Optimal Control of Scalable Quantum Devices

Dissertation
zur Erlangung des Grades
des Doktors der Naturwissenschaften
der Naturwissenschaftlich-Technischen Fakultät
der Universität des Saarlandes

von

Per J. Liebermann

Saarbrücken

2017

Tag des Kolloquiums:	21.12.2017
Dekan:	Prof. Dr. Guido Kickelbick
Mitglieder des Prüfungsausschusses:	Prof. Dr. Christoph Becher
	Prof. Dr. Frank Wilhelm-Mauch
	Prof. Dr. Christoph Bruder
	Dr. Adam Miroslaw Wysocki

> Es bleibt mir eine unerschöpfliche Quelle des Erstaunens, wenn ich sehe, wie ein paar Kritzeleien auf einer Tafel oder auf einem Blatt Papier den Lauf menschlicher Angelegenheiten verändern können.

Stanislaw Ulam

Für Zoë

Zusammenfassung

Supraleitende Qubits sind elektrische Schaltkreise die im Millikelvin-Temperaturbereich betrieben werden und auf Josephson-Kontakten basieren. Sie erlauben schnelle Operationen mittels herkömmlicher Mikrowellenelektronik und Tieftemperaturkühlung und sind ein Kandidat für die Realisierung eines universellen Quantencomputers. In dieser Arbeit verwenden wir Methoden der optimalen Steuerung um die Leistung von Quantengattern in skalierbaren Transmonqubit-Architekturen zu verbessern. Wir zeigen wie ein Multiqubitgatter für die Quantensimulation effizient angewendet wird, nutzten digitale Basisfunktionen um die Anzahl der Pulsparameter in der Optimierung zu reduzieren und simulieren Dekohärenzeffekte für analytische Pulsformen eines perfekten Verschränkers. Das Hauptaugenmerk der Arbeit liegt auf der Anpassung und Anwendung optimaler Steuerungsmethoden für die Integration des Steuerungsschaltkreises mit dem supraleitende Qubit auf einem gemeinsamen Chip. Unter Verwendung von Ideen und Bauteilen aus dem Gebiet der supraleitenden konventionellen Computer nutzen wir optimale Steuerung mit nur einem Bit an Amplitudenauflösung um Pulssequenzen für schnelle Quantengatter zu finden, leiten realisierbare Systemparameter ab und sowie Bedingungen für analytische Pulssequenzen. Dies hat das Potenzial die Verkabelung der Mikrowellensteuerung und die gesamte Wärmeentwicklung auf dem Chip zu reduzieren und erlaubt eine einfache Skalierbarkeit wenn die Anzahl der Qubits weiter wächst.

Abstract

Superconducting quantum bits are integrated circuits operated at Millikelvin temperatures and rely on Josephson junctions as their key element. They allow fast operations through standard room-temperature microwave electronics and cryogenic cooling and are currently pursed for the realization of universal quantum computers. In this thesis we use optimal control methods to improve the performance of quantum gates in scalable architectures consisting of transmon qubits. We show how to apply a multi-qubit gate for quantum simulation efficiently, use digital basis functions to reduce the number of pulse parameters in the optimization and simulate the effects of decoherence for analytical pulse shapes performing a perfect entangling gate. The main part focuses on tailoring and applying optimal control methods for an on-chip integration of the control circuitry with the superconducting qubit. This borrows ideas and devices developed in the field of superconducting conventional computers, and we use optimal control with only a single-bit of amplitude resolution to find trains of pulses for fast quantum gates, derive reasonable system parameters and extract conditions for analytic pulse trains. This has the potential to reduce the wiring of microwave control lines and the overall heat load on the chip, allowing for easier scalability when the number of qubits continues to grow.

Contents

1. Introduction

Quantum computation is a rapidly growing field, and superconducting qubits are one of the promising candidates to exhibit quantum supremacy in the near term future. This will allow us to solve certain computational problems which cannot be solved efficiently with conventional computers. We start with some basics about quantum computation, define the computational basis and introduce the gate model. This is followed by the quantization of electrical circuits and the introduction of Josephson junction qubits, which are the building blocks of superconducting quantum devices. We briefly discuss the theory of open quantum systems and conclude with optimal control theory, explaining the two numerical algorithms we use throughout this work.

1.1. Quantum computation

We give a very brief introduction to the field of quantum computation. This part mainly follows the work of Nielsen and Chuang [1], which is highly recommended for a deeper understanding of the topic. The basic element in quantum computation is a two level system with the two basis states

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = e_0 \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = e_1 \,. \tag{1.1}$$

In analogy to conventional computation this is called a qubit (quantum bit). These two states build the computational basis, and the two basis states correspond to the states of a conventional bit. However, in quantum mechanics each normalized superposition of the two basis states forms another state in the underlying Hilbert space, and two states that only differ in their global phases are indistinguishable. Therefore, we can describe each pure qubit state $|\psi\rangle$ in terms of the two angles θ and φ, such that it reads

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\varphi}\sin\left(\frac{\theta}{2}\right)|1\rangle \,. \tag{1.2}$$

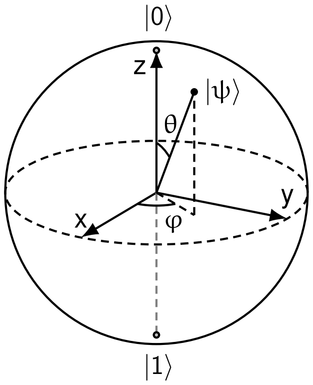

Figure 1.1.: Bloch sphere with the two computational states $|0\rangle$ and $|1\rangle$ at the poles, and a pure state $|\psi\rangle$ resting on the surface of the sphere, defined through the angles θ and φ.

The angles can take the values $0 \leqslant \theta \leqslant \pi$ and $0 \leqslant \phi < 2\pi$, and they determine the state uniquely. Written in such a way, a qubit state is described by a point on the unit sphere, which is also known as the Bloch sphere and shown in Figure 1.1. The vector pointing from the origin onto the surface is called the Bloch vector. This allows to describe the unitary dynamics of the qubit as rotations of the Bloch vector.

For an array of qubits the basis states spanning the underlying Hilbert space are formed by the tensor product of the single-qubit computational states

$$|n_1\rangle \otimes |n_2\rangle \otimes \cdots \otimes |n_N\rangle = |n_1 n_2 \ldots n_N\rangle \ . \tag{1.3}$$

Each n_i takes the values 0 and 1, and N is the number of qubits. This results in 2^N basis states, and every normalized superposition of the basis states is again a quantum state. However, this lacks a nice graphical representation such as the Bloch sphere for a single qubit, due to the vast amount of parameters that determine the state. If the N-qubit state $|\Psi\rangle$ is a product state

$$|\Psi\rangle = \prod_{i=1}^{N} \left|\psi^{(i)}\right\rangle , \tag{1.4}$$

then the state of each individual qubit $\left|\psi^{(i)}\right\rangle$ can again be represented by a Bloch vector. In general this is not the case, and states that are not product states are called entangled states. Written in the Schmidt decomposition with complex amplitudes α_j they read

$$|\Psi\rangle = \sum_{j=1}^{M} \alpha_j \prod_{i=1}^{N} \left|\psi_j^{(i)}\right\rangle , \tag{1.5}$$

such that at least two Schmidt ceofficients α_j are not zero. Otherwise $|\Psi\rangle$ would be a product state. Entangled states play a key role in quantum computation and quantum information, and they have baffled even the most brilliant minds [2]. An example of an orthogonal set of two-qubit entangled states are the four Bell states

$$\left|\Phi^{\pm}\right\rangle = \frac{|00\rangle \pm |11\rangle}{\sqrt{2}} \qquad \left|\Psi^{\pm}\right\rangle = \frac{|01\rangle \pm |10\rangle}{\sqrt{2}} . \tag{1.6}$$

These states are also known to be maximally entangled, since the absolute values $|\alpha_j|$ of the non-vanishing Schmidt coefficients are equal [3].

1.1.1. Single-qubit gates

Having defined the building block of quantum computers, we can now think about performing operations on the qubits. According to the laws of quantum mechanics, the time evolution of quantum states is unitary and so must be the gates we can apply on the qubits. For conventional computation, there are two reversible gates acting on a single bit, namely the identity operator and the NOT gate

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} . \tag{1.7}$$

The identity operator leaves the bit unaltered, and the NOT gate flips the bit from 0 to 1 and vice versa. For the qubit the following additional basic gates are defined

$$\begin{gathered} Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\ S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \quad T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} . \end{gathered} \tag{1.8}$$

The gates X, Y and Z are just the Pauli matrices which obey the commutation relations

$$[X, Y] = 2iZ \quad [Y, Z] = 2iX \quad [Z, X] = 2iY . \tag{1.9}$$

The Hadamard gate H can be understood as basis change operator between the eigenstates of Z and X. The phase gate S and the $\pi/8$ gate T change the relative phase between the basis states. It should be noted that this list is somehow redundant, i. e., two concatenated $\pi/8$ gates return the phase gate $S = T^2$. Furthermore, H and T gates are enough to construct all the Pauli gates, as well as any rotation on the Bloch sphere up to the desired precision. Therefore, H and T gate are said to be universal. An arbitrary rotation of the Bloch vector is defined through the rotation operator

$$R_{\hat{n}}(\theta) = \exp\left(-i\theta\left(n_x X + n_y Y + n_z Z\right)/2\right) . \tag{1.10}$$

Each single-qubit gate can be written as rotation operator around a normalized axis $\hat{n} = (n_x, n_y, n_z)^\top$, with rotation angle θ and omitting the global phase factor.

1.1.2. Two-qubit gates

Single-qubit gates already allow for more operations compared to conventional gates on a single-bit, but one thing that makes quantum computing special is entanglement between two qubits. We introduce perfect entanglers, i. e., gates that take some initial product state $|\psi\rangle = |\varphi\rangle|\chi\rangle$ and return a maximally entangled state $|\Psi\rangle = (|\varphi_1\rangle|\chi_1\rangle + |\varphi_2\rangle|\chi_2\rangle)/\sqrt{2}$. One example for a perfect entangler is the controlled-Z (CZ) gate

$$U_{CZ} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} , \tag{1.11}$$

which performs a phase shift of π on the state $|11\rangle$. To see that this is indeed a perfect entangler, we start with the product state

$$|\psi\rangle = \frac{1}{2}\left(|0\rangle + |1\rangle\right)\left(|0\rangle + |1\rangle\right) . \tag{1.12}$$

Applying the CZ gate turns this into the maximally entangled state

$$|\Psi\rangle = \frac{1}{2}|0\rangle\left(|0\rangle + |1\rangle\right) + \frac{1}{2}|1\rangle\left(|0\rangle - |1\rangle\right) , \tag{1.13}$$

which we cannot write as a product state because of the minus sign. We also see that the CZ gate is symmetric, i. e., control and target qubit can be

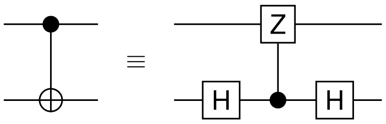

Figure 1.2.: The CNOT gate and its decomposition into CZ and Hadamard gates in the circuit representation, showing the equivalence between the two entanglers up to local rotations. Each horizontal line represents a qubit, single-qubit gates are drawn as boxes, controlled operations as vertical lines connecting the control (dot) and target qubit, and time increases from left to right.

interchanged arbitrarily. Another way to show that the CZ gate is indeed a perfect entangler is to use the geometric picture of two-qubit gates [4]. Any two-qubit gate is called locally equivalent to the CZ gate if it only differs in single-qubit operations. Such a gate is the controlled-NOT (CNOT) gate

$$U_{\text{CNOT}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \tag{1.14}$$

which swaps the state of the second qubit depending on the state of the first qubit. In Figure 1.2 we show the circuit representation of the CNOT gate and its decomposition into a CZ gate and two Hadamard gates. Each horizontal lines corresponds to a qubit, and single-qubit gates are drawn as boxes, except for the NOT gate, where we use the symbol $\oplus$ instead. A black dot marks the control qubit and is connected to the target qubit by a vertical line. By convention, the time axis points from left to right. Now we see that instead of performing the CNOT gate directly, we can first perform a Hadamard gate on the target qubit, followed by a CZ gate and another Hadamard gate on the target qubit. It turns out that the CNOT gate, together with the Hadamard and T gate, form a universal set and therefore they are sufficient to construct arbitrary multi-qubit gates up to the desired precision [1].

1.1.3. Measurement of states

Equipped with the definition of qubit states and the multi-qubit operations that can be performed on them, the last step is to readout the state of the system. We start with the general single-qubit state $|\psi\rangle = a\,|0\rangle + b\,|1\rangle$, where

a and b are complex coefficients such that $|\psi\rangle$ is normalized. The measurement of $|\psi\rangle$ is done in the computational basis $|0\rangle$ and $|1\rangle$. Measuring the state leads to an outcome of 0 with probability $|a|^2$ and an outcome of 1 with probability $|b|^2$. After the measurement the qubit is either in the state $|0\rangle$ or $|1\rangle$, with probability $|a|^2$ and $|b|^2$, respectively. Such a measurement is known as projective measurement. If we want to measure in a different basis we first apply the corresponding unitary operation onto the qubit before the measurement. For example, to measure in the eigenbasis of the X gate we apply a Hadamard gate on the qubit, which maps the state $a\,|0\rangle + b\,|1\rangle$ onto the state

$$\frac{(a+b)\,|0\rangle + (a-b)\,|1\rangle}{\sqrt{2}}\,. \tag{1.15}$$

Now the probabilities to measure $|0\rangle$ and $|1\rangle$ are $|a+b|^2/2$ and $|a-b|^2/2$, respectively. These probabilities are the same as the ones we would get by measuring the initial state in the eigenbasis of X. In case of a maximally entangled two-qubit state the measurement outcome of the first qubit determines the measurement outcome of the second qubit. Measuring the first qubit of the Bell state $|\Psi^{\pm}\rangle$ leads to an outcome of 0 with probability 1/2 and to an outcome of 1 with the same probability, such that the two-qubit state after the measurement is $|01\rangle$ or $|10\rangle$, respectively. Therefore, we can use entanglement between two qubits to determine the state of the second qubit by a single measurement of the first qubit.

1.1.4. Criteria for a universal quantum computer

Now that we have defined all the necessary operations on the qubits, we can write down the necessary requirements to realize a physical machine to perform useful quantum computations. These requirements are known as the DiVincenzo criteria [5]. First we need a scalable system with well characterized qubits, so the system parameters need be known accurately. This includes the interaction between different qubits, couplings to non-computational states and interactions with external fields. Once we have defined the mathematical subspace of coupled two-level systems, the qubits need to be initialized into a well known state for quantum computation. It is sufficient if we can prepare each qubit in the state $|0\rangle$ and apply the necessary unitary gates afterwards. To implement an algorithm a universal set of gates with low error probability is needed. We have already discussed different single- and two-qubit gates. Although H, T and CNOT gates are sufficient for quantum computation, being able to perform other single- and multi-qubit gates directly would not harm if their error is low, since decomposition into a few basic gates increases the runtime of the algorithm. Furthermore the measurement must be reliable to read out the state and retrieve the outcome of an algorithm. It is also possible to use measurement for initialization, such that we can trigger the unitary initialization operation on a measurement outcome. Finally, we need a low decoherence rate to protect the qubits long enough from interactions their environment. Every open quantum system suffers from decoherence which leads to the loss of quantum information, but quantum systems cannot be built completely isolated; otherwise we would loose the ability to control these systems from the outside. Strictly speaking, we would not be able to initialize the state, perform the required gates and measure the qubits. Quantum devices need to be designed such that calculations are safe from these environmental disturbances, and the clock rate for operations on the quantum computer must be much higher than the decoherence rate.

1.2. Superconducting qubits

Superconducting qubits are artificial atoms made of superconducting circuits with non-linear elements. The mathematical description is covered in circuit quantum electrodynamics (CQED), the superconducting analogue to conventional QED, describing the interaction between atoms and light.

1.2.1. The quantum LC-oscillator

We start with the LC-oscillator, which is a closed loop electrical circuit built out of an inductance L serially connected to a capacitance C, as can be seen in Figure 1.3. Given the voltage difference V to the ground and the circular

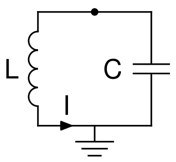

Figure 1.3.: The LC-oscillator, with voltage V against ground and current I.

current I, the energies stored in the inductor and capacitor are

$$E_L = \frac{L}{2} I^2 \quad \text{and} \quad E_C = \frac{C}{2} V^2 , \tag{1.16}$$

respectively. The flux is defined through the time integral of the voltage [6, 7]

$$\Phi(t) = \int_{-\infty}^{t} V(\xi) d\xi , \tag{1.17}$$

and the charge is the time integral of the current

$$Q(t) = \int_{-\infty}^{t} I(\xi) d\xi . \tag{1.18}$$

For the inductor, current and flux are related through the inductance of the coil

$$I = \frac{1}{L} \Phi . \tag{1.19}$$

Together with equation 1.17, the Lagrangian of the LC-oscillator reads

$$\mathcal{L} = \frac{C}{2} \dot{\Phi}^2 - \frac{1}{2L} \Phi^2 . \tag{1.20}$$

In analogy to the mechanical oscillator, the capacitance C plays the role of the "mass", the reciprocal inductance $1/L$ the one of the "spring constant", and the flux Φ takes the role of the "position". The conjugate variable of the flux is the charge

$$Q = \frac{\partial \mathcal{L}}{\partial \dot{\Phi}} = C \dot{\Phi} , \tag{1.21}$$

and therefore the Hamiltonian of the system reads

$$H = \frac{Q^2}{2C} + \frac{\Phi^2}{2L} . \tag{1.22}$$

Now we quantize the circuit, with the condition that the Heisenberg uncertainty relation between the conjugate variables

$$\Delta\Phi \Delta Q \geqslant \frac{\hbar}{2} \tag{1.23}$$

holds. We can define the raising and lowering operators $a^\dagger$ and a such that the Hamiltonian reads

$$H = \hbar\omega \left(a^\dagger a + \frac{1}{2} \right) , \tag{1.24}$$

with the minimum energy being $E_0 = \hbar\omega/2$ and the angular frequency $\omega = 1/\sqrt{LC}$ of the oscillator. With the Heisenberg uncertainty in equation 1.23 holding true for the ground state, the flux and charge operators are

$$\Phi = \sqrt{\frac{\hbar Z}{2}} \left(a^\dagger + a \right) \quad Q = i\sqrt{\frac{\hbar}{2Z}} \left(a^\dagger - a \right) , \tag{1.25}$$

where $Z = \sqrt{L/C}$ is the intrinsic impedance of the LC-oscillator. The uncertainty relation for the flux and charge operators reads

$$[\Phi, Q] = i\hbar , \tag{1.26}$$

with $\left[a, a^\dagger\right] = 1$. Therefore, the quantum LC-oscillator is the circuit analogue of the quantum mechanical oscillator. However, in quantum computing we need qubits, and these cannot be built from the LC-oscillator, since the energy gap between neighbouring energy levels is always the same. In such a case it is not possible to address a specific transition between two neighbouring energy levels exclusively. Therefore, we need non-linear elements in our circuit to break the symmetry of the LC-oscillator and to define a two-level subspace for the qubit.

1.2.2. The Josephson junction

A non-linear element in superconductors is the Josephson junction. The junction is made of a superconductor that is interrupted with a short gap of an insulating (non-superconducting) material (SIS). In Figure 1.4 the circuit diagram of the Josephson junction is shown. The junction has an intrinsic capacitance C_J shunted in parallel to the inductance L_J of the junction. The current across the Josephson junction is [8]

$$I = I_0 \sin\left(\varphi\right) , \tag{1.27}$$

with the critical current I_0 and the phase difference φ between the two sites of the junction. In the presence of a voltage the phase difference changes in time [7]

$$\hbar\dot{\varphi} = 2eV . \tag{1.28}$$

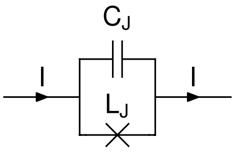

Figure 1.4.: The circuit diagram of a Josephson junction, with inductance L_J and parallel intrinsic capacitance C_J.

For superconductors the current across the junction is maintained through Cooper pairs, which carry a charge of $q = -2e$. Therefore, the flux quantum for superconductors is defined as [8]

$$\Phi_0 = \frac{\hbar}{2e} . \tag{1.29}$$

Flux Φ and phase φ are related through

$$\varphi = \frac{2e}{\hbar}\Phi = 2\pi\frac{\Phi}{\Phi_0} . \tag{1.30}$$

The Hamiltonian of the Josephson junction reads [7]

$$H = -E_J \cos(\varphi) , \tag{1.31}$$

with the Josephson coupling energy E_J which measures the ability of Cooper pairs tunnelling across the junction. The latter defines the critical current

$$I_0 = \frac{2\pi E_J}{\Phi_0} , \tag{1.32}$$

which can be obtained from $I = \partial H / \partial \Phi$.

1.2.3. The Cooper pair box

In Figure 1.5 we show the circuit diagram of the Cooper pair box qubit. The Josephson junction is capacitively coupled to a voltage source, such that the circuit forms a superconducting island between the gate capacitance C_g and the Josephson junction with intrinsic capacitance C_J. The Hamiltonian for the Cooper pair box reads [7]

$$H = E_C - E_J \cos\left(2\pi\frac{\Phi}{\Phi_0}\right) , \tag{1.33}$$

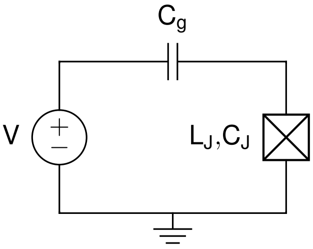

Figure 1.5.: The Cooper pair box qubit. The box with the cross is a shorthand notation for the Josephson junction with intrinsic capacitance.

with the charging energy

$$E_C = \frac{Q^2}{2C} = \frac{(2e)^2}{2C} \tag{1.34}$$

and the total capacitance $C = C_J + C_g$. The number operator of the charge reads

$$n = \frac{Q}{2e}, \tag{1.35}$$

and with the dimensionless offset gate charge

$$n_g = -\frac{C_g V}{2e} \tag{1.36}$$

the Hamiltonian reads

$$H = E_C \left(n - n_g\right)^2 - E_J \cos\left(\varphi\right) . \tag{1.37}$$

In the limit of high Josephson energies $E_C/E_J \ll 1$ we can use perturbation theory for small $\varphi \approx 0$, which is the so called transmon regime [9] of the Cooper pair box. Then the Hamiltonian reads

$$H \approx E_C \left(n - n_g\right)^2 - E_J + E_J \frac{\varphi^2}{2} - E_J \frac{\varphi^4}{24}, \tag{1.38}$$

which is basically a Duffing oscillator. It yields for the eigenenergies

$$E_m \approx -E_J + \sqrt{8E_C E_J}\left(m + \frac{1}{2}\right) - \frac{E_C}{12}\left(6m^2 + 6m + 3\right) , \tag{1.39}$$

with the energy difference between two neighbouring levels growing linearly with the state index

$$E_m - E_{m-1} = \sqrt{8E_C E_J} - E_C m . \tag{1.40}$$

The Hamiltonian can be rewritten in terms of the raising and lowering operators, and omitting all off-diagonal terms lead to

$$H_0 = \hbar \left(\omega_{10} - \frac{\Delta}{2} \right) a^\dagger a + \frac{\hbar\Delta}{2} \left(a^\dagger a \right)^2 , \tag{1.41}$$

with $\hbar\omega_{10} = E_1 - E_0$ and $\Delta = \omega_{21} - \omega_{10}$.

1.3. Optimal control theory

In optimal control we are faced with the problem of having a system with some variable parameters and the task to optimize these parameters such that a given cost function is minimized. Optimal control has been applied in a variety of systems for decades [10]. Optimization is mainly done with numerical methods, which for large system sizes is often the only choice. Analytical solutions can be retrieved from analysis of numerical solutions, or used as seeds for numerical optimization for systems derived from solved problems. In here we want to present two numerical methods of optimization we will use later in this work. The first method is a gradient ascent search for pulse shapes known as GRAPE [11], and the second method is a general gradient free global search method existing in plenty of variations in the family of genetic algorithms [12].

1.3.1. Pulse optimization with gradient ascent

Gradient based optimization algorithms have proven to often outperform gradient free methods. Here we want to introduce one of the workhorses of pulse shape optimization, which is known as GRadient Ascent Pulse Engineering (GRAPE) [11]. In its simplest form, we have a time independent drift Hamiltonian H_0 and M control Hamiltonians H_k with time dependent amplitudes $u_k(t)$. The total system Hamiltonian then reads

$$H(t) = H_0 + \sum_{k=1}^{M} u_k(t) H_k . \tag{1.42}$$

The task is to find control fields such that a given initial state $|\Psi_0\rangle$ is transformed into the desired target state $|\Psi_F\rangle$ under the unitary transformation

$$U_T = \mathcal{T} \exp \left(-i \int_0^T H(t)\, dt \right) , \tag{1.43}$$

with the time ordering operator $\mathcal{T}$ and using $\hbar = 1$. The fidelity function measures how well the final state $|\Psi_T\rangle = U_T |\Psi_0\rangle$ matches the desired target state through the state overlap $|\langle\Psi_F|\Psi_T\rangle|^2$. However, this would optimize the unitary operator only for the chosen input state. A state independent fidelity function is given by the absolute squared trace fidelity

$$\Phi_0 = \frac{1}{d^2}\left|\mathrm{Tr}\left\{U_F^\dagger U_T\right\}\right|^2 = \frac{1}{d^2}|\langle U_F|U_T\rangle|^2 , \tag{1.44}$$

with the desired unitary operator U_F and its dimension d. In the last step we have written the Hilbert-Schmidt scalar product between the two operators as inner product of the vectorized matrices (see Appendix A.1). This fidelity measure also has the advantage that it is independent of global phases in the unitary operator. In GRAPE the control vector $\{u_k\,(t)\}$ is approximated via piecewise constant (PWC) control amplitudes u_{kj}. The PWC amplitudes are samplings of the analytic shape $u_k(t)$ at the midpoint of each time interval.

Gradient search

Now we want to define a gradient for the fidelity function in terms of the PWC amplitudes u_{kj} we use in the gradient search. The time evolution of the wave function is given by the Schrödinger equation

$$\frac{\partial\,|\Psi\,(t)\rangle}{\partial t} = -i\left(H_0 + \sum_{k=1}^{M} u_k\,(t)\,H_k\right)|\Psi\,(t)\rangle\ . \tag{1.45}$$

In the following we use N discrete time steps with unit length

$$\Delta t = T/N\,. \tag{1.46}$$

Then the unitary operator at each of the N time steps is of the form

$$U_j = \exp\left[-i\Delta t\left(H_0 + \sum_{k=1}^{M} u_{kj} H_k\right)\right]\ . \tag{1.47}$$

Now we can write the fidelity function in terms of the jth time step as

$$\Phi_0 = \frac{1}{d^2}|\,\langle \underbrace{U_{j+1}^\dagger \ldots U_N^\dagger U_F}_{=\lambda_j}|\underbrace{U_j \ldots U_1}_{=\rho_j}\rangle|^2 = \frac{1}{d^2}\left|\langle\lambda_j|\rho_j\rangle\right|^2 , \tag{1.48}$$

where we have introduced the forward and backward evolution operators ρ_j and λ_j, respectively. The derivative of the fidelity function is given by

$$\begin{aligned}\frac{\partial \Phi_0}{\partial u_{kj}} &= 2i\frac{1}{d^2}\,\mathrm{Im}\left\{\langle U_F|U_T\rangle\,\frac{\partial\,\langle U_F|U_T\rangle}{\partial u_{kj}}\right\}\\ &= 2i\frac{1}{d^2}\,\mathrm{Im}\left\{\langle U_F|U_T\rangle\,\langle\lambda_j|\frac{\partial U_j}{\partial u_{kj}}\rho_{j-1}\rangle\right\}\\ &= 2\Delta t\frac{1}{d^2}\,\mathrm{Im}\big\{\langle U_F|U_T\rangle\,\langle\lambda_j\big|H_k\rho_j\rangle\big\}\end{aligned} \tag{1.49}$$

with

$$\frac{\partial U_j}{\partial u_{kj}} = -i\Delta t H_k U_j\,. \tag{1.50}$$

The GRAPE algorithm then works as follows:

1. Initialize the pulses u_{kj}.
2. Calculate the fidelity Φ and terminate if Φ lies above a given threshold.
3. Calculate the operators ρ_j, λ_j and the gradients $\frac{\partial\Phi_0}{\partial u_{kj}}$.
4. Update the controls u_{kj} and go back to 2.

The update rule for the control vector at each iteration is

$$u_{kj} \mapsto u_{kj} + \varepsilon\frac{\partial \Phi_0}{\partial u_{kj}}\,, \tag{1.51}$$

where ε is the step size of the update.

Filtering

The control vector found by GRAPE usually does not match the pulse shape that reaches the quantum systems. This is due to the limitations of the arbitrary waveform generator (AWG) used to create the pulse shapes itself, as well as damping in the control lines through the different temperature regimes from room temperature electronics to reach the cryogenic cooled system. Here we take into account this effect in terms of linear filtering of the control pulses [13]. Let s_{kl} be the controls after the filtering process. The linear filter process reads

$$s_{kl} = \sum_j w_{k,l-j}u_{kj}\,, \tag{1.52}$$

with the window function $w_{k,l-j}$, which we allow to be different for each control k. We also use a symmetric window functions, i. e. $w_l = g_{l+M/2}$, with M being the window size and g_l is a vector storing the function values of the window function. Now the unitary for each pixel reads

$$U_l = \exp\left[-i\Delta t\left(H_0 + \sum_{k=1}^{M} s_{kl}H_k\right)\right] , \tag{1.53}$$

from which we get the gradient of the fidelity with respect to the filtered controls

$$\frac{\partial\Phi_0}{\partial s_{kl}} = 2\Delta t\frac{1}{d^2}\,\mathrm{Im}\left(\langle U_F|U_T\rangle\,\langle\lambda_l|H_k\rho_l\rangle\right) . \tag{1.54}$$

With the help of the chain rule the gradient of the fidelity with respect to the unfiltered controls is accessible

$$\frac{\partial\Phi}{\partial u_{kj}} = \sum_l \frac{\partial s_{kl}}{\partial u_{kj}}\frac{\partial\Phi}{\partial s_{kl}} = \sum_l w_{k,l-j}\frac{\partial\Phi}{\partial s_{kl}} . \tag{1.55}$$

Therefore, we calculate the gradients for the filtered controls first before we get the gradients for the unfiltered controls and can apply the update rule.

1.3.2. Genetic algorithms

We have already mentioned that gradient based algorithms are faster than gradient free optimization methods. However, it is not always possible to obtain a gradient for the system. This may be due to unknown system parameters or the way the system is simulated, such that the system dynamics are not characterized well enough. Genetic algorithms [12, 14], on the other hand, are a well established optimization method. It is a very versatile tool to attack global optimization problems for complicated control landscapes. Genetic algorithms have been applied successfully in a variety of systems, like analog quantum optimal control, e. g., to optimize laser pulses to control molecules [15].

Within the genetic algorithm framework, every solution for the variational parameters of the control problem is encoded into a genome. This might be a string of boolean, integer or real numbers. For pulse shape optimizations, each real valued control amplitude u_{jk} can be encoded as bit string, which we refer to as a gene, and the latter are concatenated to a genome. Another possibility is to encode the control vector directly, such that the genome is a string of real values. We mainly focus on genomes that contain genes of boolean type with size one, such that the genome is a simple bit string. Each genome is a more or less good solution to our problem. The fitness of a genome measures how well the genome solves the problem and we choose the fidelity function as fitness measure.

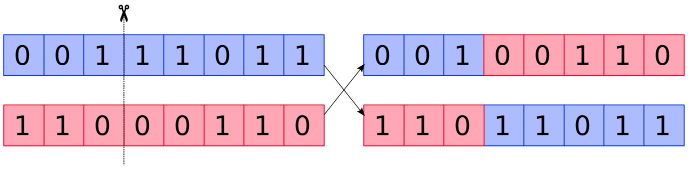

Figure 1.6.: Crossover of two genomes, with the parental genomes marked in different colours. Each genome is cut at the third position and the tails of the genomes are exchanged.

Now we can define two basic operations on the genomes. The first one includes pairs of genomes to breed new genomes. The crossover of two genomes cuts each of them at a random position and concatenates the first part of the first genome with the second part of the second genome and the first part of the second genome with the second part of the first genome. We restrict the random position of the cut to be the same for both genomes, such that each genome is of the same length. Figure 1.6 shows the crossover operation between two genomes. The crossover operation is independent of the genome type and can be extended to include multiple cuts per genome [16], and one can restrict the cut position to not destroy single genes.

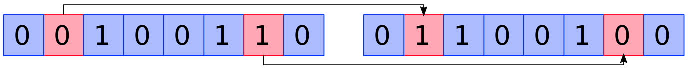

Figure 1.7.: Mutation of a bit-string genome. At the second and seventh position, marked in red, the bit in the genome is flipped.

The second, even more basic operation involving only a single genome is mutation. For each gene in the genome we throw a coin and change the gene depending on the outcome. For a simple bit, this is just a bit flip at the given position, and for real numbers we increment the value by a small amount. The mutation rate defines how strong the coin is biased and is usually chosen to be low to keep the overall pattern of a genome. Figure 1.7 shows a genome consisting of single bits undergoing some mutation.

Equipped with these operations on the genomes, the genetic algorithm works as follows:

1. Initialize a population of genomes of size N.
2. Select a subset $\leqslant N$ of genomes from the population for mating them via crossover with probability p_{mate}.
3. Mutate all genomes with probability p_{mutate}.
4. Decode the genomes to retrieve their fitness values.
5. Select, depending on their fitness value, the genomes that make it to the next generation and go back to 2.

The algorithm terminates if we find a genome with a fitness value above a certain threshold. The selection process for the crossover operation allows for multiple copies of genomes with high fitness values. Such a selection process is roulette wheel selection, where the selection probability is proportional to the fitness value. The decoding process turns the genomes into the controllable system parameters we are optimizing, with which in turn the time evolution of the system and the fidelity are calculated. The final selection process we choose is a survival of the fittest of the mutated genomes. We point out that at each step in the algorithm a variety of parameters can be changed. For example, we could mate all genomes in the population without prior selection, and only the mated genomes make it to the next generation, depending on a final fitness based selection process like roulette wheel. Also, one can choose a subset of genomes which are fixed and only get extinct if better genomes are found.

1.4. Open quantum systems

Quantum systems often cannot be viewed as isolated systems alone. As we have already discussed in Section 1.1.4, the interaction with the environment of the quantum device is unavoidable. Here we want to describe the basic model to take environmental effects into account, which is the Lindblad master equation for the density matrix [3].

1.4.1. Liouville space

Before we can start to take environmental effects into account, we define the density matrix

$$\rho = \sum_j w_j \left|\Psi_j\middle\rangle\middle\langle\Psi_j\right| \quad w_j \geqslant 0\,, \tag{1.56}$$

with weights $w_j \geqslant 0$. The normalization condition for the density matrix reads

$$\mathrm{tr}\{\rho\} = 1\,, \tag{1.57}$$

and the weights must satisfy $\sum_j w_j = 1$. If all but one w_j are zero, the state described by the density matrix is called a pure state. In this case the density matrix description of the state is equivalent to the description by the wave function. Otherwise, if at least two $w_j > 0$, the density matrix describes a mixed state. The time evolution of the density matrix is determined by the Liouville-von Neumann equation

$$\frac{\partial\rho(t)}{\partial t} = -\frac{i}{\hbar}[H, \rho]\,, \tag{1.58}$$

which follows directly from the Schrödinger equation and the definition of the density matrix in equation 1.56. The density matrix at time t starting in the initial state ρ_0 reads

$$\rho\,(t) = \mathcal{E}(t)(\rho_0) = U(t)\rho_0 U^\dagger(t)\,, \tag{1.59}$$

where $\mathcal{E}(t)$ maps density matrices onto density matrices. While the wave function is an element of the Hilbert space $\mathcal{H}$, the density matrix is an element of the Liouville space $\mathcal{L}$. The latter is defined as the space of operators for which the Hilbert-Schmidt scalar product is finite and can be written as Cartesian product of the Hilbert space with itself

$$\mathcal{L} = \mathcal{H} \otimes \mathcal{H}\,. \tag{1.60}$$

Together with the Hilbert-Schmidt scalar product the Liouville space itself forms a Hilbert space.

