ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Exploring optimized preparation schemes for logical qubit states encoded in the trapped ion motional oscillator

Semester Thesis

Chams Rutkowski

June 30, 2018

Advisors: Christa Flühmann
Prof. Dr. Jonathan Home
Co-advisor: Prof. Dr. Jürg Leuthold

Department of Information Technology and Electrical Engineering, ETH Zürich

Abstract

The purpose of this semester project was to implement a computational tool in order to create exotic harmonic oscillator states in a quasi-deterministic fashion. To achieve this goal, a set of piece-wise constant parameters of an interaction hamiltonian was considered. In practice, a gradient ascent based algorithm named GRAPE (GRadient Ascent Pulse Engineering) was employed. The algorithm allowed to produce non trivial shapes of laser pulses that must be applied to the trapped ion system in order to reach the desired target state.

Contents

Contents **iii**

1 Introduction **1**

2 Theoretical concepts **5**
- 2.1 Phase space 5
  - 2.1.1 Harmonic oscillator 6
  - 2.1.2 Coherent states 8
  - 2.1.3 Cat states 12
  - 2.1.4 Wigner function 13
- 2.2 Trapped ions 16
  - 2.2.1 State dependent force 19

3 Simulation **23**
- 3.1 GRAPE 23
  - 3.1.1 coherent state transfer 24
  - 3.1.2 Implementation 27
- 3.2 Experimental constraints 28
- 3.3 Grid state 36

4 Conclusion and outlook **47**

A GRAPE Python code **51**

Bibliography **61**

Chapter 1

Introduction

The recent experimental advancements in modular position and momentum measurements, led to the realization of a highly non trivial non classical state of a harmonic oscillator, realized using the axial motion of a trapped $^{40}Ca^+$ ion [1]. The state that we are interested in is called Grid state (see Fig.1.1) and represents an approximate codeword of GKP bosonic code [2].

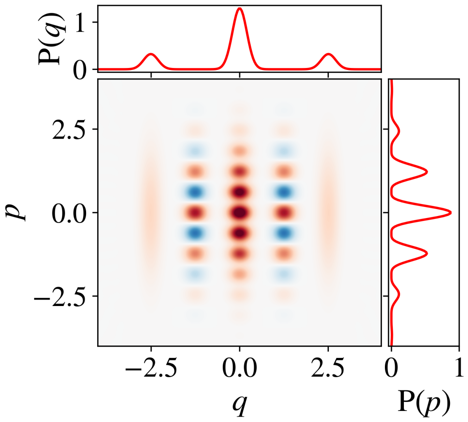

Figure 1.1: Wigner function representation of the grid state encoding a logical $|\bar{0}\rangle$ state.

As we shall see, the realization of this state in trapped ion experiments presents several difficulties. The goal of this project is to improve upon the current preparation method for such a state.

Let us consider the energy level structure of the calcium ion $^{40}Ca^+$, Fig.1.2, and let us explain how the readout of the information in such a system works.

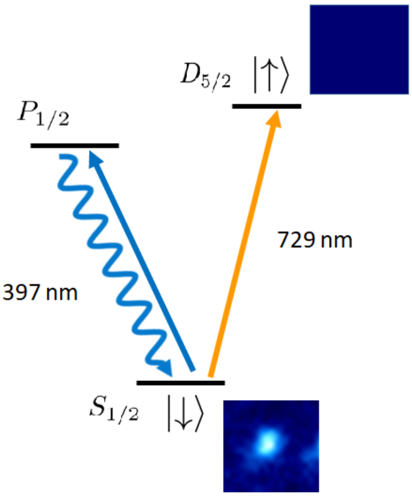

Figure 1.2: Simplified structure of energy levels of $^{40}Ca^{+}$ ion. Bottom right labels indicate the total angular momentum of the energy state, the letters stand for states with a particular total orbital angular momentum, to be added to the spin (e.g $S_{1/2}$ is the state with L=0 and S=$\frac{1}{2}$). In the fluorescence detection, the ground state will appear as a bright spot on the camera, whereas the excited state will be associated to the absence of such spot.

The two level system usually considered in $^{40}Ca^{+}$ is made up of the Zeeman sub-levels $|\downarrow\rangle = |S_{1/2}, m_j = 1/2\rangle$ and $|\uparrow\rangle = |D_{5/2}, m_j = 3/2\rangle$. This choice is particularly convenient since the optical transition $D_{5/2} \Longleftrightarrow S_{1/2}$ is coupled by a quadrupole that makes it less likely to occur; thus one can consider $D_{5/2}$ as a long living state. On the other hand, the state $P_{1/2}$, which is dipole coupled to $S_{1/2}$ and therefore short living, is used for the readout of photons scattered in the fluorescence process.
When dealing with trapped ions, one has to consider the total system composed of an ion and the harmonic motion of the ion in the trapping potential which represents the mechanical degree of freedom of the total system. Since in this work we are mainly interested in the harmonic oscillator part, we should make sure that our quantum harmonic oscillator state is not screwed up upon fluorescence detection i.e. the projective measurement. As one may imagine, the harmonic oscillator energy remains unchanged only if the system is in $|\uparrow\rangle \otimes |H.Osc\rangle$ state. The other situation $|\downarrow\rangle \otimes |H.Osc\rangle$ is unsuitable for our purposes, since photons carry a non-zero momentum during the scattering process and thus change the energy of the trap in an unpredictable manner. Consequently, the preparation of complicated harmonic oscillator states such as cat states or grid states, whose details will be explained later on, is possible only probabilistically. Moreover, as the

number of measurements needed to create a particular state increases, the probability to find the system in the state we are interested in decays, making the quantum computation with such objects unreliable.

This work will be focused on reducing the number of projective measurements needed to end up in $|\uparrow\rangle \otimes |H.Osc\rangle$ state, therefore creating it almost deterministically.

Outline :

Keeping in mind the goal of the project, let us lie down the structure of this report.
The theoretical concepts essential to understand this work are introduced in Chapter 2. Chapter 3 focuses on the analysis of GRAPE algorithm and some experimental bounds are also considered. The work is concluded with Chapter 4 in which we discuss the results of simulations and give a look at future experimental possibilities.

Chapter 2

Theoretical concepts

This chapter introduces basic notions of phase space and quantum optics [12]. Next, an example of how to create a cat state and the grid state in trapped ions will be given.

2.1 Phase space

The notion of phase space plays a central role in the Hamiltonian formulation of classical physics. Here we will just remind ourselves of some important rules of such a space.

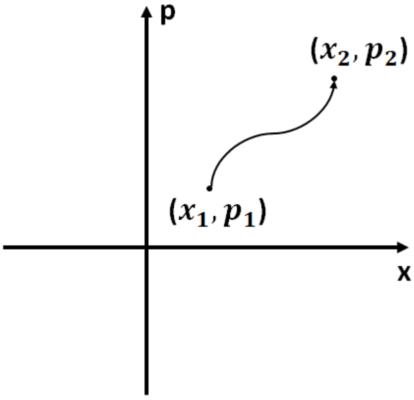

Figure 2.1: Trajectory representing a possible time evolution of a point in phase space.

As shown in the Fig.2.1 one point in phase space (2 dimensional for sim-

plicity of exposition, though the concept can be generalized to an arbitrary number of dimensions) is represented by the tuple (x, p) where x is the position of the object and p corresponds to its momentum. Knowing these 2 quantities in classical physics completely determines the state of the system. All the other quantities of interest in classical mechanics may be derived from the notion of these 2 quantities.
Another thing that one is usually interested in is the time evolution of (x, p), since the knowledge of such quantity provides us with the full description of the system over time. The rule which updates the state of the system is obtained from the notion of the hamiltonian, which is the total energy. What stated above can be summarized by the Hamilton's equations of motion:

$$\begin{aligned}\frac{dp}{dt} &= -\frac{\partial H}{\partial x}\\ \frac{dx}{dt} &= \frac{\partial H}{\partial p}\end{aligned} \tag{2.1}$$

In quantum mechanics the notion of state is replaced with the one of state vector $|\Psi(t)\rangle$ or density operator $\hat{\rho}(t) = \sum_j p_j \left|\Psi_j(t)\right\rangle \left\langle\Psi_j(t)\right|$, when dealing with open quantum systems. These objects provide full knowledge about the system and are still updated in time by the Hamiltonian (which is a hermitian operator). The time evolution of $|\Psi(t)\rangle$ is determined by the Schrödinger's equation:

$$i\hbar\frac{\partial\,|\Psi(t)\rangle}{\partial t} = \hat{H}\,|\Psi(t)\rangle \tag{2.2}$$

and for the density operator we get:

$$\frac{\partial\hat{\rho}(t)}{\partial t} = \frac{1}{i\hbar}\left[\hat{H}, \hat{\rho}(t)\right] \tag{2.3}$$

2.1.1 Harmonic oscillator

The harmonic oscillator is another concept in physics which is of great importance; let us review the basics from a heuristic point of view.
Both in classical and in quantum setting, the object we are interested in is the Hamiltonian describing the harmonic oscillator. For a classical system we may write for a free harmonic oscillator:

$$H_{h.o.} = \frac{p^2}{2m} + \frac{1}{2}m\omega^2 x^2 \tag{2.4}$$

with m being the mass of the particle and ω the resonance frequency of oscillation. It is easy to check that the phase space trajectories computed in time by particles evolving under $H_{h.o.}$ have elliptic form as shown in Fig.2.2:

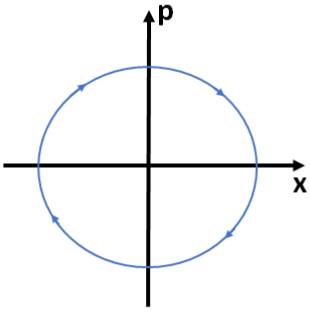

Figure 2.2: Phase space trajectory representing time evolution of a classical particle with $m = 1$ and $\omega = 1$ in a harmonic potential.

Let us now consider the bosonic quantum harmonic oscillator. In this case position and momentum will be replaced with their respective operators $\hat{x}$ and $\hat{p}$.

$$\hat{H}_{h.o.} = \frac{\hat{p}^2}{2m} + \frac{1}{2}m\omega^2\hat{x}^2 \tag{2.5}$$

One can also define a more general form of such Hamiltonian, particularly useful when dealing with bosonic quantum fields (e.g. electric field), by introducing the creation and annihilation operators $\hat{a}^\dagger$ and $\hat{a}$. The mathematical details of these operators depend on the physical system of interest, but in the case of the harmonic trap we can write $\hat{a} = \sqrt{\frac{m\omega}{2\hbar}}(\hat{x} + \frac{i}{m\omega}\hat{p})$ and $\hat{a}^\dagger = \sqrt{\frac{m\omega}{2\hbar}}(\hat{x} - \frac{i}{m\omega}\hat{p})$. The resulting Hamiltonian for a particular oscillation mode will be:

$$\hat{H}_{h.o.} = \hbar\omega(\hat{a}^\dagger\hat{a} + \frac{1}{2}) \tag{2.6}$$

In the matrix representation this object is positive definite and hermitian. Thus, $\hat{H}_{h.o.}$ has n distinct positive and real eigenvalues, which are the discrete energy levels of the system. The operators $\hat{a}$ and $\hat{a}^\dagger$ obey the following commutation relation:

$$\begin{gathered}\left[\hat{a}, \hat{a}^\dagger\right] = 1 \\ [\hat{a}, \hat{a}] = \left[\hat{a}^\dagger, \hat{a}^\dagger\right] = 0\end{gathered} \tag{2.7}$$

The Hamiltonian (2.6) together with the commutation relation in (2.7) give us a nice picture of quantized energy ladder, which can be climbed up and down by the action of field operators on the eigenstates of $\hat{H}_{h.o}$. The eigenstates $|n\rangle$ of the harmonic oscillator are called Fock states and they form a complete and orthonormal set of states with definite energy $E_n = \hbar\omega(n + \frac{1}{2})$

associated to them. The action of field operators on $|n\rangle$ is defined by:

$$\begin{aligned}\hat{a}\,|n\rangle &= \sqrt{n}\,|n-1\rangle \\ \hat{a}^{\dagger}\,|n\rangle &= \sqrt{n+1}\,|n+1\rangle\end{aligned} \tag{2.8}$$

Since the Hamiltonian is positive definite, the ladder of states will extend infinitely in the ″up″ direction and will terminate in the ″down″ direction with a state $|0\rangle$ with $n = 0$ annihilated by $\hat{a}$, called ground state. The latter has no classical analog since it is a state with no particles in it but still with a non zero energy, attributed to vacuum fluctuations. With this being said we can define a general Fock state $|n\rangle$ as:

$$|n\rangle = \frac{(\hat{a}^{\dagger})^{n}}{\sqrt{n!}}\,|0\rangle \tag{2.9}$$

Lastly, since $|n\rangle$ is an eigenstate of $\hat{H}_{h.o.}$ it evolves in time only with a phase factor and it is a stationary state. This and the uncertainty relation $\Delta x_{|n\rangle}\Delta p_{|n\rangle} = \frac{\hbar}{2}(2n+1)$ give a nice intuition on how this state might look like in phase space. The Fock state will form a ring of constant energy and area given by the Heisenberg uncertainty between x and p as shown in Fig.2.3. The fact that the energy is well defined makes the phase totally unspecified as a consequence of energy-time uncertainty relation.

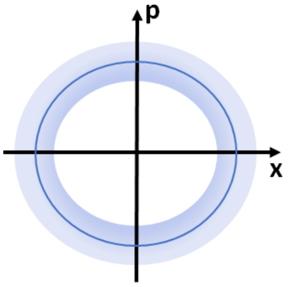

Figure 2.3: Pictorial representation of a Fock state $|n\rangle$.

Later on a more rigorous representation of such states will be explained.

2.1.2 Coherent states

Let us now consider coherent states, and motivate their importance with an example picked up from quantum optics.

