

1 Method

The pulses will be optimized using Krotov’s method [1], a gradient-based optimization algorithm available publicly as a ready-to-use Python package [2]. In this chapter Krotov’s method for quantum optimal control will be briefly introduced and an implementation as a Python package. Then the numerical simulations will be explained.

1.1 Krotov’s Method for Quantum Optimal Control

Krotov’s method for quantum control fundamentally relies on the variational principle to minimize a functional

$$J\left[\left\{\left|\phi_k^{(i)}(t)\right\rangle\right\},\left\{u_l^{(i)}(t)\right\}\right]$$

that depends on the pulse shapes and the quantum states of the system evolved under these pulse shapes [2]. A detailed explanation of this functional can be found in [1]. However a quick summary of how the method works will be presented here. Starting with some guess pulse shapes, the initial state is forward propagated in time using a Hamiltonian of the same form as ??, and the value of the functional is calculated. Then changes to the pulse shapes which will decrease the value of the functional are calculated, by propagating states with these new pulse shapes, and these updates become the new pulse shapes. This continues until convergence is reached.

The underlying mathematical principles of Krotov’s method will not be addressed in the thesis, as that is outside the scope. Therefore the method will be treated as a black box function provided by the Python package which will be presented in the next section.

1.1.1 Krotov: the Python Package

A Python implementation of the Krotov’s method is available at [2], where an abstraction layer for doing optimizations is provided. The package will be referred to as `krotov`. The reason `krotov` was chosen is primarily because it allows for optimization of multiple objectives simultaneously. This is necessary in order to create a single encoding pulse which will realise the state transfer into the cat code basis for any arbitrary state.

The package is built on top of QuTiP, a popular software package for simulating the dynamics of quantum systems [3]. That means it can leverage all the useful features of that package, which is also another reason this method was chosen. The package provides a set of submodules which are used to setup the optimization problem. The important modules will be presented here.

The `objectives` module describes the goal of the optimization, be it a state-to-state transfer or a quantum gate. It is used to specify the initial and target states, the Hamiltonian and the initial guess pulses.

The `functionals` module provides a set of functionals which can be used for Krotov’s method. In this work the functional `chis_ss` has been chosen for all optimizations.

In the `convergence` module, help functions for the stopping criteria are provided for the optimization. It exposes values such as the overlap of the forward propagated state and target state $\langle\psi(T)|\psi^{\text{tar}}\rangle$ during the optimization which can be used to check when convergence is reached. The stopping criteria chosen for this thesis is either when a minimum fidelity is obtained, the fidelity being given by [4]

$$F = \frac{1}{S} \left| \sum_{k=1}^S \langle\psi_k(T)|\psi_k^{\text{tar}}\rangle \right|^2, \quad (1.1)$$

where S is the number of (simultaneous) state transfers we want to realise, or when the change in fidelity between iterations ΔF falls below a specified value (where a local minimum can be assumed to have been reached). The values of these will be specified in Section 1.2

Finally, the `optimization` module is used to setup the whole optimization. It is possible to choose the time discretization, convergence criteria, optimization functional, and a function which will run after every iteration. Also the step size $1/\lambda$ of the pulse updates can be chosen, which is further explained in [2]. Note however that although a large step size could achieve faster convergence, it can also lead to numerical instability.

1.2 Simulations

In this chapter the numerical simulations are presented and motivated. The first simulations were done for a single qubit, in order to get familiar with the method and assess its viability, while the last simulation is a proof of concept for encoding states into the cat code basis. All optimizations were run on a Lenovo Yoga 520-14IKB PC with an Intel(R) Core(TM) i7-8550 CPU running Windows 10.

1.2.1 Qubit State Control

In this simulation, the optimization was run to realise the state evolution $|0\rangle \rightarrow |1\rangle$ and then $|0\rangle \rightarrow |2\rangle$. The first pulse shape should be close to the ideal solution of a π -pulse: a gaussian pulse which performs a bit flip on a qubit, while the second is non-trivial. Therefore they are suitable goals to test the method and package.