Now we write the density matrix as vector in Liouville space. Using the vectorization method (see Appendix A.1), the density matrix reads

$$\begin{aligned} \rho &= \sum_{kl} \rho_{kl} \left|k\right\rangle \left\langle l\right| = \sum_{jkl} w_j \alpha_{jk} \alpha^*_{jl} \left|k\right\rangle \left\langle l\right| \\ \rightarrow \left|\rho\right\rangle &= \sum_{kl} \rho_{kl} \left|l, k\right\rangle = \sum_{jkl} w_j \alpha_{jk} \alpha^*_{jl} \left|l, k\right\rangle = \sum_j w_j \left|\Psi^*_j, \Psi_j\right\rangle\,, \end{aligned} \tag{1.61}$$

where we have used the shorthand notation $|l\rangle \otimes |k\rangle = |l, k\rangle$, and the complex coefficients α_{jk} are given by the underlying wave functions

$$\left|\Psi_j\right\rangle = \sum_k \alpha_{jk} |k\rangle \quad \alpha_{jk} \in \mathbb{C} . \tag{1.62}$$

In general we have two operators A and B acting on each side of the density matrix

$$\begin{aligned} A\rho B &= \sum_{jklm} A_{jl}\rho_{lm}B_{mk} |j\rangle \langle k| \\ \to |A\rho B\rangle &= \sum_{jklm} A_{jl}\rho_{lm}B_{mk} |k, j\rangle = (B^\top \otimes A) |\rho\rangle \ . \end{aligned} \tag{1.63}$$

It follows for the unitary transformations and the effect of the Hamiltonian

$$\begin{aligned} \left|U\rho U^\dagger\right\rangle &= (U^* \otimes U) |\rho\rangle \\ |H\rho\rangle &= (\mathbb{1} \otimes H) |\rho\rangle \\ |\rho H\rangle &= (H^\top \otimes \mathbb{1}) |\rho\rangle \ . \end{aligned} \tag{1.64}$$

Therefore, the vectorized von Neumann equation reads

$$\frac{\partial |\rho\rangle}{\partial t} = -\frac{i}{\hbar} \left(\mathbb{1} \otimes H - H^\top \otimes \mathbb{1}\right) |\rho\rangle \ . \tag{1.65}$$

1.4.2. Lindblad master equation

A closed quantum system embedded into an environment is described by the Hamiltonian of the system H_S, the Hamiltonian of the bath H_B and the interaction between system and environment H_{SB}. However, while the system is approximately well characterised, the environment adds a plethora of degrees of freedom, and it is ambitious to attempt a solution of the complete system. Furthermore, we are only interested in the evolution of the small quantum system, which is described by the reduced density matrix

$$\rho_S = \mathrm{tr}_B \, \rho \, , \tag{1.66}$$

where the partial trace is taken over the bath degrees of freedom. Then we want to get a differential equation for the reduced density matrix that describes the time evolution accurately. One such equation is given by the

Lindblad master equation, although requiring strong assumptions on the system-bath interactions [3]. First, we apply the Born approximation, which requires the interaction between system and bath to be weak, such that the influence of the system on the bath is almost negligible, and we can write $\rho \approx \rho_S \otimes \rho_B$. The Markov approximation assumes that the time derivative of the density matrix at time t only depends on the state of the system at that time, i. e., no memory effects occur in the system. In a last step, fast oscillating terms are averaged out within the rotating wave approximation. This leads to the Lindblad master equation

$$\frac{\partial \rho}{\partial t} = -\frac{i}{\hbar}[H, \rho] + \sum_i \gamma_i \left(L_i \rho L_i^\dagger - \frac{1}{2} L_i^\dagger L_i \rho - \frac{1}{2} \rho L_i^\dagger L_i \right) , \tag{1.67}$$

where ρ is the density matrix of the system and H its Hamiltonian, omitting the index S for the system. The operators L_i are the Lindblad operators leading to decoherence of the system. The general effect of these is a decay of off-diagonal elements in the density matrix (dephasing) and a change of the diagonal elements (relaxation). Therefore, depending on the initial state ρ_0 and the Lindblad operators, dephasing transforms pure states into mixed states. The factor γ_i is the decoherence rate of the corresponding Lindblad operator L_i. Using the relations of the former section we get the vectorized Lindblad master equation

$$\begin{aligned} \frac{\partial |\rho\rangle}{\partial t} &= -\frac{i}{\hbar} \left(\mathbb{1} \otimes H - H^\top \otimes \mathbb{1} \right) |\rho\rangle \\ &\quad + \sum_i \gamma_i \left(L_i^* \otimes L_i - \frac{1}{2} \mathbb{1} \otimes L_i^\dagger L_i - \frac{1}{2} (L_i^\dagger L_i)^* \otimes \mathbb{1} \right) |\rho\rangle . \end{aligned} \tag{1.68}$$

The number of independent Lindblad operators is at most $N^2 - 1$, where N is the dimension of the Hilbert space. The Lindblad master equation ensures that the resulting map $\mathcal{E}(t)$ describing the time evolution of the density matrix is completely positive and trace preserving. To measure the fidelity of a map $\mathcal{E}_T$ compared to the ideal map $\mathcal{E}_F$ we start with the trace fidelity of a closed quantum system

$$\Phi_0 = \frac{1}{d^2} \langle U_F | U_T \rangle^* \langle U_F | U_T \rangle = \frac{1}{d^2} \langle U_F^* \otimes U_F | U_T^* \otimes U_T \rangle , \tag{1.69}$$

and replacing the unitary map $U_T^* \otimes U_T$ with the actual map describing the non-unitary evolution of the open system gives

$$\Phi_0 = \frac{1}{d^2} \langle \mathcal{E}_F | \mathcal{E}_T \rangle . \tag{1.70}$$

We note that the map $\mathcal{E}_{\mathsf{F}} = \mathsf{U}_{\mathsf{F}}^{*} \otimes \mathsf{U}_{\mathsf{F}}$ describes the unitary target evolution.

1.5. Outline

The outline of this thesis is as follows: In Chapter 2 we introduce single-flux quantum (SFQ) pulses to control qubits and use a genetic algorithm to optimize single-qubit gates. In Chapter 3 we derive reasonable parameters for the implementation of optimized SFQ pulses. Using GRAPE, we optimize a three-qubit gate in a von Neumann based architecture for superconducting qubits in Chapter 4. In Chapter 5 we use digital basis functions for single- and two-qubit gates within a GRAPE optimization. Finally, in Chapter 6 we compare two implementation of a perfect entangler and how well they perform under the influence of Markovian decoherence.

2. Optimal control with single flux quantum pulses

Superconductor based rapid single flux quantum (RSFQ) technology was originally pursued for ultra-high-speed conventional computing platforms [18, 19, 20, 21]. It is able to generate high frequency identical voltage pulses with high reproducibility up to operations at 750 GHz. Compared to semiconductor electronics, the digital information is accurately represented by single flux quanta (SFQ). The high frequencies are possible through the fast switching time of the Josephson junctions, that lies in the picosecond range. Furthermore, there is the availability of interconnections which allows transmission of data pulses over long distances and close to the speed of light with low dissipation. Possible applications are telecommunications, supercomputers, sensing, digital signal processing and high-precision measurements [20], to name a few.

For superconducting qubits, SFQ pulses allow on-chip control and readout for quantum computers based on Josephson devices [22, 23, 24, 25, 26, 27]. With control pulses of picosecond duration, RSFQ shows potential for fast quantum gates [28]. Then again, the timing of SFQ pulses is a major challenge [29]. For single frequency sequences of SFQ pulses DC/SFQ converters allow time equidistant pulse trains. More complex sequences can be loaded into shift registers [30, 31]. Those integration capabilities make RSFQ technology compatible with extending quantum processors [32, 33].

Present day control schemes of superconducting qubits rely heavily on room-temperature electronics; the analog signals are transmitted through filters into the cryostat where the qubit resides at Millikelvin temperatures. This leads to a heavy physical overhead, especially for extensibility from several to hundreds or thousands of qubits. But with on-chip integration the ability to create perfectly matched gate pulses via high amplitude resolution is lost. Conventional controls also have a high power consumption. On the other hand, the power consumption in RSFQ is low (eRSFQ) [34].

Some of the ideas and figures presented in this chapter have been published in [17].

In the following we develop the key ideas to use SFQ pulses to drive superconducting qubits directly. The SFQ drive is a control with only a single bit of amplitude resolution. Given a high frequency clock, we commonly decide for each time interval if there is a pulse or not. Although it is possible to change the voltage shape of SFQ pulse, i. e., changing the width to height ratio while keeping the area constant [25], and furthermore, using pulse with different area via tuning the coupling capacitance to the qubit, this does not seem to be feasible within the time scale of a single quantum gate, but in-between gates. Furthermore, this adds more complexity to our circuit, which we try to avoid. However, we want to point out that the parameters used in our simulations can give some insight into key relations between the circuit parameters for optimization of pulse sequences.

2.1. SFQ logic

In semiconductor computer science, logical 0 and 1 are defined by the voltage. In the field of RSFQ based supercomputers, the logical 0 and 1 are defined by SFQ pulses within a time frame, i. e., for a logical 1 an SFQ pulse has occurred in the given time frame. Equation 1.17 gives the relation between flux and voltage

$$\frac{d\Phi}{dt} = V\,, \tag{2.1}$$

with the phase defined through the flux divided by the flux quantum

$$\varphi = 2\pi\frac{\Phi}{\Phi_0}\,. \tag{2.2}$$

The flux itself and therefore the superconducting phase are quantized [19]

$$\Phi = n\Phi_0\,. \tag{2.3}$$

Now an SFQ pulse is defined by the voltage pulse across a Josephson junction [35, 19, 21]

$$\int V(t)\,dt = \Phi_0\,, \tag{2.4}$$

where the area of the time integrated voltage has the size of a single flux quantum. Since the SFQ pulse is defined via voltage pulse shapes, one could use them to drive a transmon via a capacitor, similar to Figure 1.5 for general voltage pulses. The simplest approximation of the transmon is a two-level system

$$H_0 = \frac{\hbar\omega_{10}}{2}(\mathbb{1} - Z) \tag{2.5}$$

with the energy splitting $\hbar\omega_{10}$. Driving the qubit with a voltage pulse leads to a control Hamiltonian of the form

$$H_1 \sim i(a^\dagger - a) = Y\,, \tag{2.6}$$

which is proportional to the Pauli-Y operator in the two-level description. The lowering and raising operator of the qubit are a and $a^\dagger$, respectively. For a voltage-driven transmon the control field reads [32]

$$u(t) = C_g V_g(t)\sqrt{\frac{\hbar\omega_{10}}{2C_B}} = \frac{\Omega(t)}{2}\,, \tag{2.7}$$

where C_g is the gate capacitance and C_B the bias capacitance of the transmon qubit [9]. Therefore the time dependent Hamiltonian reads

$$H(t) = \frac{\hbar\omega_{10}}{2}(\mathbb{1} - Z) + \frac{\hbar\Omega(t)}{2}Y\,. \tag{2.8}$$

2.2. The DANTE sequence

In the limit of very short driving times $\Delta t \ll \tau = 2\pi/\omega_{10}$, the voltage pulse leads to a rotation on the Bloch sphere around the y-axis

$$R_y(\theta) = \exp\left(-i\theta Y/2\right) = \begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{bmatrix}\,, \tag{2.9}$$

where the angle is given by

$$\theta = \Omega_0 \Delta t = C_g V_0 \sqrt{\frac{2\omega_{10}}{\hbar C_B}}\Delta t\,. \tag{2.10}$$

We refer to this type of pulse shape as a hard pulse. Let us now define the two basic operations we can perform within RSFQ, with $\hbar = 1$. First, a free evolution of the qubit for a time t

$$U_0(t) = \exp\left(-iH_0 t\right) = \begin{bmatrix} 1 & 0 \\ 0 & e^{-i\omega_{10}t} \end{bmatrix}\,, \tag{2.11}$$

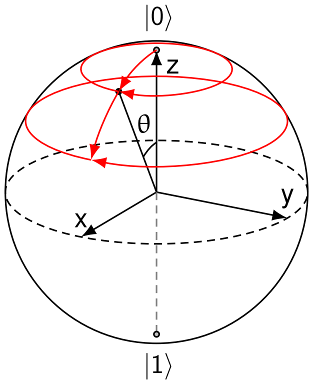

Figure 2.1.: The effect of the first two steps of the DANTE sequence on the $|0\rangle$ state. After a full precession of duration τ around the z-axis, applying an SFQ pulse to the transmon leads to a rotation of the state vector by a small angle $\theta = \pi/N$ around the y-axis. Repeating this N times performs a complete flip of the Bloch vector from $|0\rangle$ to $|1\rangle$.

which is up to a global phase just a rotation around the z-axis with angle $-\omega_{10}t$. The other operator is an applied voltage pulse, taking into account any generalization for a finite pulse width Δt for later calculations

$$\begin{aligned} U_1 &= \exp(-iH_0\Delta t)\exp(-iH_1\theta/2) \qquad (2.12) \\ &= \begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ e^{-i\omega_{10}\Delta t}\sin(\theta/2) & e^{-i\omega_{10}\Delta t}\cos(\theta/2) \end{bmatrix}. \qquad (2.13) \end{aligned}$$

Basically, we are interested to implement the following two gates: the π gate

$$U_\pi = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \qquad (2.14)$$

and the $\pi/2$ gate

$$U_{\pi/2} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}. \qquad (2.15)$$

The DANTE sequence (Delays Alternating with Nutations for Tailored Excitation) [36, 37] is a drive of the form

$$V(t) = V_0 \sum_{j=0}^{N-1} \delta(t - j\tau), \qquad (2.16)$$

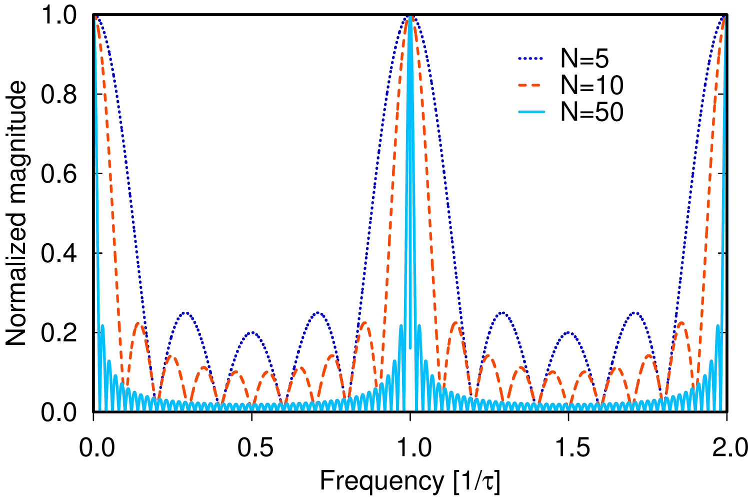

Figure 2.2.: Spectrum of the DANTE sequence with varying number of pulses N. Increasing the number of pulses narrows the bandwidth of the drive frequency and its higher harmonics, as well as decreasing the magnitudes of the side-lopes.

such that after each full precession of the Bloch vector a hard pulse is applied. This is basically an analogue of an rectangular pulse shape with the qubit energy spacing as carrier frequency. In Figure 2.1 we visualize the effect of the DANTE sequence on the Bloch vector. The nominal rotation angle is defined as $\alpha = \theta N = \Omega_0 N \Delta t$, where N is the number of pulses. Figure 2.2 shows the spectrum for different number of pulses. The more pulses are included, the narrower gets the bandwidth and the amplitudes of the sideband frequencies, such that long sequences with many pulses lead to a better drive of only the desired frequency. The resulting unitary for the qubit reads

$$U_T = [U_0 (n\tau - \Delta t) U_1]^N \, . \tag{2.17}$$

For the $\pi/2$ gate we have $N\theta = \pi/2$, and for the π gate $N\theta = \pi$. The gate time scales linearly with the number of pulses, i. e., $t_g = N\tau$.

2.3. Leakage

For a two level system the DANTE sequence works perfectly well if the delay between each pulse is exactly $n\tau$ and the number of pulses is $N = \alpha/\theta$. However, transmon qubits [9] generally show a low anharmonicity when

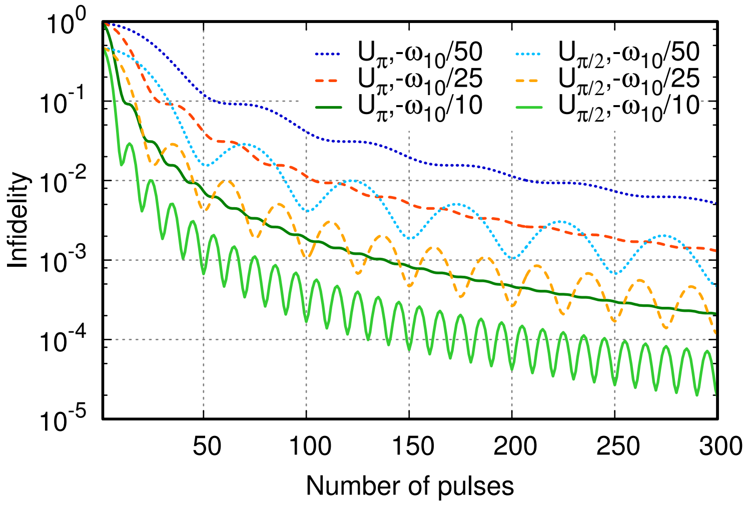

Figure 2.3.: Infidelity of the DANTE sequence for the target gates U_π and $U_{\pi/2}$ and for different values of the anharmonicity Δ, with varying number of pulses and a fixed drive frequency of $1/\tau$. Increasing the number of pulses N increases the gate time to $t_g = N\tau$ and decreases the pulse amplitudes to π/N and $\pi/2N$ for the two target gates U_π and $U_{\pi/2}$, respectively. The oscillation in the fidelity is determined through the anharmonicity and the drive frequency.

comparing the energy differences between the first and second excited state to the ground state and the first excited state. Therefore, we have to take into account at least one additional energy level in our model

$$H_0 = \left(\omega_{10} - \frac{\Delta}{2}\right) a^\dagger a + \frac{\Delta}{2}\left(a^\dagger a\right)^2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \omega_{10} & 0 \\ 0 & 0 & 2\omega_{10} + \Delta \end{bmatrix}, \tag{2.18}$$

with the anharmonicity $\Delta = \omega_{21} - \omega_{10}$ and the energy splitting ω_{21} between the second and first energy level. The last equation holds for a three level system, i. e., a qutrit. Since we have defined the drive via the qubit lowering and raising operator, its form does not change and we get for the qutrit

$$H_1 \sim i \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & -\sqrt{2} \\ 0 & \sqrt{2} & 0 \end{bmatrix}. \tag{2.19}$$

Driving with this Hamiltonian will eventually lead to leakage out of the computational subspace spanned by the two lowest states $|0\rangle$ and $|1\rangle$ into

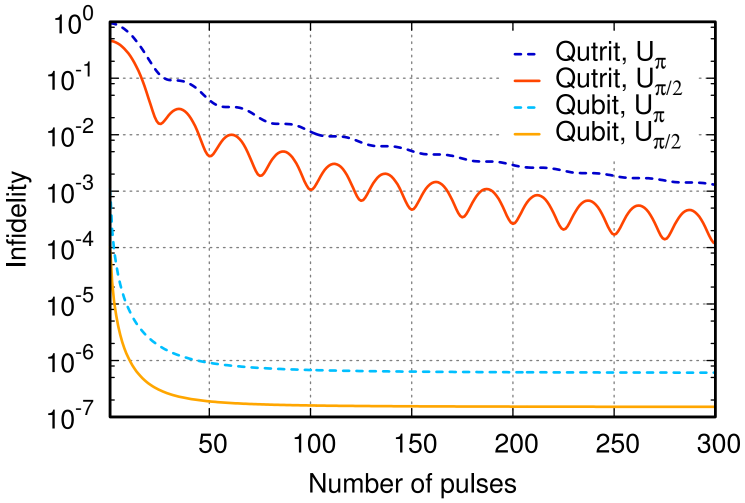

Figure 2.4.: Infidelity of the DANTE sequence for the target gates U_π and $U_{\pi/2}$, with Gaussian pulse shapes and varying number of pulses. For the qubit approximation the fidelity is limited through the Gaussian shape of the SFQ pulses, but this error is by orders of magnitude lower than the error that arises through the interaction with a third level within the qutrit model.

the second excited state $|2\rangle$ and vice versa. Our goal is then to find SFQ sequences which reduce this effect.

Besides the additional level, we also use a more physical pulse shape for our first proof of principle. We choose a Gaussian with a standard deviation of $\sigma_t = \Delta t/10$. This has the advantage that the bandwidth is no longer infinite. The pulse shape reads with $2t_c = \Delta t$

$$\Omega(t) = \theta \, \mathrm{erf}^{-1}\left(\frac{t_c}{\sqrt{2}\sigma}\right) \frac{e^{-t^2/2\sigma^2}}{\sqrt{2\pi}\sigma} \qquad \int_{-t_c}^{t_c} \Omega(t) dt = \theta \,, \tag{2.20}$$

where erf() guarantees normalization. The target gate for the qutrit is of the form

$$U_F = \Pi_Q \begin{bmatrix} \cos(\alpha/2) & -\sin(\alpha/2) & 0 \\ \sin(\alpha/2) & \cos(\alpha/2) & 0 \\ 0 & 0 & e^{i\varphi} \end{bmatrix} \Pi_Q \,, \tag{2.21}$$

with an arbitrary phase φ for the leakage level and the projector onto the computational subspace $\Pi_Q = \Pi_0 + \Pi_1$. This definition of the target gate allows us to ignore the phase of the leakage level and measure the fidelity in terms of the computational subspace of the qubit. The trace fidelity therefore reads [38]

$$\Phi = \frac{1}{4}\left|\mathrm{Tr}\left\{U_F^\dagger U_T\right\}\right|^2 , \tag{2.22}$$

which also ignores any global phase.

Figure 2.3 shows the infidelity dependence of the DANTE sequence on the number of pulses and the anharmonicity for the two gates under consideration, U_π and $U_{\pi/2}$. The rotation angle of each individual pulse is $\theta = \alpha/N$ and depends on the number of pulses N. We see that the fidelity increases with the number of pulses, and the oscillation frequency of the fidelity is proportional to the anharmonicity for fixed driving frequency $1/\tau$. Additionally, higher anharmonicities lead to better fidelities, which follows from the frequency spectrum of the DANTE sequence in Figure 2.2. Figure 2.4 shows the infidelity dependence of the number of pulses for a fixed anharmonicity $\Delta = -\omega_{10}/25$. The Gaussian pulse shape limits the reachable fidelity in the qubit case, but is still orders of magnitude better than the distortions of the ideal gate from the interaction with the leakage level.

2.4. Bit-string optimization

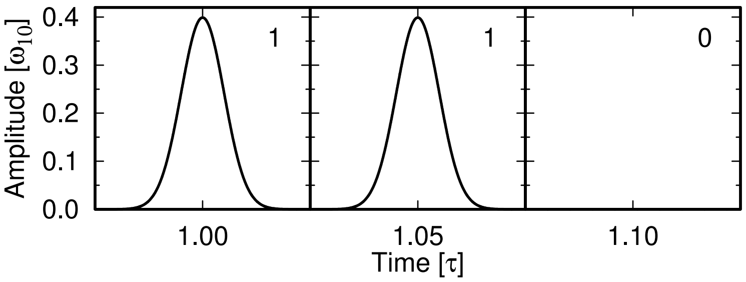

Figure 2.5.: Time slicing of the evolution. In each frame, the control amplitude $\Omega(t)$ either has a Gaussian shape (1) or vanishes (0). Therefore, only two unitary operators U_i need to be calculated, each with a duration of $2t_c$ (*here:* $t_c = \tau/40$). The delay between two consecutive pulses is always an integer multiple of $2t_c$.

In a previous section we have defined the two basic operations, i. e., $U_0(t)$ and U_1, describing the free evolution and the effect of an SFQ pulse, respectively. Here we want to introduce the bit string representation which will

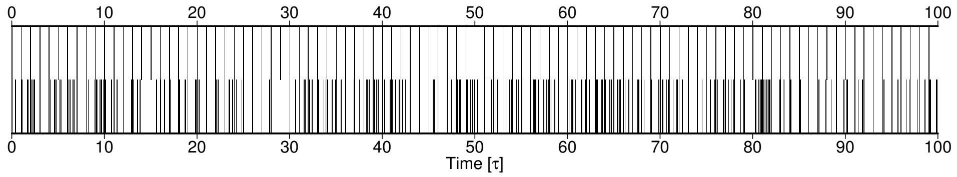

Figure 2.6.: Bit-string representation of the pulse sequence before (top) and after (bottom) the optimization for a gate time of $t_g = 100\,\tau$ to flip the Bloch vector completely. A vertical black line indicates the time when a pulse is applied and white space the time delay between consecutive pulses.

be used for optimization. Given a fast clock with frequency $f_{\text{clock}} = 1/2t_c$, in each time interval of duration $\Delta t = 2t_c$ the evolution of our transmon is either described by the free evolution operator $U_0(2t_c)$, i. e. no pulse is applied, or $U_1(2t_c)$. The total time evolution is then given by

$$U_T = \Pi_{i=N}^{1} U_{a_i} = U_{a_N} U_{a_{N-1}} \cdots U_{a_1} \tag{2.23}$$

with a_i being either 0 or 1. An example for three consecutive time intervals is shown in Figure 2.5. The database that the unitaries are chosen from contains only two entries

$$U_0 = \exp(-i2H_0t_c) = U_0(2t_c) \tag{2.24}$$

$$U_1 = U_0(t_c)\left[\mathcal{T}\exp\left(-i\int_{-t_c}^{t_c} H(t)\,dt\right)\right]U_0(-t_c)\,, \tag{2.25}$$

with $H(t) = H_0 + u(t)H_1$ and $\mathcal{T}$ the time ordering operator. Any adjustment in the experiment can be captured in the database and all characterization of the pulses has to be done once in order to find these two database entries. The operator U_1 is defined such that a hard pulse with $u(t) = \theta\delta(t)/2$ has its peak at the beginning of a new pixel, which maximizes the effect of the Y-rotation. Otherwise the whole sequence needs to be shifted in time. The total time evolution can therefore be encoded into a single bit string. For example, for the DANTE sequence [36, 32] with $t_c = \tau/40$ we get

$$a_i = \begin{cases} 1 \text{ if } & (i-1) \mod 20 = 0 \\ 0 \text{ else} & \end{cases} \quad . \tag{2.26}$$

The pulse sequence is already encoded into a bit string, and a gradient cannot be easily defined since the pulse shape is fixed. Only the pulse delays

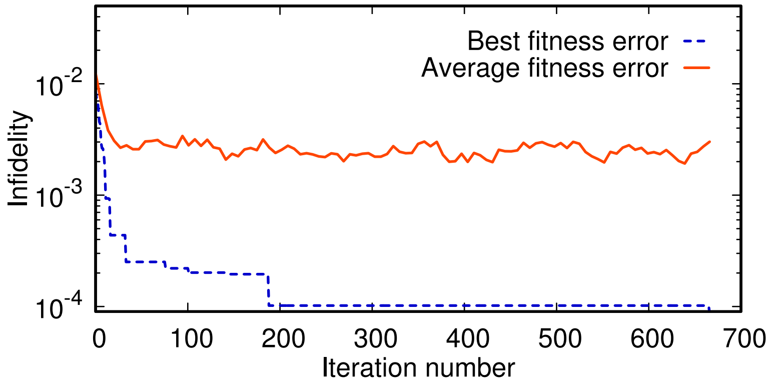

Figure 2.7.: The convergence of the genetic algorithm, starting with the DANTE sequence. At each iteration, a new population of different pulse sequences is bred from their parents, while the best solution of a generation with the lowest infidelity is kept until either the algorithm finds a better one or a given threshold is exceeded.

are variable, but then again, the number of pulses is fixed. Therefore we use a genetic algorithm to optimize the pulse sequence, which we have introduced in Section 1.3.2. We search for a local minimum in the control landscape starting with the DANTE sequence described earlier [36, 32]. If we reach the target fidelity or the maximum number of iterations the algorithm stops. The parameters of our optimization are shown in Table 2.1. Finding good values is a difficult task, partially because the effect on the search is not fully understood yet. However, it is common practice for most optimization problems to choose a high crossover and a low mutation probability [16]. We also point out that if we set the population size to one, the mutation operation changes the sequence only marginally, similar to a

Population size	70
Mutation probability	0.001
Crossover probability	0.9
Number of genomes to select for mating	64
Maximum allowed iterations	200 000
Target fitness	0.9999
Elitism	1

Table 2.1.: Parameters used in the genetic algorithm.

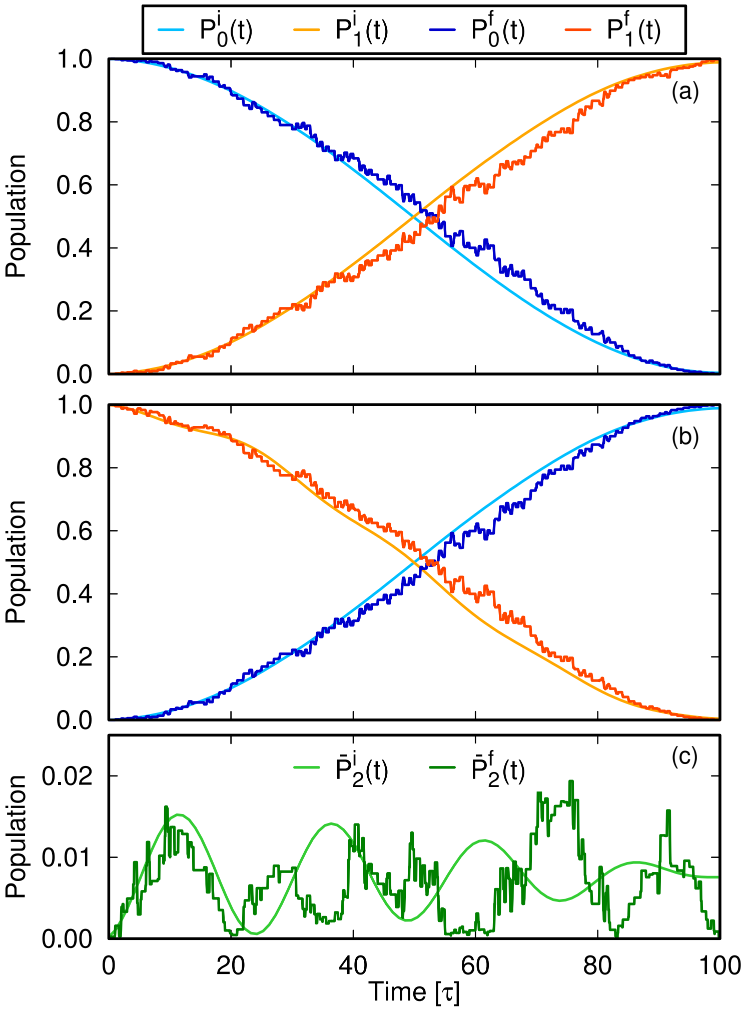

Figure 2.8.: Population dynamics of a U_π gate for the initial (index i) and optimized (index f) pulse sequence, starting in the $|0\rangle$ (a) and $|1\rangle$ state (b). The average leakage into the third level is shown in (c). While the oscillation of the leakage population with $2\pi/|\Delta|$ is apparent in the DANTE sequence, this is distorted in the optimized evolution with the gain of leakage suppression to the final gate time.

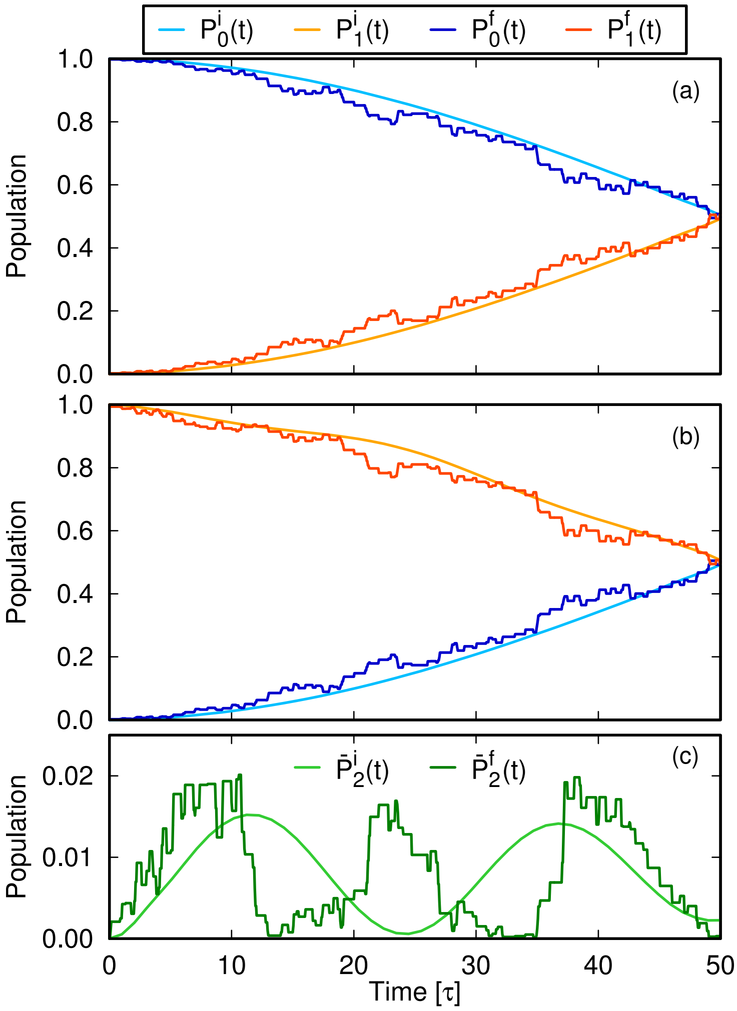

Figure 2.9.: Population dynamics of a $U_{\pi/2}$ gate for the initial (index i) and optimized (index f) pulse sequence, starting in the $|0\rangle$ (a) and $|1\rangle$ state (b). The average leakage into the third level is shown in (c).

simulated annealing algorithm [39]. The only difference left is that there is no global temperature defined that decreases with the number of iterations, and choosing the new sequence only depends on the fitness value, not on the iteration number as in simulated annealing.

Using that algorithm and setting our gate fidelity as the fitness measure for the genomes in the genetic algorithm we found the solutions for the sequence shown in Figure 2.6 and for the populations shown in Figure 2.8 and Figure 2.9. The algorithm mainly corrects for leakage into the third energy level, which always leads to an increase in the number of pulses. This is due to the fact that part of the excitation is pumped into the third level and has to get pumped back. The leakage effect also leads to phase error in the computational subspace. A minor reason is the noideal pumping at high frequencies due to the Gaussian pulse shape. We encounter a broad variation over different runs indicating the presence of an abundance of local traps. This is consistent with the common observation that, in principle, trap-free control landscape [40] develops traps when the space of the available pulse shapes is strongly constrained.

Leakage into the second level is defined as follows: we calculate the average population of being in the second state starting in the states $|0\rangle$ and $|1\rangle$, respectively. That is

$$\begin{aligned}\bar{P}_2(t) &= \frac{1}{2}\Big(\langle 2|U(t)\Pi_0 U(t)^\dagger|2\rangle + \langle 2|U(t)\Pi_1 U(t)^\dagger|2\rangle\Big) \\ &= 1 - \frac{1}{2}\,\mathrm{Tr}\Big\{A(t)^\dagger A(t)\Big\}, \qquad (2.27)\end{aligned}$$

with $A(t) = \Pi_Q U(t) \Pi_Q$ being the time evolution operator projected onto the qubit subspace. The average leakage population is shown in Figure 2.8 and Figure 2.9 as well, with oscillatory behaviour for the DANTE sequence and reduction of the final leakage of the optimized sequence.

2.4.1. Timing errors

The release of SFQ pulses timed by the clock can be distorted by inevitable timing jitter [41]. This leads to small delays of the actual time the pulse is applied. Therefore we expect a decrease in fidelity when such effects are taken into account. Here we show the influence of timing jitter for the optimized sequence for the U_π gate at $t_g = 100\tau$ and the $U_{\pi/2}$ gate at $t_g = 50\tau$.

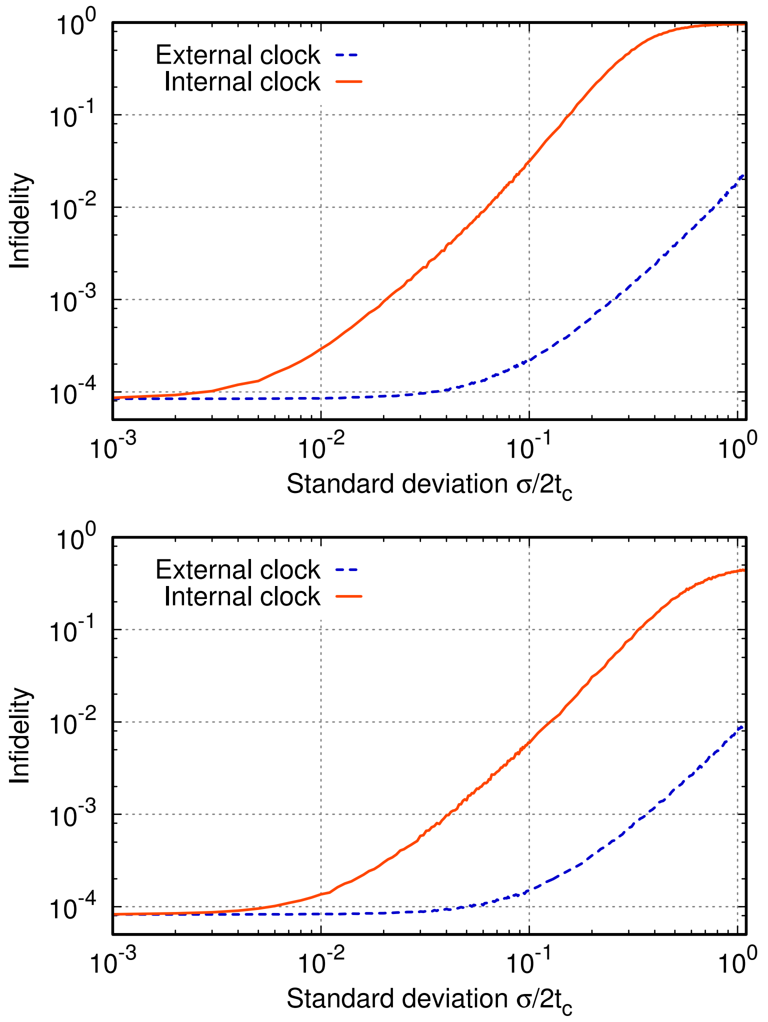

Figure 2.10.: The infidelity of the optimized sequence for a U_π gate (top) and a $U_{\pi/2}$ gate (bottom) and timing errors of the SFQ pulses, with a gate duration of 100τ (top) and 50τ (bottom). The timing jitter of each pulse either is uncorrelated, with constant variance (external clock), or depends on the last applied pulse (internal clock). For each value of σ the infidelities have been averaged over 1000 runs of the time evolution.

