R. Geller, and John M. Martinis. Emulation of a quantum spin with a superconducting phase qudit. *Science*, 325(5941):722–725, 2009.

[NC11] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, USA, 10th edition, 2011.

[NMM$^+$14] D. Nigg, M. Muller, E. A. Martinez, P. Schindler, M. Hennrich, T. Monz, M. A. Martin-Delgado, and R. Blatt. Quantum computations on a topologically encoded qubit. *Science*, 345(6194):302–305, Jun 2014.

[NRK$^+$18a] C. Neill, P. Roushan, K. Kechedzhi, S. Boixo, S. V. Isakov, V. Smelyanskiy, A. Megrant, B. Chiaro, A. Dunsworth, K. Arya, R. Barends, B. Burkett, Y. Chen, Z. Chen, A. Fowler, B. Foxen, M. Giustina, R. Graff, E. Jeffrey, T. Huang, J. Kelly, P. Klimov, E. Lucero, J. Mutus, M. Neeley, C. Quintana, D. Sank, A. Vainsencher, J. Wenner, T. C. White, H. Neven, and J. M. Martinis. A blueprint for demonstrating quantum supremacy with superconducting qubits. *Science*, 360(6385):195–199, 2018.

[NRK$^+$18b] C. Neill, P. Roushan, K. Kechedzhi, S. Boixo, S. V. Isakov, V. Smelyanskiy, A. Megrant, B. Chiaro, A. Dunsworth, K. Arya, and et al. A blueprint for demonstrating quantum supremacy with superconducting qubits. *Science*, 360(6385):195–199, Apr 2018.

[OCNP10] S Olmschenk, R Chicireanu, K D Nelson, and J V Porto. Randomized benchmarking of atomic qubits in an optical lattice. *New Journal of Physics*, 12(11):113007, nov 2010.

[OLA$^+$08] C. Ospelkaus, C. E. Langer, J. M. Amini, K. R. Brown, D. Leibfried, and D. J. Wineland. Trapped-ion quantum logic gates based on oscillating magnetic fields. *Phys. Rev. Lett.*, 101:090502, Aug 2008.

[PBE$^+$03] Marco A. Pravia, Nicolas Boulant, Joseph Emerson, Amro Farid, Evan M. Fortunato, Timothy F. Havel, R. Martinez, and David G. Cory. Robust control of quantum information. *The Journal of Chemical Physics*, 119(19):9993–10001, 2003.

[PCZ97] J. F. Poyatos, J. I. Cirac, and P. Zoller. Complete characterization of a quantum process: The two-bit quantum gate. *Physical Review Letters*, 78(2):390–393, Jan 1997.

[PGH$^+$14] Daniel Puzzuoli, Christopher Granade, Holger Haas, Ben Criger, Easwar Magesan, and D. G. Cory. Tractable simulation of error correction with honest approximations to realistic fault models. *Phys. Rev. A*, 89:022306, Feb 2014.

[PMS$^+$14] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1):4213, Jul 2014.

[Pre18] John Preskill. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, August 2018.

[PvJ11] Matteo Paris and Řeháček Jaroslav. *Quantum state estimation.* Springer, 2011.

[QC] Q-CTRL. Automate closed-loop hardware optimization of quantum devices: User guides: Boulder opal: Q-ctrl documentation.

[qis] Open-source quantum development.

[RK21] Salonik Resch and Ulya R. Karpuzcu. Benchmarking quantum computers and the impact of quantum noise, 2021.

[RLL09] C A Ryan, M Laforest, and R Laflamme. Randomized benchmarking of single- and multi-qubit control in liquid-state NMR quantum information processing. *New Journal of Physics*, 11(1):013034, jan 2009.

[RML14] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum support vector machine for big data classification. *Phys. Rev. Lett.*, 113:130503, Sep 2014.

[Rus94] Mary Beth Ruskai. Beyond Strong Subadditivity? Improved Bounds on the Contraction of Generalized Relative Entropy. *Reviews in Mathematical Physics*, 6:1147–1161, January 1994.

[SBRF93] D. T. Smithey, M. Beck, M. G. Raymer, and A. Faridani. Measurement of the wigner distribution and the density matrix of a light mode using optical homodyne tomography: Application to squeezed states and the vacuum. *Phys. Rev. Lett.*, 70:1244–1247, Mar 1993.

[SEFT20] Yasunari Suzuki, Suguru Endo, Keisuke Fujii, and Yuuki Tokunaga. Quantum error mitigation for fault-tolerant quantum computing, 2020.

[Sho95] Peter W. Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A*, 52:R2493–R2496, Oct 1995.

[SSP14] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. An introduction to quantum machine learning. *Contemporary Physics*, 56(2):172–185, Oct 2014.

[STHL10] S. Shankar, A. M. Tyryshkin, Jianhua He, and S. A. Lyon. Spin relaxation and coherence times for electrons at thesi/sio2interface. *Physical Review B*, 82(19), Nov 2010.