Hamiltonian

The anharmonic resonator (qubit) in ?? was chosen as it is a simple model for a physical qubit. As stated earlier such a system will be referred to as a qubit even though it has more than two energy levels. To induce transitions between the states of the qubit, control pulse terms were added to ??, resulting in

$$\hat{H} = \omega_q \hat{a}^\dagger \hat{a} + \frac{K_q}{2} (\hat{a}^\dagger)^2 \hat{a}^2 + \Omega(t) e^{-i\omega_q t} \hat{a} + \Omega^*(t) e^{i\omega_q t} \hat{a}^\dagger, \quad (1.2)$$

where $\Omega(t)$ is the complex amplitude of the control pulse. Looking at the Hamiltonian above, it can be argued that it can be written in the form in ??, which is needed by `krotov`, with $u_0(t) = \Omega(t) e^{i\omega_q t}$ and $u_1(t) = \Omega^*(t) e^{-i\omega_q t}$. However, there are two problems that need to be addressed. Firstly, the oscillating factors require an unnecessarily fine time discretization of the pulses. Secondly, the Krotov package expects real-valued pulse amplitudes $\{u_i(t)\}$ as inputs. The first problem was avoided by transforming the Hamiltonian into the interaction picture with respect to $\omega_q \hat{a}^\dagger \hat{a}$, also known as the rotating frame. That is the Hamiltonian is transformed with

$$\hat{U} \hat{H} \hat{U}^\dagger + i \frac{d\hat{U}}{dt} \hat{U}^\dagger, \quad (1.3)$$

where $\hat{U} = e^{-i\omega_q \hat{a}^\dagger \hat{a} t}$ which results in

$$\hat{H} \rightarrow \frac{K_q}{2} (\hat{a}^\dagger)^2 \hat{a}^2 + \Omega(t) \hat{a} + \Omega^*(t) \hat{a}^\dagger. \quad (1.4)$$

Now the pulse amplitudes $u_0(t) = \Omega(t)$ and $u_1(t) = \Omega^*(t)$, i.e. the envelope of the physical control pulse which varies significantly slower than the actual pulse. The second problem was easily fixed with a rearrangement of the terms

$$\begin{aligned} \Omega(t) \hat{a} + \Omega^*(t) \hat{a}^\dagger &= \left[\text{Re}[\Omega(t)] + i \text{Im}[\Omega(t)] \right] \hat{a} + \left[\text{Re}[\Omega(t)] - i \text{Im}[\Omega(t)] \right] \hat{a}^\dagger = \\ &= \text{Re}[\Omega(t)] (\hat{a} + \hat{a}^\dagger) + \text{Im}[\Omega(t)] i (\hat{a} - \hat{a}^\dagger). \end{aligned}$$

For intuition, $(\hat{a} + \hat{a}^\dagger)$ and $i(\hat{a} - \hat{a}^\dagger)$ correspond to Bloch sphere rotations around the x-axis and y-axis respectively. Thus the final Hamiltonian became

$$\hat{H} = \underbrace{K_q/2 (\hat{a}^\dagger)^2 \hat{a}^2}_{\hat{H}_d} + \underbrace{\text{Re}[\Omega(t)]}_{u_0(t)} \underbrace{(\hat{a} + \hat{a}^\dagger)}_{\hat{H}_0} + \underbrace{\text{Im}[\Omega(t)]}_{u_1(t)} \underbrace{i(\hat{a} - \hat{a}^\dagger)}_{\hat{H}_1}. \quad (1.5)$$

The parameters of the qubit were chosen to model real superconducting qubits with $K_q = 2\pi \times 297$ MHz (and $\omega_q = 2\pi \times 6.2815$ GHz). The system Hamiltonian in Equation (1.5) was simulated with a truncated Hilbert space size conveniently chosen to be $N_q = 3$. A smaller truncated Hilbert space, and consequently smaller matrices, requires less computations but could possibly yield a poor approximation of the Hamiltonian. Thus the solutions of the optimization were analysed with this in mind.