To take the jitter errors into account we shift each applied pulse in time by a small amount δt and each pulse U_1 is replaced by U_1', with

$$U_1' = U_0(-\delta t)U_1U_0(\delta t)\,. \tag{2.28}$$

If δt is positive, the pulse reaches the qubit with a delay, while a negative δt leads to a pulse that is applied too early. The values of δt are set randomly by a normal distribution with mean $\mu = 0$ and standard deviation σ. This σ depends on the type of the implemented clock. For an external clock, the standard deviation of each time delay is independent of the preceding pulse. Therefore we can keep it constant. If we implement the clock internally on the chip, each pulse is triggered by its predecessor. For the kth pulse the standard deviation therefore reads $\sqrt{k}\sigma$.

In Figure 2.10 the infidelity for different values of σ is shown for the U_π gate and for the $U_{\pi/2}$ gate. For each value of σ, the fidelity of the time evolution has been averaged over 1000 runs. As can be seen, the external clocking scheme is more robust by an order of magnitude of the standard deviation. It is still within the target fidelity for jitter times within 3 % of the pulse width Δt. For the internal clock, the same condition holds, but with an order of magnitude less robustness, i. e., around 0.3 %. If the jitter time is of the same order as the pulse width, the infidelity is still on the same scale as where we started our optimization, for the external clock. We therefore conclude that an external clock should be used in favour of an internal one for future devices.

2.4.2. Quantum speed limit

Every quantum system undergoes some decoherence, which leads to a loss of information in the system. Typical time scales of decoherence of transmon qubits lie in the range of several microseconds [42]. Therefore it is important to be able to perform as many gates as possible before decoherence comes into play. With the genetic algorithm we search for speed limits for the given set of parameters. The speed limit is the minimal time we need to implement the single-qubit gate with SFQ pulses for a given target fidelity.

In Figure 2.11 we show the speed limit for the system at hand for the U_π gate and the $U_{\pi/2}$ gate, which we obtained through optimization with the genetic algorithm. For each gate time the pulse width is kept constant, and the number of bits decreases with the gate duration. The genetic algorithm stops as soon as we reach a fidelity of $\Phi > 0.9999$ or the maximum number of iterations. We have also included optimizations for the doubled pulse amplitude of $\theta = \pi/50$. The speed limit lies approximately below a gate time of $t_g = 30\tau$, i. e., less than 600 bits for a clock frequency of $f_{\text{clock}} = 20/\tau$, but above the time $t_\Delta = \tau|\Delta|/\omega_{10} = 25\tau$, defined through the anharmonicity

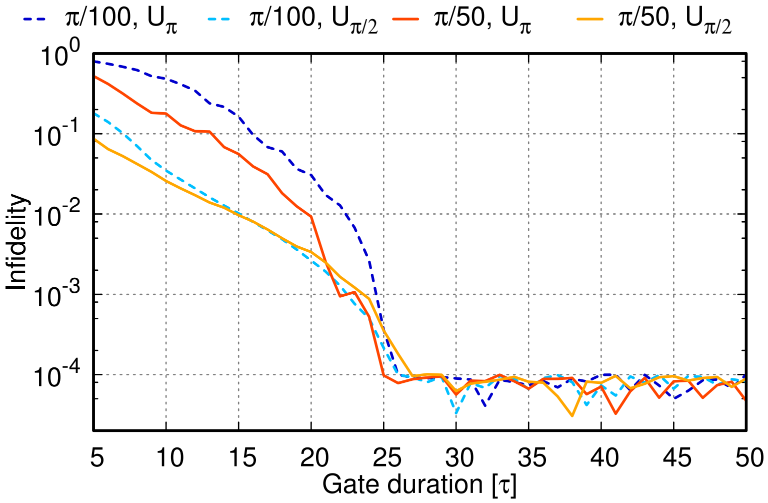

Figure 2.11.: The infidelity for different gate durations τ, with a constant pulse width of $2t_c = \tau/20$. The pulse sequences for the U_π and $U_{\pi/2}$ gates have been optimized with a genetic algorithm, for pulse amplitudes of $\pi/100$ and $\pi/50$. The algorithm halts as soon as a fidelity $\Phi > 0.9999$ has been reached. The data show that doubling the pulse amplitudes from $\pi/100$ to $\pi/50$ does not lead to faster gates. The minimal gate duration required lies below 30τ and is independent of the target gate.

Δ [38]. We also point out that the speed limit is independent of θ and the target gate.

In Figure 2.12 we show the robustness of the four optimized sequences at the approximate speed limit of $t_g = 30\tau$ for an external clock. We see that even at the speed limit the sequences show the same behaviour as they did for longer gate times. Also, driving with higher amplitudes leads to less robust sequences.

2.5. Summary

To conclude, we have successfully developed and applied an optimal control method for pulses with only a single bit of amplitude resolution which is not covered in [10]. Finding the right binary string leads to minimization of the leakage error in the transmon system and thus gate control precision compatible with the requirements of fault-tolerant quantum computing.

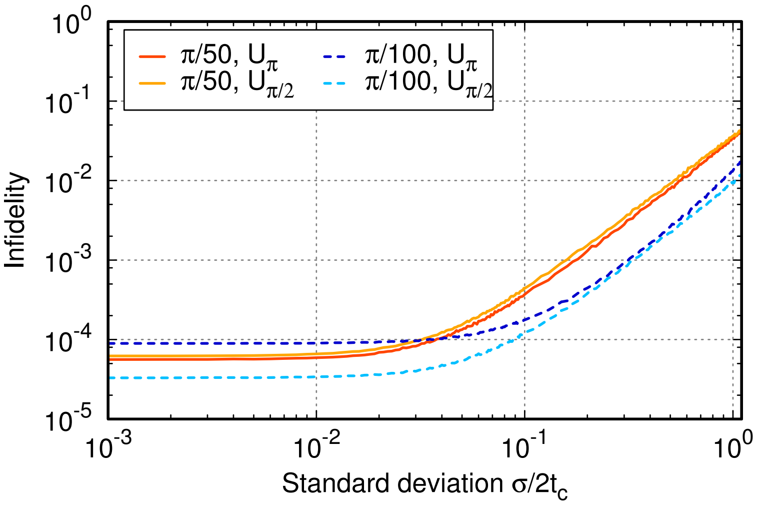

Figure 2.12.: The infidelity of the optimized sequences at a gate duration of $t_g = 30\tau$ for uncorrelated timing errors with standard deviation σ. The pulse sequences with the higher pulse amplitude $\pi/50$ are less robust against timing jitter than the sequences with the lower pulse amplitude $\pi/100$.

The results presented here show a fidelity improvement of several orders of magnitude over equal pulse spacing sequences while being robust under external timing jitter. RSFQ shift registers are needed to perform the optimized sequence and are an essential part of on-chip SFQ qubit control. This makes the underlying SFQ pulse platform together with the single-bit optimal-control theory a possible and attractive candidate for an integrated control layer in a quantum processor.

3. Implementing optimized digital control in RSFQ

The optimization technique presented in the previous chapter allowed us to apply fast and robust single-qubit gates to leakage with just a single bit control resolution. This makes high fidelity single-qubit gates reachable with RSFQ systems. But they also set some conditions for the design parameters. For example it must be possible to drive at high frequencies and to readout and apply pulses at arbitrary time slots. Here we want to use more realistic designs, derived from latest technology, to get closer to a practical single-qubit control realization. We also derive some conditions that analytic pulse sequences should meet and compare it with DANTE.

3.1. Shift registers

We study the influence of the shift register size on the qubit gate. In the most simple case, a shift register can be seen as a temporary storage device that is loaded and read out sequentially. They have been designed and tested successfully in RSFQ [30, 31]. The RSFQ clock sets the times at which a bit is read out, and every other bit is shifted forward by one position. Other types of shift registers exist as well in semiconductor electronics, like serial-in parallel-out (SIPO), parallel-in serial-out (PISO) and parallel-in parallel-out (PIPO), which read in sequentially and read out in parallel, read in parallel and read out sequentially, or read in and out in parallel, respectively. They might be available for RSFQ in the future, too. In the previous chapter, we basically operated a shift register at a frequency of $20/\tau$ and bit sizes ranging from 5 to 2000, with a minimum required size of below 600 bits for the quantum speed limit.

As a first example, we use a smaller frequency of $8/\tau$ and a bit length of 256, which leads to a gate time of $t_g = 32\tau$. We let the target angle α

Some of the ideas and figures presented in this chapter have been published in [43].

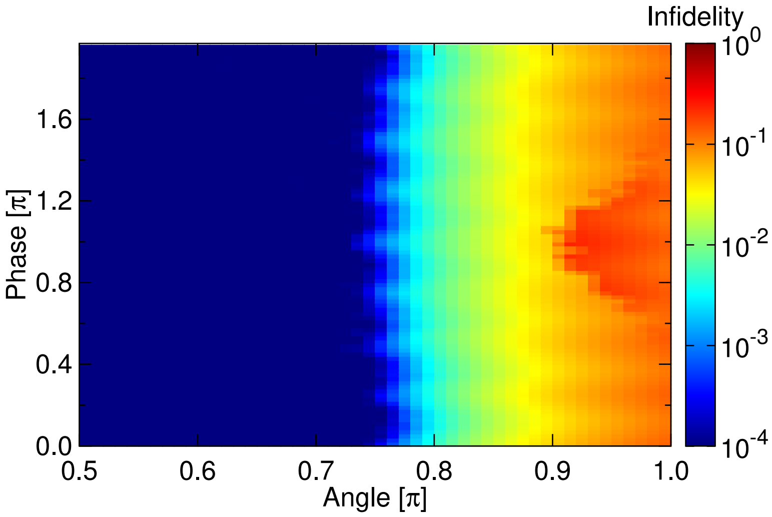

Figure 3.1.: Infidelity for different phases φ and rotation angles α. The shift register size is 256 with a pulse width of $\tau/8$ and a pulse amplitude of $\pi/100$.

and phase φ vary and optimize each sequence up to an infidelity of 10^{-4} if possible. The result is shown in Figure 3.1. The target gate reads

$$U_F = \begin{bmatrix} \cos(\alpha/2) & -(\cos\varphi + i\sin\varphi)\sin(\alpha/2) & 0 \\ (\cos\varphi - i\sin\varphi)\sin(\alpha/2) & \cos(\alpha/2) & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (3.1)$$

i. e., we implement a Y-rotation for $\varphi = 0$ and an X-rotation for $\varphi = \pi/2$. We obtain that there is a maximum rotation angle $\alpha \leqslant 3\pi/4$ and that the reachable fidelity depends on the phase φ, with a minimum for the clockwise Y-rotation. To obtain this data, we have adjusted the mutation rate to $2/n_{\text{pix}}$, where n_{pix} is the shift register size, such that a mutation of a bit string in the genetic algorithm has a Hamming distance of two, i. e., approximately two bits in a bit string are flipped.

3.1.1. Pulse width

Having seen that at low frequency there is a maximal reachable rotation angle, we want to see if there is the possibility to vary the frequency with fixed shift register size to see if we can reach a bigger angle. But first we have to change the target gate. We make the target gate independent of the

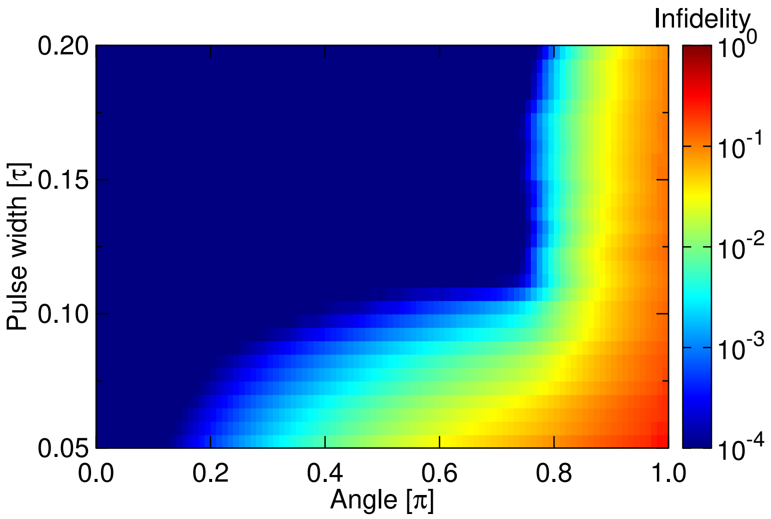

Figure 3.2.: Infidelity for different pulse widths $2t_c$ and rotation angles α for a fixed shift register size of 256 bits. We see that there is a maximal rotation angle and a minimal pixel length, where the latter is proportional to the gate time and is lower bounded by the anharmonicity.

phase φ by restricting it to Y-rotations. The target gate for arbitrary rotation angle reads

$$U_F = \begin{bmatrix} \cos(\alpha/2) & -\sin(\alpha/2) & 0 \\ \sin(\alpha/2) & \cos(\alpha/2) & 0 \\ 0 & 0 & 0 \end{bmatrix} . \tag{3.2}$$

Now, in a second step, we point out that if we want to implement the identity operation, the simplest case would be to not apply a pulse at all. However, for varying pulse widths and fixed shift register size we have gate times that are $t_g \neq n\tau$, with $n \in \mathbb{N}$. This leads to a decrease of the fidelity, which can be seen by writing out the fidelity function explicitly

$$\begin{aligned} \Phi &= \frac{1}{4}|\mathrm{tr}\, U_F^\dagger U_T|^2 = \frac{1}{4}|\langle 0|U_F^\dagger U_T|0\rangle + \langle 1|U_F^\dagger U_T|1\rangle|^2 \\ &= \frac{|\langle 0|U_F^\dagger U_T|0\rangle|^2 + |\langle 1|U_F^\dagger U_T|1\rangle|^2}{4} \\ &+ \frac{1}{2}\cos(\varphi_0 - \varphi_1)\,|\langle 0|U_F^\dagger U_T|0\rangle|\,|\langle 1|U_F^\dagger U_T|1\rangle| \,, \end{aligned} \tag{3.3}$$

with the phases φ_0 and φ_1 of $\langle 0|U_F^\dagger U_T|0\rangle$ and $\langle 1|U_F^\dagger U_T|1\rangle$, respectively. The fidelity function we are using is sensitive to relative phases between the

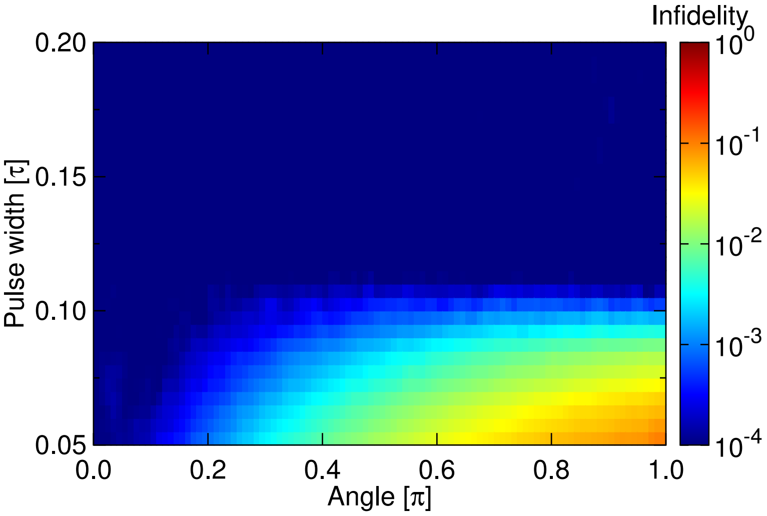

Figure 3.3.: Infidelity for different pulse widths $2t_c$ and rotation angles α for a fixed shift register size of 256 bits, but a pulse amplitude of $\pi/50$. In this case the rotation angle is only limited by the gate time, i. e., reasonable low frequencies.

levels in the computational subspace. Now setting $\alpha = 0$, i. e., setting the target to the identity, and $U_T = U_0(t_g)$, we get

$$\Phi = \frac{1}{2}\left(1 + \cos\left(2\pi t_g/\tau\right)\right) = \cos^2\left(\pi t_g/\tau\right) , \tag{3.4}$$

and we see that the fidelity oscillates with the chosen gate time. To correct for this effect we are using the following target gate instead which we obtain by multiplying the original target gate with the free evolution operator

$$U_F^* = \exp\left(-iH_0 t_g\right) U_F \, . \tag{3.5}$$

This is equivalent to implement the target gate in a frame rotating with ω_{10}. Then we just have to keep track of the gate times and waiting times between gates, i. e., correct the phase errors in software.

Again we use a fixed size shift register with 256 bits and amplitude $\pi/100$. We let the pixel duration vary from $\tau/20$ to $\tau/5$. As can be seen in Figure 3.2, there is the same limit for the maximal rotation angle we obtained from Figure 3.1, which lies around $\alpha \leqslant 3\pi/4$. Also, a lower driving frequency does not have a major effect on the maximal rotation angle. However, at frequencies that are too high the gate time falls below the critical value

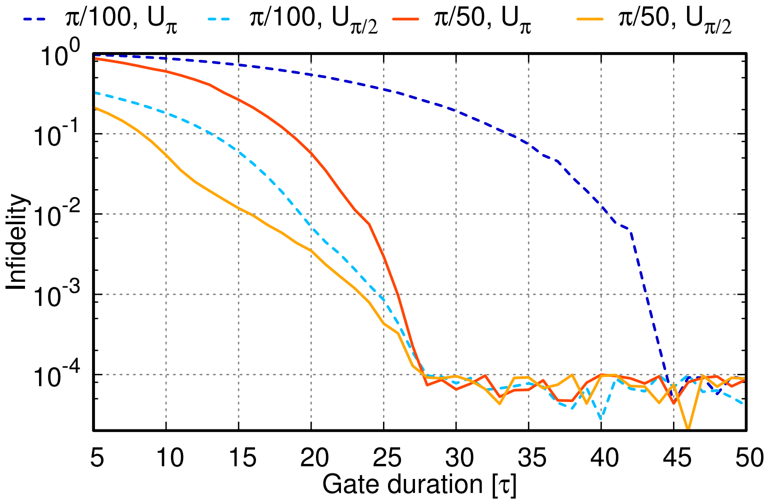

Figure 3.4.: Speed limit for the low frequency clock with $\nu = 8/\tau$. Except for the U_π gate with pulse amplitude $\pi/100$, the speed limit is still below 30τ.

defined through the anharmonicity, i. e., $t_g < 2\pi/|\Delta|$. We conclude from the data that for a pulse amplitude of $\pi/100$ we have to perform two $U_{\pi/2}$ gates to get a U_π gate with a 256 bit shift register. However, higher pulse amplitudes do not need high frequency pulses trains, and the gate time sets the only limit, as we can see in Figure 3.3. But, as we have discussed in the previous chapter, these are less robust compared to the $\pi/100$ pulses.

3.1.2. Shift register size

Now we analyse the shift register size depending on the reported speed limit. Therefore, we use three default values for the pulse width, namely $\tau/20$, $\tau/8$ and $\tau/5$. For each we optimize the sequence up to the target fidelity to find a speed limit for the pulse amplitudes $\pi/50$ and $\pi/100$ and the two target gates U_π and $U_{\pi/2}$. The results are shown in Figure 2.11, Figure 3.4 and Figure 3.5. From the numerical optimization we obtain how the constraint to lower frequency components limits the speed limits for the different gates and pulse amplitudes. While for frequency components with $20/\tau$ the speed limits where the same, we already see that for a maximal frequencies of $8/\tau$ pulse with amplitude $\pi/100$ drastically increase the gate time for the U_π gate. For a maximum frequency of $5/\tau$, we can achieve the

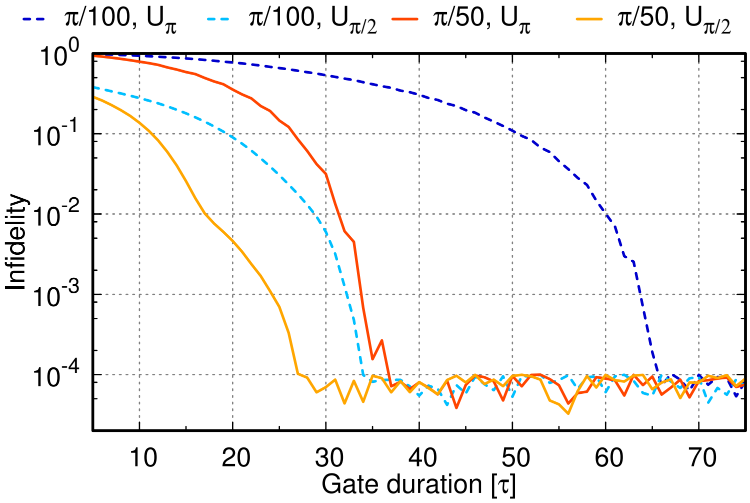

Figure 3.5.: Speed limit for a clock frequency of $\nu = 5/\tau$. The $U_{\pi/2}$ gate for the high amplitude $\pi/50$ is still below 30τ, while the $U_{\pi/2}$ gate with pulse amplitude $\pi/100$ and the U_π gate with pulse pulse amplitude $\pi/50$ have approximately the same speed limit, and the U_π gate with pulse amplitude $\pi/100$ is shifted to even further to higher gate times.

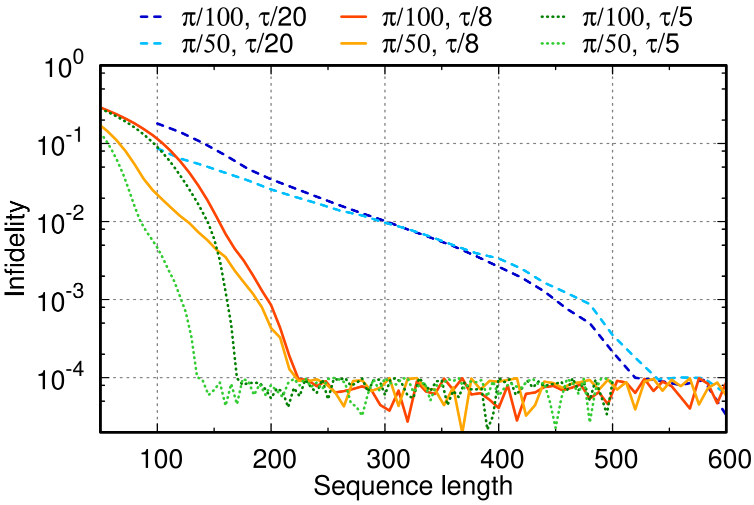

Figure 3.6.: Speed limit for the $U_{\pi/2}$ gate and three different clock frequencies in dependence of the shift register size. Lower frequencies allow for shorter shift registers, trading off increased gate times.

anharmonicity limited gate time only for the high amplitude $\pi/50$ pulses for the $U_{\pi/2}$ gate. From these we conclude that frequency limited sequences can be compensated by higher pulse amplitudes, but the anharmonicity limit always holds.

In Figure 3.6 we see that for the high frequency gates we need much longer shift registers, while low frequency optimized gates can be accomplished with smaller shift registers. Here we have only considered the $U_{\pi/2}$ gate. For smaller pulse amplitudes the shift register needs to be larger, if this effect is not compensated by higher frequencies, which then are the limiting factor for minimal shift register sizes. Therefore, there is a trade-off between short shift registers operated at low frequencies and short gate times. From the data, a shift register of 256 bits, driven at a maximum frequency of $8/\tau$ and a pulse amplitude of either $\pi/50$ or $\pi/100$ seems to be a good compromise.

3.1.3. Power consumption

Another point we have to address is the number of pulses we apply. Obviously, the number of pulses applied in the DANTE sequence per time period τ is one. However, for the optimized sequences we use more pulse per cycle, on the one hand to correct for errors in the unitary gate, on the other hand for decreasing the gate time. In Figure 3.7 we show the number of pulses per cycle time τ in dependence of the gate time for the optimized sequences, for both the U_{π} and the $U_{\pi/2}$ gate. For short pulse widths $\tau/20$ the average number of pulses slightly decreases with the gate time, while the number of pulses is usually higher than for longer pixel durations. For the latter the average number of pulses per cycle is almost independent of the gate time, showing only a slight decrease for the $U_{\pi/2}$ gate. Also the average number of pulses does not depend on the pulse amplitude. Therefore, limiting the maximal clock frequency helps to keep the average pulse number low and reducing the overall power consumption of the circuit.

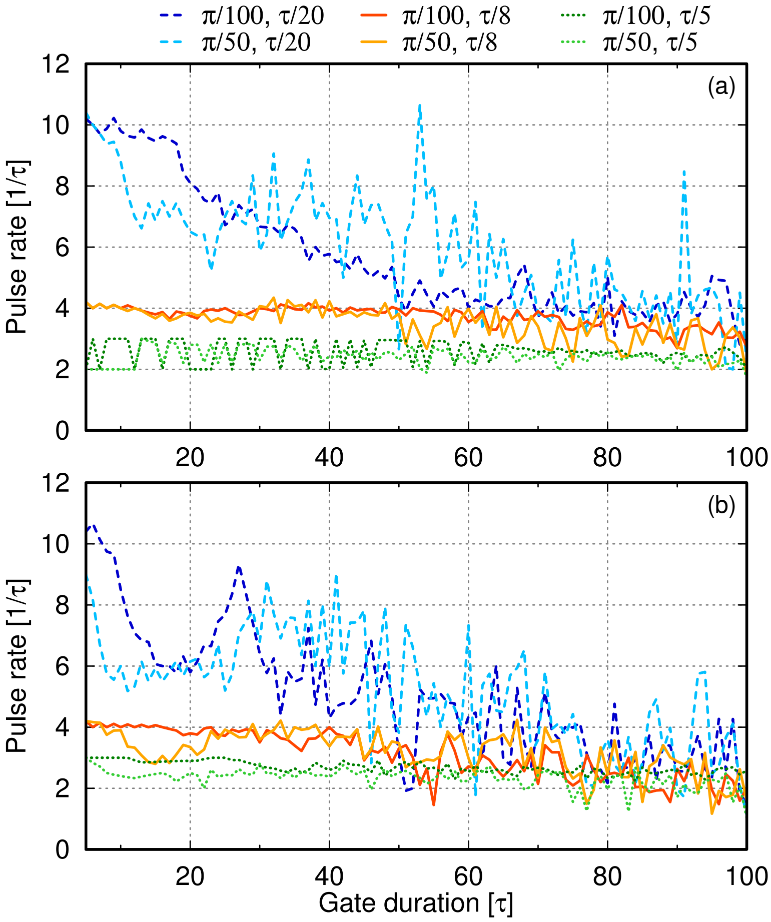

Figure 3.7.: Average number of pulses for (a) the U_π gate and (b) the $U_{\pi/2}$ gate. While short pulse widths allow for a higher number of pulses, for smaller frequencies the number of pulses is almost independent of the gate time.

3.2. Higher level leakage

So far we have only taken a single leakage level as error source into account, which we where able to correct with numerical optimized pulse sequences. Now we want to see the impact if we include higher levels into our model. For this we include the next two energy levels in the Duffing oscillator approximation of the transmon qubit. Therefore, the anharmonicity is constant

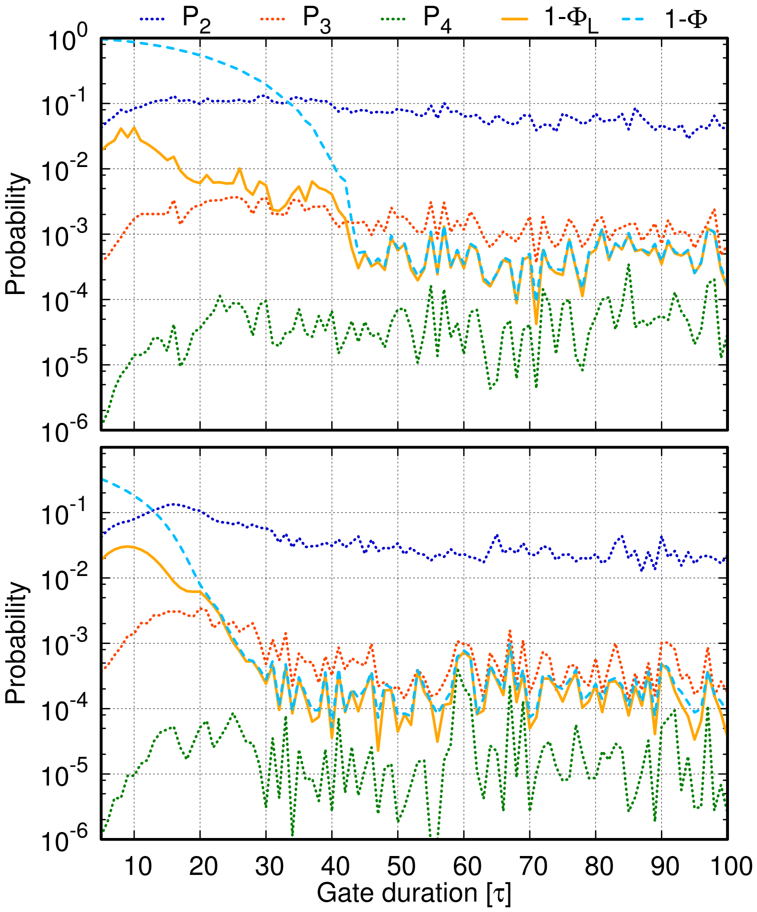

Figure 3.8.: The maximal obtainable population of each leakage level, depending on the gate duration, with a pulse amplitude of $\theta = \pi/100$ for the U_π gate (top) and the $U_{\pi/2}$ gate (bottom). Additionally, the average leakage error $1 - \Phi_L$ and the infidelity $1 - \Phi$ are shown.

and the free evolution Hamiltonian reads

$$\begin{aligned} H_0 &= \left(\omega_0 - \frac{\Delta}{2}\right) a^\dagger a + \frac{\Delta}{2}\left(a^\dagger a\right)^2 \\ &\approx \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & \omega_0 & 0 & 0 & 0 \\ 0 & 0 & 2\omega_0 + \Delta & 0 & 0 \\ 0 & 0 & 0 & 3\omega_0 + 3\Delta & 0 \\ 0 & 0 & 0 & 0 & 4\omega_0 + 6\Delta \end{bmatrix}, \end{aligned} \tag{3.6}$$

and the control Hamiltonian is

$$H_1 \sim i(a^\dagger - a) \approx i \begin{bmatrix} 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & -\sqrt{2} & 0 & 0 \\ 0 & \sqrt{2} & 0 & -\sqrt{3} & 0 \\ 0 & 0 & \sqrt{3} & 0 & -2 \\ 0 & 0 & 0 & 2 & 0 \end{bmatrix} . \tag{3.7}$$

We are using the same parameters as before, with an anharmonicity of $\Delta = -\omega_{10}/25$ and a pixel length of $\tau/8$. In Figure 3.8 we show the maximum population of each leakage level for the low pulse amplitude $\pi/100$, and in Figure 3.9 for the pulse amplitude of $\pi/50$. The maximum population is defined as the highest probability a state can be obtained in during the gate operation. Additionally, we show the average leakage error, which is defined as

$$1 - \Phi_L = 1 - \frac{1}{2} \mathrm{Tr}\left\{ A^\dagger A \right\} \qquad A = \Pi_Q U_T \Pi_Q \, , \tag{3.8}$$

and the infidelity. Comparing the figures we deduce that a smaller pulse amplitude reduces the amount of leakage into higher levels. This makes them preferable for robust gate implementations. For the given parameters, the infidelity for the larger pulse amplitude is one order of magnitude higher. Additionally, the U_π gate is more prone to leakage errors compared to the $U_{\pi/2}$ gate.

We now use the optimized sequences for the U_π and $U_{\pi/2}$ gate for both pulse amplitudes, add the two extra levels and rerun the genetic algorithm again to get a high fidelity gate for more than just one leakage level. The results are shown in Figure 3.10. One obtains that including higher levels shifts the speed limit slightly to longer gate times. In principle, higher leakage levels are not a general problem, as long as the chosen gate time is set to spare some buffer to take higher level distortions into account.

3.3. A complete set of single qubit gates

Here we show how to implement the basic single qubit gates with RSFQ. These are the Hadamard gate, the phase gate and the T gate [1]. Let us start with the Hadamard gate

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{\sqrt{2}} (Z + X) = \frac{1}{\sqrt{2}} (I - iY) \, Z \, . \tag{3.9}$$

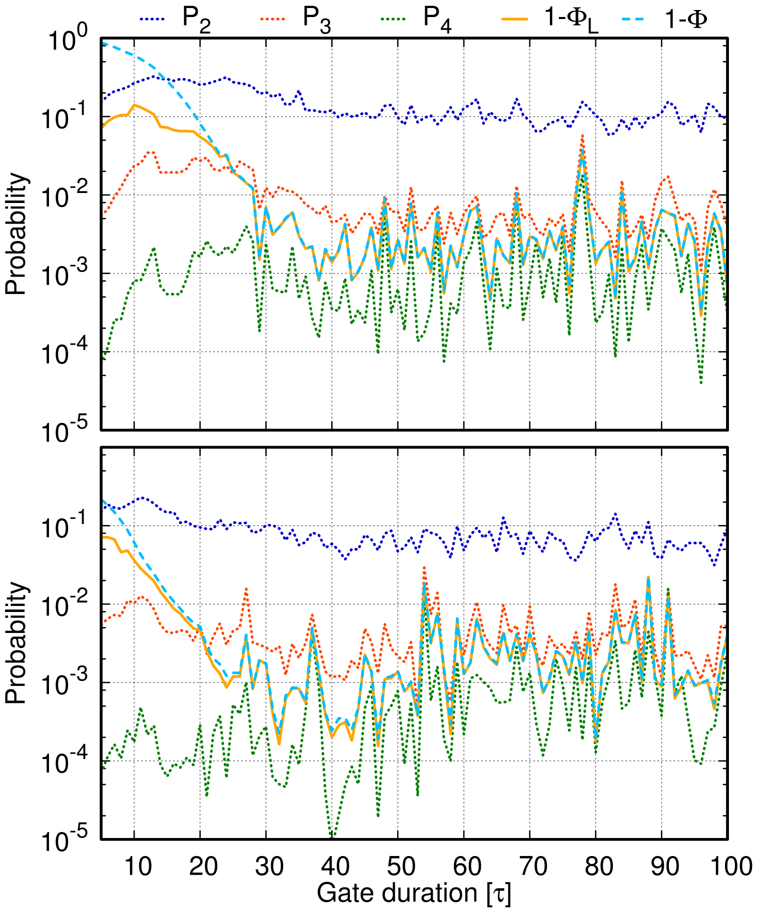

Figure 3.9.: The maximal obtainable population of each leakage level, depending on the gate duration, with a pulse amplitude of $\theta = \pi/50$ for the U_π gate (top) and the $U_{\pi/2}$ gate (bottom). Additionally, the average leakage error $1 - \Phi_L$ and the infidelity $1 - \Phi$ are shown.

As can be seen, it can be decomposed into a Z-gate, followed by a $R_y(\pi/2)$ rotation. The latter we have shown how to implement with the 256 bit – $8/\tau$ RSFQ architecture. The Z gate can be written in terms of a Z-rotations

$$Z = iR_z(\pi) . \tag{3.10}$$

Writing the free evolution operator for arbitrary durations t leads to

$$U_0(t) = e^{-i\omega_{10}t/2} R_z(-\omega_{10}t) . \tag{3.11}$$

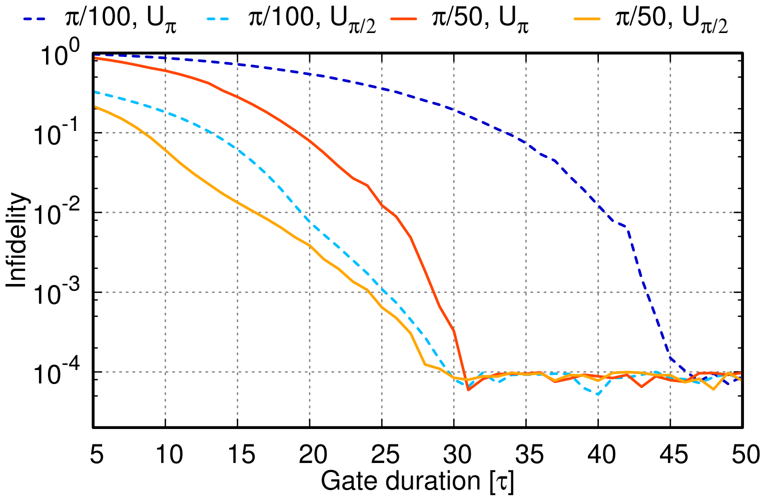

Figure 3.10.: Speed limit for the U_π and $U_{\pi/2}$ gate with the five lowest levels of the transmon. Due to higher level leakage the speed limit is slightly shifted to longer gate times.

Setting $\omega_{10}t = \pi$, i. e., $t = \tau/2$ leads to

$$U_0 = -iR_z(-\pi) = Z\,, \tag{3.12}$$

Therefore we need 256+4 bits, with the first four bits set to zero and the other 256 bits set by the optimization. Implementing the phase gate and the T gate works similar, with

$$S = e^{i\pi/4}R_z(\pi/2)\ , \tag{3.13}$$

for $\omega_{10}t = 3\pi/2$, i. e., $t = 3\tau/4$, and

$$T = e^{i\pi/8}R_z(\pi/4)\ , \tag{3.14}$$

for $\omega_{10}t = 7\pi/4$, i. e., $t = 7\tau/8$. We have to point out that this is only feasible due to the fact that the pixel duration is a multiple of eight times the qubit frequency. For other clock frequencies, where a certain number of pixels represent exactly one period, Z gate, phase gate and T gate are not directly implementable. For example, the clock frequency $20/\tau$ allows no direct T gate implementation with a shift register, and the clock frequency $5/\tau$ does not even allow a direct implementation of the Z gate. These gates can only be performed by waiting the correct amount of time, and therefore need more bookkeeping of the gate and waiting times.