Without going into details, the expectation value of an electric field operator in Fock basis yields exactly zero; this might sound suspicious since the classical measurement of such field gives a non zero value. This suggests us that we usually do not deal with fields that are in an eigenstate of the harmonic oscillator. The quantum states which represent more closely classical electromagnetic fields are called coherent states. They are defined to be a superposition of Fock states with probability amplitudes taking values according to the Poisson distribution:

$$|\alpha\rangle = e^{\frac{-|\alpha|^2}{2}} \sum_{n=0}^{\infty} \frac{\alpha^n}{\sqrt{n!}} |n\rangle \tag{2.10}$$

where $\alpha \in \mathbb{C}$ and $e^{\frac{-|\alpha|^2}{2}}$ is a normalization factor. An important property of this state is that it is an eigenstate of the annihilation operator:

$$\hat{a} |\alpha\rangle = \alpha |\alpha\rangle \tag{2.11}$$

notice that since $\hat{a}$ is not hermitian it does not have real distinct eigenvalues and $\alpha \in \mathbb{C}$ as stated before. This property gives a nice interpretation of such a state: it tells us that taking away 1 photon from the system does not change its state but only its amplitude and the global phase. We also point out that the vacuum state is just a coherent state with $\alpha = 0$. The Poisson distribution is such that the expectation value and the variance are the same, in our case the photon number operator $\hat{n} = \hat{a}^\dagger \hat{a}$ yields zero as we evaluate it in the vacuum state:

$$\langle \hat{n} \rangle_{|\alpha=0\rangle} = 0 = (\Delta \hat{n}^2)_{|\alpha=0\rangle} \tag{2.12}$$

As a consequence of the (2.12) the photon number is completely specified and it is exactly zero with no fluctuations. This in turn implies that the phase of such state will be undetermined. The location of the vacuum state in phase space will be given by:

$$\begin{aligned} \langle \hat{x} \rangle_{|0\rangle} &\approx \langle (\hat{a} + \hat{a}^\dagger) \rangle_{|0\rangle} = 0 \\ \langle \hat{p} \rangle_{|0\rangle} &\approx \langle [i(\hat{a} - \hat{a}^\dagger)] \rangle_{|0\rangle} = 0 \end{aligned} \tag{2.13}$$

The equations in (2.13) and the phase-number uncertainty provide an intuitive picture for the representation of the vacuum shown in phase space (see Fig.2.4).

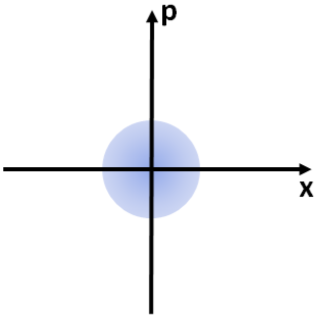

Figure 2.4: Phase space representation of vacuum state $|0\rangle$. The extension of the smearing is given by the uncertainty between x and p. In the case of a vacuum state the uncertainty relation satisfies the minimum uncertainty condition: $\Delta x \Delta p = \frac{1}{4}$. States that satisfy this condition are called minimum uncertainty wavepackets.

The vacuum state and the coherent state are related by the displacement operator defined as: $\hat{D}(\alpha) = e^{\alpha^* \hat{a} - \alpha \hat{a}^\dagger}$. $\hat{D}(\alpha)$ has the following action on $|0\rangle$:

$$|\alpha\rangle = \hat{D}(\alpha)\,|0\rangle \tag{2.14}$$

Notice though that the coherent state is not a stationary state of the harmonic oscillator Hamiltonian. Forgetting about the constant factor $\frac{1}{2}$ in the eq.(2.6) we get:

$$\begin{aligned} & e^{-i\frac{\hat{H}_{h.o.}t}{\hbar}}\,|\alpha\rangle \\ =& e^{-i\omega\hat{n}t} e^{\frac{|\alpha|^2}{2}} \sum_{n=0}^{\infty} \frac{\alpha^n}{\sqrt{n!}}\,|n\rangle \\ =& e^{\frac{|\alpha|^2}{2}} \sum_{n=0}^{\infty} \frac{(\alpha e^{-i\omega t})^n}{\sqrt{n!}}\,|n\rangle \\ =& \left| e^{-i\omega t}\alpha \right\rangle \end{aligned} \tag{2.15}$$

A phase space representation of a coherent state is depicted in Fig.2.5:

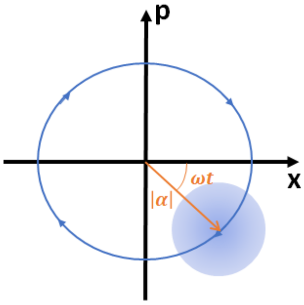

Figure 2.5: Trajectory of $|\alpha\rangle$ state as it evolves in time. The minimum uncertainty wave packet follows the classical orbit and gains a phase factor $-\omega t$ during the time evolution.

Rotating frame

In quantum optics one is usually interested in the interaction effects rather than the bare terms in the Hamiltonian. In order to get rid of usually not very interesting effects we consider the time evolution of states in the interaction picture, also called Dirac picture. In phase space this is equivalent to considering the states in a coordinate system rotating at the resonance frequency of the harmonic oscillator ω. Defining the new coordinate system as:

$$\begin{aligned} \hat{x}' &= (\frac{\hat{a}+\hat{a}^\dagger}{2})\cos(\omega t) - (\frac{\hat{a}-\hat{a}^\dagger}{2i})\sin(\omega t) \\ \hat{p}' &= -(\frac{\hat{a}+\hat{a}^\dagger}{2})\cos(\omega t) - (\frac{\hat{a}-\hat{a}^\dagger}{2i})\sin(\omega t) \end{aligned} \tag{2.16}$$

the coherent state will appear to be stationary:

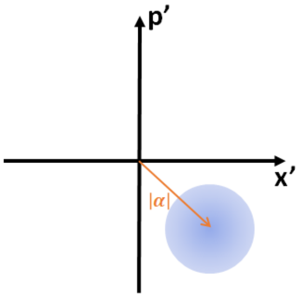

Figure 2.6: Coherent state in the rotating frame.

2.1.3 Cat states

Now that the coherent states were introduced let us focus on something more exotic. As the name already suggests, cat states are the realization of the famous thought experiment proposed in 1935 by Erwin Schrödinger. The cat state could be viewed as a quantum superposition of "macroscopic" states, which are the coherent states described in the previous subsection. The most general cat's state vector can be written as:

$$|\psi_{cat}\rangle = \frac{1}{\sqrt{N}}(e^{i\varphi_1}|\alpha_1\rangle + e^{i\varphi_2}|\alpha_2\rangle) \tag{2.17}$$

where the normalization pre-factor $\frac{1}{\sqrt{N}}$ takes into account the overlap of $|\alpha_1\rangle$ and $|\alpha_2\rangle$. If $|\alpha_1 - \alpha_2| \gg 1$, then the overlap between $|\alpha_1\rangle$ and $|\alpha_2\rangle$ is negligible and the following approximation is valid:

$$|\psi_{cat}\rangle \approx \frac{1}{\sqrt{2}}(e^{i\varphi_1}|\alpha_1\rangle + e^{i\varphi_2}|\alpha_2\rangle) \tag{2.18}$$

The quantum coherence of such state can is represented by the non vanishing off diagonal elements of the density matrix:

$$\begin{aligned} \hat{\rho}_{cat} = |\psi_{cat}\rangle\langle\psi_{cat}| \approx \frac{1}{2}(&|\alpha_1\rangle\langle\alpha_1| + |\alpha_2\rangle\langle\alpha_2| \\ &+ e^{i(\varphi_1-\varphi_2)}|\alpha_1\rangle\langle\alpha_2| + e^{i(\varphi_2-\varphi_1)}|\alpha_2\rangle\langle\alpha_1|) \end{aligned} \tag{2.19}$$

An interesting case of cat states are phase cats, which are obtained by fixing the differnece between φ_2 and φ_1 to ± 1 [3]. The phase cat states take the following form:

$$|\psi_{cat}^{\pm}\rangle = \frac{1}{\sqrt{2(1 \pm e^{-2|\alpha|})}}(|\alpha\rangle \pm |-\alpha\rangle) \tag{2.20}$$

even parity cats are denoted by the + sign whereas the - sign is attributed to the odd parity cats. Thus, such states are the eigenstates of the parity operator $\hat{P} = (-1)^{\hat{a}^\dagger\hat{a}}$ with eigenvalues ± 1:

$$\hat{P}|\psi_{cat}^{\pm}\rangle = \pm|\psi_{cat}^{\pm}\rangle \tag{2.21}$$

To conclude the discussion about the phase space, a formal way for state representations called Wigner function is introduced in the next subsection.

2.1.4 Wigner function

Previously we described intuitively how to represent quantum states of the harmonic oscillator in phase space, now we introduce a very powerful and more formal representation to this end.

The Wigner function $W(\alpha)$ is a quasi-probability distribution used for the representation of pure and mixed quantum states in phase space. As we shall see, it is real valued and can take both positive and negative values unlike classical probability distributions, which are always ≥ 0. Formally, it is the 2D Fourier Transform of the symmetric characteristic function $C_s(\lambda)$. The latter is the expectation value of the displacement operator $\hat{D}(\lambda)$ in a state described by the density operator $\hat{\rho}$. By looking at the power series expansion of $\hat{D}(\lambda)$ we note that all products of operators appearing in such a series are naturally symmetrized [3]:

$$\hat{D}(\lambda) = \mathbb{1} + (\lambda\hat{a}^\dagger - \lambda^*\hat{a}) + \lambda^2(\hat{a}^\dagger)^2 - \lambda\lambda^*(\hat{a}^\dagger\hat{a} + h.c.) + \lambda^{*2}\hat{a}^2 ... \tag{2.22}$$

from (2.22) follows the definition of the symmetric characteristic function:

$$C_s^{[\hat{\rho}]}(\lambda) = \langle\hat{D}(\lambda)\rangle = \mathrm{Tr}\left\{\hat{\rho}e^{\lambda\hat{a}^\dagger - \lambda^*\hat{a}}\right\} \tag{2.23}$$

Since $\hat{D}(\lambda)$ is a unitary operator, the absolute value of $C_s^{[\hat{\rho}]}(\lambda)$ is upper bounded by 1 and reaches its maximum value at the origin:

$$C_s^{[\hat{\rho}]}(0) = \mathrm{Tr}\{\hat{\rho}\} = 1 \tag{2.24}$$

to conclude, the conjugation property of $C_s^{[\hat{\rho}]}(\lambda)$:

$$C_s^{[\hat{\rho}]}(-\lambda) = (C_s^{[\hat{\rho}]}(\lambda))^* \tag{2.25}$$

implies that the Fourier transform of the characteristic function (i.e. W(α)) is real valued. Now that the most important properties of $C_s^{[\hat{\rho}]}(\lambda)$ are clear, let us define the Wigner function as:

$$W(\alpha) = \frac{1}{\pi^2}\int d^2\lambda C_s^{[\hat{\rho}]}(\lambda)e^{(\alpha\lambda^* - \alpha^*\lambda)} \tag{2.26}$$

with $\alpha, \lambda \in \mathbb{C}$ we can write $\lambda = \lambda' + i\lambda''$ and $\alpha = \alpha' + i\alpha'' = q + ip$. These definitions imply that (2.26) is just a Fourier relation since $\alpha\lambda^* - \alpha^*\lambda = 2i\lambda' p - 2i\lambda'' x$. The inverse Fourier transform yields:

$$C_s^{[\hat{\rho}]}(\lambda) = \int d^2\alpha W(\alpha)e^{(-\alpha\lambda^* + \alpha^*\lambda)} \tag{2.27}$$

The evaluation of (2.27) in $\lambda = 0$ leads to the conclusion that $W(\alpha)$ is normalized, as a consequence of (2.24):

$$C_s^{[\hat{\rho}]}(0) = \int d^2\alpha W(\alpha) = 1 \tag{2.28}$$

There exist two equivalent representations of the Wigner function. The first one is constructed in terms of non-diagonal elements of the density operator, the second one is expressed in terms of the parity operator $\hat{P} = (-1)^{\hat{n}}$ introduced previously. From the first representation one can see that the knowledge of $W(\alpha)$ is linked to the coherence properties of the state and that it is equivalent to the knowledge of $\hat{\rho}$. This in turn, leads to the fact that we can compute the expectation value of any field observable from $W(\alpha)$. The representation in terms of $\hat{P}$ has the following form:

$$W(\alpha) = \frac{2}{\pi} \mathrm{Tr}\left\{\hat{D}(-\alpha)\hat{\rho}\hat{D}(\alpha)\hat{P}\right\} = \frac{2}{\pi}\langle\hat{P}\rangle \tag{2.29}$$

since $W(\alpha)$ is the expectation value of an observable quantity like $\hat{P}$, it is directly measurable. Furthermore, the parity operator eigenvalues (which are ± 1) set the upper and the lower bound for $W(\alpha)$:

$$-\frac{2}{\pi} \leq W(\alpha) \leq \frac{2}{\pi} \tag{2.30}$$

The integration of $W(\alpha)$ over one quadrature yields the probability distribution of the orthogonal quadrature. This provides the marginal probability distribution of field quadratures and by writing $\alpha = \alpha' + i\alpha'' = q + ip$ we find:

$$\begin{aligned} P(q) &= \int d\alpha'' W(\alpha) \\ P(p) &= \int d\alpha' W(\alpha) \end{aligned} \tag{2.31}$$

The next two examples show how the Wigner distribution looks like for quasi classical states and non-classical states.