Optimization Setup

As the long-term goal is to use this method in physical systems some constraints were added. To simulate the constraints of physical arbitrary waveform generators (AWG) a maximum sample rate of 4 GSa s^{-1} was chosen. Recall that the physical pulses oscillate at $\omega_q = 6.2815 \text{ GHz}$, a rate which 4 GSa s^{-1} can impossibly resolve. Although AWGs with higher sample rates than 4 GSa s^{-1} exist, they are expensive. However, rotating frame transformation permits the use of a relatively low rate AWG to generate the pulse envelopes $\Omega(t)$ and a high rate tone generator to create a carrier signal at ω_q , with these signals later combined in a mixer. Further, due to the noise sensitivity of superconducting systems an amplitude constraint is needed to keep the system cool enough. To set a realistic maximum amplitude A_m some derivation was required. First the maximum amplitude was chosen with the assumption that a gaussian pulse

$$g(t) = \frac{\pi/2}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t}{\sigma})^2} \quad (1.6)$$

with $\sigma = 3 \text{ ns}$ can physically realise a π -rotation $|0\rangle \rightarrow |1\rangle$ in a planar transmon qubit. The problem with gaussian pulses however is that they do not go to zero at the edges. To fix this **krotov** provides a function for generating Blackman pulses. A Blackman pulse with a total length of 6σ is a good approximation of a Gaussian pulse, with the added benefit of going to zero at the edges. However, due to the approximation, the integral of the Blackman pulse will not exactly be equal to the integral of the corresponding gaussian pulse, which is needed to realise the π -rotation. This can be fixed by using a prefactor C which can be calculated by solving

$$C \int_0^{6\sigma} \frac{1}{\sqrt{2\pi}\sigma} B(t) dt = \int_0^{6\sigma} g(t - 3\sigma) dt \quad (1.7)$$

where $B(t)$ is the Blackman pulse. This is now used to determine the maximum amplitude

$$A(\sigma) = \frac{C}{\max(\sigma, 3)\sqrt{2\pi}} \quad (1.8)$$

with $A_m = A(\sigma = 3)$. This amplitude constraint is enforced by calling a function after every iteration which limits all pulse shape values $|\Omega| < A_m$. Conveniently, the Blackman pulse was also used as the initial guess pulse with $u_0(t) = A(T/6)B(t)$, $u_1(t) = 0$.

Another constraint was added so that the edges of the pulses were always zero. This was done with a switch-on/switch-off of 2 ns . This ramp shape is half a Blackman pulse and is provided by **krotov**'s **flattop** function.

For the $|0\rangle \rightarrow |1\rangle$ state evolution, pulse shapes were optimized with varying lengths from 4.25 ns to 30 ns with convergence criteria $F > 0.99999$ or $\Delta F < 10^{-7}$. The step size was chosen as $\lambda = \frac{1}{\frac{1}{2}A_m}$. The reason for optimizing for various pulse lengths was that faster pulse changes gives a broader support in the frequency spectrum which can ultimately induce transitions in other levels. Therefore the optimization needs to find a solution which circumvents this problem.

Lastly, for the $|0\rangle \rightarrow |2\rangle$ state evolution, pulse shapes were optimized with varying lengths from 22 ns to 30 ns with convergence criteria $F > 0.99999$ or $\Delta F < 10^{-9}$. The low ΔF was needed because the changes were very small in the beginning of the optimization. The step size was chosen as $\lambda = \frac{1}{2A_m}$ due to slow convergence. In contrast to the $|0\rangle \rightarrow |1\rangle$ state transfer both initial guess pulses were chosen to be Blackman pulses $u_0(t) = u_1(t) = A(T/6)B(T)$.

1.2.2 Cat code encoding

For the cat code encoding, six simultaneous objectives were chosen to give encoding pulses which maps the six ‘‘cardinal’’ qubit states on the Bloch sphere to the corresponding cat code states in the resonator. These initial qubit states are

$$\begin{aligned} &|0\rangle, |1\rangle, \\ &\frac{|0\rangle + |1\rangle}{\sqrt{2}}, \frac{|0\rangle - |1\rangle}{\sqrt{2}}, \\ &\frac{|0\rangle + i|1\rangle}{\sqrt{2}}, \frac{|0\rangle - i|1\rangle}{\sqrt{2}} \end{aligned}$$

with target state $|0\rangle$. The resonator initial state is $|0\rangle$ (in the Fock basis) and the corresponding target states are

$$\begin{aligned} & |C_\alpha\rangle, |C_{i\alpha}\rangle, \\ & \frac{|C_\alpha\rangle + |C_{i\alpha}\rangle}{\sqrt{2}}, \frac{|C_\alpha\rangle - |C_{i\alpha}\rangle}{\sqrt{2}}, \\ & \frac{|C_\alpha\rangle + i|C_{i\alpha}\rangle}{\sqrt{2}}, \frac{|C_\alpha\rangle - i|C_{i\alpha}\rangle}{\sqrt{2}}. \end{aligned}$$

The hope is that by optimizing these six states the pulse shapes will approximate the unitary in ??.