Instead of implementing the family of Z-rotations through the free evolution operator, we can use an X- and Y-rotations instead. An arbitrary X rotation can be performed by

$$R_x(\theta) = S^{-1} R_y(\theta) S\,, \tag{3.15}$$

i. e., the X gate is just one precession period longer than the Y gate, if the shift register bit-string is only optimized for the Y gate. Now we can decompose the T gate into

$$T = e^{i\pi/8} R_y\left(\pi/2\right) R_x\left(-\pi/4\right) R_y\left(-\pi/2\right)\,. \tag{3.16}$$

For a 256 bit shift register and a clock frequency of $8/\tau$, the T gate can therefore be implemented in less than 100τ.

3.4. Spectral decomposition

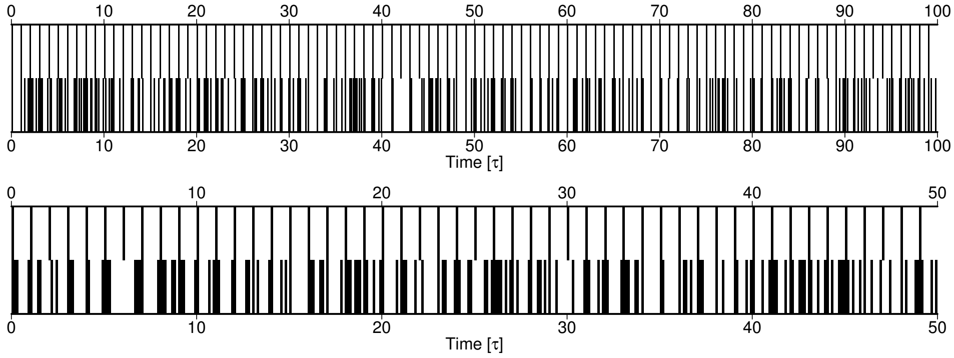

Figure 3.11.: Bit-strings for the U_π and $U_{\pi/2}$ gate at 100τ and 50τ, respectively. The clock frequency is $8/\tau$.

To get a better idea of the pulse sequence, we want to check the spectrum of it to see if we can deduce any useful information to find an analytic description. The initial and optimized bit-strings for the U_π and the $U_{\pi/2}$ gate are shown in Figure 3.11, for a clock frequency of $8/\tau$. Their spectrum is shown in Figure 3.12. Additionally, we show the bit-string and Fourier spectrum for the 32τ $U_{\pi/2}$ gate in Figure 3.12. As can be seen, the optimization does

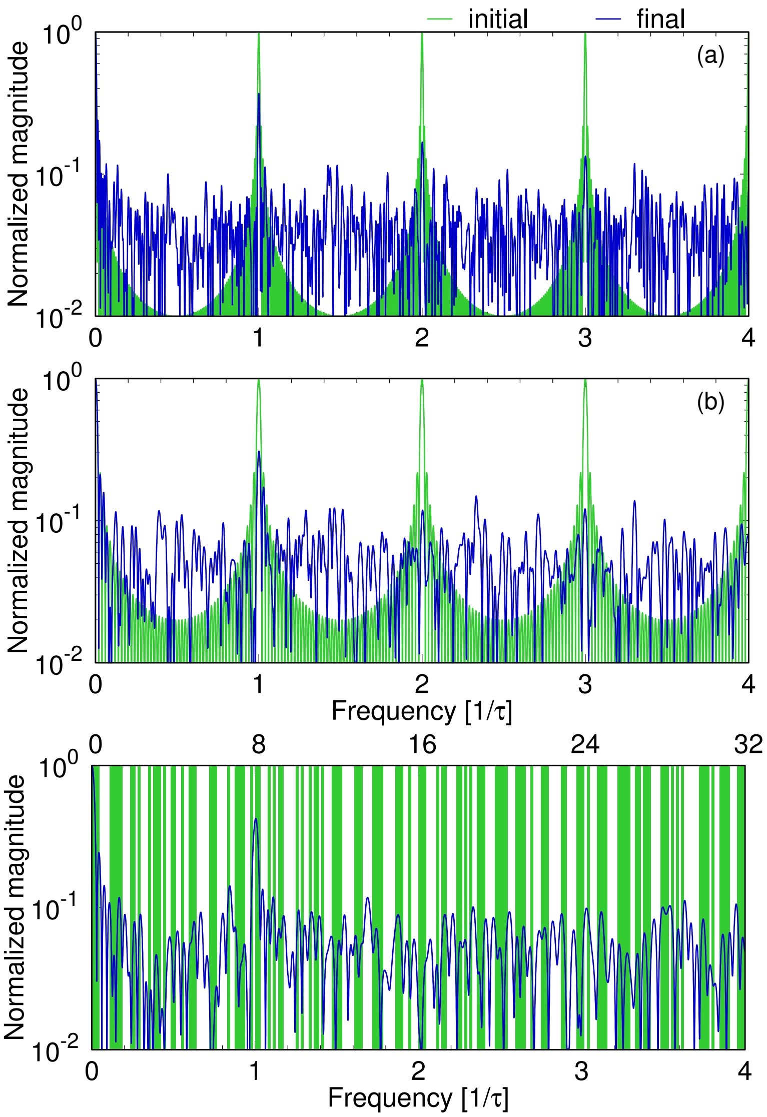

Figure 3.12.: Spectrum for the U_π gate (a) and $U_{\pi/2}$ gate (b), at 100τ and 50τ, respectively. The bottom plot shows the optimized bit-string and spectrum for the 32τ optimized $U_{\pi/2}$ gate, all for a clock frequency of 8/τ

not follow a specific pattern, such that the amplitude of the higher harmonics of the driving frequency are decreasing but still present in the spectrum, while other frequency components do not show any structure at all. The only frequency that that can be easily identified is the qubit frequency 1/τ.

It is therefore easier to use shift register instead of multi-frequency pulse sequences.

3.5. Magnus expansion

So far we have optimized the pulses sequences only numerically. While this works if we use shift registers, we could gain some insight into the dynamics if we could come up with an analytical solution. While the latter we cannot provide and this leaves some work to be done in the future, at least we can formulate the problem in a way to get some conditions the analytic sequence should satisfy. In this section we assume that the we only apply

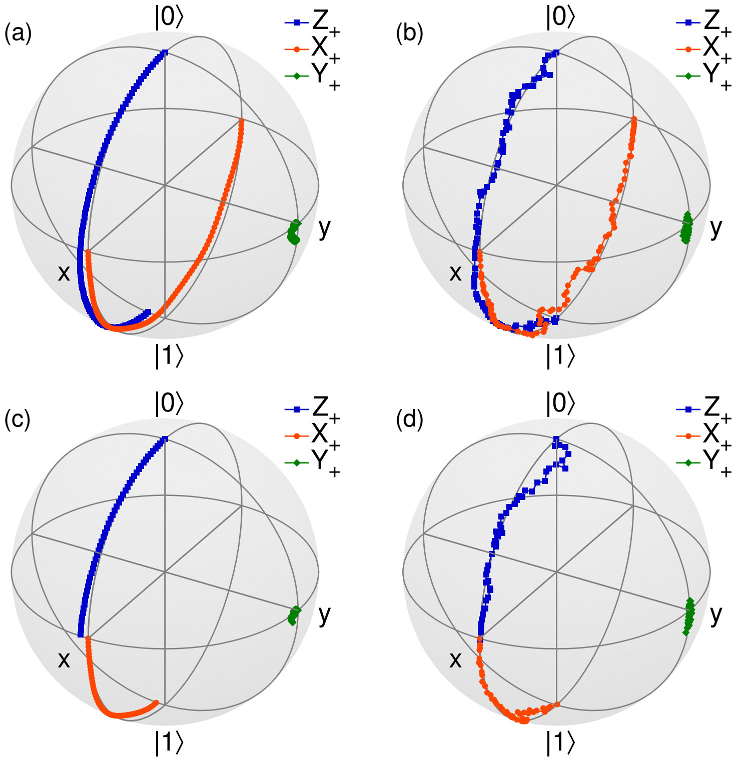

Figure 3.13.: Bloch dynamics for the U_π gate before (a) and after (b) optimisation, in a frame rotating with ω_{10} and a gate time of 100τ. For the $U_{\pi/2}$ gate (c) shows the dynamics before and (d) after optimisation for a gate time of 50τ. The three input states are sampled with period τ, and the shift register frequency is $8/\tau$.

hard pulses, although the derived method could be in principle extended for soft pulses.

We start by analysing the Bloch dynamics for the U_π and $U_{\pi/2}$ gate for the gate times $t_g = 100\tau$ and $t_g = 50\tau$, respectively. Figure 3.13 shows the time evolution for the three input states $|0\rangle$, $(|0\rangle + |1\rangle)/2$ and $(|0\rangle + i\,|1\rangle)/2$, i. e., the eigenstates of the Pauli operators with eigenvalue 1, which are noted as Z_+, X_+ and Y_+. The time evolution is sampled with a rate $1/\tau$, and the numerically optimized solutions are shown as well. As we can see from the DANTE sequence the rotation axis is slightly tilted away from the y-axis to the negative z-direction. Next to the population leakage into the second level this is the other major error the genetic algorithm optimizes away. As an example, the optimal solution for $8/\tau$ lets the Y_+ state also explore its surroundings before returning back into the initial state. The states Z_+ and X_+ do not follow a straight line on the sphere but deteriorate around the ideal line, but end up in the right target state. Obviously, the optimized solution appears to introduce some randomness in the pulse sequence to average out the effects of leakage errors.

3.5.1. Effective Hamiltonian

To analyse the origin of these errors, we use the technique of multiple pulse decoupling [44]. But instead of writing an effective Hamiltonian for the free evolution, we do seek for an effective Hamiltonian for the pulse. We note that the time evolution operator of an arbitrary sequence containing N pulses can always be written as

$$U_T = U_0(t_N - t_{N-1})U_1U_0(t_{N-1} - t_{N-2})\dots U_0(t_1 - t_0)U_1U_0(t_0)\,, \tag{3.17}$$

with $t_i - t_{i-1}$ being the time difference between between pulse i and $i+1$ (counting the pulses from 1 to N). We can expand the equation above by noting that we can replace each free evolution operators with

$$U_0(t_i - t_{i-1}) = U_0(t_i)U_0^{-1}(t_{i-1})\,. \tag{3.18}$$

This leads to the following time evolution operator of an arbitrary sequence

$$\begin{aligned} U_T &= U_0(t_N)U_0^{-1}(t_{N-1})U_1U_0(t_{N-1})\dots U(t_1)U^{-1}(t_0)U_1U(t_0) \\ &= U_0(t_N)\bar{U}_{1,N-1}\dots\bar{U}_{1,0} = U_0(t_N)\prod_{n=N-1}^{0}\bar{U}_{1,n}\,, \end{aligned} \tag{3.19}$$

with the pulses in the interaction frame

$$\bar{U}_{1,n} = U_0^{-1}(t_n)\, U_1 U_0(t_n)\ . \tag{3.20}$$

Basically, we compute the pulse sequence in the interaction frame and then go back to the lab frame with the final rotation $U_0(t_N)$. The final rotation can be omitted if we optimize the gate in the frame rotating with the qubit frequency, since we measure the fidelity in the computational subspace and allow for arbitrary phases of the leakage level. The Hamiltonian of the pulse in the interaction frame is

$$\begin{aligned} \bar{U}_{1,n} &= \exp\left(-i\bar{H}_{1,n}\Delta t\right) \\ \bar{H}_{1,n} &= U_0^{-1}(t_n)\, H_1 U_0(t_n)\ , \end{aligned} \tag{3.21}$$

with $H_1 = \Omega_0 Y/2$ and $\theta = \Omega_0 \Delta t$. While this is up to the hard pulse approximation exact, we want to find an effective Hamiltonian such that the we can write

$$\prod_{n=N-1}^{0} \bar{U}_{1,n} = \exp\left(-iH_{\text{eff}} N\Delta t\right)\ . \tag{3.22}$$

The Magnus expansion [45] allows to get the effective Hamiltonian as a power series of time-ordered commutators of the Hamiltonian $\bar{H}_{1,n}$, such that

$$H_{\text{eff}} = \sum_{k=1} H_{\text{eff}}^{(k)}\ . \tag{3.23}$$

We only go to second order in the Magnus expansion, although we should be aware that it is not guaranteed that the Magnus expansion converges. The first to two orders of the Magnus expansion read

$$\begin{aligned} H_{\text{eff}}^{(1)} &= \frac{1}{N} \sum_{n=0}^{N-1} \bar{H}_{1,n} \\ H_{\text{eff}}^{(2)} &= \frac{-i\Delta t}{2N} \sum_{m=1}^{N-1} \sum_{n=0}^{j-1} \left[\bar{H}_{1,m}, \bar{H}_{1,n}\right]. \end{aligned} \tag{3.24}$$

The first order is the time average of the interaction Hamiltonian and the second order is the time average of the ordered commutators. From these we want to derive some conditions that an analytic solution should match, such that the pulse sequence works perfectly in second order. For the qutrit the interaction Hamiltonian reads

$$\bar{H}_{1,n} = -\frac{i\Omega_0}{2}\begin{bmatrix} 0 & e^{-i\omega_{10}t_n} & 0 \\ -e^{i\omega_{10}t_n} & 0 & \sqrt{2}e^{-i\omega_{21}t_n} \\ 0 & -\sqrt{2}e^{i\omega_{21}t_n} & 0 \end{bmatrix}. \tag{3.25}$$

Therefore the effective first order Hamiltonian is

$$H_{\text{eff}}^{(1)} = -\frac{i\Omega_0}{2N}\sum_{j=0}^{N-1}\begin{bmatrix} 0 & e^{-i\omega_{10}t_j} & 0 \\ -e^{i\omega_{10}t_j} & 0 & \sqrt{2}e^{-i\omega_{21}t_j} \\ 0 & -\sqrt{2}e^{i\omega_{21}t_j} & 0 \end{bmatrix}. \tag{3.26}$$

In first order we derive the following two conditions:

$$\sum_{m=0}^{N-1} e^{-i\omega_{10}t_m} = \frac{\alpha}{\theta}e^{i\varphi}, \qquad \sum_{m=0}^{N-1} e^{-i\omega_{21}t_m} = 0, \tag{3.27}$$

with the number of pulses N, the rotation angle θ of each pulse and the target angle α. We allow a phase φ to occur in the first condition, such that we can perform rotations around an arbitrary axis on the equator of the Bloch sphere. The first condition leads to the desired rotation with the right angle, and the second condition ensures that no transition from the first to the second level occurs. If both conditions hold the time evolution operator in first order of the effective Hamiltonian reads

$$e^{-iH_{\text{eff},1}N\Delta t} = \begin{bmatrix} \cos(\alpha/2) & -e^{i\varphi}\sin(\alpha/2) & 0 \\ e^{-i\varphi}\sin(\alpha/2) & \cos(\alpha/2) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.28}$$

which for $\varphi = 0$ is the rotation around the y-axis and for $\varphi = \pi/2$ around the x-axis. For the second order effective Hamiltonian we have to calculate the commutators of $\bar{H}_{1,n}$, which are

$$\begin{aligned} &\left[\bar{H}_{1,m}, \bar{H}_{1,n}\right] = \\ &\frac{\Omega_0^2}{4}\begin{bmatrix} -2i\sin\left(\omega_{10}(t_m - t_n)\right) & 0 & -\sqrt{2}e^{-i(\omega_{10}t_m+\omega_{21}t_n)} \\ 0 & 2i\sin\left(\omega_{10}(t_m - t_n)\right) & 0 \\ \sqrt{2}e^{i(\omega_{10}t_m+\omega_{21}t_n)} & 0 & 0 \end{bmatrix} \\ &+\frac{\Omega_0^2}{4}\begin{bmatrix} 0 & 0 & \sqrt{2}e^{-i(\omega_{10}t_n+\omega_{21}t_m)} \\ 0 & -4i\sin\left(\omega_{21}(t_m - t_n)\right) & 0 \\ -\sqrt{2}e^{i(\omega_{10}t_n+\omega_{21}t_m)} & 0 & 4i\sin\left(\omega_{21}(t_m - t_n)\right) \end{bmatrix}. \end{aligned} \tag{3.29}$$

Then we get from equation 3.24 the following three conditions:

$$\sum_{m=1}^{N-1}\sum_{n=0}^{m-1}\sin(\omega_{10}(t_m - t_n)) = \sum_{m=1}^{N-1}\sum_{n=0}^{m-1}\sin(\omega_{21}(t_m - t_n)) = 0$$
$$\sum_{m=1}^{N-1}\sum_{n=0}^{m-1} e^{-i(\omega_{10}t_m+\omega_{21}t_n)} = \sum_{m=1}^{N-1}\sum_{n=0}^{m-1} e^{-i(\omega_{21}t_m+\omega_{10}t_n)}\,. \tag{3.30}$$

The first two conditions ensure that there is no rotation axis tilt in the z-direction, and the last condition prevents direct transitions from the ground state to the second level. Strictly speaking, the average over the ordered commutators has to vanish, such that $H^{(2)}_{\text{eff}} = 0$. We should mention that the third order effective Hamiltonian has only non-vanishing coefficients for $|0\rangle\langle 1|$, $|1\rangle\langle 0|$, $|1\rangle\langle 2|$ and $|2\rangle\langle 1|$, the same case as for $H^{(1)}_{\text{eff}}$. This pattern continues for higher orders, so if we would take all orders into account the sum over the effective even Hamiltonians has to vanish,

$$\sum_{i=1} H^{(2i)}_{\text{eff}} = 0\,, \tag{3.31}$$

and for the sum over the effective odd Hamiltonians we get

$$\sum_{i=0}\langle 0|H^{(2i+1)}_{\text{eff}}|1\rangle = \frac{\alpha}{\theta}e^{i\varphi} \qquad \sum_{i=0}\langle 1|H^{(2i+1)}_{\text{eff}}|2\rangle = 0\,. \tag{3.32}$$

However, as we have mentioned earlier, we approximate the effective Hamiltonian only to second order in the Magnus expansion.

3.5.2. The DANTE sequence

Now we want to see how well the DANTE sequence performs in terms of the Magnus expansion. Therefore we set the times to $t_n = mn\tau$ and the rotation angle to $\theta = \Omega_0\Delta t = \alpha/N$, where we allow for lower driving frequencies by setting the free evolution time to $m\tau$, with $m \in \mathbb{N}$. The effective Hamiltonian in first order reads

$$H^{(1)}_{\text{eff}} = \frac{-i\alpha}{2N^2\Delta t}\begin{bmatrix} 0 & N & 0 \\ -N & 0 & \sqrt{2}\frac{1-e^{-2\pi i\delta N}}{1-e^{-2\pi i\delta}} \\ 0 & -\sqrt{2}\frac{1-e^{2\pi i\delta N}}{1-e^{2\pi i\delta}} & 0 \end{bmatrix}, \tag{3.33}$$

with the dimensionless anharmonicity $\delta = \Delta/\omega_{10}$. Choosing $\delta = -1/25$ and setting $N = 25, 50, \ldots$ lets the coupling terms between the first and second state vanish, and so in first order of the Magnus expansion the DANTE sequence works perfectly. We also obtain that the oscillation of the fidelity in Figure 2.3 are proportional to $1/m\delta$. The unitary operator is the desired Y-rotation

$$e^{-iH_{\text{eff},1}N\Delta t} = \begin{bmatrix} \cos(\alpha/2) & -\sin(\alpha/2) & 0 \\ \sin(\alpha/2) & \cos(\alpha/2) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.34}$$

Note that if we drive at times such that $\delta m \in \mathcal{N}$ holds, we also drive the second transition resonantly. The commutator for the DANTE sequence reads

$$\left[\bar{H}_{1,m}, \bar{H}_{1,n}\right] = \tag{3.35}$$
$$\frac{\Omega_0^2}{4}\begin{bmatrix} 0 & 0 & \sqrt{2}e^{-i2\pi\delta m} - \sqrt{2}e^{-i2\pi\delta n} \\ 0 & -4i\sin\left(2\pi\delta(m-n)\right) & 0 \\ \sqrt{2}e^{i2\pi\delta n} - \sqrt{2}e^{i2\pi\delta m} & 0 & 4i\sin\left(2\pi\delta(m-n)\right) \end{bmatrix}.$$

Then we get for the effective second order Hamiltonian

$$H_{\text{eff},2} = \frac{-i\alpha^2}{8N^3\Delta t}$$
$$\times \begin{bmatrix} 0 & 0 & 2\sqrt{2}\frac{1-e^{-i2\pi\delta N}-N+Ne^{-i2\pi\delta}}{(1-e^{-i2\pi\delta})^2} \\ 0 & 2i\frac{\sin(2\pi\delta N)-N\sin(2\pi\delta)}{1-\cos(2\pi\delta)} & 0 \\ 2\sqrt{2}\frac{N-Ne^{i2\pi\delta}-1+e^{i2\pi\delta N}}{(1-e^{i2\pi\delta})^2} & 0 & 2i\frac{N\sin(2\pi\delta)-\sin(2\pi\delta N)}{1-\cos(2\pi\delta)} \end{bmatrix}$$
$$\approx \frac{-i\alpha^2}{8N^2\Delta t}\begin{bmatrix} 0 & 0 & \frac{2\sqrt{2}}{e^{-i2\pi\delta}-1} \\ 0 & -\frac{2i\sin(2\pi\delta)}{1-\cos(2\pi\delta)} & 0 \\ \frac{2\sqrt{2}}{1-e^{2\pi i\delta}} & 0 & \frac{2i\sin(2\pi\delta)}{1-\cos(2\pi\delta)} \end{bmatrix}. \tag{3.36}$$

The last term is approximated for large N. This terms scales with $1/N\delta$, and we see that indeed the rotation axis is tilted and we have leakage into the second level. If we want to improve the DANTE sequence we have to make sure to let these terms decay faster than $1/N\delta$, while keeping the perfect result from the effective first order.

3.6. Summary

We have seen that restricting the maximal frequency to $8/\tau$ in the genetic algorithm still allows us to apply high-fidelity $\pi/2$-rotations with short shift register of 256 bits. For such short gate times a resonant pulse train would require much higher pulse amplitudes and lead to strong leakage and low fidelities. Including higher levels,i. e., moving from the qutrit to a five level qudit model, increases the speed limits slightly, but higher level leakage in the original sequence are a minor concern and can be easily captured with another run of the genetic algorithm. The frequency analysis of the optimized sequences reveals that the qubit frequency $1/\tau$ is always present in the spectrum, but the amplitudes of higher harmonics in a resonant pulse train decrease over the course of the optimization. On average the optimized SFQ pulse trains contain around three pulses per qubit cycle, i. e., a pulse repetition rate of $3/\tau$. With the help of the Magnus expansion we derived sufficient conditions that a pulse train needs to satisfy in the second order approximation of the effective Hamiltonian. Resonant pulse trains such as DANTE are first order approximations at best.

Possible extensions of the single-qubit SFQ driving scheme could include a tunable coupler between voltage drive and transmon qubit [46], which would allow different pulse sequence envelopes compared to the rectangular window we have used in this work, such as a Gaussian or a Hann window. The area of the pulse shape that drives the qubit would then depend on the time, although changes of the coupling strength would happen on time scales $\gtrsim \tau$, while the pulse width is still $\ll \tau$. Two-qubit gates could probably be implemented within the cross-resonance scheme [47, 48], which only requires a single rectangular voltage pulse of the control qubit resonant with the target qubit to perform an entangling gate locally equivalent to the CNOT gate.

4. A three-qubit gate for quantum simulations

Quantum simulation is based on the idea to simulate a complex quantum system with another engineered quantum system. Unlike universal quantum computers, these devices are build to solve specific problems and might be the first realizable systems to take advantage of quantum computing principles. Using quantum mechanics to simulate nature has originally been proposed by Feynman [50]. Here we want to implement the iFredkin gate on a quantum device, which is a key requirement for the simulation of the Fermi-Hubbard model proposed in [51]. We focus on an implementation with tunable transmon qubits and fixed coupling strengths. However, with the emergence of the tunable bus for fixed frequency qubits developped at IBM [52], this might be also a potential candidate for the desired quantum simulator.

4.1. The iFredkin gate

The iFredkin gate is a three-qubit entangling gate which performs a controlled-iSwap operation on two target qubits:

$$U_{\mathrm{iFredkin}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & i & 0 \\ 0 & 0 & 0 & 0 & 0 & i & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{4.1}$$

The name of the gate is chosen due to its similarity to the Fredkin gate [1], which performs a controlled-swap operation. Like the swap gate the iSwap

Some of the ideas and figures presented in this chapter have been published in [49].

gate is a two-qubit gate that flips the excitation from the first to the second qubit and vice versa, but with an additional phase shift of $\pi/2$ for the flipped states:

$$U_{\mathrm{iSwap}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & i & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} . \tag{4.2}$$

In contrast to the swap gate the iSwap gate is a perfect entangler [53, 54]. This can be seen by starting with the product state

$$|\Psi_{\mathrm{in}}\rangle = (a\,|0\rangle + b\,|1\rangle)\,(c\,|0\rangle + d\,|1\rangle) \tag{4.3}$$

and performing the iSwap operation to obtain the final state

$$|\Psi_{\mathrm{out}}\rangle = (c\,|0\rangle + id\,|1\rangle)\,a\,|0\rangle + (ic\,|0\rangle + d\,|1\rangle)\,b\,|1\rangle \ , \tag{4.4}$$

which is maximally entangled if $a = b = c = d = 1/\sqrt{2}$. In contrast, applying the swap operation on the input state returns again a product state

$$|\Psi_{\mathrm{out}}\rangle = (c\,|0\rangle + d\,|1\rangle)\,(a\,|0\rangle + b\,|1\rangle) \ . \tag{4.5}$$

Another property of the iSwap gate is that it conserves the number of excitations in the system, and this holds for the iFredkin gate as well. Now the goal is to implement the iFredkin gate on a three-qubit system, using numerical optimization of pulse shapes within GRAPE (see Section 1.3.1).

4.2. Directly coupled qubits

As a first step we capacitively couple three superconducting transmon qubits together to get the basic triangle structure shown in Figure 4.1. This is the basic element of the quantum simulator where the iFredkin operation is performed. It consist of the control qubit P and two register qubits S_1 and S_2, with controllable frequencies and fixed coupling strengths. In the full simulator an array of register qubits S with nearest neighbour couplings would be used, and the control qubit P is coupled to all of them to perform the necessary iFredkin gates for the simulation of the Fermi-Hubbard model [51]. For now we focus on the three-qubit model in the rotating wave

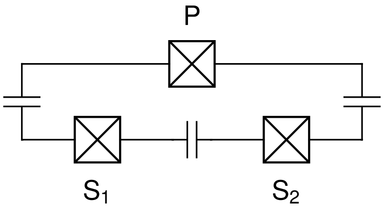

Figure 4.1.: Coupling three qubits directly with each other to implement the iFredkin-gate, where qubit P takes the role of the control qubit. Not shown are the flux control lines of the qubits to change their frequencies.

approximation. The Hamiltonian of this system conserves the number of excitations and for tunable transmon qubits reads

$$H = \sum_{i=1}^{3} \left[\left(\omega_i(t) - \frac{\Delta_i}{2} \right) b_i^\dagger b_i + \frac{\Delta_i}{2} \left(b_i^\dagger b_i \right)^2 \right] + \sum_{i<j} g_{ij} \left(b_i^\dagger b_j + b_i b_j^\dagger \right) . \tag{4.6}$$

Each qubit is coupled to the remaining two qubits with coupling strength g_{ij}, has a controllable frequency $\omega_i(t)/2\pi$ and a fixed anharmonicity Δ_i in the Duffing oscillator approximation of the transmon qubit [9]. The operators $b_i^\dagger$ and b_i are the raising and lowering operator of the ith qubit. Since both the iFredkin and the Hamiltonian are number conserving, it should be possible to implement the iFredkin gate with this architecture. Furthermore, optimal control methods have been demonstrated successfully on three qubit gates [55]. But we point out that the system is not fully controllable, such that arbitrary single-qubit bit flip operations would need additional control lines and tailored microwave pulses.

To find the optimal detuning sequence of the three qubits we work in a frame rotating with the control qubit $\omega_R = \omega_1$ (see Appendix A.2), with the system parameters we use for the simulations listed in Table 4.1. The initial

	$\omega_i/2\pi$	$\Delta_i/2\pi$	$g_{iP}/2\pi$	$g_{iS_1}/2\pi$	$g_{iS_2}/2\pi$
Qubit P	6.5 GHz	−200 MHz	-	30 MHz	30 MHz
Qubit S_1	7.5 GHz	−300 MHz	30 MHz	-	30 MHz
Qubit S_2	8.5 GHz	−400 MHz	30 MHz	30 MHz	-

Table 4.1.: Parameters used in the direct coupling case to implement the iFredkin gate.

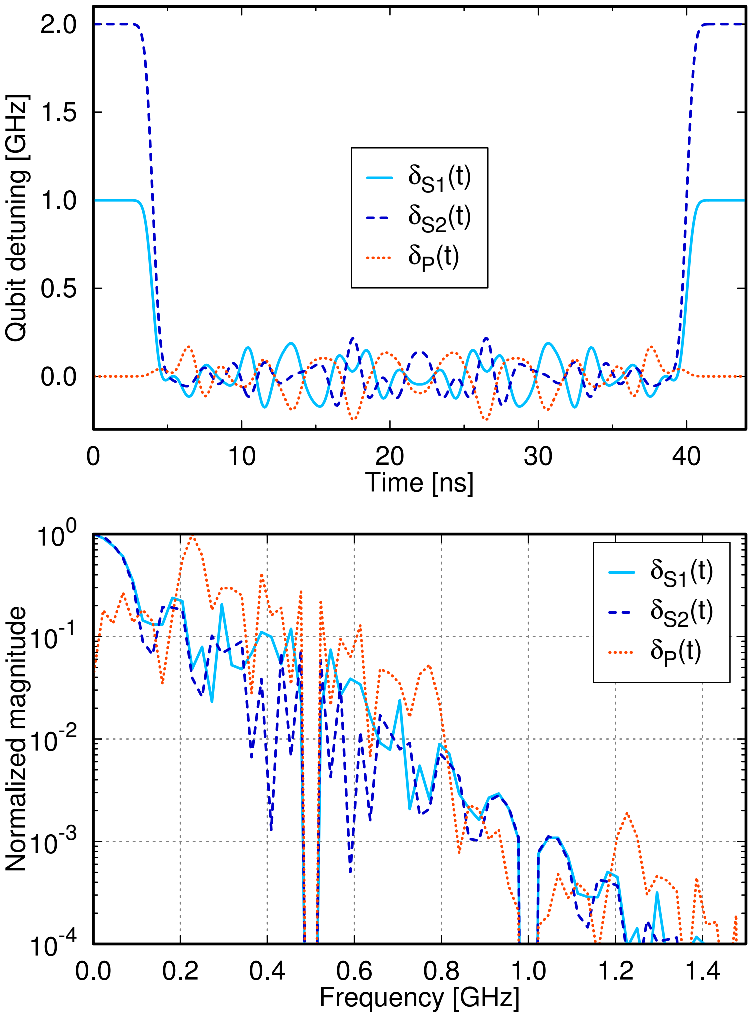

Figure 4.2.: The three Z-controls, including a 4 ns buffer on each side and filtered by a Gaussian window function with standard deviation $\sigma = 0.4$ ns. We allow the P qubit to be tunable as well. The bottom picture shows the spectrum of the controls.

frequency of the control qubit P is $\omega_1/2\pi$ and therefore the initial and final detuning of P from the rotating frame frequency is $\delta_P/2\pi = 0$. Likewise, the detunings of the register qubits are defined as $\delta_{S_i}(t) = \omega_{S_i}(t) - \omega_1$. The time steps in the simulation have a duration of $\delta t = 0.1$ ns and the control pulses to tune the qubit frequencies have a resolution of 1 ns, typical for arbitrary waveform generators (AWG). We are also filtering the controls

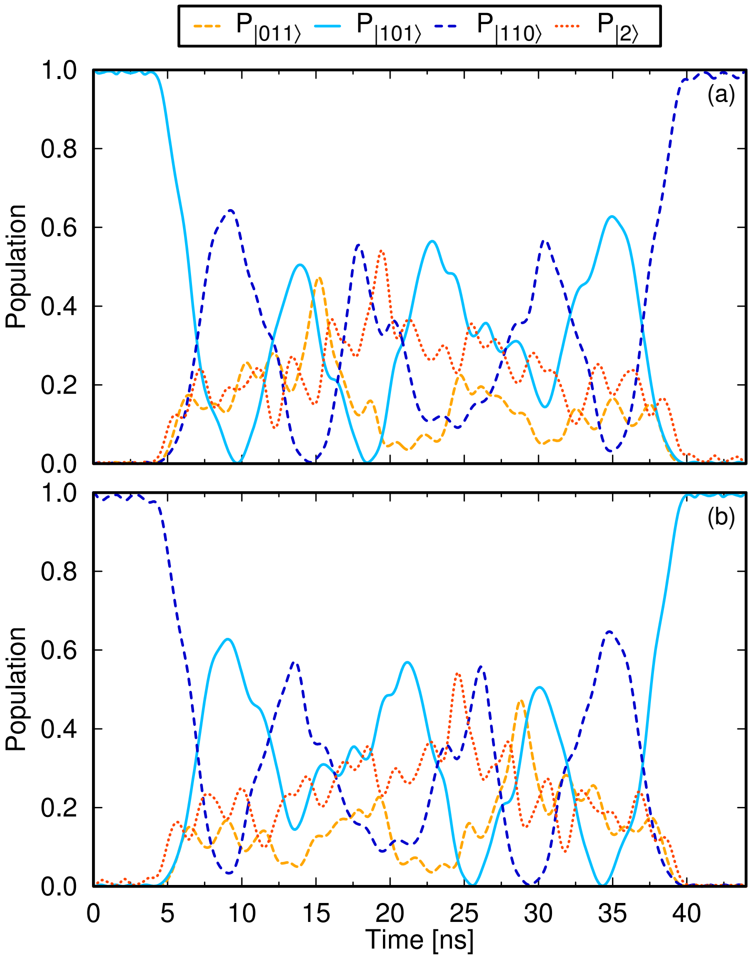

Figure 4.3.: Time evolution of the populations in the two-excitation subspace, starting in (a) $|101\rangle$ and (b) $|110\rangle$, respectively. Shown are the swap of population between the states $|110\rangle$ and $|101\rangle$, the population of the state $|011\rangle$, and the probability that one of the qubits is excited into its second level $P_{|2\rangle} = |200\rangle + |020\rangle + |002\rangle$.

with a Gaussian window function with a standard deviation of $\sigma = 0.4$ ns. To ensure that the pulse shapes start and end at the initial qubit frequencies we add a buffer of 4 ns at the beginning and the end of each pulse. Since the number of excitations is a conserved quantity, the maximum excitation of the system is $n = 3$, and for each qubit the three lowest energy levels are taken into account. Therefore, the dimension of the simulated Hilbert space

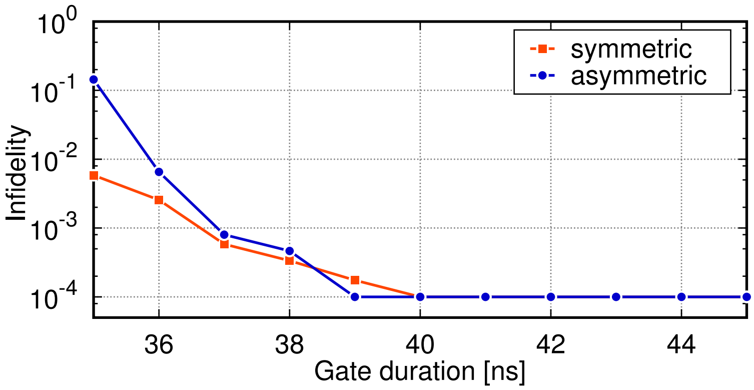

Figure 4.4.: Speed limit of the iFredkin gate for the $\Phi = 0.9999$ threshold. The asymmetric case has been calculated for the parameters in Table 4.1, while for the symmetric case all anharmonicities are set to $-200\,\mathrm{MHz}$.

is $d = 17$, and the dimension of the computational subspace is $N = 8$, such that the the trace fidelity reads

$$\Phi = \frac{1}{N^2} \left| \mathrm{Tr}\left\{ U^{\dagger}_{\mathrm{iFredkin}} \Pi_Q U\left(t_g\right) \Pi_Q \right\} \right|^2 . \tag{4.7}$$

The operators Π_Q project the total time evolution $U\left(t_g\right)$ onto the computational subspace.

Figure 4.2 shows the optimized control pulses an their spectrum found by GRAPE to perform the iFredkin operation with a trace fidelity of $\Phi \geqslant 0.9999$. Once the qubits are brought close to resonance, the pulse shapes oscillate with various frequencies and low amplitudes around the resonance point. In Figure 4.3 we show the populations of the qubit states in the two-excitation subspace in which the excitation swap occurs. The states are labelled as $|n_P n_{S_1} n_{S_2}\rangle$, where each n_i is the number of excitations in the ith qubit. We point out that over the course of the time evolution almost every state carries a large amount of excitation probability at some point, including the second excited state of each qubit. Finally, we can approximately find a speed limit for the iFredkin gate, shown in Figure 4.4. As can be seen, the speed limit is just above 40 ns, which is very close to two-qubit gates for qubits with tunable frequencies [56]. One effect of approaching the speed limit is the symmetry breaking of the control pulses in Figure 4.2. We also calculate the the speed limit where each qubit has the same anharmonicity of $\Delta = -200\,\mathrm{MHz}$, which we refer to as the symmetric case. For a fidelity threshold of $\Phi = 0.9999$ this shows no major difference for the best achievable gate duration. But we see that the speed limit does not reveal a sharp cut-off. Taking into account that the duration of the buffer could possibly be reduced, which we can see from Figure 4.2 and Figure 4.3, the speed limit of the iFredkin gate seems to lie just above $2\pi/g = 33.3\,\text{ns}$.