Example 1: Coherent states Let us consider a coherent state $|\beta\rangle$ for which the characteristic function is:

$$C_s^{[|\beta\rangle\langle\beta|]}(\lambda) = \langle\beta|\,\hat{D}(\lambda)\,|\beta\rangle = e^{-\frac{|\lambda|^2}{2}} e^{\beta^*\lambda - \lambda^*\beta} \tag{2.32}$$

the corresponding Wigner function will be a gaussian of width $\frac{1}{2}$ centered at the classical amplitude β:

$$W(\alpha)^{[|\beta\rangle\langle\beta|]} = \frac{2}{\pi} e^{-2|\alpha-\beta|^2} \tag{2.33}$$

The coherent state represents the classical behavior of the harmonic oscillator and unsurprisingly the Wigner function shows properties of a classical probability distribution, which is always positive as shown in the Fig(2.7):

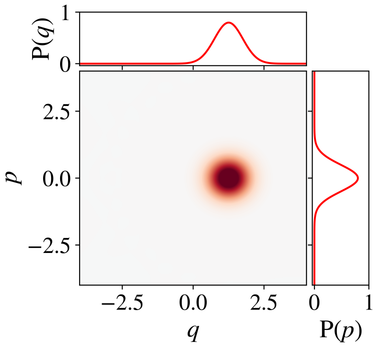

Figure 2.7: Wigner function of a coherent state $|\beta\rangle$. The marginal probability distributions for field quadratures are plotted on the sides.

Example 2: Phase cat states The phase cat state has no classical analog so we expect the Wigner function to behave in a different way. For a pure cat state $\left|\psi_{cat}^{\pm}\right\rangle$ the Wigner function is:

$$\begin{aligned} W(\alpha)^{[cat,\pm]} = \frac{1}{\pi^2 N_\pm} \int d^2\lambda e^{\alpha\lambda^* - \lambda^*\alpha} (\langle\beta|\, \hat{D}(\lambda)\, |\beta\rangle + \langle-\beta|\, \hat{D}(\lambda)\, |-\beta\rangle \\ \pm \langle\beta|\, \hat{D}(\lambda)\, |-\beta\rangle \pm \langle-\beta|\, \hat{D}(\lambda)\, |\beta\rangle) \end{aligned} \tag{2.34}$$

assuming $\beta \in \mathbb{R}$ and $\alpha = \alpha' + i\alpha''$ we get a Wigner function that takes both positive and negative values:

$$W(\alpha)^{[cat,\pm]} = \frac{4}{\pi N_\pm} (e^{-2|\alpha-\beta|^2} + e^{-2|\alpha+\beta|^2} \pm 2e^{-2|\alpha|} \cos 4\alpha''\beta) \tag{2.35}$$

Such a $W(\alpha)$ cannot represent a classical probability distribution and is associated with non classical states. The plot of $W(\alpha)^{[cat,+]}$ i.e. the even parity cat state for $N_+ \approx \frac{1}{\sqrt{2}}$ is shown in the Fig.(2.8)

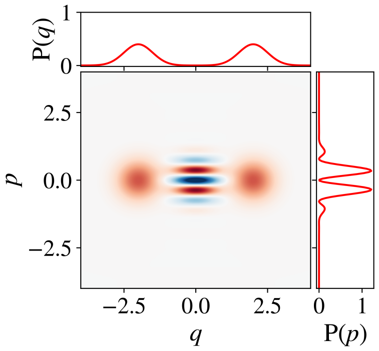

Figure 2.8: Wigner function of an even pure cat state $\left|\psi_{cat}^{+}\right\rangle$. The central part of the plot represents the quantum interference, with red standing for positive values of $W(\alpha)$ and blue for negative ones.

Notice however that the negativity of the Wigner function alone is not sufficient to tell whether we are dealing with a purely quantum state. As a matter of fact, squeezed states are genuine quantum states with no classical analog but their Wigner function is always ≥ 0.

2.2 Trapped ions

All the concepts introduced in the previous section have to be realized by some physical system. The states of harmonic oscillator that we want to explore can be implemented in trapped ion. In this section we want to give a closer look at the techniques used to realized the states we are interested in.

The system that we want to consider consists of a two level system built on the quadrupole transition between Zeeman sub-levels $|\downarrow\rangle = |S_{1/2}, m_j = 1/2\rangle$ and $|\uparrow\rangle = |D_{5/2}, m_j = 3/2\rangle$ of the ${}^{40}Ca^+$ ion coupled to the axial mode of the quantized harmonic motion of the trapping potential through a laser field (see Fig. 2.10):

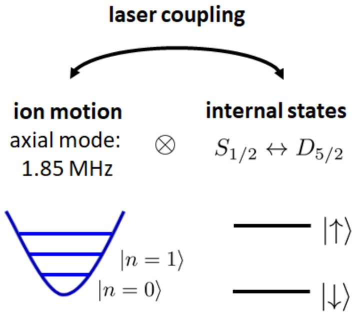

Figure 2.9: Schematic representation of a trapped ion system.

Such a system can be described by the Hamiltonian [9]:

$$\begin{aligned}\hat{H}_I &= -\frac{1}{2} i\hbar\Omega_0 e^{-i(\omega_L t+\varphi)} (\hat{\sigma}_+ e^{i\omega_0 t}) e^{i\eta(\hat{a}e^{-i\omega_z t}+\hat{a}^\dagger e^{i\omega_z t})} + H.c \\ &= -\frac{1}{2} i\hbar\Omega_0 e^{i((\omega_0-\omega_L)t-\varphi)} \hat{\sigma}_+ e^{i\eta(\hat{a}e^{-i\omega_z t}+\hat{a}^\dagger e^{i\omega_z t})} + H.c\end{aligned} \tag{2.36}$$

which is the interaction Hamiltonian with respect to:

$$\hat{H}_0 = \hbar\omega_z(\hat{a}^\dagger\hat{a} + \frac{1}{2}) + \hbar\omega_0 \frac{\hat{\sigma}_z}{2} \tag{2.37}$$

in (2.36) the phase between the laser and the atomic polarization is represented by φ and ω_0 is the transition frequency of the two level system.

$\hat{\sigma}_+ = |\uparrow\rangle\langle\downarrow|$ and $\hat{\sigma}_- = |\downarrow\rangle\langle\uparrow|$ are the rising or lowering operator acting on the internal levels of the state whereas $\hat{a}$ and $\hat{a}^\dagger$ are the annihilation and creation operators of the motional part of the state (to not be confused with the annihilation and creation operators of the photon field). The Rabi frequency Ω_0 incorporates the laser field intensity (represented by a coherent state) and and can be thought as the frequency at which we exchange the population in the atomic levels. We also define the Lamb-Dicke parameter η ,coming from the momentum recoil of the laser field, as:

$$\eta = \underbrace{\frac{2\pi}{\lambda}\cos\theta}_{k_z}\underbrace{\sqrt{\frac{\hbar}{2m\omega_z}}}_{z_0} \tag{2.38}$$

where k_z is the projection of the wave vector of the laser field in the direction of the axial mode and z_0 is the extension of the ion's wave function in the motional ground state. Next, we make an approximation and expand the hamiltonian in (2.37) to the 1^{st} order in η. Such approximation is called Lamb-Dicke regime and it is valid if $\eta \ll 1$ and $\eta^2(2n+1) \ll 1$ are both satysfied. The first condition is satysfied if proper experimental parameters are chosen. The second condition can be fulfilled if the system has a sufficiently small phonon occupancy which can be achieved by cooling down the system first with doppler cooling then with sideband cooling. The resulting interaction hamiltonian reads:

$$\hat{H}_I \approx -\frac{1}{2}i\hbar\Omega_0 e^{i((\omega_0-\omega_L)t-\varphi)}\hat{\sigma}_+(\mathbb{1} + i\eta(\hat{a}e^{-i\omega_z t} + \hat{a}^\dagger e^{i\omega_z t})) + H.c \tag{2.39}$$

Let us now examine the different transitions that we may drive with our laser.

Carrier transition :
In the following discussion we set the laser frequency to match the atomic transition frequency i.e. $\omega_L = \omega_0$. The interaction hamiltonian has the following form:

$$\hat{H}_{I,carrier} \approx -\frac{1}{2}i\hbar\Omega_0 e^{-i\varphi}\hat{\sigma}_+(\mathbb{1} + \underbrace{i\eta(\hat{a}e^{-i\omega_z t} + \hat{a}^\dagger e^{i\omega_z t})}_{=0}) + H.c \tag{2.40}$$

here we performed the rotating wave approximation (RWA) and dropped the off-resonant terms. Thus, $\hat{H}_{I,carrier}$ acts on the Hilbert space of the two level system with $\hat{\sigma}_\pm$ and on the Hilbert space of the phonons with the identity $\mathbb{1}$ driving resonant transitions at the rabi frequency Ω_0:

$$\hat{H}_{I,carrier} \approx -i\frac{1}{2}\hbar\Omega_0(e^{-i\varphi}\hat{\sigma}_+ \otimes \mathbb{1} - e^{i\varphi}\hat{\sigma}_- \otimes \mathbb{1}) \tag{2.41}$$

Red sideband transition :
We set the laser frequency to $\omega_L = \omega_0 - \omega_z$ to get:

$$\begin{aligned}\hat{H}_{I,RSB} &\approx -\frac{1}{2}i\hbar\Omega_0 e^{i(\omega_z t-\varphi)}\hat{\sigma}_+(\mathbb{1} + i\eta(\hat{a}e^{-i\omega_z t} + \hat{a}^\dagger e^{i\omega_z t})) + H.c \\ &= -\frac{1}{2}i\hbar\Omega_0 e^{i(\omega_z t-\varphi)}\hat{\sigma}_+ \otimes \mathbb{1} + \frac{1}{2}\hbar\Omega_0\eta e^{-i\varphi}\hat{\sigma}_+ \otimes \hat{a} \\ &+ \frac{1}{2}\hbar\Omega_0\eta e^{i2\omega_z t}\hat{\sigma}_+ \otimes \hat{a}^\dagger + H.c\end{aligned} \tag{2.42}$$

we again apply the RWA and drop the off resonant terms obtaining:

$$\hat{H}_{I,RSB} \approx \frac{1}{2}\hbar\Omega_0\eta(\hat{\sigma}_+ \otimes \hat{a}e^{-i\varphi} + \hat{\sigma}_- \otimes \hat{a}^\dagger e^{i\varphi}) \tag{2.43}$$

for $\varphi = 0$ this expression corresponds to the Janes-Cummings model but the role of the electromagnetic field is now played by the phonons.

Blue sideband transition :
With similar arguments the laser pulse is $\omega_L = \omega_0 + \omega_z$, this leads to:

$$\begin{aligned}\hat{H}_{I,BSB} &\approx -\frac{1}{2}i\hbar\Omega_0 e^{i(-\omega_z t-\varphi)}\hat{\sigma}_+(\mathbb{1} + i\eta(\hat{a}e^{-i\omega_z t} + \hat{a}^\dagger e^{i\omega_z t})) + H.c \\ &= -\frac{1}{2}i\hbar\Omega_0 e^{i(-\omega_z t-\varphi)}\hat{\sigma}_+ \otimes \mathbb{1} + \frac{1}{2}\hbar\Omega_0\eta e^{-i(\varphi+2\omega_z)}\hat{\sigma}_+ \otimes \hat{a} \\ &+ \frac{1}{2}\hbar\Omega_0\eta e^{-i\varphi}\hat{\sigma}_+ \otimes \hat{a}^\dagger + H.c\end{aligned} \tag{2.44}$$

by applying the RWA and dropping the off-resonant terms we get:

$$\hat{H}_{I,BSB} \approx \frac{1}{2}\hbar\Omega_0\eta(\hat{\sigma}_+ \otimes \hat{a}^\dagger e^{-i\varphi} + \hat{\sigma}_- \otimes \hat{a}^\dagger e^{i\varphi}) \tag{2.45}$$

(2.46) can be regarded as anti-Janes-Cummings model provided that $\varphi = 0$.

As a last remark we point out that the eigenstates of the uncoupled system described by (2.37) are the dressed states $|\downarrow, n\rangle$ and $|\uparrow, n\rangle$. Those states are coupled by (2.39) with different strengths depending on the laser frequency ω_0. The figure below shows schematically how the dressed states ladder gets coupled through the laser field:

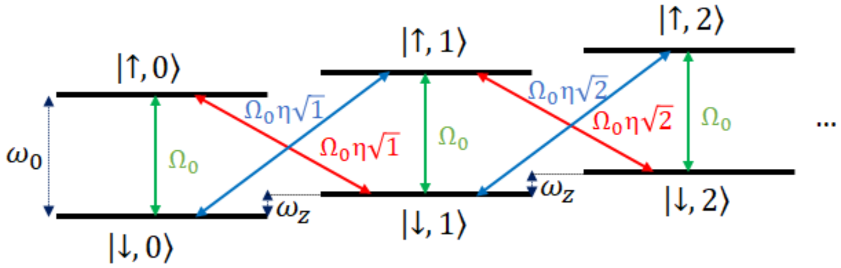
$|\uparrow,0\rangle$
$|\uparrow,1\rangle$
$|\uparrow,2\rangle$
$|\downarrow,0\rangle$
$|\downarrow,1\rangle$
$|\downarrow,2\rangle$
ω_0
Ω_0
$\Omega_0\eta\sqrt{1}$
$\Omega_0\eta\sqrt{1}$
Ω_0
$\Omega_0\eta\sqrt{2}$
$\Omega_0\eta\sqrt{2}$
Ω_0
ω_z
ω_z
...

Figure 2.10: Ladder of dressed states with laser couplings marked with different colours. Green represents the carrier, red/blue stands for red/blue sideband respectively.

the action of the interaction hamiltonians in (2.41),(2.43) and (2.46) on dressed states provides us with the coupling strengths:

$$\begin{aligned}\hat{H}_{I,carrier}\left|\downarrow,n\right\rangle &= \frac{1}{2}\hbar\Omega e^{-i\varphi}\left|\uparrow,n\right\rangle \\ \hat{H}_{I,RSB}\left|\downarrow,n\right\rangle &= \frac{1}{2}\hbar\Omega\eta\sqrt{n}e^{-i\varphi}\left|\uparrow,n-1\right\rangle \\ \hat{H}_{I,BSB}\left|\downarrow,n\right\rangle &= \frac{1}{2}\hbar\Omega\eta\sqrt{n+1}e^{-i\varphi}\left|\uparrow,n+1\right\rangle\end{aligned} \tag{2.46}$$

this explains the Fig.2.10 and leads to the conclusion that the carrier pulse $\hat{H}_{carrier}$ couples the states $|\downarrow,n\rangle$ and $|\uparrow,n\rangle$ with the rabi frequency Ω_0. The red sideband pulse couples $|\downarrow,n\rangle$ and $|\uparrow,n-1\rangle$ with an effective rabi frequency $\Omega_{eff,R} = \Omega_0\eta\sqrt{n}$ and the blue sideband pulse couples $|\downarrow,n\rangle$ and $|\uparrow,n+1\rangle$ states with $\Omega_{eff,R} = \Omega_0\eta\sqrt{n+1}$. Since we are in the Lamb-Dicke regime ($\eta^2(2n+1) \ll 1$) the sideband transitions are much slower than the carrier and they are sensitive to the occupancy of the motional states.

