1 Method

In this chapter Krotov's method for quantum optimal control will be briefly introduced and an implementation as a Python package. Then the numerical experiments will be explained.

1.1 Krotov's Method for quantum optimal control

Krotov's method fundamentally relies on the variational principle to minimize a functional

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

where the constraints are included as Lagrange multipliers [1]. A detailed explanation of this functional when the method is applied to quantum systems can be found in [2].

1.2 Krotov: the Python package

A Python implementation of the Krotov package is available at `https://krotov.readthedocs.io/en/latest/`. It provides simple functions and objects to

1.3 Optimization Experiments

In this chapter the numerical optimization experiments are presented and motivated.

1.3.1 Hamiltonian

To test the method, the anharmonic oscillator in **??** will be chosen as it is often used as the physical realisation of a qubit. Throughout this thesis, such a system will be referred to as a qubit even though it has more than two energy levels. Consequently, the resonance frequency of the qubit ω_{01} refers to the transition between $|0\rangle$ and $|1\rangle$. To induce transitions between these states, control pulse terms are added to **??**

$$\hat{H} = \omega_{01}\hat{a}^\dagger\hat{a} - \frac{\kappa}{2}\left(\hat{a}^\dagger\hat{a}\right)^2 + \Omega(t)\mathrm{e}^{i\omega_{01}t}\hat{a} + \Omega^*(t)\mathrm{e}^{-i\omega_{01}t}\hat{a}^\dagger \tag{1.1}$$

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 **??** with $u_0(t) = \Omega(t)\mathrm{e}^{i\omega_{01}t}$ and $u_1(t) = \Omega^*(t)\mathrm{e}^{-i\omega_{01}t}$. However, there are two problems that need to be adressed. Firstly, the oscillating factors will 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 can be avoided by transforming the Hamiltonian into the interaction picture. Choosing $H_A = \omega_{01}\hat{a}^\dagger\hat{a}$, eq. (1.1) transforms[1] into

$$\hat{H} \rightarrow\ -\frac{\kappa}{2}\left(\hat{a}^\dagger\hat{a}\right)^2 + \Omega(t)\hat{a} + \Omega^*(t)\hat{a}^\dagger. \tag{1.2}$$

Now the pulse amplitudes are $u_0(t) = \Omega(t)$ and $u_1(t) = \Omega^*(t)$, i.e. the envelope of the physical driving pulse which varies significantly slower. The second problem can now be easily fixed with a rearrangment of the terms

$$\Omega(t)\hat{a} + \Omega^*(t)\hat{a}^\dagger = \Big[\mathrm{Re}[\Omega(t)] + i\mathrm{Im}[\Omega(t)]\Big]\hat{a} + \Big[\mathrm{Re}[\Omega(t)] - i\mathrm{Im}[\Omega(t)]\Big]\hat{a}^\dagger =$$

$$= \mathrm{Re}[\Omega(t)]\left(\hat{a} + \hat{a}^\dagger\right) + \mathrm{Im}[\Omega(t)]i\left(\hat{a} - \hat{a}^\dagger\right).$$

For intuition, $\left(\hat{a} + \hat{a}^\dagger\right)$ and $i\left(\hat{a} - \hat{a}^\dagger\right)$ correspond to Bloch sphere rotations around the x-axis and y-axis respectively. This leaves us with the final Hamiltonian

$$\hat{H} = \underbrace{-\kappa/2\left(\hat{a}^\dagger\hat{a}\right)^2}_{\hat{H}_d} + \underbrace{\mathrm{Re}[\Omega(t)]}_{u_0(t)}\underbrace{(\hat{a} + \hat{a}^\dagger)}_{\hat{H}_0} + \underbrace{\mathrm{Im}[\Omega(t)]}_{u_1(t)}\underbrace{i(\hat{a} - \hat{a}^\dagger)}_{\hat{H}_1}. \tag{1.3}$$

[1] Full derivation is shown in **??**

1.3.2 Optimization Setup

The system Hamiltonian (1.3) is simulated using its matrix representation and a chosen Hilbert space size, which in this case is conveniently chosen to be $L = 3$. A smaller Hilbert space, and consequently smaller matrices, requires less computations however could possibly be a poor approximation of the Hamiltonian. This, however, is not a problem in this case as we can neglect

The goal of the optimization is to realise state transfers Rotating frame

4 gigasamples/s

Amplitude constraint (with pi pulse calibration)

Guess pulses half amplitude (actually blackman pulses)

convergence criteria fidelity F change between iterations falls below a certain critera ΔF

1.3.3 $|0\rangle \rightarrow |1\rangle$ state transfer

Pulse shapes were optimized with varying lengths from 4.25 ns to 30 ns with convergence criteria $F > 0.99999$ or $\Delta F < 10^{-7}$. Step size $\lambda = \frac{1}{\frac{1}{2}A_m}$

1.3.4 $|0\rangle \rightarrow |2\rangle$ state transfer

Pulse shapes were optimized with varying lengths from 22 ns to 30 ns with convergence criteria $F > 0.99999$ or $\Delta F < 10^{-9}$. Step size $\lambda = \frac{1}{2A_m}$