[SWS15] Yuval R Sanders, Joel J Wallman, and Barry C Sanders. Bounding quantum gate error rate based on reported average fidelity. *New Journal of Physics*, 18(1):012002, dec 2015.

[SZ77] M Sass and D Ziessow. Error analysis for optimized inversion recovery spin-lattice relaxation measurements. *Journal of Magnetic Resonance (1969)*, 25(2):263–276, 1977.

[SZL$^+$19] Lian-He Shao, Yu-Ran Zhang, Yu Luo, Zhengjun Xi, and Shao-Ming Fei. Quantifying quantum non-markovianity based on quantum coherence via skew information. *Laser Physics Letters*, 17(1):015202, Dec 2019.

[TBG17] Kristan Temme, Sergey Bravyi, and Jay M. Gambetta. Error mitigation for short-depth quantum circuits. *Physical Review Letters*, 119(18), Nov 2017.

[Tea] The Qiskit Team.

[Tea20] The Qiskit Team, Dec 2020.

[Tea21a] The Qiskit Team, Jul 2021.

[Tea21b] The Qiskit Team. Measurement error mitigation, Feb 2021.

[Tol18] J Tolar. On clifford groups in quantum computing. *Journal of Physics: Conference Series*, 1071:012022, Aug 2018.

[VHY$^+$14] M. Veldhorst, J. C. C. Hwang, C. H. Yang, A. W. Leenstra, B. de Ronde, J. P. Dehollain, J. T. Muhonen, F. E. Hudson, K. M.

Itoh, A. Morello, and A. S. Dzurak. An addressable quantum dot qubit with fault-tolerant control-fidelity. *Nature Nanotechnology*, 9(12):981–985, Dec 2014.

[VR89] K. Vogel and H. Risken. Determination of quasiprobability distributions in terms of probability distributions for the rotated quadrature phase. *Phys. Rev. A*, 40:2847–2849, Sep 1989.

[WBD$^+$19] K. Wright, K. M. Beck, S. Debnath, J. M. Amini, Y. Nam, N. Grzesiak, J.-S. Chen, N. C. Pisenti, M. Chmielewski, C. Collins, and et al. Benchmarking an 11-qubit quantum computer. *Nature Communications*, 10(1), Nov 2019.

[WBE16] Joel J Wallman, Marie Barnhill, and Joseph Emerson. Robust characterization of leakage errors. *New Journal of Physics*, 18(4):043021, apr 2016.

[WBL12] Nathan Wiebe, Daniel Braun, and Seth Lloyd. Quantum algorithm for data fitting. *Phys. Rev. Lett.*, 109:050505, Aug 2012.

[WER$^+$20] Max Werninghaus, Daniel J. Egger, Federico Roy, Shai Machnes, Frank K. Wilhelm, and Stefan Filipp. Leakage reduction in fast superconducting qubit gates via optimal control, 2020.

[WG18] Christopher J. Wood and Jay M. Gambetta. Quantification and characterization of leakage errors. *Physical Review A*, 97(3), Mar 2018.

[WGHF15] Joel Wallman, Chris Granade, Robin Harper, and Steven T Flammia. Estimating the coherence of noise. *New Journal of Physics*, 17(11):113020, nov 2015.

[WHE$^+$04] Yaakov S. Weinstein, Timothy F. Havel, Joseph Emerson, Nicolas Boulant, Marcos Saraceno, Seth Lloyd, and David G. Cory. Quantum process tomography of the quantum fourier transform. *The Journal of Chemical Physics*, 121(13):6117–6133, 2004.

[WKS16] Nathan Wiebe, Ashish Kapoor, and Krysta M Svore. Quantum perceptron models, 2016.

[WSM20] Ellis Wilson, Sudhakar Singh, and Frank Mueller. Just-in-time quantum circuit transpilation reduces noise. In *2020 IEEE International Confer-*

ence on Quantum Computing and Engineering (QCE), pages 345–355, 2020.

[WZY05] X. R. Wang, Y. S. Zheng, and Sun Yin. Spin relaxation and decoherence of two-level systems. *Phys. Rev. B*, 72:121303, Sep 2005.

[YN05] J. Q. You and Franco Nori. Superconducting circuits and quantum information. *Physics Today*, 58(11):42–47, Nov 2005.

[ZMM17] Ming-Jing Zhao, Teng Ma, and YuQuan Ma. Coherence evolution in two-qubit system going through amplitude damping channel, 2017.

[ZPW19] Alwin Zulehner, Alexandru Paler, and Robert Wille. An efficient methodology for mapping quantum circuits to the ibm qx architectures. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 38(7):1226–1236, 2019.

VITA

December 1, 1997	Born, Miami, Florida
2021	Ingeniería de Sistemas Universidad de Antioquia Medellín, Antioquia

PUBLICATIONS AND PRESENTATIONS

David Quiroga, Prasanna Date, (2020). *Discriminating Quantum States with Quantum Machine Learning.* IEEE International Conference on Quantum Computing and Engineering (In review process).