Now we are going to explain how cat states, introduced previously, can be created using the trapped ions.

2.2.1 State dependent force

If we combine the interaction hamiltonians of the blue and red sidebands through a bichromatic laser pulse, we get a total interaction hamiltonian of the form:

$$\begin{aligned}\hat{H}_{I,SDF} &= \hat{H}_{I,RSB} + \hat{H}_{I,BSB}\\ &\approx \frac{\hbar\Omega_B\eta}{2}(\hat{\sigma}_+\hat{a}^\dagger e^{-i(\delta t+\varphi_B)} + \hat{\sigma}_-\hat{a}e^{i(\delta t+\varphi_B)})\\ &+ \frac{\hbar\Omega_R\eta}{2}(\hat{\sigma}_+\hat{a}e^{i(\delta t-\varphi_R)} + \hat{\sigma}_-\hat{a}^\dagger e^{-i(\delta t-\varphi_R)})\end{aligned} \tag{2.47}$$

such hamiltonian in general contains a detuning of the laser frequency from the sidebands δ which in an ideal resonance situation can be set to zero. For the simulation purposes we will consider $\delta = 0$ getting rid of the time dependence, in an experiment though this term plays a role. If $\Omega_B = \Omega_R = \Omega$ and $\Delta\varphi = \frac{\varphi_R-\varphi_B}{2}$ and $\bar{\varphi} = \frac{\varphi_R+\varphi_B}{2}$ the hamiltonian in (2.47) can be rewritten as:

$$\hat{H}_{I,SDF} = \frac{\hbar\Omega\eta}{2}\hat{\sigma}_{\bar{\varphi}}(\hat{a}e^{i\Delta\varphi} + \hat{a}^\dagger e^{-i\Delta\varphi}) \tag{2.48}$$

where $\hat{\sigma}_{\bar{\varphi}} = \hat{\sigma}_x \cos\bar{\varphi} + \hat{\sigma}_y \sin\bar{\varphi}$. As long as this hamiltonian is time independent (which is the case if the phases and the amplitudes of the sidebands are time independent over the time evolution), the time propagator is given by:

$$\hat{U}(t) = e^{-i\frac{\hat{H}_{I,SDF}t}{\hbar}} = e^{\hat{\sigma}_{\bar{\varphi}}(\alpha(t)\hat{a}-\alpha(t)^*\hat{a}^\dagger)} = \hat{\mathcal{D}}(\alpha(t)\hat{\sigma}_{\bar{\varphi}}) \tag{2.49}$$

where $\alpha(t) = -i\frac{\Omega\eta t}{2}e^{i\Delta\varphi}$. The displacement operator in (2.49) can be interpreted as a state dependent force (SDF) which displaces the motional part of the state according to the sign set by the $\hat{\sigma}_{\bar{\varphi}}$ eigenstate of the two level system, let us see how it acts on a state prepared in $|{\downarrow},0\rangle$ when $\Delta\varphi = \bar{\varphi} = 0$:

$$\begin{aligned}&\hat{\mathcal{D}}(\alpha(t)\hat{\sigma}_x)\,|{\downarrow},0\rangle = \exp\Big\{\hat{\sigma}_x(\alpha(t)\hat{a} - \alpha(t)^*\hat{a}^\dagger)\Big\}\frac{|+\rangle + |-\rangle}{\sqrt{2}}\,|0\rangle\\ &= \frac{1}{\sqrt{2}}(\hat{\mathcal{D}}(\alpha(t))\,|+\rangle\,|0\rangle + \hat{\mathcal{D}}(-\alpha(t))\,|-\rangle\,|0\rangle)\\ &= \frac{1}{4}(\underbrace{|{\downarrow}\rangle}_{bright}\underbrace{(|\alpha(t)\rangle + |-\alpha(t)\rangle)}_{even} + \underbrace{|{\uparrow}\rangle}_{dark}\underbrace{(|\alpha(t)\rangle - |-\alpha(t)\rangle)}_{odd})\end{aligned} \tag{2.50}$$

thus if $\alpha(t)$ is sufficiently big, upon measurement we collapse the state into an even parity cat state or an odd parity cat state. However, the measurement itself in a trapped ion system is a huge problem. As already mentioned in the introduction, if the system is detected in the ground state, the momentum of several (around 1000) photons scattered in the process of the fluorescence detection randomize the motional part of the state. This obligates us to work only with odd parity cats. It also means that we can only create odd cats probabilistically by post selecting the dark counts from the fluorescence detection. Thus, the dark state detection is not invasive and one can analyze the motional degree of freedom further and apply another SDF for example.

Notice though that after the application of sequential displacement operators the state acquires a geometric phase independent of the motional state proportional to the area spanned by the trajectories of the displacement in the phase space [10]. For 2 sqeuential displacements we get:

$$\hat{\mathcal{D}}(\alpha)\hat{\mathcal{D}}(\beta) = \hat{\mathcal{D}}(\alpha+\beta)\exp\{i\,\mathrm{Im}\{\alpha\beta^*\}\} \tag{2.51}$$

the geometric phase acquired by the state is $\Phi = \mathrm{Im}\{\alpha\beta^*\}$ and constitutes another degree of freedom when creating states in phase space which involve more than one displacement.

Now we will give an example in which the SDF hamiltonian is used to create the grid state.

Grid state :
In order to get the logical zero grid state, reported in the introduction, we need to start with the phononic part of the state in the sqezed vacuum. Such a state can be achieved with high fidelity by using the reservoir engineering [11]. The initial state can be represented by $|\psi\rangle_{init} = \mathbb{1}\otimes\hat{S}(r)\,|\downarrow,0\rangle$, where the sqeueezing operator is $\hat{S}(r) = \exp\{r(\hat{a}^2 - \hat{a}^{\dagger 2})/2\}$ with the squeezing parameter r being real and ≈ 0.9. The Wigner function of the motional part is shown in the figure below:

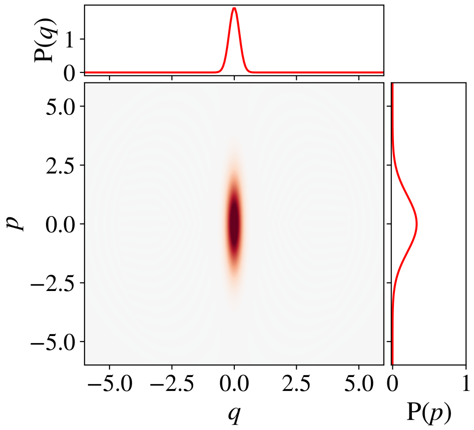

Figure 2.11: Wigner function of the squeezed vacuum state.

Then, we apply the SFD with $\bar{\varphi} = \Delta\varphi = 0$, $\alpha = l = \sqrt{2\pi}/2$ (parameters used in the simulation) to $|\psi\rangle_{init}$. By post selecting the dark counts in the measurement we get, with probability 1/2 the state:

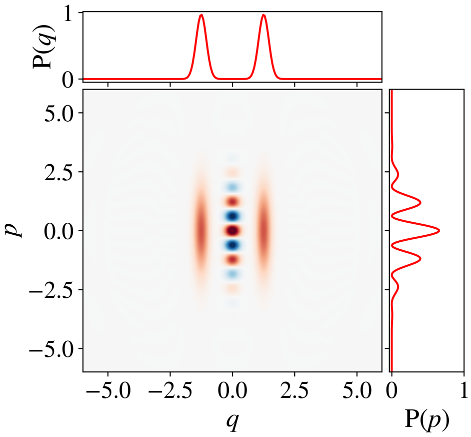

Figure 2.12: Even superposition of sqeezed states obtained after the application of the SDF and post-selection of dark counts.

to get the smallest possible grid state we have to apply the operations above again to the state we obtained and post select again the dark counts to get, with probability 1/4, the state whose Wigner function is shown in the figure below:

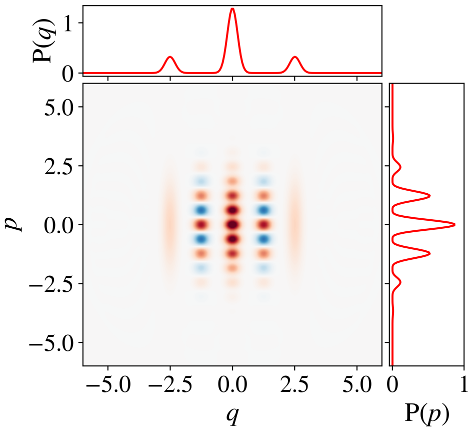

Figure 2.13: Wigner function of a grid state.

Thus, we see that experimentally the problem lies in the fact that the grid state can be created only probabilistically by post selecting the data from many measurements. In the next chapter, we try to solve this issue by implementing a gradient ascent based algorithm in order to find an optimal sequence of sideband pulses which produce the grid state without post selection of data.

Chapter 3

Simulation

In this chapter the control technique known as Gradient Ascent Pulse Engineering (GRAPE) and its implementation are presented. In the first part we analyze the algorithm and explain how to properly choose the parameters for the simulation of the state coherent transfer. Next, we consider the experimental constraints to be taken into account in order to produce the target state in an experimentally affordable manner. The last section shows the results of the simulation for diverse harmonic oscillator states.

3.1 GRAPE

Gradient ascent based algorithms are a standard tool in numerical optimization problems. The main idea behind such algorithms exploits the fact that the gradient of any function is a vector pointing in the direction of maximal increase, such information is then used to update the parameters of interest. The problem can be translated into a maximization of an objective function $f(\mathbf{x})$ which depends on certain parameters $\mathbf{x} \in \mathbb{R}^n$. In the unconstrained case, the aim is to solve:

$$\max_{\mathbf{x}} f(\mathbf{x}) \tag{3.1}$$

in other words, we want to find the parameters $\mathbf{x}$ that maximize $f(\mathbf{x})$. A numerical solution to (3.1) could be found by calculating $\nabla f(\mathbf{x})$ and use the latter to update the parameters as follows:

$$\mathbf{x}_{new} = \mathbf{x}_{old} + \varepsilon \nabla f(\mathbf{x}) \tag{3.2}$$

where ε is chosen such that the incremental change of $\mathbf{x}_{old}$ is small. The procedure results to be computationally efficient if there exist an efficient method to evaluate the gradient.

3.1.1 coherent state transfer

The case for which we want to employee the gradient ascent algorithm is called coherent state transfer. The idea to use this control technique in a quantum system was originally formalized and proposed by N.Khaneja et.al.[4], where a series of optimized pulses was used in the context of NMR spectroscopy. Our goal is to reach a desired quantum state $\hat{\rho}_{target}$, by applying opportune laser pulses to our trapped ion system prepared in an initial state $\hat{\rho}_0$ at time $T = 0$.

To start the discussion, we need a formal way of measuring the distance between the states represented by a density matrix $\hat{\rho}(T)$, at any given time T, and the target state density matrix $\hat{\rho}_{target}$. Such a distance ,which we will denote with $\mathcal{F}(\cdot)$, is the trace distance between two states:

$$\mathcal{F}(\hat{\rho}_{target}, \hat{\rho}(T)) = \mathrm{Tr}\left\{\hat{\rho}^{\dagger}_{target}\hat{\rho}(T)\right\} \tag{3.3}$$

Let us lie down few properties of (3.3) which will be important in the implementation of the algorithm [5]. The Frobenius inner product, defined in (3.3) can be re-written for two complex $n \times n$ matrices X and Y as:

$$\langle X|Y\rangle_F = \mathrm{Tr}\left\{X^{\dagger}Y\right\} = \sum_i \sum_j x^*_{ji} y_{ij} \tag{3.4}$$

and it naturally induces a norm on X:

$$\|X\|_F = \sqrt{\langle X|X\rangle} = \sqrt{\sum_i \sum_j |x_{ij}|^2} \tag{3.5}$$

The main properties of such norm are:

$$\begin{gathered} \langle X|Y\rangle_F = \overline{\langle Y|X\rangle_F} \\ \left\langle X^{\dagger}\middle|Y\right\rangle_F = \left\langle Y^{\dagger}\middle|X\right\rangle_F \\ X = X^{\dagger}, Y = Y^{\dagger} \implies \langle X|Y\rangle_F \in \mathbb{R} \end{gathered} \tag{3.6}$$

The last property in (3.6) is very important since it implies that if X and Y are square hermitian matrices their overlap according to the Frobenius inner product yields a real number. At this point it is reasonable to pick as a figure of merit for the optimization problem, the distance between X and Y:

$$\Phi = \|X - Y\|^2_F \tag{3.7}$$

Now we want to show that minimizing $\Phi = \|X - Y\|^2_F$ is equivalent to maximizing $\Phi_0 = \mathrm{Re}\{\langle X|Y\rangle_F\}$. Exploiting the linearity of the inner product we write:

$$\begin{aligned}
\|X - Y\|_F^2 &= \langle X - Y | X - Y \rangle_F = \langle X - Y | X \rangle_F - \langle X - Y | Y \rangle_F \\
&= \langle X | X \rangle_F - \langle X | Y \rangle_F - \langle Y | X \rangle_F + \langle Y | Y \rangle_F \\
&= \|X\|_F^2 - 2\,\mathrm{Re}\{\langle X | Y \rangle_F\} + \|Y\|_F^2
\end{aligned} \tag{3.8}$$

with $\|X\|_F^2$ and $\|Y\|_F^2$ being positive by definition, the problem can be turned into the maximization of $\Phi_0 = \mathrm{Re}\{\langle X|Y\rangle_F\}$ as stated previously. Furhermore, in our case we are dealing with density matrices which are hermitian, such that by the last property in (3.6) we find $\Phi_0 = \mathrm{Re}\{\langle X|Y\rangle_F\} = \langle X|Y\rangle_F$ and thus we want to maximize:

$$\Phi_0 = \langle X|Y\rangle_F \tag{3.9}$$

With this being said, let us focus on how to steer the initial state $\hat{\rho}_0$ into the target state $\hat{\rho}_{target}$ over time. In absence of relaxation and in the Schrödinger picture, the time evolution of a density matrix is given by the Liouville-Von Neumann equation:

$$\frac{\partial \hat{\rho}(t)}{\partial t} = \frac{1}{i\hbar}\left[\hat{H}, \hat{\rho}(t)\right] \tag{3.10}$$