Hamiltonian

In this simulation the coupled qubit-resonator system in ?? was used. As the operators \hat{a} and b commute, the rewriting used in Equation (1.5) was also used for the resonator. The rotating frame transformation is done here with respect to $\omega_r \hat{a}^\dagger \hat{a} + \omega_q \hat{b}^\dagger \hat{b}$. However, as the coupling term has no time dependence and does not commute with the operators $\hat{a}^\dagger \hat{a}$ and $\hat{b}^\dagger \hat{b}$ there will still be a time dependence in the rotating frame

$$g(\hat{a}\hat{b}^\dagger + \hat{a}^\dagger\hat{b}) \rightarrow g(\hat{a}\hat{b}^\dagger e^{i(\omega_q - \omega_r)t} + \hat{a}^\dagger\hat{b}e^{-i(\omega_q - \omega_r)t}). \quad (1.9)$$

This was a problem because `krotov` expects \hat{H}_d to be time independent. There is a way to “simulate” the coupling term (1.9) by inserting it as an external time dependent pulse and omit it from optimization which was done in this case. The coupling term was then split into a real and imaginary part which is, as stated before, required by `krotov`.

The final Hamiltonian provided to `krotov` was then

$$\begin{aligned} \hat{H} = & \underbrace{K_r/2(\hat{a}^\dagger)^2\hat{a}^2 + K_q/2(\hat{b}^\dagger\hat{b})^2}_{\hat{H}_d} + \\ & + \underbrace{\text{Re}[\Omega_r(t)](\hat{a} + \hat{a}^\dagger)}_{u_0(t)\hat{H}_0} + \underbrace{\text{Im}[\Omega_r(t)]i(\hat{a} - \hat{a}^\dagger)}_{u_1(t)\hat{H}_1} + \underbrace{\text{Re}[\Omega_q(t)](\hat{b} + \hat{b}^\dagger)}_{u_2(t)\hat{H}_2} + \underbrace{\text{Im}[\Omega_q(t)]i(\hat{b} - \hat{b}^\dagger)}_{u_3(t)\hat{H}_3} + \\ & + \underbrace{[\cos((\omega_q - \omega_r)t)]g(\hat{a}\hat{b}^\dagger + \hat{a}^\dagger\hat{b})}_{u_4(t)\hat{H}_4} + \underbrace{[\sin((\omega_q - \omega_r)t)]ig(\hat{a}\hat{b}^\dagger - \hat{a}^\dagger\hat{b})}_{u_5(t)\hat{H}_5} \end{aligned} \quad (1.10)$$

The parameters of this system is presented in Table 1.1. These were chosen to resemble the physical values measured in [4] except for g which was chosen at relatively large value to reduce the pulse length needed to realise the transfer.

Table 1.1: Hamiltonian parameters

Param.	Value
$\omega_q/(2\pi)$	6.2815 GHz
$K_q/(2\pi)$	297 MHz
$\omega_r/(2\pi)$	8.3056 GHz
$K_r/(2\pi)$	4.5 kHz
$g/(2\pi)$	0.225 GHz

The truncated Hilbert space size for the resonator was chosen to be $N_r = 8$ with the mean photon number $\alpha = 2$. Due to the large system size, the qubit's truncated Hilbert space size was set to $N_q = 2$.

Optimization Setup

The optimization was setup in the same way as the previous simulation, however due to the rapid oscillations at $|\omega_q - \omega_r|$ a finer time discretization of 24 GSa s^{-1} was chosen. The step size was chosen as $\lambda = \frac{1}{\frac{1}{2}A_m}$ and the pulse length $T = 60 \text{ ns}$ to allow the coupling to affect the system. Further, convergence criteria was chosen as $F > 0.999999$ or $\Delta F < 10^{-6}$. Another important step in optimizing in the rotating frame was to convert the target states into the rotating frame in order for the final states to be correct in the lab frame. Initial guess pulses were chosen to be the same Blackman pulses in the setup for the $|0\rangle \rightarrow |2\rangle$ transfer.