4.3. Coupling qubits via a resonator

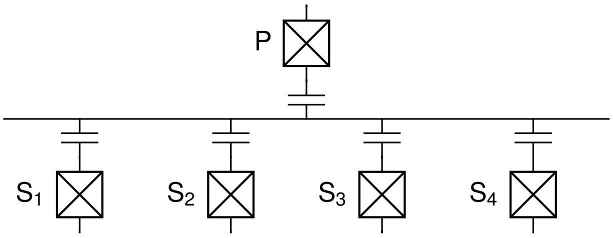

Figure 4.5.: The RezQu-architecture to implement the iFredkin-gate, where qubit P takes the role of the control qubit. Each qubit is coupled to a common transmission line, and is shifted in and out of resonance to perform the desired gate. Not shown are the flux control lines of the qubits to change their detuning from the bus.

In the previous section we have shown that is possible to implement the iFredkin gate with tunable transmon qubits and fixed couplings only. However, it is not straightforward to scale this system to include a larger system register S. While nearest-neighbour coupling between the S qubits is possible within a linear chain and therefore iSwaps between the S qubits, the control qubit P needs to be coupled to each S qubit as well to perform the iFredkin gates. Coupling a single qubit to every other qubit over large distances directly is a nontrivial task. Besides, we want to avoid crosstalk

	Qubit P	Qubit S_1	Qubit S_2
$\omega_i/2\pi$	7.5 GHz	8.0 GHz	8.5 GHz
$\delta_i/2\pi$	1.0 GHz	1.5 GHz	2.0 GHz
$\Delta_i/2\pi$	−200 MHz	−300 MHz	−400 MHz
$g_i/2\pi$	30 MHz	45 MHz	60 MHz

Table 4.2.: RezQu parameters used for the iFredkin gate with a bus frequency at $\omega_B/2\pi = 6.5\,\text{GHz}$.

between qubits in the register S that do not take part in the iFredkin operation, such that the coupling strength between register qubits should be tunable. Therefore, we should think about a modified architecture. One way to couple superconducting qubits over long distances can be done with transmission line resonators. The RezQu architecture [57, 58, 59, 60] is built out of frequency tunable qubits coupled to a common transmission line. This allows to couple the control qubit P to every S qubits with such a transmission line, by tuning their frequencies to bring them on resonance for interaction and turn of the coupling by bringing them out of resonance. Here we use the RezQu architecture as a potential candidate for a scalable simulation device. Pulse shapes found by numerical methods, such as GRAPE, have proven to be faster than analytical control pulses on this architecture [56]. We will work in the resonant regime, i. e., the qubits are brought on resonance with the transmission resonator to interact with each other. The difference to the dispersive regime, where the bus is always detuned from the qubits is discussed in Chapter 6. Again we will work in the rotating frame, this time rotating with the bus angular frequency $\omega_R = \omega_B$. Each transmon qubit frequency is tunable and the coupling strengths are fixed, as has been the case in the previous section. The Hamiltonian of the RezQu architecture, where the coupling between each qubit is mediated by a common bus, reads

$$H = \sum_{i=1}^{3} \left(\delta_i \left(t \right) - \frac{\Delta_i}{2} \right) b_i^\dagger b_i + \frac{\Delta_i}{2} \left(b_i^\dagger b_i \right)^2 + g_i \left(a^\dagger b_i + a b_i^\dagger \right) . \tag{4.8}$$

Here, a and b_i are the bus and qubit lowering operators, respectively. Each of the detunings $\delta_i = \omega_i - \omega_B$ of qubit i from the bus is controllable. The parameters of the device used for the simulation are listed in Table 4.2. In all runs the controls have a time resolution of typical AWGs of 1 ns, with fine steps of 0.1 ns for the time evolution. Additionally, the pulse shapes are filtered by a Gaussian window with a bandwidth of 331 MHz (standard deviation $\sigma = 0.4$ ns). For each qubit, the first three energy levels are taken into account. Again we are only interested in the correct evolution of the computational subspace and use the same trace fidelity function as in the previous section.

In Figure 4.6 we show the optimized qubit-bus detuning for the implementation of a iFredkin gate. The time evolution of the populations is shown in Figure 4.7 and the speed limit for the 0.9999 fidelity in Figure 4.8. The

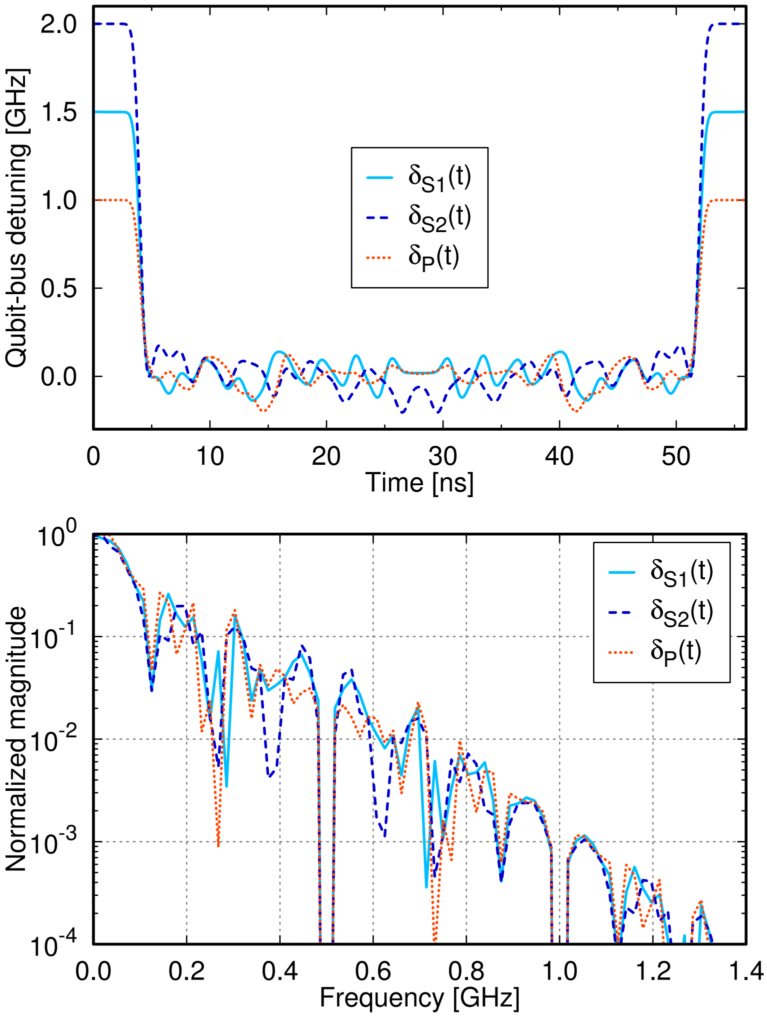

Figure 4.6.: The three Z-controls, including a 4 ns buffer on each side and filtered by a Gaussian window function with standard deviation $\sigma = 0.4$ ns. The coupling between the qubits is mediated by a bus resonator. The bottom picture shows the spectrum of the controls.

states are labelled as $|n_B; n_P n_{S_1} n_{S_2}\rangle$, with the number of excitation in the bus n_B. As can be seen, the pulse shapes are highly symmetric [56], as are the resulting time evolutions of the populations. The pattern of the controls is similar to that of Figure 4.2, showing low amplitude oscillations once all qubits are close to resonance. Approaching the speed limit leads

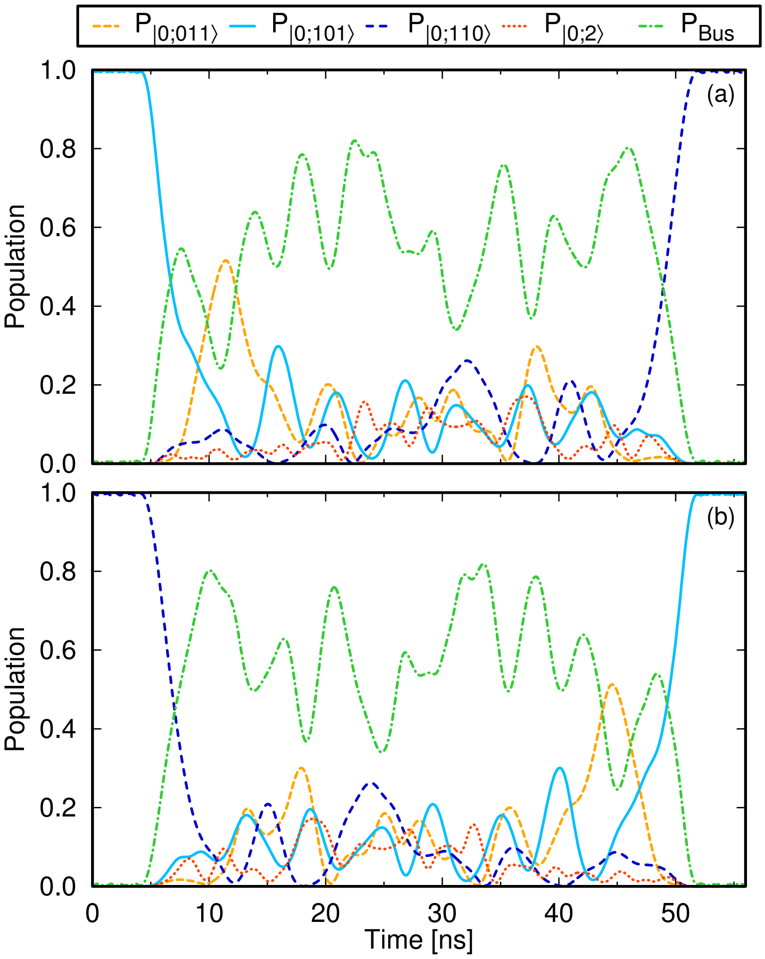

Figure 4.7.: Time evolution of the populations in the two-excitation subspace, starting in (a) $|0;101\rangle$ and (b) $|0;110\rangle$, respectively. Shown are the swap of population between the states $|0;110\rangle$ and $|0;101\rangle$, the population of the state $|0;011\rangle$, the probability that one of the qubits is excited into its second level $P_{|0;2\rangle} = |0;200\rangle + |0;020\rangle + |0;002\rangle$, and the probability that the bus is excited $P_{\mathrm{Bus}} = |1;001\rangle + |1;010\rangle + |1;100\rangle + |2;000\rangle$. It can be clearly seen that the iFredkin implementation makes heavy use of excitations in the resonator.

to the typical deterioration of the pulse shape symmetry. As can be seen from Figure 4.7, most of the population resides in the bus resonator while the qubits are on resonance with the bus. Leakage into the second excited state of each qubit plays also an important role in the gate implementation

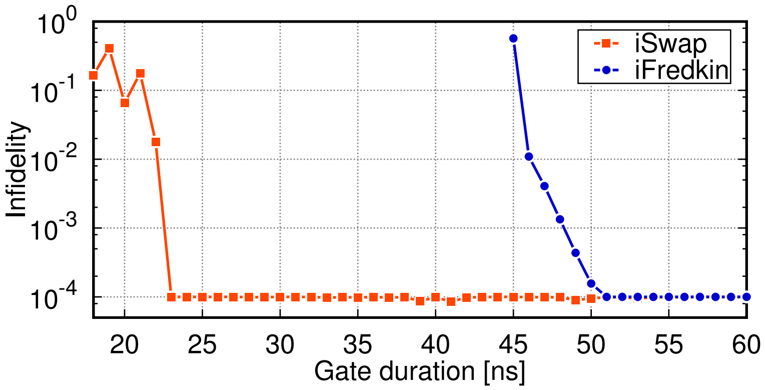

Figure 4.8.: Speed limit for the iFredkin and iSwap gate for the three frequency tunable qubits in the RezQu architecture for the fidelity threshold $\Phi = 0.9999$. In case of the iSwap the P qubit is kept out of resonance with the bus and the S qubits, showing that the unconditioned two-qubit gate can be implemented in less than half the time of the iFredkin gate.

of the numerical pulse and is of the same order as the qubit populations, which we have already seen in Figure 4.3. The speed limit shown in Figure 4.8 proves that the iFredkin gate can easily be implemented below a gate time of $t_g = 60\,\text{ns}$. This time scale is typical for analytic two-qubit pulse shapes, i.e., the Strauch sequence [61, 56]. As a comparison we also show the speed limit of a simple iSwap between the two S qubits, where the control qubit P is brought to a parking frequency of 10 GHz. This shows the typical achievable two-qubit speed limit, and that the time critical operation is the conditioning of the iSwap with P in the iFredkin gate. It might be possible to decrease the gate time with non-gradient based global search methods, such as genetic algorithms (see Section 1.3.2), which requires a lot more system resources for optimization. On the other hand, swapping two bits is limited by the coupling strength g. Taking the lowest value $g = 30\,\text{MHz}$ as a reference, i.e. the coupling of the control qubit to the bus, the speed limit of the iFredkin gate lies approximately at $\sqrt{2}2\pi/g = 47.14\,\text{ns}$.

4.4. Summary

We have implemented the iFredkin gate for quantum devices with only Z-controls. As the results show, fast three-qubit gates are reachable with optimal control methods, with gate times that compare to two-qubit gates. Within the RezQu architecture it is therefore possible to implement the required iFredkin gates on a system register S with a single control qubit P, needed for efficient simulation of the Fermi-Hubbard model [51]. The disadvantage of the presented methods is that we have to detune all remaining qubits far away of the resonator, since the latter contains most of the excitation. Furthermore, it is not possible to parallelize iFredkin operation on different pairs of system qubits, due to the crosstalk over the bus. But, as coherence times keep increasing, sequential iFredkin gates are less harmful and perform faster than their decomposition into single- and two-qubit gates, so it should be possible to use the RezQu architecture in the near term future for quantum simulation.

5. Optimal control with digital basis functions

Optimal pulse shapes found by GRAPE often lack a simplified description, since every control pixel is optimized individually. Using a set of basis functions to parametrize the control pulse can help to understand the optimal solution, and to find a description where much less parameters need to be optimized. A standard parametrization is the decomposition into a Fourier series. Depending on the hardware, amplitudes, phases and frequencies can be optimized. With the help of the chain rule it is straightforward to implement the GRAPE algorithm using analytic functions [62]. Here, we parametrize the control pulse through a basis set of digital sine waves, which are known as Walsh functions [63]. These functions can only take the values ±1 within the function domain.

5.1. Walsh functions

Before we define the Walsh functions through an explicit construction formula we introduce the Rademacher functions [64] first. These are digital sine functions and have the form

$$R_n(x) = \mathrm{sgn}\left(\sin\left(2^n \pi x\right)\right) \qquad n \geqslant 0\,, \tag{5.1}$$

where the domain is the open interval $x \in [0, 1)$. The function values are either 1 or -1 and constant within each subinterval $[l/2^n, (l+1)/2^n)$, where l ranges from 0 to $2^n - 1$. The scalar product on the full interval is defined as

$$\langle f|g\rangle = \int_0^1 f(x)\, g(x)\, dx\,, \tag{5.2}$$

and the Rademacher functions are orthonormal

$$\int_0^1 R_i(x)\, R_j(x)\, dx = \delta_{ij}\,. \tag{5.3}$$

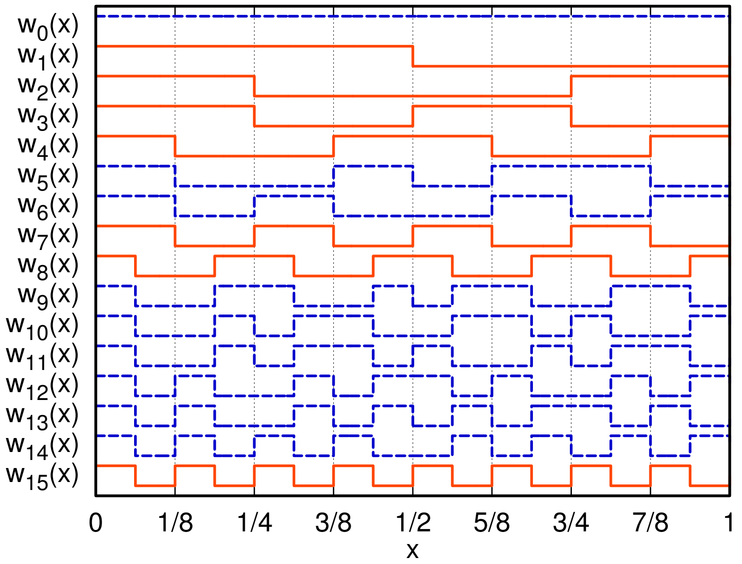

Figure 5.1.: The first 16 Walsh functions in sequency order. Incrementing the function index adds another sign flip and changes the symmetry relative to the midpoint $x = 1/2$. The digital sine and cosine functions are marked in red.

However, the Rademacher functions are not complete and therefore do not form a basis set. This can be seen by choosing the piecewise constant (PWC) function $f(x) = R_2(x)R_1(x)$ and decompose it into Rademacher functions. The function $f(x)$ is obviously not 0, but using $R_i^2(x) = 1$ we see that its decomposition reads

$$\int_0^1 f(x)R_i(x)\,dx = \delta_{i1}\underbrace{\int_0^1 R_2(x)\,dx}_{=0} + \delta_{i2}\underbrace{\int_0^1 R_1(x)\,dx}_{=0} = 0\,. \tag{5.4}$$

As this example shows, multiplying two different Rademacher functions returns a new function. If we do all possible multiplications of the first m Rademacher functions we get the definition of the Walsh functions in Parley order [65]

$$W_n(x) = \prod_{j=1}^{m} R_j(x)^{b_j}\,. \tag{5.5}$$

The values $b_j \in \{0,1\}$ are determined through the binary representation of the function index $n = \sum_{j=1}^{m} b_j 2^{j-1}$, and the number of subintervals is

$N = 2^m$. The orthonormal property of the Walsh functions follows directly from the Rademacher functions and reads

$$\int_0^1 W_i(x) W_j(x)\,dx = \delta_{ij}\,. \tag{5.6}$$

Multiplying two different Walsh functions returns again a Walsh function, which we deduce from equation 5.5 and $R_i^2(x) = 1$. For N subintervals we get N orthonormal Walsh functions, and therefore the Walsh functions form a basis set of the PWC functions with step length $1/N$.

Despite the simple construction formula in equation 5.5, it is worth to sort the Walsh functions differently, such that the index of the function is equal to the number of sign changes in the interval $[0, 1)$. The Walsh functions in sequency order can be constructed by the following recursive formula [63, 66]

$$\begin{aligned} w_{2j}(x) &= w_j(2x) + (-1)^j w_j(2x-1) \\ w_{2j+1}(x) &= w_j(2x) - (-1)^j w_j(2x-1)\,, \end{aligned} \tag{5.7}$$

with $w_0(x) = W_0(x) = R_0(x)$ and $w_1(x) = W_1(x) = R_1(x) = H(x)$. The latter is also known as the Haar wavelet [67]

$$H(x) = \begin{cases} 1 & 0 \leqslant x < \frac{1}{2} \\ -1 & \frac{1}{2} \leqslant x < 1 \\ 0 & \text{else} \end{cases} \tag{5.8}$$

and gives rise to the wavelet decomposition of the interval $[0, 1)$. Similar to equation 5.5, one can also find an explicit construction for the sequency ordered Walsh functions. Therefore, we introduce the digital cosine functions

$$r_n(x) = \operatorname{sgn}(\cos(2^n \pi x)) \qquad n \geqslant 0\,. \tag{5.9}$$

Using the relation $R_n(x)\, r_n(x) = R_{n+1}(x)$ one easily gets

$$w_n(x) = \prod_{j=1}^{m} r_{j-1}(x)^{b_j}\,, \tag{5.10}$$

where n is again decomposed into its binary representation. We note that using the Gray code [65] decomposition of n in equation 5.5 instead of the binary decomposition would also lead to the sequency ordered Walsh functions. Figure 5.1 shows the first 16 Walsh functions in sequency order. Indeed, the number of sign changes grows linearly with the index of the function, and neighbouring functions have even and odd symmetry relative to the midpoint $x = 1/2$ of the interval. This symmetry relation can be expressed through

$$w_n(1-x) = (-1)^n w_n(x)\,. \tag{5.11}$$

Therefore, even and odd indices label the even and odd Walsh functions, respectively, as well as an even and odd number of sign changes. Given an analytic function $u(x)$, it can be decomposed into an infinite Walsh series

$$u(x) = \sum_{n=0}^{\infty} c_n w_n(x)\,, \tag{5.12}$$

where the nth Walsh coefficient reads

$$c_n = \int_0^1 u(x) w_n(x)\,dx\,. \tag{5.13}$$

5.2. Fast Walsh transformation

For the following sections we discretize the open interval $[0, 1)$ into $N = 2^m$ equally sized subintervals $[l/N, (l+1)/N)$, with $l = 0, \dots, N-1$. Then the integral of the orthonormal condition in equation 5.6 turns into the sum

$$\frac{1}{N} \sum_{l=0}^{N-1} w_i(x_l) w_j(x_l) = \delta_{ij}\,, \tag{5.14}$$

Parley	Sequency	Hadamard
$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$

Table 5.1.: Different orderings of the Walsh functions and their transformation matrices for $N = 4$. Each matrix row corresponds to a discrete Walsh function. Note that all three matrices are symmetric.

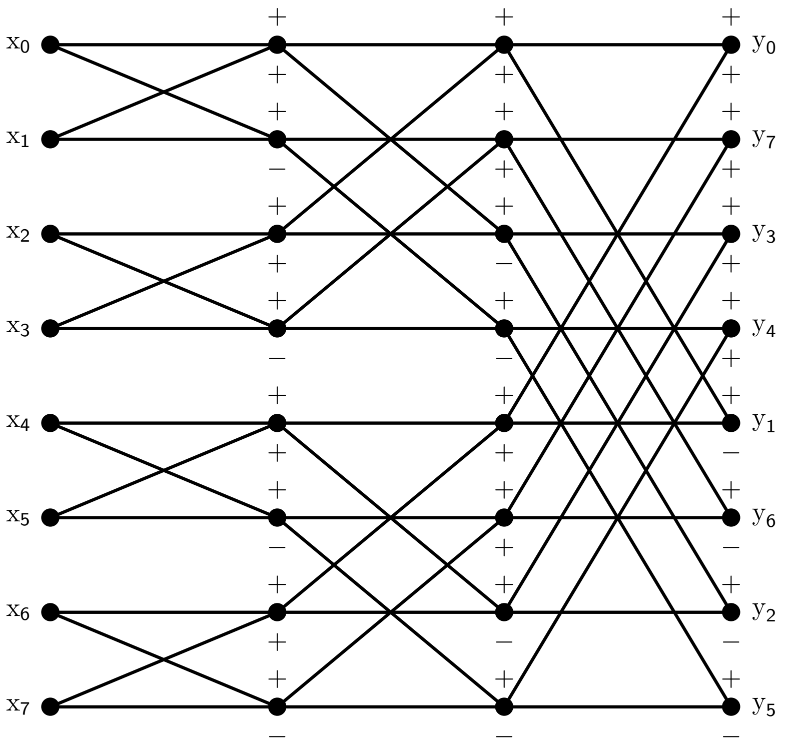

Figure 5.2.: The fast Walsh transformation of a vector x with the final vector y in sequency order. The $+$ and $-$ at each node indicate the sign of the addend. The alignment of the components of y gives rise to a third scheme to label the Walsh functions, namely the Hadamard order.

where $x_l = (l + 1/2)/N$ is the midpoint of the corresponding subinterval. Given a vector $x = [x_0, x_1, \dots, x_{N-1}]^\top$, the transformation into the Walsh basis reads

$$y = Tx\,, \tag{5.15}$$

with the transformation matrix

$$T = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & -1 & \cdots & -1 \end{bmatrix}. \tag{5.16}$$

The nth row of this matrix is the nth Walsh function, sampled at the midpoint of each subinterval. We point out that the transformation matrix is symmetric $T^\top = T$, which is independent of the specific ordering, and the Walsh transform is its own inverse up to normalization

$$x = \frac{1}{N} T y = \frac{1}{N} T^2 x \,. \tag{5.17}$$

Analogue to the fast Fourier transformation (FFT) a fast Walsh transformation (FWT) can be defined [66]. As an example, we apply the algorithm for $N = 4$ on the input vector $x = [x_0, x_1, x_2, x_3]^\top$:

$$\left.\begin{aligned} a_0 &= x_0 + x_1 \\ a_1 &= x_0 - x_1 \\ a_2 &= x_2 + x_3 \\ a_3 &= x_2 - x_3 \end{aligned}\right\} \rightarrow \quad \begin{aligned} y_0 &= a_0 + a_2 \\ y_1 &= a_0 - a_2 \\ y_2 &= a_1 - a_3 \\ y_3 &= a_1 + a_3 \end{aligned} \,. \tag{5.18}$$

The generalization to higher dimensions $N = 2^m$ for an arbitrary $m \geqslant 1$ should be straightforward and is done graphically for $N = 8$ in Figure 5.2. The alignment of the output vector y in Figure 5.2 is called Hadamard order of the Walsh functions, while the entries of y are labelled corresponding to the sequency order. In the Hadamard order the transformation matrix T takes the form of a general Hadamard matrix of dimension $N = 2^m$. Table 5.1 lists the three orderings encountered for the Walsh functions and their corresponding transformation matrix for $N = 4$. If requested, one can normalize the output vector y by multiplying with $1/\sqrt{N}$, making forward and backward transform fully symmetric. The FWT reduces the computational cost from $\mathcal{O}(N^2)$ for the direct application of T to $\mathcal{O}(N \log(N))$. Also, only pairwise addition and subtraction occurs, and no multiplication with phase factors is needed, mapping real vectors onto real vectors.

5.3. Gradient ascent with Walsh functions

To use the Walsh decomposition in the GRAPE algorithm, the size of the control vector u_{kj} must be $N = 2^m$, which matches the number of intervals of the $N/2$ highest Walsh functions $w_{N/2}(x) \dots w_{N-1}(x)$. The time spacing between two successive sample points is $\tau = t_g/N$, with t_g being the gate time. Now the kth control vector can be decomposed into a finite series of Walsh functions with the help of the asymmetric FWT

$$u_{kj} = \sum_{n=0}^{N-1} a_{kn} w_n(x_j) = \sum_{n=0}^{N-1} T_{jn} a_{kn} \,, \tag{5.19}$$

where the coefficients of the Walsh decomposition read

$$a_{kn} = \frac{1}{N} \sum_{j=0}^{N-1} u_k(x_j) w_n(x_j) = 2^{-m} \sum_{j=0}^{N-1} T_{jn} u_{kj} \,. \tag{5.20}$$

The update rule for the Walsh coefficients is

$$a_{kn} \to a_{kn} + \epsilon \frac{\partial \Phi}{\partial a_{kn}} \,, \tag{5.21}$$

with the gradient of the fidelity function Φ with respect to the Walsh coefficients. The gradient can be calculated with the help of the chain rule

$$\frac{\partial \Phi}{\partial a_{kn}} = \sum_{j=0}^{N-1} \frac{\partial u_{kj}}{\partial a_{kn}} \frac{\partial \Phi}{\partial u_{kj}} = \sum_{j=0}^{N-1} w_n \left(x_j \right) \frac{\partial \Phi}{\partial u_{kj}} \,. \tag{5.22}$$

The last equation is the FWT of the gradient of the fidelity function with respect to the control vector u_{kj}. At each iteration of the GRAPE algorithm the gradient is calculated and the Walsh amplitudes a_{kn} are updated according to equation 5.21, before the control amplitudes u_{kj} are set through the FWT of a_{kn}. We can also use some filter process of the piecewise constant controls and include a buffer at the beginning and end of the pulses to allow smooth pulse shapes. However, the buffer pixels are excluded from the Walsh transformation and the chain rule in equation 5.22.

The decomposition in equation 5.12 is not limited to Walsh functions, and also allows us to use analytic controls [62]. Furthermore, instead of using the midpoint sampling method in GRAPE and calculating the unitary matrix U_j for each pixel through exponentiation, one could solve the Schrödinger equation with any ordinary differential eqaution (ODE) solver, for example the Runge-Kutta method [68]. This can reduce the total amount of pixels needed to compute the time evolution operators with high accuracy. Additionally, instead of doing a backwards evolution to compute the gradient of $\partial U / \partial u_{kn}$, one can solve the dynamic equation for the gradient $\partial U / \partial u_{kn}$ in parallel with the Schrödinger equation. This is done in the Gradient Optimization for Analytic conTrols (GOAT) algorithm [69]. However, the choice of the parametrization is a priori not affected by the algorithm itself, but GOAT can reduce the computational cost dramatically if fast oscillating analytic functions are used and high accuracy is required.

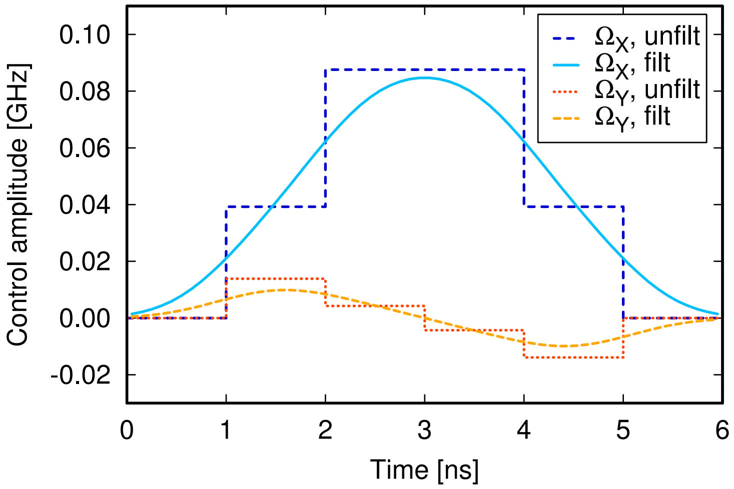

Figure 5.3.: The control amplitudes resembling the DRAG pulse shape. The four inner pixels are determined by the Walsh amplitudes, and on each side a buffer pixel is added (unfilt), before the controls are filtered by a Gaussian window with $\sigma = 0.53\,\text{ns}$ (filt).

5.3.1. The numerical DRAG pulse

As a first demonstration, we use GRAPE and the Walsh parametrization to find a numerically optimized pulse shape for the driven single-qubit system that lead to the developement of the analytic Derivative Removal by Adiabatic Gate (DRAG) method [70]. The Hamiltonian in the rotating wave approximation (RWA) in a frame rotating with the frequency $\omega_Q/2\pi$ of the qubit (see Appendix A.2) reads

$$H(t) = \frac{\Delta}{2} a^\dagger a^\dagger a a + \Omega_X(t)\left(a^\dagger + a\right) + i\Omega_Y(t)\left(a^\dagger - a\right). \tag{5.23}$$

The qubit is driven by two control pulses with envelopes $\Omega_X(t)$ and $\Omega_Y(t)$ and at a frequency of $\omega_Q/2\pi$. The anharmonicity between the second and third level is set to $\Delta/2\pi = -400\,\text{MHz}$. The goal is to implement the NOT gate between the two lowest energy levels, and we are using the fidelity function

$$\Phi = \frac{1}{4}\left|\text{Tr}\, U_F^\dagger U(t_g)\right|^2, \quad \text{with} \quad U_F = \begin{bmatrix} X & 0 \\ 0 & 0 \end{bmatrix}. \tag{5.24}$$

The target fidelity is set to $\Phi = 0.9999$ and the gate time is $t_g = 6\,\text{ns}$. For each control only two of the four possible Walsh functions are used:

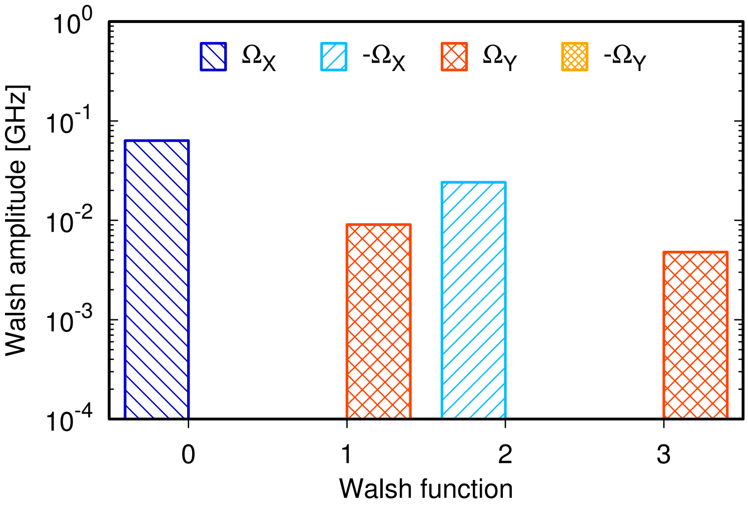

Figure 5.4.: The absolute Walsh amplitudes of the DRAG pulses, color-coded for positive and negative values. For Ω_X and Ω_Y only the even and odd functions are used, respectively. The X-control has a positive and a negative Walsh component, while the Y-control only has positive Walsh components.

the even pair $w_0(x)$ and $w_2(x)$ for the X-control and the odd pair $w_1(x)$ and $w_3(x)$ for the Y-control. A buffer pixel is added on each side, and a Gaussian filter is applied on the controls with standard deviation of $\sigma = 0.53\,\text{ns}$ (bandwidth of 250 MHz) to allow for a smooth beginning and end of the pulse. The time resolution of the Walsh functions is 1 ns, and 10 subpixels per control pixel are used for the time evolution of the system. Figure 5.3 shows the result of the GRAPE optimization for the pulse shapes. As can be seen, the filtered controls of the numerical pulse shapes resemble the DRAG solution of the system, with the Y-control being approximately the scaled derivative of the X-control. In Figure 5.4 we show the Walsh amplitudes of the optimized controls. The number of Walsh functions for each pulse is sufficient to achieve the desired fidelity, and we are able to reduce the number of optimization parameters to four compared to the standard GRAPE method which uses eight parameters.

5.3.2. The CZ gate

As another demonstration we want to implement the CZ gate in the RezQu architecture [59], which we have introduced in Section 4.3. For two qubits

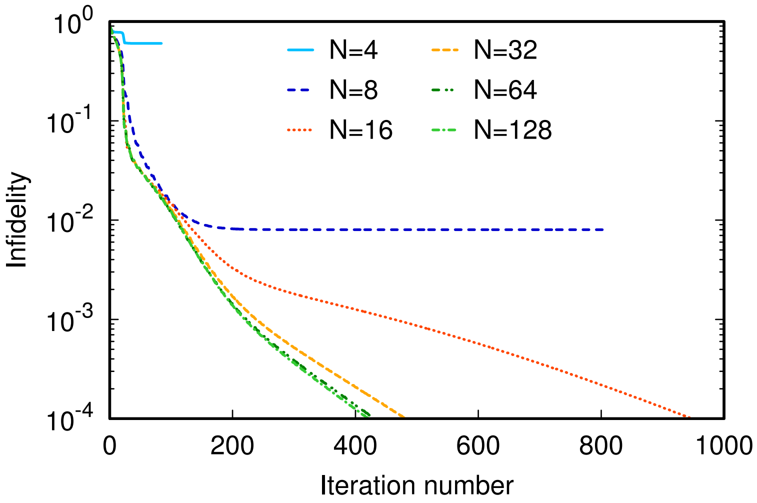

Figure 5.5.: Infidelities for different numbers of Walsh functions N. GRAPE converges for at least $N = 16$, corresponding to a time resolution of $\tau = 2$ ns. Higher time resolutions allow for more Walsh functions and a faster convergence.

the Hamiltonian in a frame rotating with the frequency of the bus $\omega_B/2\pi$ (see Appendix A.2) reads

$$H = \sum_{i=1}^{2} \left(\delta_i(t) - \frac{\Delta_i}{2}\right) b_i^\dagger b_i + \frac{\Delta_i}{2}\left(b_i^\dagger b_i\right)^2 + g_i\left(a^\dagger b_i + a b_i^\dagger\right), \qquad (5.25)$$

with the qubit-bus detuning $\delta_i = \omega_i - \omega_B$. The system parameters are listed in Table 5.2. For the optimization we allow up to 128 Walsh functions. Setting the gate time to 32 ns, this requires a time resolution of $\tau = 0.25$ ns for the control pixels, and we set the subpixel duration to $dt = 0.125$ ns. Again, we use the same Gaussian filter as in the previous case and add a

	Qubit 1	Qubit 2
$\omega_i/2\pi$	6.6 GHz	6.5 GHz
$\delta_i/2\pi$	0.5 GHz	0.4 GHz
$\Delta_i/2\pi$	-200 MHz	-167 MHz
$g_i/2\pi$	30 MHz	25 MHz

Table 5.2.: RezQu parameters used in the Walsh example with a bus frequency of $\omega_B/2\pi = 6.1$ GHz.