The exact solution of such an equation is not easy to evaluate in the case of time dependent hamiltonians. To solve the issue more easily we suppose that we can write the time dependent hamiltonian in the following way:

$$\hat{H} = \hat{H}_0 + \sum_{k=1}^{m} u_k(t)\hat{H}_k \tag{3.11}$$

where $\hat{H}_0$ is the bare hamiltonian and $\hat{H}_k$ are the control hamiltonians. The set of $u_k(t)$ represent the control amplitudes which we are going to vary in order to solve:

$$\max_{u_k(t)} \mathcal{F}(\hat{\rho}_{target}, \hat{\rho}(T)) \tag{3.12}$$

Further, we assume that the transfer time T is discretized in N equal steps such that a small time step $\Delta t = \frac{T}{N}$ is defined. Another assumption we make is that all the control amplitudes $u_k(t)$ are constant over the time interval Δt for each time step j. Now we are allowed to treat the total hamiltonian of the system in (3.11) as time independent and the time evolution during the time step j is given by the propagator:

$$\hat{U}_j = \exp\left\{-i\Delta t(\hat{H}_0 + \sum_{k=1}^{m} u_k(j)\hat{H}_k)\right\} \tag{3.13}$$

Thus, after the transfer time T the density operator $\hat{\rho}_0$ will read:

$$\hat{\rho}(T) = \hat{U}_N \cdots \hat{U}_1 \hat{\rho}_0 \hat{U}_1^\dagger \cdots \hat{U}_N^\dagger \tag{3.14}$$

by substituting the equation above into the definition of (3.3) we get:

$$\begin{aligned}\Phi_0 &= \mathcal{F}(\hat{\rho}_{target}, \hat{\rho}(T)) = \mathrm{Tr}\Big\{\hat{\rho}^{\dagger}_{target}\hat{\rho}(T)\Big\} \\ &= \mathrm{Tr}\Big\{\hat{\rho}^{\dagger}_{target}\hat{U}_N \cdots \hat{U}_1\hat{\rho}_0\hat{U}_1^{\dagger} \cdots \hat{U}_N^{\dagger}\Big\}\end{aligned} \tag{3.15}$$

in addition to that, we use the trace invariance under cyclic permutations in order to re-write Φ_0 in a convenient way:

$$\Phi_0 = \langle \underbrace{\hat{U}^{\dagger}_{j+1} \cdots \hat{U}^{\dagger}_N \hat{\rho}_{target}\hat{U}_N \cdots \hat{U}_{j+1}}_{\hat{\lambda}_j} \,|\, \underbrace{\hat{U}_j \cdots \hat{U}_1\hat{\rho}_0\hat{U}_1^{\dagger} \cdots \hat{U}_j^{\dagger}}_{\hat{\rho}_j} \rangle_F \tag{3.16}$$

we can interpret $\hat{\rho}_j$ as the system density matrix propagated in time from $t = 0$ to $t = j\Delta t$ and $\hat{\lambda}_j$ as the target state density matrix back-propagated from $t = T$ to $t = j\Delta t$.

Let us now explore the crucial part of the algorithm, the gradient calculation. Consider the small perturbation of the kth control amplitude at time step j:

$$u_k(j) \longrightarrow u_k(j) + \delta u_k(j) \tag{3.17}$$

such variation provokes a change in $\hat{U}_j$. Noticing that in general, when taking derivatives of exponentials, the following formula holds [6]:

$$\frac{\partial e^{z(\lambda)}}{\partial \lambda} = \int_0^1 d\alpha e^{(1-\alpha)z}\frac{\partial z}{\partial \lambda}e^{\alpha z} \tag{3.18}$$

where z is an analytic function and $\lambda \in \mathbb{C}$. Extending this definition to complex $n \times n$ matrices and assuming perturbations up to the first order in $\delta u_k(j)$ and $\left\|\hat{H}_0 + \sum_{k=1}^{m} u_k(j)\hat{H}_k\right\|\Delta t \ll 1$ we find:

$$\frac{\partial \hat{U}_j}{\partial u_k(j)} \approx -i\Delta t\hat{H}_k\hat{U}_j + \mathcal{O}(\Delta t^2) \tag{3.19}$$

considering the expansion in Δt up to the 1^{st} order. Knowing this we can derive the expression for the gradient of the figure of merit Φ_0 [5]:

$$\begin{aligned}
\frac{\partial \Phi_0}{\partial u_k(j)} &= \frac{\partial}{\partial u_k(j)} \langle \underbrace{\hat{U}^\dagger_{j+1} \cdots \hat{U}^\dagger_N \hat{\rho}_{target} \hat{U}_N \cdots \hat{U}_{j+1}}_{\hat{\lambda}_j} \,|\, \underbrace{\hat{U}_j \cdots \hat{U}_1 \hat{\rho}_0 \hat{U}^\dagger_1 \cdots \hat{U}^\dagger_j}_{\hat{\rho}_j} \rangle_F \\
&= \mathrm{Tr}\left\{ \hat{\lambda}^\dagger_j \frac{\partial \hat{\rho}_j(u_k(j))}{\partial u_k(j)} \right\} \\
&= \mathrm{Tr}\left\{ \hat{\lambda}^\dagger_j (\hat{U}_j \cdots \hat{U}_1 \hat{\rho}_0 \hat{U}^\dagger_1 \cdots \frac{\partial \hat{U}^\dagger_j}{\partial u_k(j)} + \frac{\partial \hat{U}_j}{\partial u_k(j)} \cdots \hat{U}_1 \hat{\rho}_0 \hat{U}^\dagger_1 \cdots \hat{U}^\dagger_j) \right\} \\
&\approx \mathrm{Tr}\left\{ \hat{\lambda}^\dagger_j i \Delta t (\hat{U}_j \cdots \hat{U}_1 \hat{\rho}_0 \hat{U}^\dagger_1 \cdots \hat{U}^\dagger_j \hat{H}_k - \hat{H}_k \hat{U}_j \cdots \hat{U}_1 \hat{\rho}_0 \hat{U}^\dagger_1 \cdots \hat{U}^\dagger_j) \right\} \\
&= \mathrm{Tr}\left\{ \hat{\lambda}^\dagger_j i \Delta t (\hat{\rho}_j \hat{H}_k - \hat{H}_k \hat{\rho}_j) \right\} \\
&= -\left\langle \hat{\lambda}_j \middle| i \Delta t [\hat{H}_k, \hat{\rho}_j] \right\rangle_F
\end{aligned} \tag{3.20}$$

where the approximation made in the 4^{th} line results from (3.19). If we update the control amplitudes according to:

$$u_k(j) \longrightarrow u_k(j) + \varepsilon \frac{\partial \Phi_0}{\partial u_k(j)} \tag{3.21}$$

we will observe an increase in the objective function Φ_0 and solve approximately the problem defined in (3.12). The choice of an appropriate ε will be explained in the last section, for now ε is just a constant such that the change in $u_k(j)$ is small.

3.1.2 Implementation

The main steps necessary to implement the GRAPE algorithm can be summarized in five steps:

1. Guess initial control amplitudes $u_k(j)$.
2. Start from $\hat{\rho}_0$ and compute $\hat{\rho}_j = \hat{U}_j \cdots \hat{U}_1 \hat{\rho}_0 \hat{U}^\dagger_1 \cdots \hat{U}^\dagger_j$ for all $j \leq N$.
3. Start from $\hat{\rho}_{target}$ and compute $\hat{\lambda}_j = \hat{U}^\dagger_{j+1} \cdots \hat{U}^\dagger_N \hat{\rho}_{target} \hat{U}_N \cdots \hat{U}_{j+1}$ for all $j \leq N$.
4. Compute the approximate gradient $\frac{\partial \Phi_0}{\partial u_k(j)}$ according to (3.20) and update all the $m \times N$ control amplitudes with the rule defined in (3.21).
5. Take the new $m \times N$ control amplitudes and repeat the algorithm from point 2. until the convergence.

Notice that in order to terminate the algorithm a threshold to Φ_0 must be set. Moreover, in the first step, a well educated guess for the initial control amplitudes ease the convergence. Another aspect concerning the implementation is the fact that if we want to use the algorithm with $u_k(j) \in \mathbb{R}$, we have to make sure that the gradient is real valued. The comparison between the last property of the Frobenius inner product in (3.6) and the approximate gradient of Φ_0 in (3.20) leads to the conclusion that the $\frac{\partial \Phi_0}{\partial u_k(j)} \in \mathbb{R}$ if $\hat{H}_k^\dagger = \hat{H}_k$. In other words, we need to write down the hamiltonian in such a way that all individual control hamiltonians $\hat{H}_k$ are hermitian operators.

In this work GRAPE was implemented in Python with QuTiP, a quantum toolbox used for simulating the dynamics of open quantum systems. The code of the algorithm is provided in the Appendix A.

3.2 Experimental constraints

In this section we want to analyze the main experimental constraints and include them in the algorithm in the form of penalty terms.

To see why certain bounds are needed, let us first consider the unconstrained case and see how the algorithm performs in absence of experimental bounds. For simplicity of treatment we choose the cat state, introduced in the previous chapter, as the target state. The initial state is a vacuum state with phonon occupancy set to N=50, such that the Hilbert space of the motional part of the state is big enough and when applying the state dependent force we do not encounter the "hard wall" set by the Lamb-Dicke approximation. The state dependent force hamiltonian, which governs the time evolution, is written in the form:

$$\begin{aligned} \hat{H}_{SDF} =& g_R(\hat{\sigma}_+ \hat{a} e^{-i\varphi_R} + \hat{\sigma}_- \hat{a}^\dagger e^{i\varphi_R}) + g_B(\hat{\sigma}_- \hat{a} e^{i\varphi_B} + \hat{\sigma}_+ \hat{a}^\dagger e^{-i\varphi_B}) \\ =& g_R \cos\varphi_R \underbrace{(\hat{\sigma}_- \hat{a}^\dagger + \hat{\sigma}_+ \hat{a})}_{\hat{H}_1} + g_R \sin\varphi_R \underbrace{[i(\hat{\sigma}_- \hat{a}^\dagger - \hat{\sigma}_+ \hat{a})]}_{\hat{H}_2} \\ &+ g_B \cos\varphi_B \underbrace{(\hat{\sigma}_- \hat{a} + \hat{\sigma}_+ \hat{a}^\dagger)}_{\hat{H}_3} + g_B \sin\varphi_B \underbrace{[i(\hat{\sigma}_- \hat{a} - \hat{\sigma}_+ \hat{a}^\dagger)]}_{\hat{H}_4} \end{aligned} \tag{3.22}$$

where $g_{B,R} = \frac{\hbar \Omega_{B,R} \eta}{2}$. The equation above is a simplified expression of $\hat{H}_{SDF}$ where we have already neglected all non resonant terms along with the nearly resonant carrier term. Furthermore, we assumed to be exactly on resonance, setting the detuning δ from the energy levels to zero. The last line in (3.22) shows how $\hat{H}_{SDF}$ may be written such that all its separate terms are hermitian. The control amplitudes for the simulation will be the phases and the amplitudes of the blue and the red side band laser pulses. Since all the pre-factors of each $\hat{H}_k$ appearing in (3.22) are products of two control parameters that we want to update, we divide them into the quadrature Q and in-phase I components:

$$\begin{aligned} I_{B,R} &= g_{B,R} \cos \varphi_{B,R} \\ Q_{B,R} &= g_{B,R} \sin \varphi_{B,R} \end{aligned} \tag{3.23}$$

then we use the rule in (3.21) to update I and Q:

$$\begin{aligned} I'_{B,R} &= I_{B,R} + \alpha \\ Q'_{B,R} &= Q_{B,R} + \beta \end{aligned} \tag{3.24}$$

where $\alpha = \varepsilon \frac{\partial \Phi_0}{\partial I}$ and $\beta = \varepsilon \frac{\partial \Phi_0}{\partial Q}$. At this point, I' and Q' are fixed numbers and we can use the following formula to recover the updated values of $\varphi_{B,R}$ and $g_{B,R}$:

$$\begin{aligned} \varphi'_{B,R} &= \arctan\left(\frac{Q'_{B,R} - \beta}{I'_{B,R} - \alpha}\right) \\ g'_{B,R} &= \sqrt{(Q'_{B,R} - \beta)^2 + (I'_{B,R} - \alpha)^2} \end{aligned} \tag{3.25}$$

The control amplitudes resulting from the unconstrained simulation are shown in the Fig.3.1:

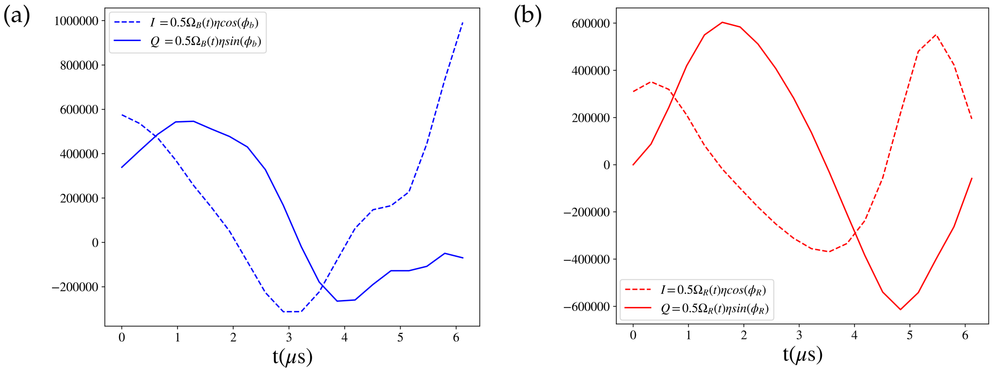

Figure 3.1: In-phase and quadrature components of the control parameters in the unconstrained case: a) blue side band control amplitudes b) red side band control amplitudes.

For this simulation the time axis was divided into 20 time steps of fixed size $\Delta t = 0.322\mu s$. Notice that the parameters ε and Δt are related. Since Δt appears directly in the expression of the gradient in (3.20) and ε multiplies the gradient when updating the control amplitudes, the product $\varepsilon \times \Delta t$ will set by how much $u_k(j)$ will grow (3.21). Thus one has to be careful in the choice of these two parameters, an excessive value of the product causes the overshoots in the control amplitude $u_k(j)$, on the other hand if the product is too small the number of steps needed for the convergence will increase. Furthermore it is very unlikely to get a monotone convergence for ε fixed. Ideally one could solve the problem with the line search algorithm, this method though could be very time consuming and difficult to implement in practice for optimizing Φ_0. In our case the dimension of the parameter space is small enough to allow us to adjust the value by hand. The parameter ε used in the update rule (3.17) was set to:

$$\varepsilon = \begin{cases} 5 \times 10^{10}, & \text{if } \Phi_0 \leq 0.3 \\ 0.6 \times 10^{10}, & \text{if } \Phi_0 > 0.3 \end{cases} \tag{3.26}$$

in this example, such choice guarantees a monotone convergence of the objective Φ_0 (see Fig3.2):

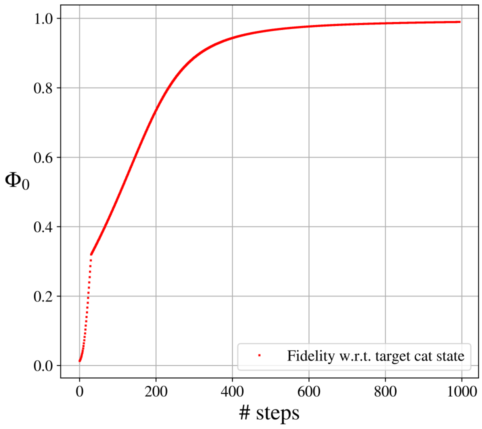

Figure 3.2: Φ_0 vs. number of steps of the algorithm. In this case $\#steps = 995$ and $\Phi_0(\#steps = 995) \approx 0.99$

For the sake of completeness, the Wigner functions of $\hat{\rho}_{target}$ and $\hat{\rho}(T)$ are shown below:

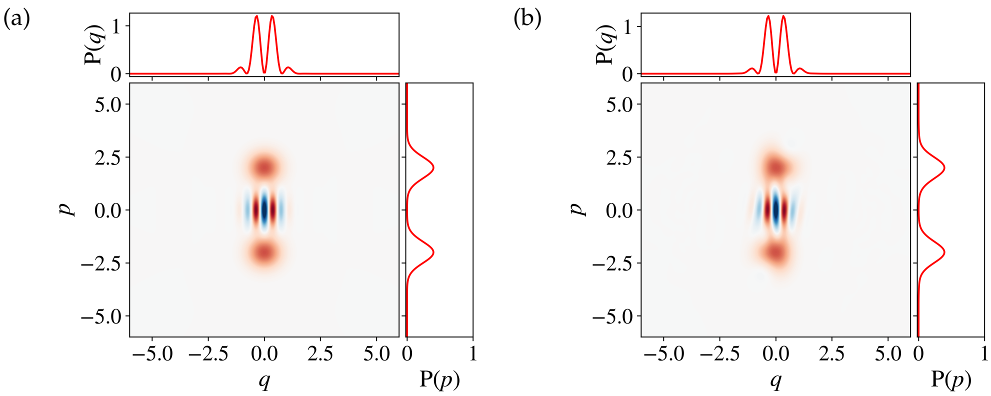

Figure 3.3: Wigner functions of the cat state a)$\hat{\rho}_{target}$ and b)$\hat{\rho}(T)$.

Hence, with this procedure we are able to avoid the projective measurement and create a cat state deterministically. The final remark on this simulation is that it is experimentally unfeasible. The physical limit to the values of in-phase and quadrature components of the side bands is $I_{max} = Q_{max} \approx 310000 a.u.$, by looking at the maximum values reached by the side band pulses in the Fig.3.1 it is clear that the physical limit is exceeded by far. Moreover, the smallest time resolution available with the lab instruments is $\Delta t \approx 1.4\mu s$, such lower bound is violated since for the simulation we choose $\Delta t = 0.0.322\mu s$. Lastly, no limitations were imposed on the derivative of the control amplitudes implying ,that in more complicated cases than our example, the Fourier spectrum broadens. To overcome these limitations we need to constrain the amplitude and the spectrum. Following the approach adopted in [7] instead of maximizing Φ_0 only, we include the Lagrange mutipliers in the optimization problem:

$$\max_{u_k(t)}(\mathcal{F}(\hat{\rho}_{target}, \hat{\rho}(T)) - \sum_i \lambda_i \gamma_i(u_k(j))) \tag{3.27}$$

where $\lambda_i \geq 0$ are the Lagrange multipliers and $\gamma_i \geq 0$ are the constraints to the solution. The latter act as penalty terms with the strength set by the magnitude of λ_i. The constraints that we want to include, should punish the gradient directions which lead to an overshoot in the amplitude and sharp variations in time of the control parameters. Thus, the constraint functions should have the following form:

$$\begin{aligned} \gamma_{amplitude} &= \sum_j (|u_k(j\Delta t)| - u_{max})^2 \Theta(|u_k(j\Delta t)| - u_{max}) \\ \gamma_{slope} &= \sum_j (u_k((j+1)\Delta t) - u_k(j\Delta t))^2 \end{aligned} \tag{3.28}$$

where $\Theta(\cdot)$ is the Heaviside step function. The update of the control parameters in (3.21) is modified to:

$$u_k(j) \longrightarrow u_k(j) + \varepsilon \frac{\partial \Phi_0}{\partial u_k(j)} - \sum_i \lambda_i \frac{\partial \gamma_i(u_k(j))}{\partial u_k(j)} \tag{3.29}$$

The derivatives of the constraint functions in (3.28) with respect to the control amplitudes are:

$$\begin{aligned}
\frac{\partial \gamma_{amplitude}}{\partial u_k(j)} &= 2sgn(u_k(j\Delta t))(|u_k(j\Delta t)| - u_{max})\Theta(|u_k(j\Delta t)| - u_{max}) \\
&\quad + \underbrace{(|u_k(j\Delta t)| - u_{max})^2 \delta(|u_k(j\Delta t)| - u_{max})}_{\equiv 0} \\
\frac{\partial \gamma_{slope}}{\partial u_k(j)} &= -2u_k((j+1)\Delta t) + 4u_k(j\Delta t) - 2u_k((j-1)\Delta t)
\end{aligned} \tag{3.30}$$

here we notice that the second term of $\frac{\partial \gamma_{amplitude}}{\partial u_k(j)}$ identically vanishes. The second thing we may notice is that $\frac{\partial \gamma_{slope}}{\partial u_k(j)}$ has a non causal expression involving the knowledge of the control amplitudes at times $t = (j \pm 1)\Delta t$. This causes problems when evaluating the control amplitudes at $t = 0$ and $t = N\Delta t$. In order to overcome this issue we assume that the slope of $u_k(j)$ does not change between the intervals at the end $[(N-1)\Delta t, N\Delta t]$, $[N\Delta t, (N+1)\Delta t]$ and at the beginning $[0, \Delta t]$, $[-\Delta t, 0]$. With this adaptation we can extend the definition of $u_k(j)$ defining its values at $t = -\Delta t$ and $t = (N+1)\Delta t$:

$$\begin{aligned}
u_k((-\Delta t)) &= u_k(0) - (u_k(\Delta t) - u_k(0)) \\
u_k((N+1)\Delta t) &= u_k(N\Delta t) + (u_k(N\Delta t) - u_k((N-1)\Delta t))
\end{aligned} \tag{3.31}$$

We also respected the resolution allowed by the experiment by dividing the time axis into 8 steps of equal size $\Delta t = 1.61\mu s$. Once again, the value of the parameter ε was regulated by hand to:

$$\varepsilon = 5 \times 10^9 \tag{3.32}$$

in addition to that, the values of 8 Lagrange multipliers (4 per each side band, since we want to limit both the amplitude and the derivative of all $u_k(j)$) were also set by hand such that the punishment by violation of the constraints is big enough:

		λ_{I_B}	λ_{Q_B}	λ_{I_R}	λ_{Q_R}
	$\gamma_{amplitude}$	1	1	1	1
	γ_{slope}	0.1	0.1	0.1	0.1

the control amplitudes produced with this setting are shown in the figure below:

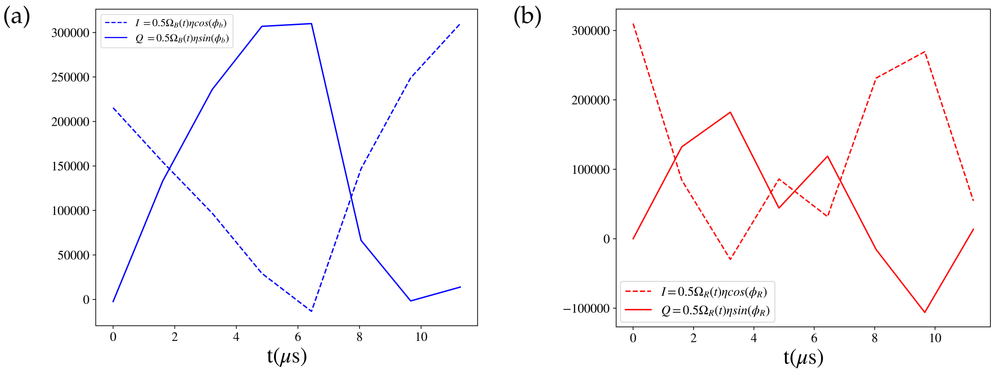

Figure 3.4: In-phase and quadrature components of the control parameters with constraints: a) blue side band control amplitudes b) red side band control amplitudes.

the amplitude overshoots of the side band signals are gone in this case. Using (3.25) we can recover the coupling strengths and the phases of the sideband pulses:

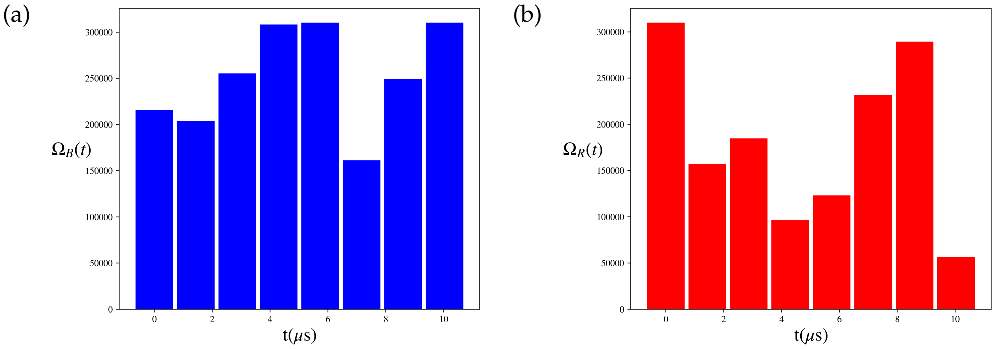

Figure 3.5: Coupling strengths of the sidebands a) blue sideband b) red sideband.

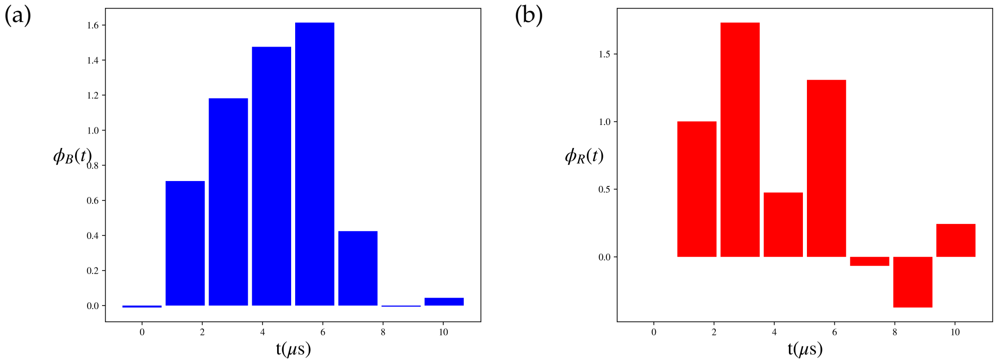

Figure 3.6: Phases of the sidebands a) blue sideband b) red sideband.

The time evolution of the population of the internal states and the phonon population is shown in the figure below:

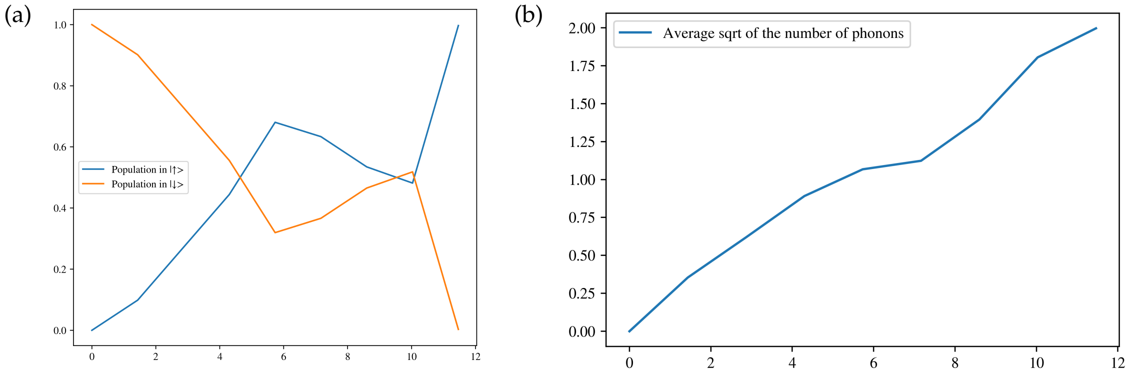

Figure 3.7: a) All the population is transferred from the ground state to the excited state durig the transfer time b) Avarage number of phonons upon the application of sidebands.