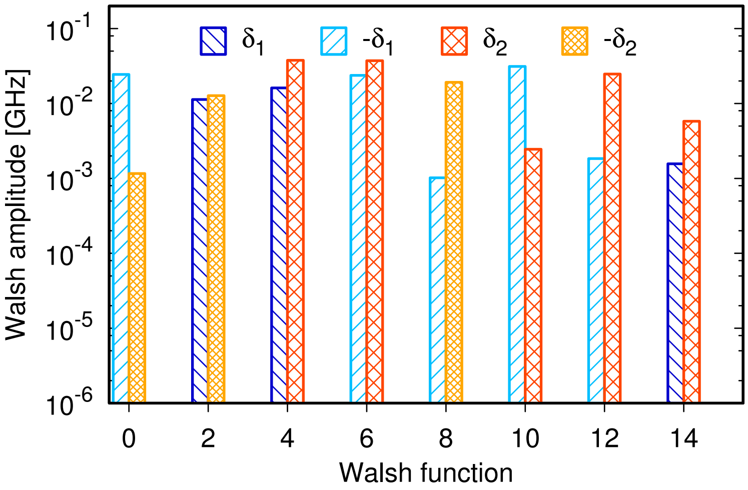

Figure 5.6.: The absolute Walsh amplitudes for $N = 16$, color-coded for positive and negative values. The odd components do not contribute to the pulse shape and can be neglected for further optimizations, leading to at most 16 parameters that GRAPE needs to optimize. The PWC ansatz needs 32 parameters for the same time resolution.

buffer of 4 ns duration on each side, for a total gate time of $t_g = 40$ ns. The CZ gate embedded into the system reads

$$U_F = |0;00\rangle\langle 0;00| + |0;01\rangle\langle 0;01| + |0;10\rangle\langle 0;10| - |0;11\rangle\langle 0;11| \,, \tag{5.26}$$

such that the bus resonator remains in the ground state. Therefore follows for the trace fidelity

$$\Phi = \frac{1}{16}\left|\mathrm{Tr}\, U_F^\dagger U(t_g)\right|^2 . \tag{5.27}$$

The first step is to figure out how coarse we can make the time resolution τ before GRAPE does not converge any more. This reduced the number of Walsh functions each pulse can be made of. Figure 5.5 shows the convergence of GRAPE for different numbers of Walsh functions ranging from $N = 4$ to $N = 128$, which relates to time resolutions between $\tau = 8$ ns and $\tau = 0.25$ ns. The GRAPE algorithm succeeds in finding optimal pulses if at least the first 16 Walsh functions are given. Including higher order Walsh functions allows GRAPE to converge faster, but at the cost of more parameters to optimize. A time resolution $\tau = 2$ ns is therefore sufficient, which is

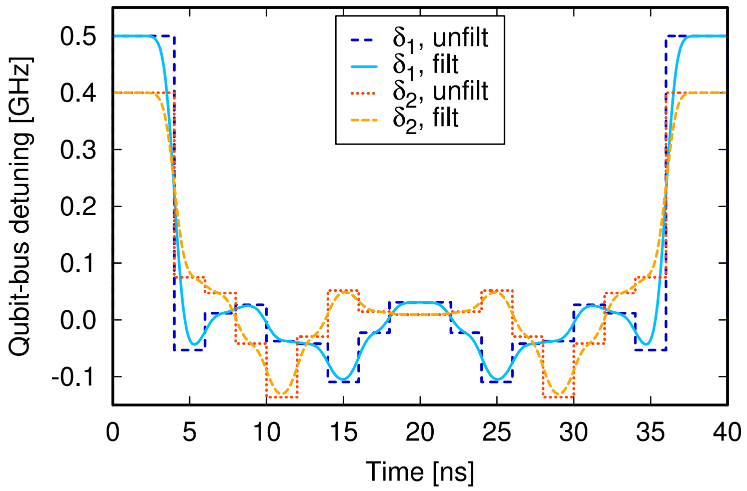

Figure 5.7.: Control pulses for $N = 16$, using only even Walsh functions in the GRAPE optimization. Both the PWC controls with buffer (unfilt) and the Gaussian filtered controls (filt) are shown.

easily accessible with microwave control hardware. Examining the Walsh amplitudes for $N = 16$ in Figure 5.6 shows that only the even functions contribute to the control pulse. Therefore the pulse shapes are symmetric relative to half the gate time $t_g/2$, and the odd Walsh functions $w_{2n+1}(t)$ are dismissed in further GRAPE optimizations. This reduces the number of parameters by a factor of two compared to the standard piecewise constant approach for the same time resolution. Figure 5.7 shows a GRAPE optimization of the pulses where only even Walsh functions have been used.

Now we are trying to reduce the number of parameters even further. Depending on the absolute value of the Walsh amplitudes in Figure 5.6, we reduce the number of Walsh functions step-by-step. We use the optimal result from Figure 5.7 as a starting point for each optimization, set the Walsh amplitudes we want to dismiss to zero and run GRAPE again. This process is continued until the algorithm does not succeed to find an optimal pulse shape anymore. The results for the pulse shapes are shown in Figure 5.8 and for the Walsh amplitudes in Figure 5.9. We manage to skip three Walsh functions for δ_1 and one for δ_2. This leaves us with 12 parameters to optimize, compared to the 32 parameters we have for the standard PWC optimization for a $\tau = 2\,\text{ns}$ time resolution.

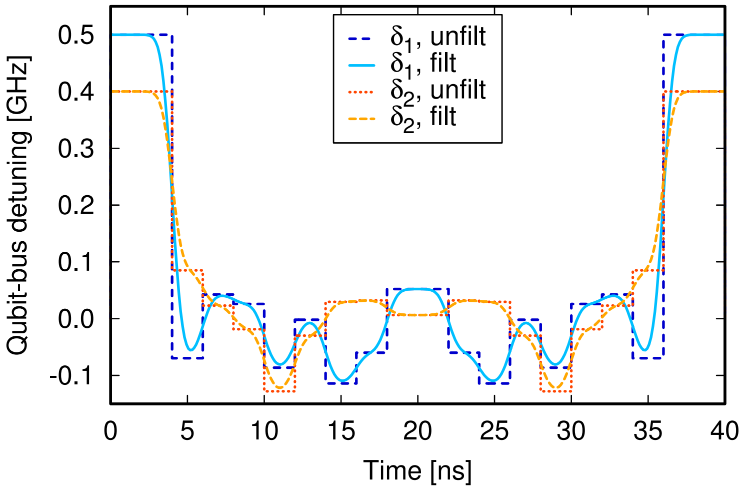

Figure 5.8.: Control pulses for $N = 16$, using only five (δ_1) and seven (δ_2) Walsh functions. Both the Gaussian filtered (filt) and unfiltered (unfilt) pulses are shown.

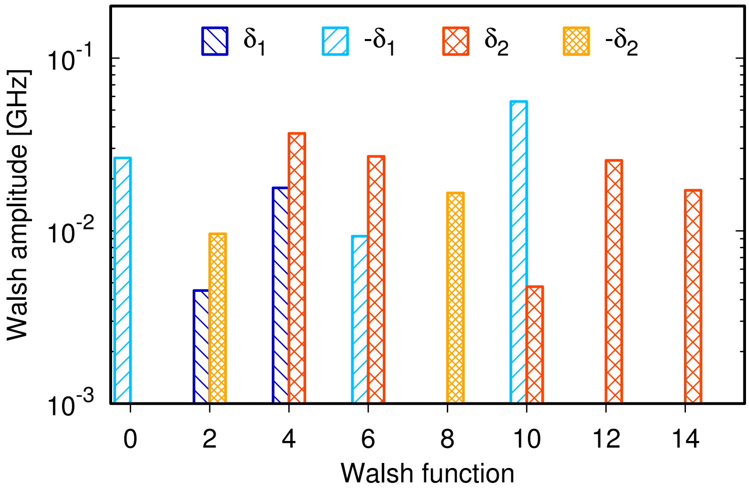

Figure 5.9.: The absolute value of the Walsh amplitudes for $N = 16$, color-coded for positive and negative values. Odd and some even components have been neglected, leading to a total number of twelve parameters; five for the first control (δ_1) and seven for the second (δ_2), which are sufficient for GRAPE to succeed.

5.4. Summary

We have introduced the Walsh functions and extended GRAPE to use the Walsh parametrization for control pulses, with the goal to reduce the number of optimization parameters. The given examples show that a suitable parametrization can lead to a significant simplification of the optimization problem and pulse description for the problem at hand. Due to their digital nature, Walsh functions are easy to implement in conventional electronics. Only the time resolution which is dictated by the number of intervals and the gate time sets a limit for possible implementations. Furthermore, it should be straightforward to exchange the GRAPE algorithm with GOAT for optimizing the Walsh amplitudes.

6. Resonant and dispersive CZ gate

We present two implementations of the CZ gate on the RezQu architecture. The gate is performed by carefully tuning the qubit frequencies and using the second excited level of one transmon to accumulate the necessary phases. Such a sequence is known as the Strauch gate [61] for freuqency tunable qubits and is adapted from the Cirac-Zoller gate [71] in ion traps. In the resonant implementation we perform the gate between the bus and the first qubit, preceded and succeeded by a swap of excitations between the second qubit and the bus. In the second case we work in the dispersive regime and implement the gate directly between the qubits, making use of the virtual photon coupling mediated by the resonator. While in the first case the gate can be performed faster due to the stronger coupling, the second case is more robust against Purcell decay in the resonator.

6.1. Resonant gate

As we have already discussed in Chapter 4, the RezQu architecture consists of a two ore more transmon qubits coupled to a common transmission line [57, 58, 59, 60]. For each qubit the detuning from the bus can be controlled, such that they can be brought in and out of resonance with the bus.

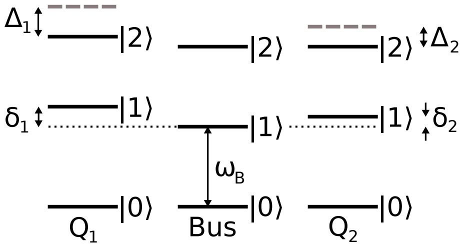

Figure 6.1.: Energy level diagram of the two-qubit RezQu architecture. The detunings can be controlled to bring the qubit levels in resonance with the bus, while the anharmonicities stay fixed.

Figure 6.1 shows the energy diagram with the variable detunings. We use a three-level approximation for each transmon, such that the Hamiltonian in a frame rotating with ω_B (see Chapter 4) is given by

$$H(t) = \sum_{i,j=1}^{2} \delta_j^{(i)}(t)\Pi_j^{(i)} + g_j^{(i)}\left(a^\dagger \Pi_{j-1,j}^{(i)} + a\Pi_{j,j-1}^{(i)}\right), \tag{6.1}$$

with the projector $\Pi_j^{(i)}$ and the hopping element $\Pi_{j,k}^{(i)}$. Label i refers to the qubit and the labels j and k to the energy level. The controllable detuning from the bus is defined as

$$\delta_j^{(i)}(t) = \delta_j^{(i)} + ju^{(i)}(t) = \omega_j^{(i)} - j\omega_B + ju^{(i)}(t), \tag{6.2}$$

with the ith control field $u^{(i)}(t)$. For the transmon the coupling strength to the bus is set to $g_j^{(i)} = \sqrt{j}g_i$, and in Table 6.1 we list the parameters of the system we use in our simulations. The states are labelled as $|n_B; n_1 n_2\rangle$, where the first index is the bus excitation, the second index the excitation of the first qubit Q_1 and the last index the excitation of the second qubit Q_2.

Since the qubits are not directly coupled to each other, the CZ gate is implemented as follows: First, qubit Q_2 is brought in resonance with the bus. This leads to an iSwap gate between the bus and the qubit when the timing is done right, and the excitation from the qubit is transferred to the resonator. Next, the detuning of qubit Q_1 is changed such that its second level is on resonance with the bus. If so, the state $|1;10\rangle$ exchanges its energy with $|0;20\rangle$, and after twice the swap time the state $|1;10\rangle$ has accumulated a phase shift of π. Therefore, a CZ gate between bus and qubit Q_1 is performed. Swapping the excitation between a bus/qubit and the second level of a qubit is known as Strauch sequence [61]. Finally, we have to swap back the excitation from the bus to the qubit Q_2, again by detuning

	Qubit 1	Qubit 2
$\omega_i/2\pi$	6.6 GHz	6.5 GHz
$\delta_i/2\pi$	0.5 GHz	0.4 GHz
$\Delta_i/2\pi$	−200 MHz	−167 MHz
$g_i/2\pi$	30 MHz	25 MHz

Table 6.1.: The RezQu parameters for the Strauch gate implementation with a bus frequency at $\omega_B/2\pi = 6.1$ GHz.

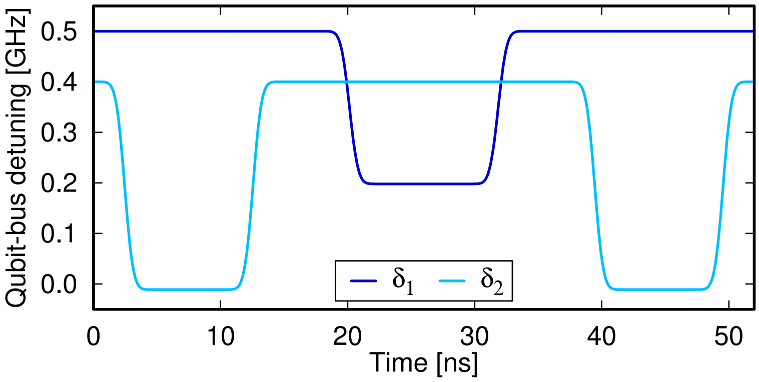

Figure 6.2.: The pulse shape for a resonant Strauch sequence. First, the second qubit is brought in resonance with the bus for a swap of qubit excitation, then the second level of the first qubit is brought in resonance with the bus to accumulated the necessary phase shift. Finally, the excitation is swapped back from the resonator to the second qubit.

the frequency of the second qubit. Then, up to local phases, we implemented a CZ gate between qubit Q_1 and Q_2. Figure 6.2 shows the time dependent detunings of this sequence. We refer to this sequence as resonant Strauch gate, since qubits and bus are always brought onto resonance to perform the desired swap and phase operations.

For the simulation we use erf-function pulse shapes with a standard deviation of $\sigma = 0.53\,\text{ns}$ and set the gate time to $t_g = 52\,\text{ns}$. The difficulty lies in tuning to the right frequency values of the qubits to bring then in resonance with the bus and timing the duration of the pulses, since interaction with other levels leads to perturbations of the ideal case [60] through unwanted couplings and phase accumulation. The target gate takes into account local phases and is set to

$$U_{CZ} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{i\varphi_2} & 0 & 0 \\ 0 & 0 & e^{i\varphi_1} & 0 \\ 0 & 0 & 0 & -e^{i(\varphi_1+\varphi_2)} \end{bmatrix}, \tag{6.3}$$

where φ_1 and φ_1 are the local phases of qubit Q_1 and Q_2, respectively. By tuning the parameters carefully we are able to reach a trace fidelity of $\Phi = 0.989$.

6.2. Dispersive gate

Another way to apply a CZ gate is to work in the dispersive regime [72]. In general this is a unitary transformation of the system such that the qubit-bus coupling terms are replaced by an effective qubit-qubit coupling. In most cases it is sufficient to approximate the transformation into second order of a small parameter. This approximation can be applied if the coupling strengths are much smaller than the detunings of each transmon level, i. e., if the condition

$$g_j^{(i)} \ll \left|\delta_j^{(i)} - \delta_{j-1}^{(i)}\right| \tag{6.4}$$

holds. However, the approximation is no longer unitary. The Hamiltonian (6.1) can be split into two parts [73]

$$H = H_0 + V\,, \tag{6.5}$$

where H_0 is diagonal with the energies of the bare states $|n_B; n_1 n_2\rangle$

$$H_0 = \sum_{i,j=1}^{2} \delta_j^{(i)}(t)\Pi_j^{(i)} \tag{6.6}$$

and V is the interaction term of the qubits with the bus

$$V = \sum_{i,j=1}^{2} g_j^{(i)} \left(a^\dagger \Pi_{j-1,j}^{(i)} + a\Pi_{j,j-1}^{(i)}\right)\,. \tag{6.7}$$

We want to point out that the Hamiltonian is block diagonal and commutes with the number operator of the system

$$N = a^\dagger a + \sum_{i,j=1}^{2} j\Pi_j^{(i)} \qquad [H_0, N] = [V, N] = 0\,. \tag{6.8}$$

In the dispersive limit the interaction term V is assumed to be a small perturbation of the bare states. Now we apply the dispersive transformation [73, 72] onto the Hamiltonian

$$H_D = e^{-S} H e^{S} \tag{6.9}$$

where the operator S is the generator of the transformation and defined through

$$S = \sum_{i,j=1}^{2} \lambda_j^{(i)} \left(a^\dagger \Pi_{j-1,j}^{(i)} - a\Pi_{j,j-1}^{(i)}\right)\,. \tag{6.10}$$

The generator also commutes with the number operator $[S, N] = 0$. Using the Baker-Campbell-Hausdorff formula one gets

$$H_D = e^{-S} H e^{S} = H + [H, S] + \frac{1}{2}[[H, S], S] \cdots = \sum_{j=0} \frac{1}{j!}[H, S]^{(j)} . \tag{6.11}$$

This transformation is also known as Schrieffer-Wolff-Transformation [74]. Setting the evolution parameters to

$$\lambda_j^{(i)} = \frac{g_j^{(i)}}{\delta_j^{(i)} - \delta_{j-1}^{(i)}} \qquad (\delta_0^{(i)} = 0) \tag{6.12}$$

yields for the first commutator in (6.11)

$$V = -[H, S] . \tag{6.13}$$

Up to second order in the small coupling $g_j^{(i)}$ the effective Hamiltonian then reads

$$H_{\text{eff}} = H_0 + \frac{1}{2}[V, S] . \tag{6.14}$$

Introducing the energy shift

$$\chi_j^{(i)} = g_j^{(i)} \lambda_j^{(i)} , \tag{6.15}$$

the effective Hamiltonian contains the following four terms

$$\begin{aligned} H_{\text{eff}} = &\sum_{i,j=1}^{2} \chi_j^{(i)} a^\dagger a \left(\Pi_j^{(i)} - \Pi_{j-1}^{(i)} \right) + \left(\delta_j^{(i)} + \chi_j^{(i)} \right) \Pi_j^{(i)} \\ &+ \frac{1}{2} \sum_{j,l=1}^{2} \left(g_j^{(1)} \lambda_l^{(2)} + g_l^{(2)} \lambda_j^{(1)} \right) \left(\Pi_{j-1,j}^{(1)} \Pi_{l,l-1}^{(2)} + \Pi_{j,j-1}^{(1)} \Pi_{l-1,l}^{(2)} \right) \\ &+ \frac{1}{2} \sum_{i=1}^{2} \left(g_1^{(i)} \lambda_2^{(i)} - g_2^{(i)} \lambda_1^{(i)} \right) \left(a^\dagger a^\dagger \Pi_{0,2}^{(i)} + a a \Pi_{2,0}^{(i)} \right) . \end{aligned} \tag{6.16}$$

The first and second term are the ac-Stark and Lamb shift, respectively, the third term is the virtual photon interaction between the two qubits and the last term describes two-photon interactions between the qubits and the resonator. We observe that for $N = 0$ the transformation does not change the energy, and the ground state remains dark to radiation processes [75].

However, this changes for $N > 0$ when the system is excited. Writing out the Hamiltonian for $N = 1$

$$H_{\mathrm{eff}}^{N=1} = \begin{bmatrix} \delta_1^{(2)} + \chi_1^{(2)} & \lambda_1^{(1)}\lambda_1^{(2)}\frac{\delta_1^{(1)}+\delta_1^{(2)}}{2} & 0 \\ & \delta_1^{(1)} + \chi_1^{(1)} & 0 \\ \text{h.c.} & & -\chi_1^{(1)} - \chi_1^{(2)} \end{bmatrix} \tag{6.17}$$

reveals that we get a negative energy shift of the bus level if none of the qubits is excited, and a positive energy shift for a qubit excitation, which is used for system characterization and readout [76, 77]. Furthermore, we see that the coupling between the qubits scales with g^2/δ. For the Strauch sequence we want to accumulate the π-phase through the states $|0;11\rangle$ and $|0;20\rangle$. In the second order dispersive regime, the $N = 2$ subspace Hamiltonian splits into two blocks: The first block is spanned by the states where one excitation resides in the bus and the other excitation in one of the qubits, with the Hamiltonian

$$H_{\mathrm{eff}}^{N=2} = \begin{bmatrix} \delta_1^{(2)} + 2\chi_1^{(2)} - \chi_2^{(2)} - \chi_1^{(1)} & \lambda_1^{(1)}\lambda_1^{(2)}\frac{\delta_1^{(1)}+\delta_1^{(2)}}{2} \\ \text{h.c.} & \delta_1^{(1)} + 2\chi_1^{(1)} - \chi_2^{(1)} - \chi_1^{(2)} \end{bmatrix} \tag{6.18}$$

and the same coupling strength as in the $N = 1$ case. The second block contains the remaining four states, including the two-photon couplings

$$H_{\mathrm{eff}}^{N=2} = \tag{6.19}$$
$$\begin{bmatrix} \delta_2^{(2)} + \chi_2^{(2)} & \lambda_1^{(1)}\lambda_2^{(2)}\frac{\delta_1^{(1)}+\delta_2^{(2)}}{2} & 0 & \lambda_1^{(2)}\lambda_2^{(2)}\frac{\delta_1^{(2)}-\delta_2^{(2)}}{2} \\ & \delta_1^{(1)} + \chi_1^{(1)} + \delta_1^{(2)} + \chi_1^{(2)} & \lambda_2^{(1)}\lambda_1^{(2)}\frac{\delta_2^{(1)}+\delta_1^{(2)}}{2} & 0 \\ \vdots & & \delta_2^{(1)} + \chi_2^{(1)} & \lambda_1^{(1)}\lambda_2^{(1)}\frac{\delta_1^{(1)}-\delta_2^{(1)}}{2} \\ \text{h.c.} & \cdots & & -2\chi_1^{(1)} - 2\chi_1^{(2)} \end{bmatrix}.$$

We note that the effect of the two-photon coupling between the qubits and the bus is rather small, and the major contribution to the perturbation stems from the unwanted coupling of the $|0;11\rangle$ to the $|0;02\rangle$ state.

Figure 6.3 shows the pulse shape for the Strauch gate for a gate time of $t_g = 207.8\,\mathrm{ns}$, where the resonant condition reads

$$\delta_1^{(1)} + \chi_1^{(1)} + \delta_1^{(2)} + \chi_1^{(2)} = \delta_2^{(1)} + \chi_2^{(1)} \tag{6.20}$$

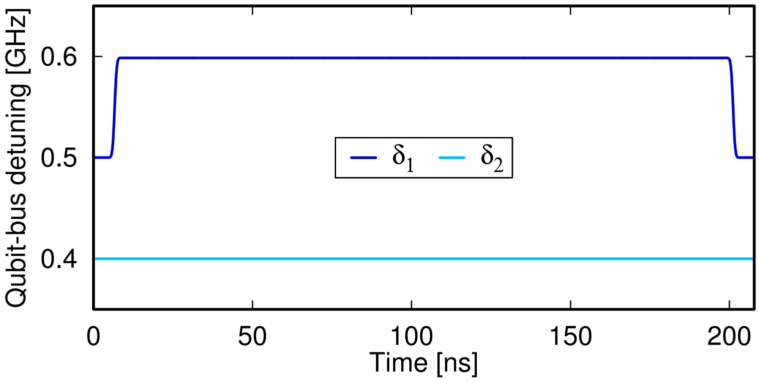

Figure 6.3.: Strauch sequence in the dispersive regime. The second level of the first qubit is brought in resonance with the first level of the second qubit, such that a Strauch gate is performed.

and the on resonance time is approximately

$$t_{\mathrm{on}} \approx \frac{\pi}{g_{\mathrm{eff}}} = \frac{2\pi\delta_2^{(1)}\delta_1^{(2)}}{\sqrt{2}g_2g_1(\delta_2^{(1)} + \delta_1^{(2)})}\,. \tag{6.21}$$

The detuning of the second qubit Q_2 is kept constant, and equation 6.20 gives an iterative formula for the detuning of the first qubit $\delta_1^{(1)}$, with the detuning of the second level $\delta_2^{(1)} = 2\delta_1^{(1)} + \Delta^{(i)}$. The simulation is done with the full Hamiltonian to minimize the effects of the dispersive approximation. With careful adjustment of the parameters we achieve a trace fidelity of $\Phi = 0.998$.

6.3. Decoherence

Every quantum system undergoes some decoherence. Here we want to compare how the two CZ gate implementations suffer from Purcell decay of the resonator and qubit relaxation. We model the decoherence mechanism by a Lindblad master equation [3]

$$\frac{\partial\rho}{\partial t} = -i[H,\rho] + D_{\mathrm{Purcell}}(\rho) + \sum_{i=1}^{2} D_1^{(i)}(\rho) + D_2^{(i)}(\rho) \tag{6.22}$$

and make the assumptions of a cold bath,i. e., we are only taking the decay of excitations into account. For finite temperatures the interaction with the environment would also lead to random excitations of the system with small probability. The dissipators are defined as

$$D(\rho) = L\rho L^\dagger - \frac{1}{2}L^\dagger L\rho - \frac{1}{2}\rho L^\dagger L\,, \tag{6.23}$$

with the Lindblad operator L. For the Purcell decay of the resonator the Lindblad operator reads

$$L_{\text{Purcell}} = \sqrt{\kappa}a \qquad L^\dagger_{\text{Purcell}}L_{\text{Purcell}} = \kappa a^\dagger a\,. \tag{6.24}$$

Therefore κ describes the photon loss rate of the resonator. The qubits are prone to decoherence as well. First, the T_1 decay due to radiationless relaxation for qubit i is given by

$$L_1^{(i)} = \sqrt{\gamma_1^{(i)}}\sum_{j=1}^{2}\sqrt{j}\Pi_{j-1,j}^{(i)} \qquad L_1^\dagger L_1^{(i)} = \gamma_1^{(i)}\sum_{j=1}^{2} j\Pi_j^{(i)}\,, \tag{6.25}$$

and $T_1 = 1/\gamma_1$ is the relaxation time constant. Additionally, we have a pure dephasing process T_φ with Lindblad operators

$$L_2^{(i)} = \sqrt{\gamma_2^{(i)}}\sum_{j=1}^{2}\sqrt{j}\Pi_j^{(i)} \qquad L_2^\dagger L_2^{(i)} = \gamma_2^{(i)}\sum_{j=1}^{2} j\Pi_j^{(i)}\,, \tag{6.26}$$

where $T_\varphi = 2/\gamma_2$ is the dephasing time and the dephasing operator is chosen such that

$$2T_1 L_1^\dagger L_1 = T_\varphi L_2^\dagger L_2 \tag{6.27}$$

holds, omitting the qubit index for simplicity in the last equation. Now the T_2 time is given by the relation

$$\frac{1}{T_2} = \frac{1}{2T_1} + \frac{1}{T_\varphi}\,. \tag{6.28}$$

We perform two simulations with $T_2 = 0$ and $T_2 = T_1$ and let κ and γ_1 loop through several orders of magnitude. Figure 6.4 shows the increase of infidelity of the resonant CZ gate for $T_2 = 0$ and for $T_2 = T_1$, and Figure 6.5 shows the infidelities of the dispersive CZ gate for different values of the Purcell decay rate and the qubit relaxation rate for $T_2 = 0$ and for $T_2 = T_1$. In the dispersive regime the Lindblad operators change due to the dispersive

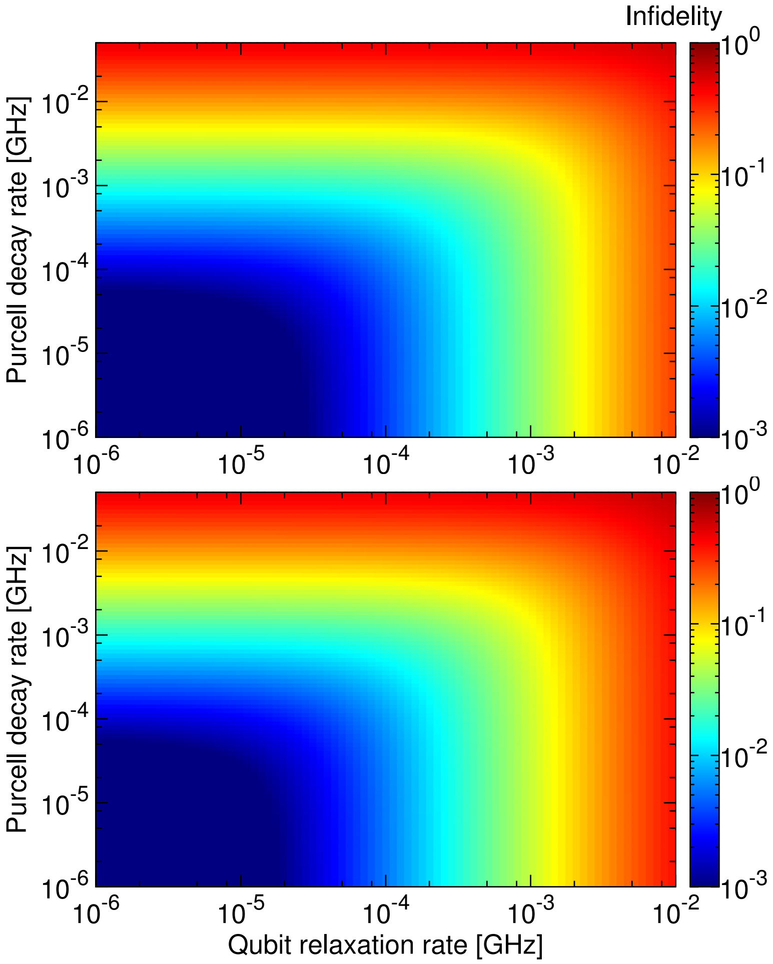

Figure 6.4.: Increase of infidelity for different parameter values for the resonant CZ gate, with $T_2 = 0$ (top) and $T_2 = T_1$ (bottom).

transformation of the system. In second order we get for the effective Purcell decay of the resonator

$$L_{\text{Purcell,eff}} = \sqrt{\kappa}\left[a + \sum_{i,j=1}^{2} \lambda_j^{(i)} \Pi_{j-1,j}^{(i)} + \frac{1}{2}\lambda_j^{(i)2} a \left(\Pi_j^{(i)} - \Pi_{j-1}^{(i)}\right)\right] \quad (6.29)$$

We see that through the transformation the Purcell decay also leads to a small relaxation of both qubits in first order of $\lambda_j^{(i)}$, and in second order the decay rate depends on the qubit state. The number operator of the bus

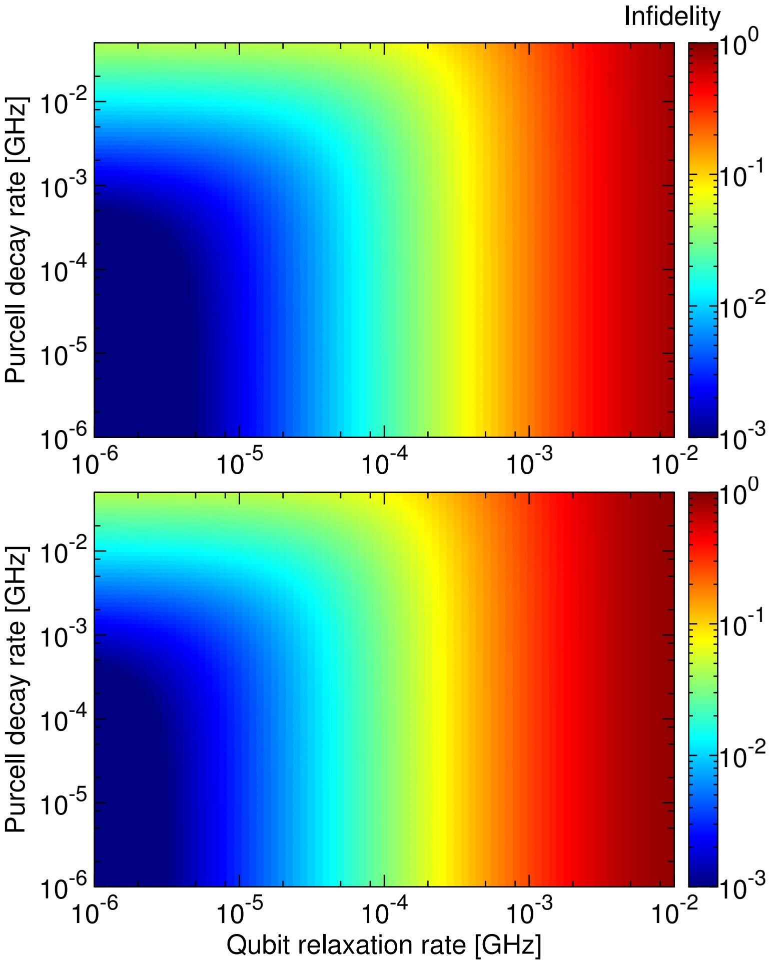

Figure 6.5.: Increase of infidelity for different parameter values for the dispersive CZ gate, with $T_2 = 0$ (top) and $T_2 = T_1$ (bottom).

resonator transforms to

$$\begin{aligned}\left(L^\dagger L\right)_{\text{Purcell,eff}} &= \kappa a^\dagger a + \kappa \sum_{i,j=1}^{2} \lambda_j^{(i)} \left(a^\dagger \Pi_{j-1,j}^{(i)} + a \Pi_{j,j-1}^{(i)}\right) \\ &+ \kappa \sum_{j,l=1}^{j+l=3} \lambda_j^{(1)} \lambda_l^{(2)} \left(\Pi_{j,j-1}^{(1)} \Pi_{l-1,l}^{(2)} + \Pi_{j-1,j}^{(1)} \Pi_{l,l-1}^{(2)}\right) \\ &+ \kappa \sum_{i,j=1}^{2} \lambda_j^{(i)2} \left(\Pi_j^{(i)} + a^\dagger a \left(\Pi_j^{(i)} - \Pi_{j-1}^{(i)}\right)\right) . \qquad (6.30)\end{aligned}$$

Here we observe a similar pattern to the Hamiltonian, with qubit-bus interactions in first order and in second order qubit-qubit interactions and dressed resonator energy. The T_1 Lindblad operator with $k \neq i$ reads

$$\begin{aligned}
L_{1,\text{eff}}^{(i)} = &\sqrt{\gamma_1^{(i)}} \sum_{j=1}^{2} \left[\sqrt{j}\Pi_{j-1,j}^{(i)} + \sqrt{j}\lambda_j^{(i)} a \left(\Pi_j^{(i)} - \Pi_{j-1}^{(i)}\right)\right] \\
&+ \sqrt{\gamma_1^{(i)}} \left(\lambda_2^{(i)} - \sqrt{2}\lambda_1^{(i)}\right) a^\dagger \Pi_{02}^{(i)} \\
&+ \sqrt{\gamma_1^{(i)}} \left(\lambda_2^{(i)} - \sqrt{2}\lambda_1^{(i)}\right) \left(\lambda_1^{(i)} a a \Pi_{10}^{(i)} + \frac{1}{2}\lambda_1^{(k)} \Pi_{02}^{(i)} \Pi_{10}^{(k)}\right) \\
&+ \frac{1}{2}\sqrt{\gamma_1^{(i)}} \sum_{j,l=1}^{j+l=3} \lambda_j^{(i)} \lambda_l^{(k)} \left(\Pi_j^{(i)} - \Pi_{j-1}^{(i)}\right) \Pi_{l-1,l}^{(k)} \\
&- \frac{1}{2}\sqrt{\gamma_1^{(i)}} \lambda_1^{(i)2} \sum_{j=1}^{2} \sqrt{j}\Pi_{j-1,j}^{(i)} \\
&- \frac{1}{2}\sqrt{\gamma_1^{(i)}} (\lambda_2^{(i)} - \sqrt{2}\lambda_1^{(i)})^2 a^\dagger a \Pi_{01}^{(i)} \\
&- \frac{1}{2}\sqrt{\gamma_1^{(i)}} \sqrt{2}\lambda_2^{(i)} (\lambda_2^{(i)} - \sqrt{2}\lambda_1^{(i)}) \Pi_{12}^{(i)} \,,
\end{aligned} \tag{6.31}$$

and we see that this leads to various decay mechanism in first order, like dressed decay of the resonator and decay mechanisms that include two-photon transitions. In second order appears a decay channel of the second qubit depending on the state of the first qubit and the first qubit decay channel depends on the resonator state. The number operator of the ith qubit transforms to

$$\begin{aligned}
\left(L^\dagger L\right)_{1,\text{eff}}^{(i)} = &\gamma_1^{(i)} \sum_{j=1}^{2} \left[j\Pi_j^{(i)} - \lambda_j^{(i)} \left(a^\dagger \Pi_{j-1,j}^{(i)} + a\Pi_{j,j-1}^{(i)}\right)\right] \\
&- \frac{1}{2}\gamma_1^{(i)} \sum_{j,l=1}^{j+l=3} \lambda_j^{(1)} \lambda_l^{(2)} \left(\Pi_{j-1,j}^{(1)} \Pi_{l,l-1}^{(2)} + \Pi_{j,j-1}^{(1)} \Pi_{l-1,l}^{(2)}\right) \\
&- \gamma_1^{(i)} \sum_{j=1}^{2} \lambda_j^{(i)2} \left(\Pi_j^{(i)} + a^\dagger a \left(\Pi_j^{(i)} - \Pi_{j-1}^{(i)}\right)\right) \,.
\end{aligned} \tag{6.32}$$

Similar to the number operator of the bus we get qubit-bus interaction terms, qubit-qubit interaction terms and dressed bus energy terms. The T_φ dephasing operator transforms to

$$
\begin{aligned}
L_{2,\mathrm{eff}}^{(i)} = {} & \sqrt{\gamma_2^{(i)}} \sum_{j=1}^{2} \left[\sqrt{j} \Pi_j^{(i)} + \lambda_j^{(i)} \left(\sqrt{j-1} - \sqrt{j} \right) \left(a^\dagger \Pi_{j-1,j}^{(i)} + a \Pi_{j,j-1}^{(i)} \right) \right] \\
& + \sqrt{\gamma_2^{(i)}} \frac{1}{2} \sum_{j,l=1}^{j+l=3} \lambda_j^{(i)} \lambda_l^{(k)} \left(\sqrt{j-1} - \sqrt{j} \right) \left(\Pi_{j-1,j}^{(i)} \Pi_{l,l-1}^{(k)} + \Pi_{j,j-1}^{(i)} \Pi_{l-1,l}^{(k)} \right) \\
& + \sqrt{\gamma_2^{(i)}} \sum_{j=1}^{2} \lambda_j^{(i)2} \left(\sqrt{j-1} - \sqrt{j} \right) \left(a^\dagger a \left(\Pi_j^{(i)} - \Pi_{j-1}^{(i)} \right) + \Pi_j^{(i)} \right) \\
& + \sqrt{\gamma_2^{(i)}} \lambda_1^{(i)} \lambda_2^{(i)} \left(1 - \sqrt{2} \right) \left(a a \Pi_{20}^{(i)} + a^\dagger a^\dagger \Pi_{02}^{(i)} \right) . \qquad (6.33)
\end{aligned}
$$

Again qubit-bus interactions appear in first order and qubit-qubit interactions in second order. Besides the dressed resonator energy in second order we also get two-photon interaction terms between the qubit and the bus.