The Fig.3.8 shows the evolution of Φ_0 as a function of number of times the algorithm is executed, such number has increased from the unconstrained case:

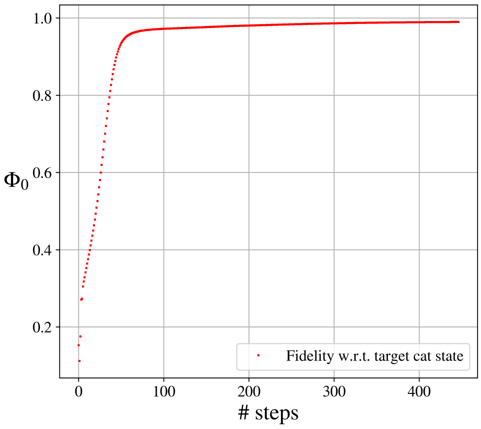

Figure 3.8: Φ_0 vs. number of steps of the algorithm. In this case $\#steps = 447$ and $\Phi_0(\#steps = 447) \approx 0.99$

The Wigner functions of the target cat state and $\hat{\rho}(T)$ are shown below:

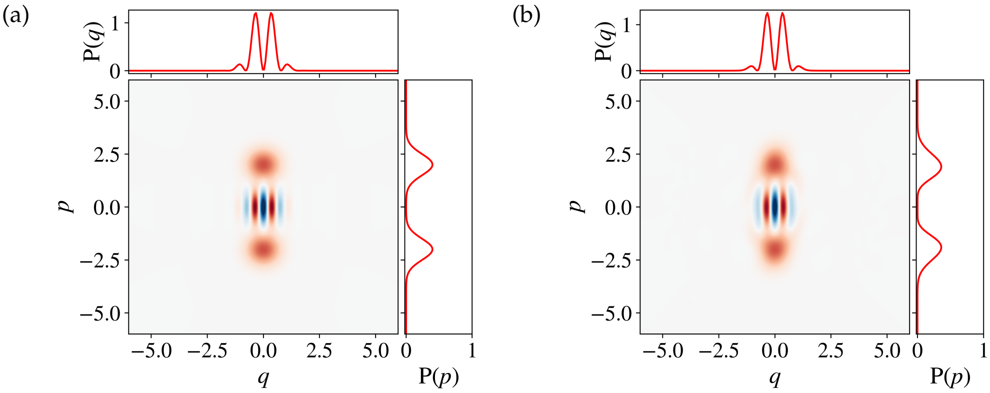

Figure 3.9: Wigner functions of the cat state a)$\hat{\rho}_{target}$ and b)$\hat{\rho}(T)$. The time evolution of the Wigner function can be seen in the GIF attached to the PDF of the report.

This concludes the comparison between the simulation with the physical constraints and the unconstrained case. The next section analyzes the grid state creation which is a more challenging case. However all the procedures being presented in this chapter still apply.

3.3 Grid state

Before we start to discuss the creation of the grid state, let us point out a few aspects of the GRAPE algorithm. The first thing to notice is that GRAPE is a 1^{st} order algorithm and does not provide any information about the curvature of Φ_0, making the achievement of the global maximum of such function improbable. The second aspect to look at is the strong dependence of Φ_0 on the choice of the initial conditions. Some initial conditions lead to local maxima: since the gradient becomes progressively smaller as we approach the local maximum and no stochastic component exists in the actual implementation of the algorithm, we are trapped in such region. One way to overcome such problem could be the introduction of a stochastic part in the algorithm, which gives a "kick" in a random direction as we stay in the region of a local minimum (which can be spotted by setting a threshold to the value of Φ_0) for too long (in terms of the number of iterations of the algorithm). A less sophisticated strategy to solve the problem, much simpler from the implementation point of view, consists in the employment of the pseudo-random number generator used to produce the initial conditions for the algorithm. In the latter case, we run the algorithm many times and keep as the "good" result the initial conditions which lead to a higher value of Φ_0. In this work we will follow the latter trial and error approach to implement the coherent state transfer. Of course the choice of appropriate parameters such as ε and λ_i is also important for the convergence as already explained in the previous chapter.

In order to create the logical zero grid state we chose to start in the sqeezed vacuum state with squeezing parameter $r \approx 0.9$. We divide the time axis into 18 steps of size $\Delta t = 1.43\mu s$ which results in a transfer time T which is 4 times higher than the time needed to create a cat state without constraints. The parameter ε was set to:

$$\varepsilon = \begin{cases} 1 \times 10^{10}, & \text{if } \Phi_0 \leq 0.62 \\ 1 \times 10^{9}, & \text{if } \Phi_0 > 0.62 \end{cases} \tag{3.33}$$

The corresponding Lagrange multipliers are reported in the table below:

		λ_{I_B}	λ_{Q_B}	λ_{I_R}	λ_{Q_R}
$\Phi_0 \leq 0.62$	$\gamma_{amplitude}$	1	1	1	1
	γ_{slope}	0.1	0.1	0.1	0.1
$\Phi_0 > 0.62$	$\gamma_{amplitude}$	1	1	1	1
	γ_{slope}	0.09	0.09	0.09	0.09

Table 3.1: Lagrange multipliers changed accordingly with ε.

To give an illustrative example of the creation of the grid state, we run the algorithm for 7 times for 5285 iterations (this number was chosen to be big enough to observe the convergence of the algorithm) with 7 different initial conditions. The initial values of φ_B, φ_R, g_B and g_R were generated randomly and all simulations were run with settings in the table 3.1 and equation (3.33).The plot of the objective function Φ_0 for these diverse initial conditions is shown in the next figure:

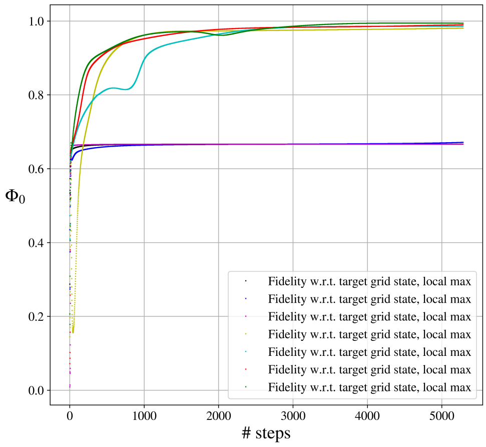

Figure 3.10: Φ_0 vs. number of steps of the algorithm. Different initial conditions are marked with different colours.

as we can observe form the Fig.3.10, there are two main values at which Φ_0 saturates, the lower value corresponds to $\Phi_0 \approx 0.667$ and the bigger value is $\Phi_0 \approx 0.99$. If we look more carefully at the plot, we notice that Φ_0 is not monotone in the initial part of the algorithm:

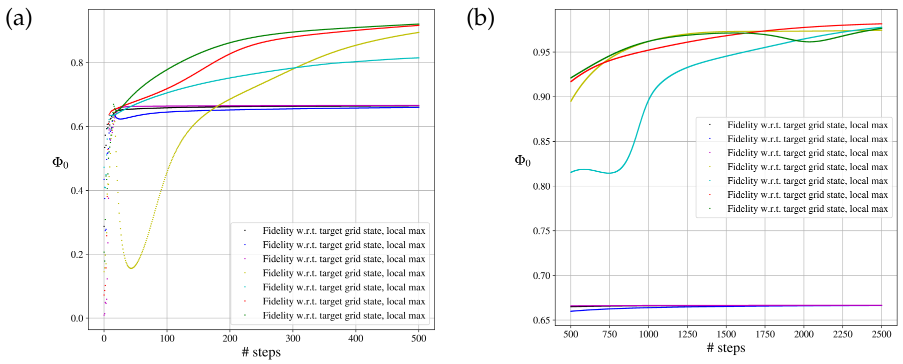

Figure 3.11: a) In the interval #*steps* $= [10, 50]$ we see a dramatic decrease in the yellow curve, which has a monotone increase afterwards. The blue curve also experiences a small decrease in the region #*steps* $= [10, 30]$ and saturates to $\Phi_0 \approx 066$. b) The cyan curve shows a small dip in the region around #*steps* ≈ 750 and the green curve starts to decrease at #*steps* ≈ 1500 and increases again after #*steps* ≈ 2000.

Thus, we see that finding a local maximum is not obvious and it is not easy to say after how many iterations of the algorithm this maximum does not change anymore. This leads to the conclusion that the implementation of a stochastic gradient descent algorithm or an adaptive parameter ε would not be a straight forward task. In order to select the best local maximum among all the simulated values of Φ_0 we can look at the last part of the algorithm:

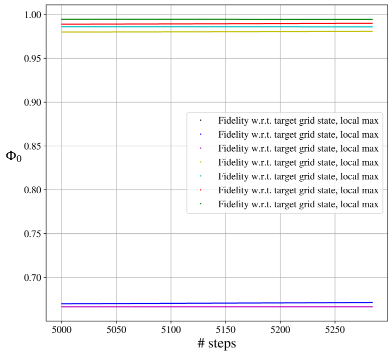

Figure 3.12: Φ_0 vs. number of steps of the algorithm in the interval #*steps* $= [5000, 5285]$. The black curve coincides with the purple one.

Now, since there are multiple initial conditions which lead $\Phi_0 \approx 0.99$ and $\Phi_0 \approx 0.667$, we can ask ourselves whether the values of control amplitudes converge to the same final configurtion when the trace distance approaches

$\Phi_0 \approx 0.99$ or $\Phi_0 \approx 0.667$. To answer this question we can compare the control amplitudes produced by the green curve and red curve, they both approach $\Phi_0 \approx 0.99$ in Fig.3.12. The next figures shows such comparison:

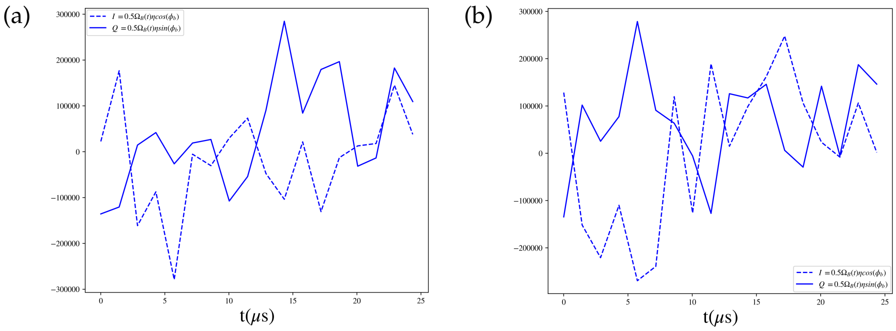

Figure 3.13: In-phase and quadrature components of the blue sidebands in their final configuration for: a)green curve b)red curve.

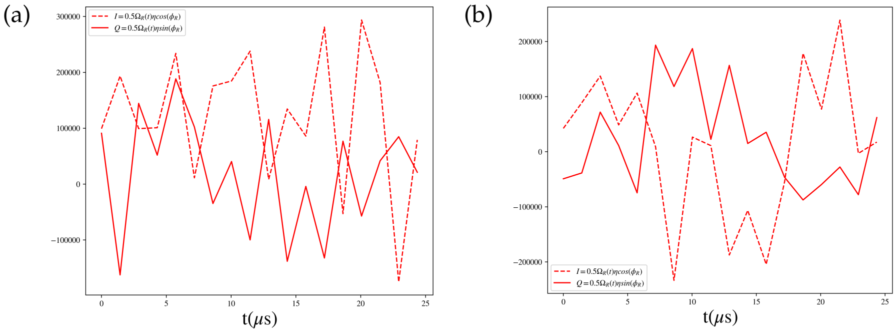

Figure 3.14: In-phase and quadrature components of the red sidebands in their final configuration for: a)green curve b)red curve.

As we can observe in Fig.3.13 and in Fig.3.14, the green and the red curve produce different values for the control parameters even though they both approach $\Phi_0 \approx 0.99$. We can also compare the black and the purple curve since their Φ_0 is the same in the last part of the algorithm as shown in Fig.3.12. The plots of the sideband pulses for such curves are shown below:

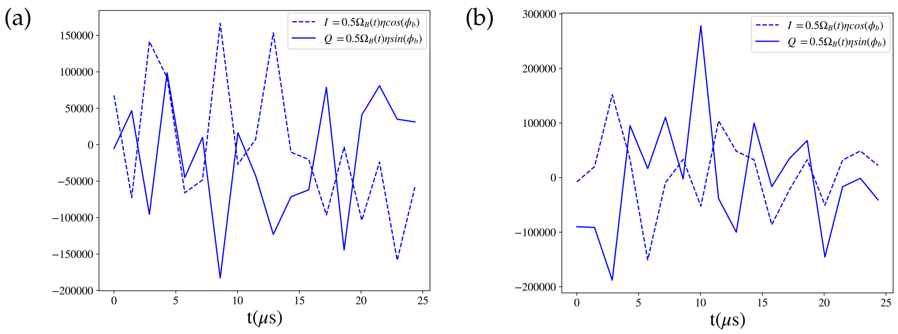

Figure 3.15: In-phase and quadrature components of the blue sidebands in their final configuration for: a)black curve b)purple curve.