Figure 6.6 shows the difference in fidelity of the resonant and the dispersive gate. We see that the dispersive gate only works better if the Purcell decay rate is $\kappa > 10^{-5}$ and qubit relaxation processes are slow. Otherwise the resonant gate outperforms the dispersive gate. The disadvantage of the resonant gate is that it needs more parameter tuning and is more error prone to unwanted transitions out of the computational subspace. But the dispersive gate lacks of speed (in our simulations four times longer) and the allowed detunings are restricted to keep the qubit frequencies far detuned from the resonator.

6.4. Extended dispersive regime

With either the tunable detuning or higher transmon levels we can reach the regime where the approximation $\left|\delta_j^{(i)} - \delta_{j-1}^{(i)}\right| \gg g_j^{(i)}$ is no longer valid. Therefore we have to account higher order terms of the transformation to find an analytic expression that allows us to decouple the qubits from the bus for all regimes. We will not show the complete diagonalization here, which requires careful analysis of very high orders of the transformation. These terms can get very long, so instead we write out the effective Hamiltonian in fourth order for the two transmons coupled to the resonator. First we start with the exact diagonalization for a qubit coupled to a resonator.

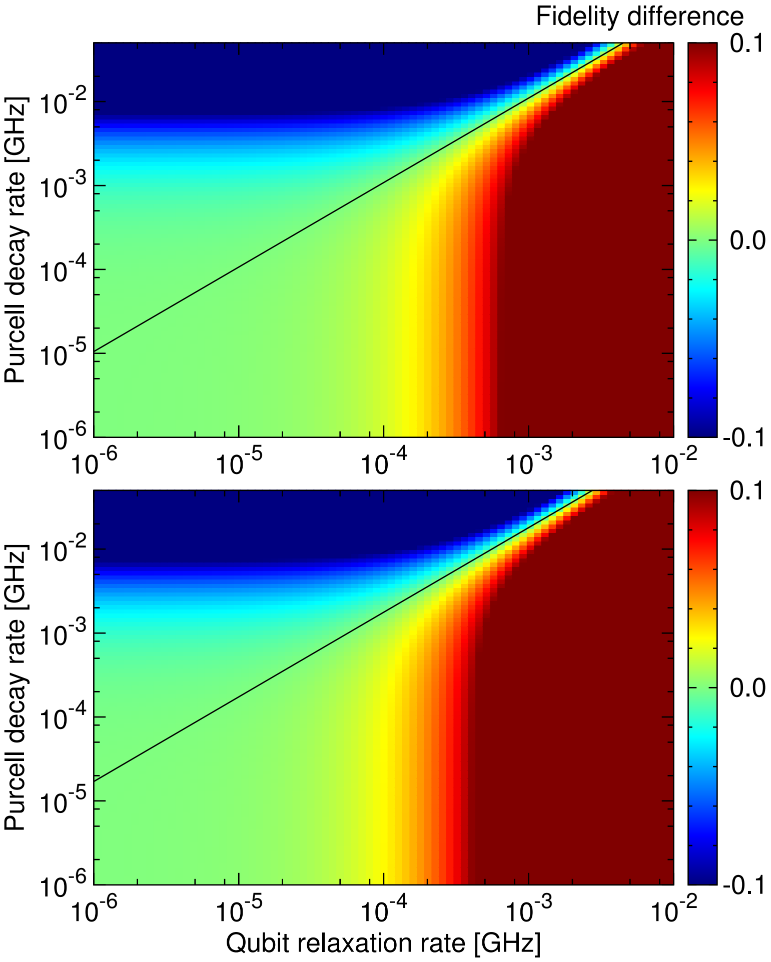

Figure 6.6.: Difference of the fidelity for different values of the Purcell decay rate and the qubit relaxation rate, for $T_2 = 0$ (top) and $T_2 = T_1$ (bottom). The solid lines show the cut-off where the dispersive gate works better than the resonant gate for strong Purcell decay rates.

Using the dispersive transformation [73, 72] up to arbitrary order it reads

$$\lambda = \frac{1}{2}\arctan\left(2\sqrt{N}g/\delta\right)/\sqrt{N}, \tag{6.34}$$

and if g is small one gets the approximation we used earlier in Section 6.2. We note that this solutions still holds if g/δ diverges. Now for two transmons we focus on the $N = 1$ case, i. e. the transmons can be interpreted as two-level systems, making this the first generalization of the effective decoupling of the bus. The second generalization stems from higher levels in the transmons, which will only affect the decoupling for $N \geqslant 2$. With the number operators defined as

$$N_0 = a^\dagger a \quad N_i = \sum_j j\Pi_j^{(i)} \quad N = N_0 + \sum_i N_i \tag{6.35}$$

We note that the following conditions hold

$$[S_i, N_0] = [N_i, S_i] \quad [V_i, N_0] = [N_i, V_i] \quad [H_0, N_0] = [H_0, N_i] = 0\,, \tag{6.36}$$

with S_i and V_i the generator and interaction term for each qubit, i. e., the generator reads $S = S_1 + S_2$ and the interaction term $V = V_1 + V_2$. Now we give the full dispersive transformation for the RezQu architecture up to 4th order in λ_i, assuming g as (large) perturbation of the diagonal Hamiltonian H_0 and therefore counting it as first order term

$$\begin{aligned} H_{\text{eff}} \approx\ & H_0 + V + [H_0, S] + [V, S] + \frac{1}{2}[H_0, S]^{(2)} \\ & + \frac{1}{2}[V, S]^{(2)} + \frac{1}{6}[H_0, S]^{(3)} + \frac{1}{6}[V, S]^{(3)} + \frac{1}{24}[H_0, S]^{(4)}\,. \end{aligned} \tag{6.37}$$

In here, we use the convention $\lambda_1^{(i)} = \lambda_i$ (not be confused with the notation we used earlier). For the $N = 1$ subspace we get for the diagonal terms

$$\begin{aligned} \langle 0;01|H_{\text{eff}}|0;01\rangle \approx\ & \delta_2 + 2g_2\lambda_2 - \delta_2\lambda_2^2 - (g_1\lambda_1 + g_2\lambda_2)\lambda_2^2 - \frac{g_2\lambda_2}{3}\left(\lambda_1^2 + \lambda_2^2\right) \\ & + \frac{\delta_1\lambda_1^2 + \delta_2\lambda_2^2}{4}\lambda_2^2 + \frac{\delta_2\lambda_2^2}{12}\left(\lambda_1^2 + \lambda_2^2\right) \\ \langle 0;10|H_{\text{eff}}|0;10\rangle \approx\ & \delta_1 + 2g_1\lambda_1 - \delta_1\lambda_1^2 - (g_1\lambda_1 + g_2\lambda_2)\lambda_1^2 - \frac{g_1\lambda_1}{3}\left(\lambda_1^2 + \lambda_2^2\right) \\ & + \frac{\delta_1\lambda_1^2 + \delta_2\lambda_2^2}{4}\lambda_1^2 + \frac{\delta_1\lambda_1^2}{12}\left(\lambda_1^2 + \lambda_2^2\right) \\ \langle 1;00|H_{\text{eff}}|1;00\rangle \approx\ & -2(g_1\lambda_1 + g_2\lambda_2) + \delta_1\lambda_1^2 + \delta_2\lambda_2^2 \\ & + \frac{4}{3}(g_1\lambda_1 + g_2\lambda_2)\left(\lambda_1^2 + \lambda_2^2\right) - \frac{\delta_1\lambda_1^2 + \delta_2\lambda_2^2}{3}\left(\lambda_1^2 + \lambda_2^2\right)\,, \end{aligned} \tag{6.38}$$

which only contain even orders. The qubit-bus coupling terms contain only odd order terms and read

$$\begin{aligned}\langle 0;01|H_{\text{eff}}|1;00\rangle &\approx g_2 - \delta_2\lambda_2 - \frac{3}{2}\left(g_1\lambda_1 + g_2\lambda_2\right)\lambda_2 - \frac{g_2}{2}\left(\lambda_1^2 + \lambda_2^2\right)\\ &+ \frac{\delta_1\lambda_1^2 + \delta_2\lambda_2^2}{2}\lambda_2 + \frac{\delta_2\lambda_2}{6}\left(\lambda_1^2 + \lambda_2^2\right)\\ \langle 0;10|H_{\text{eff}}|1;00\rangle &\approx g_1 - \delta_1\lambda_1 - \frac{3}{2}\left(g_1\lambda_1 + g_2\lambda_2\right)\lambda_1 - \frac{g_1}{2}\left(\lambda_1^2 + \lambda_2^2\right)\\ &+ \frac{\delta_1\lambda_1^2 + \delta_2\lambda_2^2}{2}\lambda_1 + \frac{\delta_1\lambda_1}{6}\left(\lambda_1^2 + \lambda_2^2\right), \end{aligned} \tag{6.39}$$

and the qubit-qubit coupling is again of even order and reads

$$\begin{aligned}\langle 0;01|H_{\text{eff}}|0;10\rangle &\approx g_2\lambda_1 + g_1\lambda_2 - \frac{(\delta_1 + \delta_2)}{2}\lambda_1\lambda_2\\ &- \left(g_1\lambda_1 + g_2\lambda_2\right)\lambda_1\lambda_2 - \frac{g_2\lambda_1 + g_1\lambda_2}{6}\left(\lambda_1^2 + \lambda_2^2\right)\\ &+ \frac{\delta_1\lambda_1^2 + \delta_2\lambda_2^2}{4}\lambda_1\lambda_2 + \frac{\delta_1 + \delta_2}{24}\lambda_1\lambda_2\left(\lambda_1^2 + \lambda_2^2\right). \end{aligned} \tag{6.40}$$

Setting the general solution for a single qubit coupled to a bus (for all ratios g_i/δ_i)

$$\lambda_i = \frac{1}{2}\arctan\left(2g_i/\delta_i\right) \approx \frac{g_i}{\delta_i} - \frac{4g_i^3}{3\delta_i^3} \tag{6.41}$$

and going to third order still leaves some coupling terms between the bus and the qubits, i. e.

$$\begin{aligned}\langle 0;01|H_{\text{eff}}|1;00\rangle &\approx -\frac{g_1^2 g_2}{3\delta_1^2} - \frac{g_1^2 g_2}{\delta_1\delta_2}\\ \langle 0;10|H_{\text{eff}}|1;00\rangle &\approx -\frac{g_1 g_2^2}{\delta_1\delta_2} - \frac{g_1 g_2^2}{3\delta_2^2}, \end{aligned} \tag{6.42}$$

which cannot be compensated since higher orders cannot cancel these terms. Therefore the analytic expression of λ_i must contain some parameters from the other qubit $j \neq i$, such that it is a function of the three number operators N_k, both coupling constants g_k and the detunings δ_k.

For $N = 2$ the general expressions are getting a bit more complicated. Beside having two qubits coupled to the bus, each of them contains a second energy level. We have already seen in Section 6.2 that in second order terms appear that couple the second excited state of each qubit to the second excited state of the bus resonator, i. e., the two-photon transition terms. To compensate for this we have to add a another term to the generator which is of second order [78]

$$T = \sum_{i=1}^{2} \eta^{(i)} \left(a^\dagger a^\dagger \Pi_{02}^{(i)} - aa\Pi_{20}^{(i)} \right) , \tag{6.43}$$

such that the effective Hamiltonian for $N = 2$ in second order is obtained through

$$H_{\text{eff}} \approx H_0 + V + [H_0, S] + [V, S] + \frac{1}{2}[H_0, S]^{(2)} + [H_0, T] \,. \tag{6.44}$$

We not that in the $N = 1$ subspace the generator T vanishes and terms in that subspace are not affected. Setting $\lambda_1^{(i)} = \lambda_i$, $\lambda_2^{(i)} = \Lambda_i$ and $\eta^{(i)} = \eta_i$, we get for the six energy terms

$$\begin{aligned}
\langle 0;02|H_{\text{eff}}|0;02\rangle &\approx 2\delta_2 + \Delta_2 + 2\sqrt{2}g_2\Lambda_2 - (\delta_2 + \Delta_2)\Lambda_2^2 \\
\langle 0;11|H_{\text{eff}}|0;11\rangle &\approx \delta_1 + \delta_2 + 2\left(g_1\lambda_1 + g_2\lambda_2\right) - \left(\delta_1\lambda_1^2 + \delta_2\lambda_2^2\right) \\
\langle 0;20|H_{\text{eff}}|0;20\rangle &\approx 2\delta_1 + \Delta_1 + 2\sqrt{2}g_1\Lambda_1 - (\delta_1 + \Delta_1)\Lambda_1^2 \\
\langle 1;01|H_{\text{eff}}|1;01\rangle &\approx \delta_2 + 2g_2\left(2\lambda_2 - \sqrt{2}\Lambda_2\right) - 2g_1\lambda_1 \\
&\quad - 2\delta_2\lambda_2^2 + (\delta_2 + \Delta_2)\Lambda_2^2 + \delta_1\lambda_1^2 \\
\langle 1;10|H_{\text{eff}}|1;10\rangle &\approx \delta_1 + 2g_1\left(2\lambda_1 - \sqrt{2}\Lambda_1\right) - 2g_2\lambda_2 \\
&\quad - 2\delta_1\lambda_1^2 + (\delta_1 + \Delta_1)\Lambda_1^2 + \delta_2\lambda_2^2 \\
\langle 2;00|H_{\text{eff}}|2;00\rangle &\approx -4\left(g_1\lambda_1 + g_2\lambda_2\right) + 2\left(\delta_1\lambda_1^2 + \delta_2\lambda_2^2\right) \,,
\end{aligned} \tag{6.45}$$

and we see that in second order the diagonal elements are not affected by the extension of the generator. For the coupling terms between bus and qubits we get

$$\begin{aligned}
\langle 0;02|H_{\text{eff}}|1;01\rangle &\approx \sqrt{2}g_2 - (\delta_2 + \Delta_2)\Lambda_2 \\
\langle 0;20|H_{\text{eff}}|1;10\rangle &\approx \sqrt{2}g_1 - (\delta_1 + \Delta_1)\Lambda_1 \\
\langle 0;11|H_{\text{eff}}|1;01\rangle &\approx g_1 - \delta_1\lambda_1 \\
\langle 0;11|H_{\text{eff}}|1;10\rangle &\approx g_2 - \delta_2\lambda_2 \\
\langle 1;01|H_{\text{eff}}|2;00\rangle &\approx \sqrt{2}g_2 - \sqrt{2}\delta_2\lambda_2 \\
\langle 1;10|H_{\text{eff}}|2;00\rangle &\approx \sqrt{2}g_1 - \sqrt{2}\delta_1\lambda_1 \,,
\end{aligned} \tag{6.46}$$

and the virtual photon couplings between the qubits are

$$\begin{aligned}\langle 0;02|H_{\text{eff}}|0;11\rangle &\approx g_1\Lambda_2 + \sqrt{2}g_2\lambda_1 - \frac{\delta_1+\delta_2+\Delta_2}{2}\Lambda_2\lambda_1\\ \langle 0;11|H_{\text{eff}}|0;20\rangle &\approx g_2\Lambda_1 + \sqrt{2}g_1\lambda_2 - \frac{\delta_2+\delta_1+\Delta_1}{2}\Lambda_1\lambda_2\\ \langle 1;01|H_{\text{eff}}|1;10\rangle &\approx g_2\lambda_1 + g_1\lambda_2 - \frac{\delta_1+\delta_2}{2}\lambda_1\lambda_2\,. \end{aligned} \tag{6.47}$$

Now the virtual two-photon transition terms contain the second order coefficients η_i

$$\begin{aligned}\langle 0;02|H_{\text{eff}}|2;00\rangle &\approx -g_2\left(2\lambda_2 - \sqrt{2}\Lambda_2\right) + \frac{\Delta_2}{\sqrt{2}}\lambda_2\Lambda_2 - \sqrt{2}\eta_2(2\delta_2+\Delta_2)\\ \langle 0;20|H_{\text{eff}}|2;00\rangle &\approx -g_1\left(2\lambda_1 - \sqrt{2}\Lambda_1\right) + \frac{\Delta_1}{\sqrt{2}}\lambda_1\Lambda_1 - \sqrt{2}\eta_1(2\delta_1+\Delta_1)\,. \end{aligned} \tag{6.48}$$

If we would stop here, we can determine η_i such that the two-photon transition terms vanish, and the $\lambda_j^{(i)}$ are chosen according to Section 6.2. Now we include third and fourth order terms as well. For this we have to extend the generator even further

$$S = S^{(1)} + S^{(2)} + S^{(3)} + S^{(4)}\,, \tag{6.49}$$

with $S^{(1)} = S$ and $S^{(2)} = T$ and the index indicates the lowest order of the generator. The third order effective Hamiltonians therefore reads

$$\begin{aligned}H_{\text{eff}}^3 = &\left[H_0, S^{(3)}\right] + \frac{1}{2}\left[\left[H_0, S^{(2)}\right], S^{(1)}\right] + \frac{1}{2}\left[\left[H_0, S^{(1)}\right], S^{(2)}\right]\\ &+ \frac{1}{6}\left[H_0, S^{(1)}\right]^3 + \left[V, S^{(2)}\right] + \frac{1}{2}\left[V, S^{(1)}\right]^2, \end{aligned} \tag{6.50}$$

with the third order generator

$$S^{(3)} = \xi_1\left(a^\dagger\Pi_{02}^{(1)}\Pi_{10}^{(2)} - a\Pi_{20}^{(1)}\Pi_{01}^{(2)}\right) + \xi_2\left(a^\dagger\Pi_{10}^{(1)}\Pi_{02}^{(2)} - a\Pi_{01}^{(1)}\Pi_{20}^{(2)}\right)\,, \tag{6.51}$$

which compensates for two-photon two-qubit-bus coupling terms. The fourth order generator reads

$$S^{(4)} = \zeta\left(a^\dagger a^\dagger\Pi_{01}^{(1)}\Pi_{01}^{(2)} - aa\Pi_{10}^{(1)}\Pi_{10}^{(2)}\right)\,, \tag{6.52}$$

which is yet another two-qubit-bus coupling term. Therefore the forth order effective Hamiltonian is

$$\begin{aligned}H^4_{\text{eff}} =& \left[H_0, S^{(4)}\right] + \frac{1}{2}\left[\left[H_0, S^{(3)}\right], S^{(1)}\right] + \frac{1}{2}\left[\left[H_0, S^{(2)}\right], S^{(2)}\right] \\ &+ \frac{1}{2}\left[\left[H_0, S^{(1)}\right], S^{(3)}\right] + \frac{1}{6}\left[\left[\left[H_0, S^{(2)}\right], S^{(1)}\right], S^{(1)}\right] \\ &+ \frac{1}{6}\left[\left[\left[H_0, S^{(1)}\right], S^{(2)}\right], S^{(1)}\right] + \frac{1}{6}\left[\left[\left[H_0, S^{(1)}\right], S^{(1)}\right], S^{(2)}\right] \\ &+ \frac{1}{24}\left[H_0, S^{(1)}\right]^4 + \left[V, S^{(3)}\right] + \frac{1}{2}\left[\left[V, S^{(2)}\right], S^{(1)}\right] \\ &+ \frac{1}{2}\left[\left[V, S^{(1)}\right], S^{(2)}\right] + \frac{1}{6}\left[V, S^{(1)}\right]^3 . \end{aligned} \tag{6.53}$$

For $N = 2$ the third and fourth order terms of the effective Hamiltonian are given in Appendix A.3. We note that in fourth order every single- and two-photon coupling term appears in the effective Hamiltonian at least once, such that the given set of parameters λ_i, Λ_i, η_i, ξ_i and ζ are sufficient to decouple the two qubits from the bus for $N = 2$, and they have to be chosen properly such that the unwanted coupling terms vanish for all orders. However, finding a general analytic solution is a non-trivial task if no approximations are made.

6.5. Summary

We have seen that with the help of the dispersive transformation we can implement effective CZ gates between qubits directly by tuning their frequency on resonance with each other. But they are more time consuming compared to the resonant implementation and therefore need better robustness of the qubits against decoherence. Using X- and Y-controls and optimal control methods can reduce the gate time of locally equivalent perfect entanglers in the dispersive regime for fixed frequency qubits [79]. Leaving the dispersive regime and engineering the qubit frequencies close on resonance with the bus decreases the gate time even further [80]. Therefore, optimal control methods will eventually allow finding the global optimum for gates robust against decoherence error for only frequency tunable qubits as well. But without an analytic decoupling transformation valid for resonant, dispersive and intermediate regimes these numerical pulse shapes always lack an explanation for their effective qubit interaction strength.

7. Conclusion

Superconducting qubits have been pursed for quantum computation for several decades. Coherence times continue to increase and at the time of this writing superconducting devices contain around tens of qubits. Still, building a universal quantum computer is far from realization, even a logical qubit has not been realized yet. As the number of qubits keeps growing, scaling the electronics for control and readout surrounding the device becomes more and more of an issue.

In Chapter 2 we have presented a novel method to optimize single-qubit gates with SFQ pulses and intensified their analysis in Chapter 3. The SFQ controls allow for better scalability of superconducting qubits and combine the latter with the field of conventional superconducting computers, where SFQ pulses are the building block of the computing architecture, defining the logical bits. The nature of SFQ pulses needs novel optimal control methods, since they do not allow for pulse shape optimization like in external microwave controls. Therefore, we have used digital optimal control with only a single-bit of amplitude resolution to compensate for leakage effects in the qubit. Speed limits of this control method are almost as close to the limits set by the system parameters as in AWG controlled systems. Furthermore, a reasonable set of parameters is given to use shift register for loading optimized pulse sequences, which enable to apply a complete set of single-qubit gates on SFQ driven transmon qubits. An analytic approximation for pulse sequences through the Magnus expansion allows us to derive conditions for composite SFQ pulse trains. Two-qubit gates like the cross-resonance gate should in principle be straightforward to implement. Therefore, is is possible to restrict controls to SFQ pulse trains and to achieve almost the same controllability as for microwave pulse shapes used in nowadays control of superconducting qubits. This could pave the way for a scalable quantum device with low power consumption and less wiring through control lines.

As mentioned before, using external microwave controls is state of the art for gate operations on superconducting qubits. In Chapter 4 we have shown how to implement the iFredkin gate in two superconducting architectures with only frequency tunable qubits. The direct coupling of qubits shows only a small speed advantage compared to the RezQu architecture where the qubits are all coupled through a common transmission line. The latter allows for easier scalability and gives a reasonable solution to the requirement of coupling a specific single control qubit to all other qubits, while being able to control the interaction strength between the remaining qubits as well. We also see that optimized control pulses make the three-qubit gates as fast as two-qubit gates, and that the weak point in terms of gate duration is having the gate between two register qubits controlled by the state of the third qubit. In all, the direct implementation of three-qubit gates outperforms decomposition into single- and two-qubit gates due to tailored microwave pulses. It is therefore possible to see such a device being realized as part of a quantum simulator for the Fermi-Hubbard model in the near term future.

Parametrization of numerical pulse shapes can help to understand the underlying processes during the gate performance. In Chapter 5 we have introduced the Walsh functions and combined them with the GRAPE method. Restricting to digital basis function can approximate AWG with their limited time resolution. Still, Walsh functions show symmetries similar to Fourier decompositions. The results of the numerical optimization make it clear that reducing the number of parameters and choosing specific digital functions with proper symmetries whenever possible is feasible. Furthermore, digital functions are easy to produce with electronics and this might lead to the development of more specialized hardware for microwave controlled superconducting qubits.

In Chapter 6 we have investigated the differences of a resonant and a dispersive implementation of the CZ gate. Simulations of the Lindblad master equation for the RezQu architecture reveal that the resonant implementation of the analytic gate in most cases works better than its dispersive counterpart, except when the cavity has a very low quality factor. While in the dispersive regime less parameters need to be tuned, they are also more constraint to specific frequency ranges. The resonant gate on the other hand allows for much more variety in the control parameters and is less prone to decoherence of the qubits due to its faster operational speed. We have also shown how to possibly extend the dispersive transformation to a general decoupling transformation to work in the dressed frame for both gate implementations. This leaves some work for the future to find analytical expressions for parameters in the generators of the transformation that allow to retrieve the effective qubit-qubit interaction strengths. Numerical optimal control allows to find faster gates already, but analytic methods require a proper decoupling transformation, also to find a suitable parametrization of the pulse shapes and in return a faster and more robust optimization of the gates.

In all, this work summarizes several techniques to improve various quantum gates with optimal control methods for standard microwave control and digital control for extensible superconducting architectures with high-fidelity gates. While quantum simulation of small systems might soon be realizable with tailored microwave pulses found by optimal control, universal quantum computation requires still a lot of improvements on fabrication and scalability of quantum devices, as well as on external electronics and cooling of large chips. Superconducting conventional computation shows how to reliably fabricate large numbers of Josephson junctions for RSFQ circuitry, and with the number of qubits increasing on-chip integration of SFQ control could solve parts of the scalability issues.

A. Methods

A.1. Vectorization

We use the column-major order for vectorization of a matrix, i. e., each column of the matrix is stacked to a column vector

$$\begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{bmatrix} \to \begin{bmatrix} a_{00} & a_{10} & a_{20} & a_{01} & a_{11} & a_{21} & a_{02} & a_{12} & a_{22} \end{bmatrix}^T \tag{A.1}$$

For an operator A acting on elements of the Hilbert space $\mathcal{H}$ the vectorization reads

$$A = \sum_{jk} a_{jk} |j\rangle\langle k| \quad \to \quad |A\rangle = \sum_{jk} a_{jk} |k, j\rangle \ . \tag{A.2}$$

The vectorization of the Hilbert-Schmidt scalar product reads

$$\langle A|B\rangle = \operatorname{Tr} A^\dagger B = \sum_{jk} a^*_{jk} b_{jk} \,. \tag{A.3}$$

Since operators can be transformed into row or column vectors, writing out the scalar product explicitly

$$\langle A|B\rangle = \sum_{jklm} a^*_{lm} b_{jk} \langle m, l|k, j\rangle \tag{A.4}$$

reveals for the adjoint operator

$$\langle A| = |A\rangle^\dagger = \sum_{jk} a^*_{jk} \langle k, j| = \left(\sum_{jk} a_{jk} |k, j\rangle \right)^\dagger \ . \tag{A.5}$$

The product of three operators transforms into

$$\begin{aligned} ABC &= \sum_{ijkl} a_{ij} b_{jk} c_{kl} |i\rangle\langle l| \\ \to |ABC\rangle &= \sum_{ijklmn} c_{kl} a_{ij} b_{jk} \underbrace{|l\rangle\langle k| \otimes |i\rangle\langle j| \, |k, j\rangle}_{=|l,i\rangle} = C^T \otimes A \, |B\rangle \ . \end{aligned} \tag{A.6}$$

A.2. Rotating Frame

The rotating frame is a special transformation from the Schrödinger or lab frame into an interaction frame [81]. The wave function at time t in the rotating frame is the wave function in the lab frame without the trivial (non-interacting or static) rotation accumulated over time t

$$\left|\psi^{R}(t)\right\rangle = R^{\dagger}(t)\left|\psi(t)\right\rangle . \tag{A.7}$$

Now the time derivative of the wave function in the rotating frame obeys the Schrödinger equation

$$\begin{aligned}\left|\dot{\psi}^{R}(t)\right\rangle &= \dot{R}^{\dagger}(t)\left|\psi(t)\right\rangle - iR^{\dagger}(t)H(t)\left|\psi(t)\right\rangle \\ &= \dot{R}^{\dagger}(t)R(t)\left|\psi^{R}(t)\right\rangle - iR^{\dagger}(t)H(t)R(t)\left|\psi^{R}(t)\right\rangle\end{aligned} \tag{A.8}$$

with the Hamilton operator

$$\begin{aligned}H^{R}(t) &= R^{\dagger}(t)H(t)R(t) + i\dot{R}^{\dagger}(t)R(t) \\ &= R^{\dagger}(t)H(t)R(t) - iR^{\dagger}(t)\dot{R}(t) .\end{aligned} \tag{A.9}$$

For a system consisting of a single element with dimension d, for example a qudit, and a rotating frame frequency ω_R we get

$$R(t) = \sum_{n=0}^{d-1} e^{-in\omega_R t}\Pi_n , \tag{A.10}$$

with the projector Π_n onto the nth state. The time derivative reads

$$-iR^{\dagger}(t)\dot{R}(t) = -\omega_R \sum_{n=0}^{d-1} n\Pi_n . \tag{A.11}$$

In the general case for N qudits, each rotating with its own frequency, we get

$$R(t) = \bigotimes_{i=1}^{N} R_i(t) = \bigotimes_{i=1}^{N}\sum_{n=0}^{d_i-1} e^{-in\omega_i t}\Pi_n^{(i)} , \tag{A.12}$$

and the time derivative reads

$$\begin{aligned}R^{\dagger}(t)\dot{R}(t) &= \sum_{i=1}^{N}\bigotimes_{k=1}^{i-1}\mathbb{1}_k \otimes R_i^{\dagger}(t)\dot{R}_i(t) \otimes \bigotimes_{j=i+1}^{N}\mathbb{1}_j \\ -iR_i^{\dagger}(t)\dot{R}_i(t) &= -\omega_i \sum_{n=0}^{d_i-1} n\Pi_n^{(i)} .\end{aligned} \tag{A.13}$$

If we set $\omega_i = \omega_R$, i. e., every frame rotates with the same frequency, we get

$$-\, iR^{\dagger}(t)\,\dot{R}(t) = -\omega_R n\,, \tag{A.14}$$

with the number operator $n = \sum_{i=1}^{N} n_i$. For a two-level system with Hamiltonian

$$H = \nu_1 X + \nu_2 Y + \nu_3 \frac{1}{2}(\mathbb{1} - Z) \tag{A.15}$$

the rotating frame Hamiltonian reads

$$\begin{aligned} H^R &= (\nu_1 \cos(\omega_R t) - \nu_2 \sin(\omega_R t))\, X + (\nu_1 \sin(\omega_R t) + \nu_2 \cos(\omega_R t))\, Y \\ &\quad + (\nu_3 - \omega_R)\frac{1}{2}(\mathbb{1} - Z)\,. \end{aligned} \tag{A.16}$$

A.3. Dispersive Hamiltonian

Here we give the third and fourth order effective Hamiltonians of the dispersive transformation in the subspace $N = 2$, which is defined in Section 6.4. The six eigenstates of the unperturbed Hamiltonian are $|0;02\rangle$, $|0;11\rangle$, $|0;20\rangle$, $|1;01\rangle$, $|1;10\rangle$ and $|2;00\rangle$. Note that we can obtain the two-level approximation of the qubits from the three-level case by setting $\Lambda_i = 0$, $\eta_i = 0$, $\xi_i = 0$ and $\Delta_i = 0$, and ignoring all terms that include the states $|0;02\rangle$ and $|0;20\rangle$. In third order we have six single-photon qubit-bus coupling terms. These are the same coupling types as in the interaction Hamiltonian V, and they should vanish if the parameters are properly chosen. The coupling terms including the second level of the qubits read

$$\begin{aligned} \langle 0;02|H^3_{\text{eff}}|1;01\rangle &= \frac{6g_2\lambda_2 - 3g_1\lambda_1}{2}\Lambda_2 - \frac{\sqrt{2}g_2}{2}\left(\lambda_1^2 + 2\lambda_2^2 + 4\Lambda_2^2\right) \\ &\quad - \frac{2\delta_2\lambda_2^2 - \delta_1\lambda_1^2}{2}\Lambda_2 + \frac{\delta_2 + \Delta_2}{6}\Lambda_2\left(\lambda_1^2 + 2\lambda_2^2 + 4\Lambda_2^2\right) \\ &\quad + 2g_2\eta_2 - (3\delta_2 + \Delta_2)\lambda_2\eta_2 \end{aligned} \tag{A.17}$$

$$\begin{aligned} \langle 0;20|H^3_{\text{eff}}|1;10\rangle &= \frac{6g_1\lambda_1 - 3g_2\lambda_2}{2}\Lambda_1 - \frac{\sqrt{2}g_1}{2}\left(\lambda_2^2 + 2\lambda_1^2 + 4\Lambda_1^2\right) \\ &\quad - \frac{2\delta_1\lambda_1^2 - \delta_2\lambda_2^2}{2}\Lambda_1 + \frac{\delta_1 + \Delta_1}{6}\Lambda_1\left(\lambda_2^2 + 2\lambda_1^2 + 4\Lambda_1^2\right) \\ &\quad + 2g_1\eta_1 - (3\delta_1 + \Delta_1)\lambda_1\eta_1\,. \end{aligned} \tag{A.18}$$

Next we have the coupling terms where both qubits are excited

$$\begin{aligned}\langle 0;11|H^3_{\text{eff}}|1;01\rangle &= 3\frac{g_1\lambda_2 + g_2\lambda_1}{2}\lambda_2 - \frac{3\sqrt{2}g_2}{2}\Lambda_2\lambda_1 - \frac{g_1}{2}\left(\Lambda_2^2 + 2\lambda_2^2 + 4\lambda_1^2\right)\\ &\quad - \frac{\delta_1+\delta_2}{2}\lambda_1\lambda_2^2 + \frac{\delta_2+\Delta_2}{2}\Lambda_2^2\lambda_1 + \frac{\delta_1\lambda_1}{6}\left(\Lambda_2^2 + 2\lambda_2^2 + 4\lambda_1^2\right)\end{aligned} \tag{A.19}$$

$$\begin{aligned}\langle 0;11|H^3_{\text{eff}}|1;10\rangle &= 3\frac{g_2\lambda_1 + g_1\lambda_2}{2}\lambda_1 - \frac{3\sqrt{2}g_1}{2}\Lambda_1\lambda_2 - \frac{g_2}{2}\left(\Lambda_1^2 + 2\lambda_1^2 + 4\lambda_2^2\right)\\ &\quad - \frac{\delta_1+\delta_2}{2}\lambda_2\lambda_1^2 + \frac{\delta_1+\Delta_1}{2}\Lambda_1^2\lambda_2 + \frac{\delta_2\lambda_2}{6}\left(\Lambda_1^2 + 2\lambda_1^2 + 4\lambda_2^2\right),\end{aligned} \tag{A.20}$$

and finally the transitions involving the second resonator level

$$\begin{aligned}\langle 1;01|H^3_{\text{eff}}|2;00\rangle &= -3\sqrt{2}\frac{2g_2\lambda_2 + g_1\lambda_1}{2}\lambda_2 + 3g_2\Lambda_2\lambda_2\\ &\quad - \frac{\sqrt{2}g_2}{2}\left(2\lambda_2^2 + \Lambda_2^2 + \lambda_1^2\right) + \sqrt{2}\frac{\delta_1\lambda_1^2 + 2\delta_2\lambda_2^2}{2}\lambda_2\\ &\quad - \sqrt{2}\frac{\delta_2+\Delta_2}{2}\Lambda_2^2\lambda_2 + \frac{\sqrt{2}\delta_2\lambda_2}{6}\left(\lambda_2^2 + \Lambda_2^2 + \lambda_1^2\right)\\ &\quad - 2g_2\eta_2 + \frac{3\delta_2 + 2\Delta_2}{\sqrt{2}}\Lambda_2\eta_2\end{aligned} \tag{A.21}$$

$$\begin{aligned}\langle 1;10|H^3_{\text{eff}}|2;00\rangle &= -3\sqrt{2}\frac{2g_1\lambda_1 + g_2\lambda_2}{2}\lambda_1 + 3g_1\Lambda_1\lambda_1\\ &\quad - \frac{\sqrt{2}g_1}{2}\left(2\lambda_1^2 + \Lambda_1^2 + \lambda_2^2\right) + \sqrt{2}\frac{\delta_2\lambda_2^2 + 2\delta_1\lambda_1^2}{2}\lambda_1\\ &\quad - \sqrt{2}\frac{\delta_1+\Delta_1}{2}\Lambda_1^2\lambda_1 + \frac{\sqrt{2}\delta_1\lambda_1}{6}\left(2\lambda_1^2 + \Lambda_1^2 + \lambda_2^2\right)\\ &\quad - 2g_1\eta_1 + \frac{3\delta_1 + 2\Delta_1}{\sqrt{2}}\Lambda_1\eta_1\,.\end{aligned} \tag{A.22}$$

In third order appear two-photon coupling terms which include the second level of one qubit and the first level of the other qubit and the bus. To compensate for this unwanted terms we have added the third order generator $S^{(3)}$ (defined in Section 6.4) and get

$$\begin{aligned}\langle 0;02|H^3_{\text{eff}}|1;10\rangle &= -\frac{3g_2}{2}\left(\sqrt{2}\lambda_2 - \Lambda_2\right)\lambda_1 + \frac{\Delta_2\lambda_2\Lambda_2}{2}\lambda_1\\ &\quad + 2g_1\eta_2 - (\delta_1 + 2\delta_2 + \Delta_2)\lambda_1\eta_2 + (\delta_1 - 2\delta_2 - \Delta_2)\xi_2\end{aligned} \tag{A.23}$$

$$\begin{aligned}\langle 0;20|H^3_{\text{eff}}|1;01\rangle = &-\frac{3g_1}{2}\left(\sqrt{2}\lambda_1 - \Lambda_1\right)\lambda_2 + \frac{\Delta_1\lambda_1\Lambda_1}{2}\lambda_2 \\ &+ 2g_2\eta_1 - (\delta_2 + 2\delta_1 + \Delta_1)\lambda_2\eta_1 + (\delta_2 - 2\delta_1 - \Delta_1)\xi_1\,. \end{aligned} \tag{A.24}$$

Now we continue with the fourth order terms. We start with the diagonal elements of the Hamiltonian. For the second level of each qubit we get

$$\begin{aligned}\langle 0;02|H^4_{\text{eff}}|0;02\rangle = &\left(2g_2\lambda_2 - g_1\lambda_1\right)\Lambda_2^2 - \frac{\sqrt{2}g_2\Lambda_2}{3}\left(\lambda_1^2 + 2\lambda_2^2 + 4\Lambda_2^2\right) \\ &+ \frac{\delta_1\lambda_1^2 - 2\delta_2\lambda_2^2}{4}\Lambda_2^2 + \frac{\delta_2 + \Delta_2}{12}\Lambda_2^2\left(\lambda_1^2 + 2\lambda_2^2 + 4\Lambda_2^2\right) \\ &- 2g_2\left(\sqrt{2}\lambda_2 - 2\Lambda_2\right)\eta_2 - 2\delta_2\lambda_2\Lambda_2\eta_2 - 2(2\delta_2 + \Delta_2)\eta_2^2 \end{aligned} \tag{A.25}$$

$$\begin{aligned}\langle 0;20|H^4_{\text{eff}}|0;20\rangle = &\left(2g_1\lambda_1 - g_2\lambda_2\right)\Lambda_1^2 - \frac{\sqrt{2}g_1\Lambda_1}{3}\left(\lambda_2^2 + 2\lambda_1^2 + 4\Lambda_1^2\right) \\ &+ \frac{\delta_2\lambda_2^2 - 2\delta_1\lambda_1^2}{4}\Lambda_1^2 + \frac{\delta_1 + \Delta_1}{12}\Lambda_1^2\left(\lambda_2^2 + 2\lambda_1^2 + 4\Lambda_1^2\right) \\ &- 2g_1\left(\sqrt{2}\lambda_1 - 2\Lambda_1\right)\eta_1 - 2\delta_1\lambda_1\Lambda_1\eta_1 - 2(2\delta_1 + \Delta_1)\eta_1^2\,. \end{aligned} \tag{A.26}$$

With both qubits excited the corresponding element reads

$$\begin{aligned}\langle 0;11|H^4_{\text{eff}}|0;11\rangle = &-g_1\lambda_1\left(\lambda_1^2 - \lambda_2^2\right) - \sqrt{2}g_1\Lambda_1\lambda_2^2 - \frac{g_1\lambda_1}{3}\left(\Lambda_2^2 - \lambda_2^2 + \lambda_1^2\right) \\ &- g_2\lambda_2\left(\lambda_2^2 - \lambda_1^2\right) - \sqrt{2}g_2\Lambda_2\lambda_1^2 - \frac{g_2\lambda_2}{3}\left(\Lambda_1^2 - \lambda_1^2 + \lambda_2^2\right) \\ &+ \frac{\delta_1\lambda_1^2}{4}\left(\lambda_1^2 - \lambda_2^2\right) + \frac{\delta_1 + \Delta_1}{4}\Lambda_1^2\lambda_2^2 + \frac{\delta_1\lambda_1^2}{12}\left(\Lambda_2^2 - \lambda_2^2 + \lambda_1^2\right) \\ &+ \frac{\delta_2\lambda_2^2}{4}\left(\lambda_2^2 - \lambda_1^2\right) + \frac{\delta_2 + \Delta_2}{4}\Lambda_2^2\lambda_1^2 + \frac{\delta_2\lambda_2^2}{12}\left(\Lambda_1^2 - \lambda_1^2 + \lambda_2^2\right)\,, \end{aligned} \tag{A.27}$$

and for a single excitation in the bus we have

$$\begin{aligned}\langle 1;01|H^4_{\mathrm{eff}}|1;01\rangle =& -\frac{8g_2\lambda_2}{3}\left(2\lambda_2^2+\Lambda_2^2+\lambda_1^2\right)+\frac{4\sqrt{2}g_2\Lambda_2}{3}\left(2\lambda_2^2+\Lambda_2^2+\lambda_1^2\right)\\ &+\left(g_2\lambda_1-g_1\lambda_2\right)\lambda_1\lambda_2+\frac{4g_1\lambda_1}{3}\left(\Lambda_2^2-\lambda_2^2+\lambda_1^2\right)\\ &+\frac{2\delta_2\lambda_2^2}{3}\left(2\lambda_2^2+\Lambda_2^2+\lambda_1^2\right)-\frac{\delta_2+\Delta_2}{3}\Lambda_2^2\left(2\lambda_2^2+\Lambda_2^2+\lambda_1^2\right)\\ &-\frac{\delta_2-\delta_1}{4}\lambda_1^2\lambda_2^2-\frac{\delta_1\lambda_1^2}{3}\left(\Lambda_2^2-\lambda_2^2+\lambda_1^2\right)\\ &-2g_2\left(\sqrt{2}\lambda_2+\Lambda_2\right)\eta_2+2(2\delta_2+\Delta_2)\lambda_2\Lambda_2\eta_2\end{aligned}\tag{A.28}$$

$$\begin{aligned}\langle 1;10|H^4_{\mathrm{eff}}|1;10\rangle =& -\frac{8g_1\lambda_1}{3}\left(2\lambda_1^2+\Lambda_1^2+\lambda_2^2\right)+\frac{4\sqrt{2}g_1\Lambda_1}{3}\left(2\lambda_1^2+\Lambda_1^2+\lambda_2^2\right)\\ &+\left(g_1\lambda_2-g_2\lambda_1\right)\lambda_1\lambda_2+\frac{4g_2\lambda_2}{3}\left(\Lambda_1^2-\lambda_1^2+\lambda_2^2\right)\\ &+\frac{2\delta_1\lambda_1^2}{3}\left(2\lambda_1^2+\Lambda_1^2+\lambda_2^2\right)-\frac{\delta_1+\Delta_1}{3}\Lambda_1^2\left(2\lambda_1^2+\Lambda_1^2+\lambda_2^2\right)\\ &-\frac{\delta_1-\delta_2}{4}\lambda_1^2\lambda_2^2-\frac{\delta_2\lambda_2^2}{3}\left(\Lambda_1^2-\lambda_1^2+\lambda_2^2\right)\\ &-2g_1\left(\sqrt{2}\lambda_1+\Lambda_1\right)\eta_1+2(2\delta_1+\Delta_1)\lambda_1\Lambda_1\eta_1\,.\end{aligned}\tag{A.29}$$

Finally, for both excitations in the resonator we get the element

$$\begin{aligned}\langle 2;00|H^4_{\mathrm{eff}}|2;00\rangle =& \frac{8}{3}\left(g_1\lambda_1+g_2\lambda_2\right)\left(\lambda_1^2+\lambda_2^2\right)-2\sqrt{2}g_1\Lambda_1\lambda_1^2\\ &-2\sqrt{2}g_2\Lambda_2\lambda_2^2+\frac{2g_1\lambda_1}{3}\left(4\lambda_1^2+\Lambda_1^2\right)+\frac{2g_2\lambda_2}{3}\left(4\lambda_2^2+\Lambda_2^2\right)\\ &-\frac{2}{3}\left(\delta_1\lambda_1^2+\delta_2\lambda_2^2\right)\left(\lambda_1^2+\lambda_2^2\right)+\frac{\delta_1+\Delta_1}{2}\Lambda_1^2\lambda_1^2\\ &+\frac{\delta_2+\Delta_2}{2}\Lambda_2^2\lambda_2^2-\frac{\delta_1\lambda_1^2}{6}\left(4\lambda_1^2+\Lambda_1^2\right)-\frac{\delta_2\lambda_2^2}{6}\left(4\lambda_2^2+\Lambda_2^2\right)\\ &+2g_1(2\sqrt{2}\lambda_1-\Lambda_1)\eta_1+2g_2(2\sqrt{2}\lambda_2-\Lambda_2)\eta_2\\ &-2(\delta_1+\Delta_1)\lambda_1\Lambda_1\eta_1-2(\delta_2+\Delta_2)\lambda_2\Lambda_2\eta_2\\ &+2(2\delta_1+\Delta_1)\eta_1^2+2(2\delta_2+\Delta_2)\eta_2^2\,.\end{aligned}\tag{A.30}$$

We continue with the coupling terms describing single-photon transitions. For the virtual qubit-qubit couplings we get the three terms

$$\begin{aligned}
\langle 0;02|H^4_{\text{eff}}|0;11\rangle = {} & (2g_2\lambda_2 - g_1\lambda_1)\,\Lambda_2\lambda_1 - \sqrt{2}g_2\lambda_1\left(\lambda_2^2 + \Lambda_2^2\right) \\
& - \frac{g_1\Lambda_2 + \sqrt{2}g_2\lambda_1}{6}\left(\Lambda_2^2 - \lambda_2^2 + \lambda_1^2\right) \\
& - \frac{2\delta_2\lambda_2^2 - \delta_1\lambda_1^2}{4}\Lambda_2\lambda_1 + \frac{\delta_2 + \Delta_2}{4}\Lambda_2\lambda_1\left(\lambda_2^2 + \Lambda_2^2\right) \\
& + \frac{\delta_1 + \delta_2 + \Delta_2}{24}\Lambda_2\lambda_1\left(\Lambda_2^2 - \lambda_2^2 + \lambda_1^2\right) \\
& + (g_1\lambda_2 + g_2\lambda_1)\eta_2 - \frac{5\delta_2 + 2\Delta_2 + \delta_1}{3}\lambda_1\lambda_2\eta_2 \\
& + g_2\xi_2 - \frac{3\delta_2 + \Delta_2 - \delta_1}{2}\lambda_2\xi_2 \qquad \text{(A.31)}
\end{aligned}$$

$$\begin{aligned}
\langle 0;11|H^4_{\text{eff}}|0;20\rangle = {} & (2g_1\lambda_1 - g_2\lambda_2)\,\Lambda_1\lambda_2 - \sqrt{2}g_1\lambda_2\left(\lambda_1^2 + \Lambda_1^2\right) \\
& - \frac{g_2\Lambda_1 + \sqrt{2}g_1\lambda_2}{6}\left(\Lambda_1^2 - \lambda_1^2 + \lambda_2^2\right) \\
& - \frac{2\delta_1\lambda_1^2 - \delta_2\lambda_2^2}{4}\Lambda_1\lambda_2 + \frac{\delta_1 + \Delta_1}{4}\Lambda_1\lambda_2\left(\lambda_1^2 + \Lambda_1^2\right) \\
& + \frac{\delta_2 + \delta_1 + \Delta_1}{24}\Lambda_1\lambda_2\left(\Lambda_1^2 - \lambda_1^2 + \lambda_2^2\right) \\
& + (g_1\lambda_2 + g_2\lambda_1)\eta_1 - \frac{5\delta_1 + 2\Delta_1 + \delta_2}{3}\lambda_1\lambda_2\eta_1 \\
& + g_1\xi_1 - \frac{3\delta_1 + \Delta_1 - \delta_2}{2}\lambda_1\xi_1 \qquad \text{(A.32)}
\end{aligned}$$

and

$$\begin{aligned}
\langle 1;01|H_{\mathrm{eff}}^{4}|1;10\rangle = &-\frac{g_2\lambda_1+g_1\lambda_2}{2}\left(\lambda_1^2+\lambda_2^2\right)+\frac{g_2\lambda_1+g_1\lambda_2}{6}\left(\Lambda_1^2+\Lambda_2^2\right)\\
&-g_2\lambda_1\left(3\lambda_2^2+\Lambda_2^2\right)+2\sqrt{2}g_2\Lambda_2\lambda_1\lambda_2\\
&-g_1\lambda_2\left(3\lambda_1^2+\Lambda_1^2\right)+2\sqrt{2}g_1\Lambda_1\lambda_1\lambda_2\\
&+\frac{\delta_1+\delta_2}{8}\lambda_1\lambda_2\left(\lambda_1^2+\lambda_2^2\right)-\frac{\delta_1+\delta_2}{24}\lambda_1\lambda_2\left(\Lambda_1^2+\Lambda_2^2\right)\\
&+\frac{\delta_2\lambda_2\lambda_1}{4}\left(3\lambda_2^2+\Lambda_2^2\right)-\frac{\delta_2+\Delta_2}{2}\Lambda_2^2\lambda_1\lambda_2\\
&+\frac{\delta_1\lambda_1\lambda_2}{4}\left(3\lambda_1^2+\Lambda_1^2\right)-\frac{\delta_1+\Delta_1}{2}\Lambda_1^2\lambda_1\lambda_2\\
&-\left(\sqrt{2}g_1\lambda_2+g_2\Lambda_1\right)\eta_1-\left(\sqrt{2}g_2\lambda_1+g_1\Lambda_2\right)\eta_2\\
&+\frac{5\delta_1+3\Delta_1+\delta_2}{3}\lambda_2\Lambda_1\eta_1+\frac{5\delta_2+3\Delta_2+\delta_1}{3}\lambda_1\Lambda_2\eta_2\\
&-\sqrt{2}\left(g_1\xi_1+g_2\xi_2\right)\\
&+\frac{3\delta_1+2\Delta_1-\delta_2}{2}\Lambda_1\xi_1+\frac{3\delta_2+2\Delta_2-\delta_1}{2}\Lambda_2\xi_2\,. \qquad \text{(A.33)}
\end{aligned}$$

Now we have a look at the virtual two-photon transitions, starting with the qubit-qubit term

$$\begin{aligned}
\langle 0;02|H_{\mathrm{eff}}^{4}|0;20\rangle = &-\frac{g_2\lambda_1\Lambda_1}{2}\left(\sqrt{2}\lambda_2-\Lambda_2\right)-\frac{g_1\lambda_2\Lambda_2}{2}\left(\sqrt{2}\lambda_1-\Lambda_1\right)\\
&+\frac{\Delta_1+\Delta_2}{8}\lambda_1\Lambda_1\lambda_2\Lambda_2\\
&+g_1\left(2\Lambda_1-\sqrt{2}\lambda_1\right)\eta_2+g_2\left(2\Lambda_2-\sqrt{2}\lambda_2\right)\eta_1\\
&-\frac{2\delta_2+\Delta_2+\delta_1-\Delta_1}{3}\lambda_1\Lambda_1\eta_2-\frac{2\delta_1+\Delta_1+\delta_2-\Delta_2}{3}\lambda_2\Lambda_2\eta_1\\
&-(2\delta_1+\Delta_1+2\delta_2+\Delta_2)\eta_1\eta_2+\sqrt{2}(g_1\xi_2+g_2\xi_1)\\
&-\frac{2\delta_1+\Delta_1+\Delta_2}{2}\Lambda_2\xi_1-\frac{2\delta_2+\Delta_2+\Delta_1}{2}\Lambda_1\xi_2\,. \qquad \text{(A.34)}
\end{aligned}$$

The two-photon qubit-bus couplings which should vanish for properly chosen parameters read

$$\begin{aligned}\langle 0;02|H^4_{\mathrm{eff}}|2;00\rangle &= 2g_2\left(\Lambda_2-\sqrt{2}\lambda_2\right)\lambda_2\Lambda_2\\ &+\frac{g_2}{6}\left(2\lambda_2-\sqrt{2}\Lambda_2\right)\left(\Lambda_2^2+2\lambda_2^2+4\lambda_1^2\right)\\ &+\frac{2\delta_2\lambda_2^2-\left(\delta_2+\Delta_2\right)\Lambda_2^2}{2\sqrt{2}}\lambda_2\Lambda_2-\frac{\Delta_2\lambda_2\Lambda_2}{12\sqrt{2}}\left(\Lambda_2^2+2\lambda_2^2+4\lambda_1^2\right)\\ &-3\left(\sqrt{2}g_2\lambda_2+g_2\Lambda_2+\sqrt{2}g_1\lambda_1\right)\eta_2\\ &+\frac{(6\delta_1+4\delta_2+2\Delta_2)\lambda_1^2+2(5\delta_2+\Delta_2)\lambda_2^2+(5\delta_2+4\Delta_2)\Lambda_2^2}{3\sqrt{2}}\eta_2\\ &+\sqrt{2}g_1\xi_2+\frac{2\delta_2+\Delta_2-2\delta_1}{\sqrt{2}}\lambda_1\xi_2\end{aligned}\tag{A.35}$$

$$\begin{aligned}\langle 0;20|H^4_{\mathrm{eff}}|2;00\rangle &= 2g_1\left(\Lambda_1-\sqrt{2}\lambda_1\right)\lambda_1\Lambda_1\\ &+\frac{g_1}{6}\left(2\lambda_1-\sqrt{2}\Lambda_1\right)\left(\Lambda_1^2+2\lambda_1^2+4\lambda_2^2\right)\\ &+\frac{2\delta_1\lambda_1^2-\left(\delta_1+\Delta_1\right)\Lambda_1^2}{2\sqrt{2}}\lambda_1\Lambda_1-\frac{\Delta_1\lambda_1\Lambda_1}{12\sqrt{2}}\left(\Lambda_1^2+2\lambda_1^2+4\lambda_2^2\right)\\ &-3\left(\sqrt{2}g_1\lambda_1+g_1\Lambda_1+\sqrt{2}g_2\lambda_2\right)\eta_1\\ &+\frac{(6\delta_2+2\Delta_1+4\delta_1)\lambda_2^2+2(5\delta_1+\Delta_1)\lambda_1^2+(5\delta_1+4\Delta_1)\Lambda_1^2}{3\sqrt{2}}\eta_1\\ &+\sqrt{2}g_2\xi_1+\frac{2\delta_1+\Delta_1-2\delta_2}{\sqrt{2}}\lambda_2\xi_1\,.\end{aligned}\tag{A.36}$$

Finally, there is a term appearing in fourth order which couples both first levels of the qubit with the second level of the bus, for which we have designed the generator $S^{(4)}$

$$\begin{aligned}
\langle 0;11|\mathrm{H}^{4}_{\mathrm{eff}}|2;00\rangle = {} & g_1\left(\Lambda_1-\sqrt{2}\lambda_1\right)\lambda_1\lambda_2+g_2\left(\Lambda_2-\sqrt{2}\lambda_2\right)\lambda_1\lambda_2 \\
& -\frac{1}{3}\left(g_1\sqrt{2}\lambda_1+g_2\sqrt{2}\lambda_2\right)\lambda_1\lambda_2 \\
& -\sqrt{2}\frac{g_2\lambda_1+g_1\lambda_2}{3}\left(\lambda_1^2+\lambda_2^2\right)+\sqrt{2}\frac{g_2\lambda_1-g_1\lambda_2}{6}\left(\Lambda_1^2-\Lambda_2^2\right) \\
& +\sqrt{2}\frac{\delta_1\lambda_1^2+\delta_2\lambda_2^2}{3}\lambda_1\lambda_2+\sqrt{2}\frac{\delta_1+\delta_2}{12}\left(\lambda_1^2+\lambda_2^2\right)\lambda_1\lambda_2 \\
& -\sqrt{2}\frac{\delta_1-\delta_2}{24}\left(\Lambda_1^2-\Lambda_2^2\right)\lambda_1\lambda_2 \\
& -\sqrt{2}\frac{(\delta_1+\Delta_1)\Lambda_1^2+(\delta_2+\Delta_2)\Lambda_2^2}{4}\lambda_1\lambda_2 \\
& -\frac{4g_1\lambda_2+\sqrt{2}g_2\Lambda_1}{2}\eta_1-\frac{4g_2\lambda_1+\sqrt{2}g_1\Lambda_2}{2}\eta_2 \\
& +\sqrt{2}\frac{4\delta_1+3\Delta_1+\delta_2}{6}\lambda_2\Lambda_1\eta_1+\sqrt{2}\frac{4\delta_2+3\Delta_2+\delta_1}{6}\lambda_1\Lambda_2\eta_2 \\
& -\sqrt{2}(\delta_1+\delta_2)\zeta\,.
\end{aligned}\tag{A.37}$$

Publication List

In This Thesis

Some ideas of this thesis have appeared in the following publications:

Journal Articles

- P. J. Liebermann, and F. K. Wilhelm
  Optimal qubit control using single-flux quantum pulses
  Phys. Rev. Applied, 6, 024022 (2016), arXiv:1512.05495

Submitted

- P. J. Liebermann, P.-L. Dallaire-Demers, and F. K. Wilhelm
  Implementation of the iFREDKIN gate in scalable superconducting architecture for the quantum simulation of fermionic systems
  arXiv:1701.07870, (2017)
- R. McDermott, M. G. Vavilov, B. L. T. Plourde, F. K. Wilhelm, P. J. Liebermann, O. A. Mukhanov, T. A. Ohki
  Quantum–Classical Interface Based on Single Flux Quantum Digital Logic
  arXiv:1710.04645

Other Publications

Submitted

- S. Kirchhoff, T. Keßler, P. J. Liebermann, E. Assémat, S. Machnes, F. Motzoi, and F. K. Wilhelm
  Optimized cross-resonance gate for coupled transmon systems
  arXiv:1701.01841, (2017)

Bibliography

[1] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press 2010).

[2] A. Einstein, B. Podolsky, and N. Rosen, Can quantum-mechanical description of physical reality be considered complete?, Phys. Rev. **47**, 777 (1935).

[3] H. P. Breuer and F. Petruccione, *The theory of open quantum systems* (Oxford University Press 2002).

[4] J. Zhang, J. Vala, S. Sastry, and K. B. Whaley, Geometric theory of nonlocal two-qubit operations, Phys. Rev. A **67**, 042313 (2003).

[5] D. P. DiVincenzo, The physical implementation of quantum computation (2000), `arXiv:quant-ph/0002077`.

[6] M. H. Devoret, Quantum fluctuations in electrical circuits, in *Quantum Fluctuations* (Elsevier 1997).

[7] S. M. Girvin, Circuit QED: Superconducting qubits coupled to microwave photons, in *Quantum Machines: Measurement and Control of Engineered Quantum Systems* (Oxford University Press 2011).

[8] J. Clarke and F. K. Wilhelm, Superconducting quantum bits, Nature **453**, 1031 (2008).

[9] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. 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 (2007).

[10] S. J. Glaser, U. Boscain, T. Calarco, C. P. Koch, W. Köckenberger, R. Kosloff, I. Kuprov, B. Luy, S. Schirmer, T. Schulte-Herbrüggen, D. Sugny, and F. K. Wilhelm, Training Schrödinger's cat: quantum optimal control, Eur. Phys. J. D **69**, 279 (2015).

[11] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms, J. Magn. Reson. **172**, 296 (2005).

[12] P. Sutton and S. Boyden, Genetic algorithms: A general search procedure, Am. J. Phys. **62**, 549 (1994).

[13] F. Motzoi, J. M. Gambetta, S. T. Merkel, and F. K. Wilhelm, Optimal control methods for rapidly time-varying Hamiltonians, Phys. Rev. A **84**, 022307 (2011).

[14] D. Whitley, A genetic algorithm tutorial, Stat. Comput. **4**, 65 (1994).

[15] R. S. Judson and H. Rabitz, Teaching lasers to control molecules, Phys. Rev. Lett. **68**, 1500 (1992).

[16] A. K. Hartmann and H. Rieger, *Optimization Algorithms in Physics* (Wiley-VCH, Weinheim 2002).

[17] P. J. Liebermann and F. K. Wilhelm, Optimal qubit control using single-flux quantum pulses, Phys. Rev. Applied **6**, 024022 (2016).

[18] K. K. Likharev and V. K. Semenov, RSFQ logic/memory family: a new Josephson-junction technology for sub-terahertz-clock-frequency digital systems, IEEE Trans. Appl. Supercond. **1**, 3 (1991).

[19] P. Bunyk, K. Likharev, and D. Zinoviev, RSFQ technology: Physics and devices, Int. J. High Speed Electron. Syst. **11**, 257 (2001).

[20] S. Anders, M. Blamire, F.-I. Buchholz, D.-G. Crété, R. Cristiano, P. Febvre, L. Fritzsch, A. Herr, E. Il'ichev, J. Kohlmann, J. Kunert, H.-G. Meyer, J. Niemeyer, T. Ortlepp, H. Rogalla, T. Schurig, M. Siegel, R. Stolz, E. Tarte, H. ter Brake, H. Toepfer, J.-C. Villegier, A. Zagoskin, and A. Zorin, European roadmap on superconductive electronics – status and perspectives, Physica C: Supercond. **470**, 2079 (2010).

[21] K. K. Likharev, Superconductor digital electronics, Physica C Supercond. **482**, 6 (2012).

[22] T. P. Orlando, S. Lloyd, L. S. Levitov, K. K. Berggren, M. J. Feldman, M. F. Bocko, J. E. Mooij, C. J. P. Harmans, and C. H. van der Wal, Flux-based superconducting qubits for quantum computation, Physica C Supercond. **372–376**, 194 (2002).

[23] M. J. Feldman and M. F. Bocko, A realistic experiment to demonstrate macroscopic quantum coherence, Physica C Supercond. **350**, 171 (2001).

[24] T. A. Ohki, M. Wulf, and M. J. Feldman, Low-Jc rapid single flux quantum (RSFQ) qubit control circuit, IEEE Trans. Appl. Supercond. **17**, 154 (2007).

[25] V. K. Semenov and D. V. Averin, SFQ control circuits for Josephson junction qubits, IEEE Trans. Appl. Supercond. **13**, 960 (2003).

[26] A. Fedorov, A. Shnirman, G. Schön, and A. Kidiyarova-Shevchenko, Reading out the state of a flux qubit by Josephson transmission line solitons, Phys. Rev. B **75**, 224504 (2007).

[27] K. G. Fedorov, A. V. Shcherbakova, M. J. Wolf, D. Beckmann, and A. V. Ustinov, Fluxon readout of a superconducting qubit, Phys. Rev. Lett. **112**, 160502 (2014).

[28] T. A. Ohki, M. Wulf, and M. F. Bocko, Picosecond on-chip qubit control circuitry, IEEE Trans. Appl. Supercond. **15**, 837 (2005).

[29] K. Gaj, E. G. Friedman, and M. J. Feldman, Timing of multi-gigahertz rapid single flux quantum digital circuits, J. VLSI Signal Process. Syst. Signal Image Video Technol. **16**, 247 (1997).

[30] O. A. Mukhanov, Rapid single flux quantum (RSFQ) shift register family, IEEE Trans. Appl. Supercond. **3**, 2578 (1993).

[31] O. A. Mukhanov, RSFQ 1024-bit shift register for acquisition memory, IEEE Trans. Appl. Supercond. **3**, 3102 (1993).

[32] R. McDermott and M. G. Vavilov, Accurate qubit control with single flux quantum pulses, Phys. Rev. Applied **2**, 014007 (2014).

[33] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, Phys. Rev. A **86**, 032324 (2012).

[34] O. A. Mukhanov, Energy-efficient single flux quantum technology, IEEE Transactions on Applied Superconductivity **21**, 760 (2011).

[35] D. K. Brock, RSFQ technology: Circuits and systems, Int. J. High Speed Electron. Syst. **11**, 307 (2001).

[36] G. Bodenhausen, R. Freeman, and G. A. Morris, A simple pulse sequence for selective excitation in Fourier transform NMR, J. Magn. Reson. **23**, 171 (1976).

[37] G. A. Morris and R. Freeman, Selective excitation in Fourier transform nuclear magnetic resonance, J. Magn. Reson. **29**, 433 (1978).

[38] P. Rebentrost and F. K. Wilhelm, Optimal control of a leaking qubit, Phys. Rev. B **79**, 060507 (2009).

[39] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, Optimization by simulated annealing, Science **220**, 671 (1983).

[40] H. A. Rabitz, M. M. Hsieh, and C. M. Rosenthal, Quantum optimally controlled transition landscapes, Science **303**, 1998 (2004).

[41] A. V. Rylyakov and K. K. Likharev, Pulse jitter and timing errors in RSFQ circuits, IEEE Trans. Appl. Supercond. **9**, 3539 (1999).

[42] R. Barends, J. Kelly, A. Megrant, D. Sank, E. Jeffrey, Y. Chen, Y. Yin, B. Chiaro, J. Mutus, C. Neill, P. O'Malley, P. Roushan, J. Wenner, T. C. White, A. N. Cleland, and J. M. Martinis, Coherent josephson qubit suitable for scalable quantum integrated circuits, Phys. Rev. Lett. **111**, 080502 (2013).

[43] R. McDermott, M. G. Vavilov, B. L. T. Plourde, F. K. Wilhelm, P. J. Liebermann, O. A. Mukhanov, and T. A. Ohki, Quantum–classical interface based on single flux quantum digital logic (2017), `arXiv:1710.04645`.

[44] L. M. K. Vandersypen and I. L. Chuang, NMR techniques for quantum control and computation, Rev. Mod. Phys. **76**, 1037 (2005).

[45] S. Blanes, F. Casas, J. Oteo, and J. Ros, The Magnus expansion and some of its applications, Phys. Rep. **470**, 151 (2009).

[46] R. C. Bialczak, M. Ansmann, M. Hofheinz, M. Lenander, E. Lucero, M. Neeley, A. D. O'Connell, D. Sank, H. Wang, M. Weides, J. Wenner, T. Yamamoto, A. N. Cleland, and J. M. Martinis, Fast tunable coupler for superconducting qubits, Phys. Rev. Lett. **106**, 060501 (2011).

[47] C. Rigetti and M. Devoret, Fully microwave-tunable universal gates in superconducting qubits with linear couplings and fixed transition frequencies, Phys. Rev. B **81**, 134507 (2010).

[48] J. M. Chow, A. D. Córcoles, J. M. Gambetta, C. Rigetti, B. R. Johnson, J. A. Smolin, J. R. Rozen, G. A. Keefe, M. B. Rothwell, M. B. Ketchen, and M. Steffen, Simple all-microwave entangling gate for fixed-frequency superconducting qubits, Phys. Rev. Lett. **107**, 080502 (2011).

[49] P. J. Liebermann, P.-L. Dallaire-Demers, and F. K. Wilhelm, Implementation of the iFREDKIN gate in scalable superconducting architecture for the quantum simulation of fermionic systems (2017), `arXiv:1701.07870`.

[50] R. P. Feynman, Simulating physics with computers, Int. J. Theor. Phys. **21**, 467 (1982).

[51] P.-L. Dallaire-Demers and F. K. Wilhelm, Quantum gates and architecture for the quantum simulation of the Fermi-Hubbard model, Phys. Rev. A **94**, 062304 (2016).

[52] D. C. McKay, S. Filipp, A. Mezzacapo, E. Magesan, J. M. Chow, and J. M. Gambetta, Universal gate for fixed-frequency qubits via a tunable bus, Phys. Rev. Applied **6**, 064007 (2016).

[53] G. Haack, F. Helmer, M. Mariantoni, F. Marquardt, and E. Solano, Resonant quantum gates in circuit quantum electrodynamics, Phys. Rev. B **82**, 024514 (2010).

[54] F. A. Calderon-Vargas and J. P. Kestner, Directly accessible entangling gates for capacitively coupled singlet-triplet qubits, Phys. Rev. B **91**, 035301 (2015).

[55] E. Zahedinejad, J. Ghosh, and B. C. Sanders, High-fidelity single-shot Toffoli gate via quantum control, Phys. Rev. Lett. **114**, 200502 (2015).

[56] D. J. Egger and F. K. Wilhelm, Optimized controlled-Z gates for two superconducting qubits coupled through a resonator, Supercond. Sci. Technol. **27**, 014001 (2014).

[57] M. Mariantoni, H. Wang, T. Yamamoto, M. Neeley, R. C. Bialczak, Y. Chen, M. Lenander, E. Lucero, A. D. O'Connell, D. Sank, M. Weides, J. Wenner, Y. Yin, J. Zhao, A. N. Korotkov, A. N. Cleland, and J. M. Martinis, Implementing the quantum von Neumann architecture with superconducting circuits, Science **334**, 61 (2011).

[58] A. Galiautdinov, Single-step implementation of the controlled-Z gate in a qubit/bus/qubit device (2011), `arXiv:1103.4641`.

[59] A. Galiautdinov, A. N. Korotkov, and J. M. Martinis, Resonator-zero-qubit architecture for superconducting qubits, Phys. Rev. A **85**, 042321 (2012).

[60] J. Ghosh, A. Galiautdinov, Z. Zhou, A. N. Korotkov, J. M. Martinis, and M. R. Geller, High-fidelity controlled-σ^Z gate for resonator-based superconducting quantum computers, Phys. Rev. A **87**, 022309 (2013).

[61] F. W. Strauch, P. R. Johnson, A. J. Dragt, C. J. Lobb, J. R. Anderson, and F. C. Wellstood, Quantum logic gates for coupled superconducting phase qubits, Phys. Rev. Lett. **91**, 167005 (2003).

[62] T. E. Skinner and N. I. Gershenzon, Optimal control design of pulse shapes as analytic functions, J. Magn. Reson. **204**, 248 (2010).

[63] J. L. Walsh, A closed set of normal orthogonal functions, Am. J. Math. **45**, 5 (1923).

[64] H. Rademacher, Einige Sätze über Reihen von allgemeinen Orthogonalfunktionen, Math. Ann. **87**, 112 (1922).

[65] D. Hayes, K. Khodjasteh, L. Viola, and M. J. Biercuk, Reducing sequencing complexity in dynamical quantum error suppression by Walsh modulation, Phys. Rev. A **84**, 062323 (2011).

[66] T. Beer, Walsh transforms, Am. J. Phys. **49**, 466 (1981).

[67] A. Haar, Zur Theorie der orthogonalen Funktionensysteme, Math. Ann. **69**, 331 (1910).

[68] I. N. Bronstein, K. A. Semendjajew, G. Musiol, and H. Mühlig, *Taschenbuch der Mathematik* (Verlag Harri Deutsch 2008).

[69] S. Machnes, D. J. Tannor, F. K. Wilhelm, and E. Assémat, Gradient optimization of analytic controls: the route to high accuracy quantum optimal control (2015), `arXiv:1507.04261`.

[70] F. Motzoi, J. M. Gambetta, P. Rebentrost, and F. K. Wilhelm, Simple pulses for elimination of leakage in weakly nonlinear qubits, Phys. Rev. Lett. **103**, 110501 (2009).

[71] J. I. Cirac and P. Zoller, Quantum computations with cold trapped ions, Phys. Rev. Lett. **74**, 4091 (1995).

[72] M. Boissonneault, J. Gambetta, and A. Blais, Nonlinear dispersive regime of cavity QED: The dressed dephasing model, Phys. Rev. A **77**, 060305 (2008).

[73] P. Carbonaro, G. Compagno, and F. Persico, Canonical dressing of atoms by intense radiation fields, Phys. Lett. A **73**, 97 (1979).

[74] J. R. Schrieffer and P. A. Wolff, Relation between the Anderson and Kondo Hamiltonians, Physical Review **149**, 491 (1966).

[75] A. Blais, J. Gambetta, A. Wallraff, D. I. Schuster, S. M. Girvin, M. H. Devoret, and R. J. Schoelkopf, Quantum-information processing with circuit quantum electrodynamics, Phys. Rev. A **75**, 032329 (2007).

[76] A. Blais, R.-S. Huang, A. Wallraff, S. M. Girvin, and R. J. Schoelkopf, Cavity quantum electrodynamics for superconducting electrical circuits: An architecture for quantum computation, Phys. Rev. A **69**, 062320 (2004).

[77] M. J. Peterer, S. J. Bader, X. Jin, F. Yan, A. Kamal, T. J. Gudmundsen, P. J. Leek, T. P. Orlando, W. D. Oliver, and S. Gustavsson, Coherence and decay of higher energy levels of a superconducting transmon qubit, Phys. Rev. Lett. **114**, 010501 (2015).

[78] S. Poletto, J. M. Gambetta, S. T. Merkel, J. A. Smolin, J. M. Chow, A. D. Córcoles, G. A. Keefe, M. B. Rothwell, J. R. Rozen, D. W. Abraham, C. Rigetti, and M. Steffen, Entanglement of two superconducting qubits in a waveguide cavity via monochromatic two-photon excitation, Phys. Rev. Lett. **109**, 240505 (2012).

[79] M. H. Goerz, G. Gualdi, D. M. Reich, C. P. Koch, F. Motzoi, K. B. Whaley, J. c. v. Vala, M. M. Müller, S. Montangero, and T. Calarco, Optimizing for an arbitrary perfect entangler. II. Application, Phys. Rev. A **91**, 062307 (2015).

[80] M. H. Goerz, F. Motzoi, K. B. Whaley, and C. P. Koch, Charting the circuit QED design landscape using optimal control theory (2016), `arXiv:1606.08825`.

[81] M. Levitt, *Spin Dynamics: Basics of Nuclear Magnetic Resonance* (Wiley 2001).

Danksagung

Zuallererst möchte ich mich bei Professor Frank Wilhelm-Mauch für die Möglichkeit bedanken, in seiner Arbeitsgruppe die vorliegende Arbeit zu verfassen. Er war stets eine Quelle der Motivation und des Antriebs. Ich möchte meiner Mutter Brigitte und meiner Schwester Julia für ihre jahrelange Unterstützung danken. Besonders danke ich Zoë Ashwood für ihre Liebe und Geduld in all den Jahren. Auch möchte ich mich bei meinen Freunden und Kollegen Daniel Egger, Luke Govia und Pierre-Luc Dallaire-Demers, sowie Emily Pritchett, Bruno Taketani, Markku Stenberg, Felix Motzoi und Elie Assémat für die anregenden Diskussionen und die gemeinsame Zeit in Saarbrücken bedanken. Des weiteren Danke ich der gesamten Arbeitsgruppe, insbesondere Anette Messinger, Raphael Schmit, Peter Schuhmacher, Marius Schöndorf und Michael Kaicher fürs Probelesen. Zuletzt möchte ich mich bei Monika Francois bedanken, die stets ein offenes Ohr für Hilfe bei administrativen Aufgaben hatte.

Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form in einem Verfahren zur Erlangung eines akademischen Grades vorgelegt.

Ort, Datum

____

Unterschrift

____