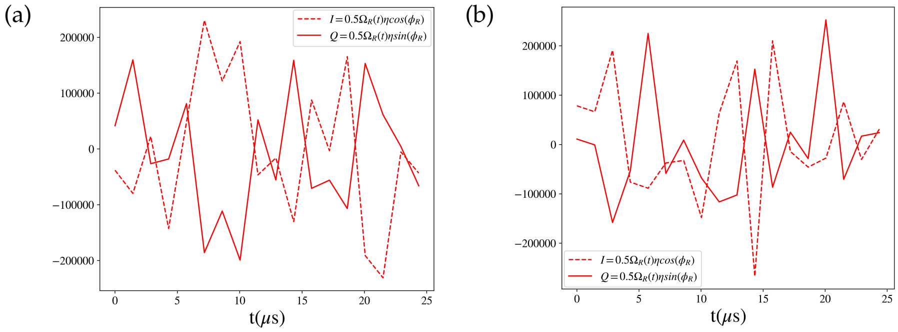

Figure 3.16: In-phase and quadrature components of the red sidebands in their final configuration for: a)black curve b)purple curve.

also in this case we get different pulseshapes for the sidebands. With this approach we can reach different local maxima in the parameter space of φ_B, φ_R, g_B and g_R.

The next thing to notice is that in the interval considered in the Fig.3.12 all the values of the maxima do not change in an appreciable manner anymore. The highest value of Φ_0 is reached by the green curve and it corresponds to $\Phi_{0,green}(5285) = 0.99444185941754 65$. The initial conditions which produced the green curve allow us to reach a high value of Φ_0, so they are a good configuration to start with. We can now add a small perturbation to the initial conditions which generated the green curve and we can check whether it is possible to reach a higher value of objective function. Such perturbations are chosen randomly between 0 and 1% of the maximum values allowed for $[\varphi_{B,green}, \varphi_{R,green}, g_{B,green}, g_{R,green}]$ (i.e 1% of $[2\pi, 2\pi, 310000, 310000]$) and between 0 and 10% of such values. The figure below shows the result after 5285 cycles of the algorithm with the perturbations in the initial conditions and the original curve:

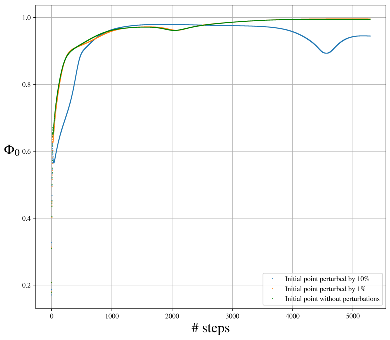

Figure 3.17: Φ_0 vs. number of steps of the algorithm. The unperturbed initial conditions produce Φ_0 marked with green, the blue and the orange curves represent Φ_0 created with the initial conditions perturbed by 10% and 1% respectively.

If we look at the last part of the simulation, we can see that the orange curve, whose initial conditions were perturbed by 1% in a random direction, outperforms the green curve with no perturbations:

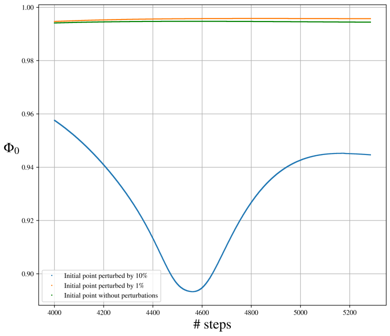

Figure 3.18: Φ_0 vs. number of steps of the algorithm in the interval $\#steps = [5000, 5285]$.

In our example we keep the orange curve as the ″good″ result which in the last point of the simulation reaches $\Phi_{0,orange}(5285) = 0.9957544827350344$. The sideband pulses corresponding to this curve are shown below:

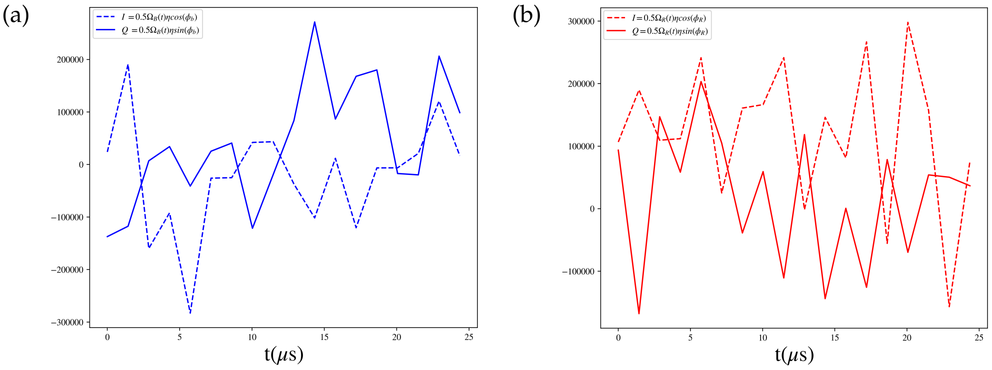

Figure 3.19: In-phase and quadrature components of the control parameters corresponding to the orange curve: a) blue sideband control amplitudes b) red sideband control amplitudes.

Using the eqauation in (3.25) we can obtain the coupling strengths and the phases of the sidebands relative to the in-phase and quadrature components in Fig.3.19:

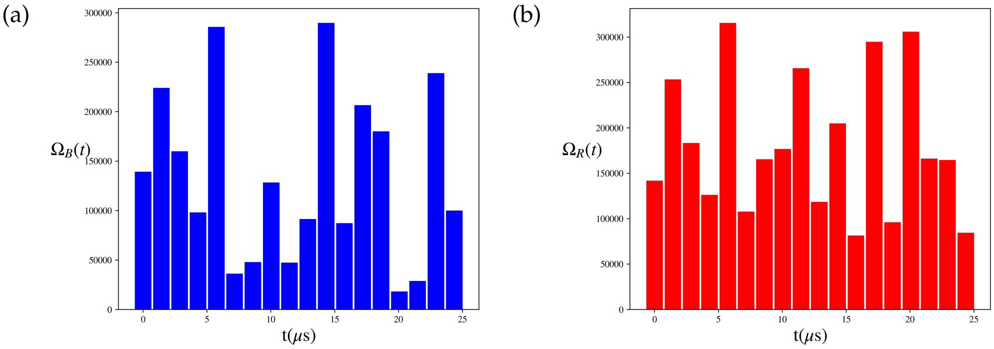

Figure 3.20: Coupling strengths of sidebands corresponding to the orange curve a) blue sideband b) red sideband.

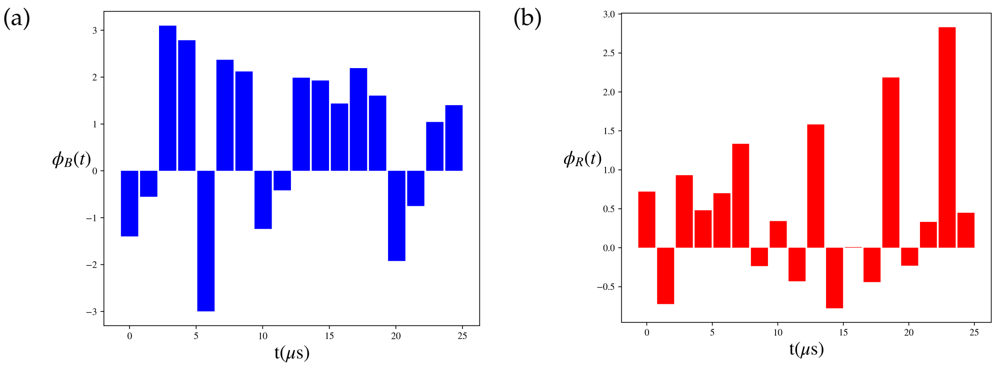

Figure 3.21: Phase pulses of sidebands corresponding to the orange curve a) blue sideband b) red sideband.

The evolution of the populations of the internal states and the phonons, upon application of the sidebands are shown below:

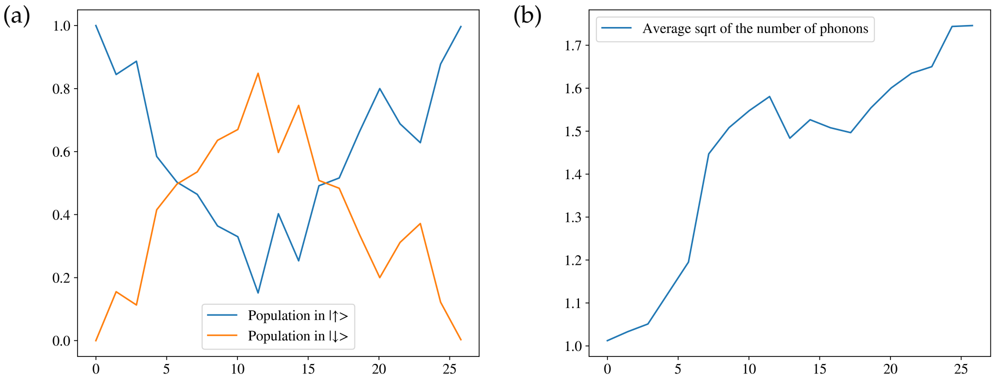

Figure 3.22: a) Time evolution of the populations, the system was initialized in the excited state and after the transfer time all the population ends up in the excited state b) Time evolution of the avarage number of phonons.

the Wigner function of the logical zero grid state obtained with such settings and the ideal grid target state are shown below:

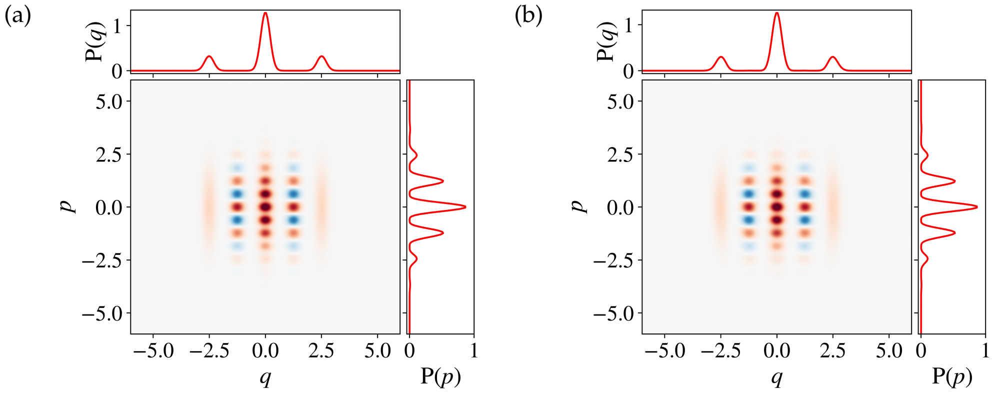

Figure 3.23: Wigner functions of the grid state a)$\hat{\rho}_{target}$ and b)$\hat{\rho}(T)$. The time evolution of the Wigner function can be seen in the GIF attached to the PDF of the report.

The last thing we want to analyze is the stability of the configuration reached by the orange curve. To do so, we add a perturbation to $[\varphi^{final}_{B,orange}, \varphi^{final}_{R,orange}, g^{final}_{B,orange}, g^{final}_{R,orange}]$ randomly chosen between 0 and 10% of $[2\pi, 2\pi, 310000, 310000]$. The next plot shows the result for 40 different perturbations added to the final control amplitudes:

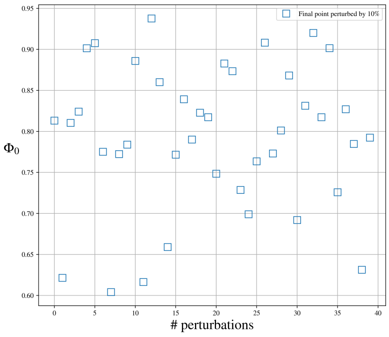

Figure 3.24: Φ_0 vs. number of perturbations of the final configuration of sideband pulses shown in Fig.3.19.

The mean value of the objective function with perturbed configuration in 40 iterations is $\Phi_{0,mean} = 0.794627576428$ whereas the smallest value assumed by Φ_0 in the simulation is $\Phi_{0,min} = 0.6042165395821802$. The Wigner function which corresponds to $\Phi_{0,min}$ is plotted in the next figure:

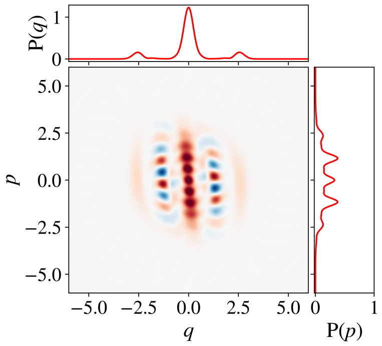

Figure 3.25: Wigner function of the perturbed grid state which corresponds to the lowest value of the overlap $\Phi_{0,min}$ with the target grid.

Even though the values of Φ_0 reached in this example are not extremely high, we illustrated the principle to follow in order to find many local maxima with the 1^{st} order GRAPE algorithm. With more simulations and a different choice of ε and Lagrange multipliers higher values of Φ_0 might be obtained.

Chapter 4

Conclusion and outlook

This chapter concludes the semester project by presenting the reached results reached and lies down suggestions for possible further research.

Despite being far from the experimental implementation, the concepts introduced in the previous chapters allow us to create and study complicated states of harmonic oscillator such as the grid state. The algorithm was analyzed in a way that enabled us to find different local maxima and pick up a high value for Φ_0 and create the desired states faithfully.

Future possibilities :
The coherent state transfer considered in this semester thesis could be improved both physically and numerically.

From a physical point of view, the simulated time evolution is based on the hamiltonian in (3.22). The latter neglects all the off resonant terms, one of which is the carrier term. This term coming from the sidebands is nearly resonant, and starts to play a role when the amplitude of the signals $g_{B,R} \propto \Omega_{B,R}$ becomes large with respect to the trap frequency ω_z. The higher order terms are unimportant as long as we are in the Lamb-Dicke regime. Thus in a future work, a time dependent hamiltonian including the carrier term has to be studied. Another physical aspect neglected in the simulation is the decoherence of the system, which could be included by defining the collapse operators and simulate the dissipative dynamics.

For what concerns the implementation of the algorithm, the adaptive choice of ε and λ_i should be taken into account in order to get an unsupervised algorithm. Moreover, one can also consider the use of higher order derivatives and implement a 2^{nd} order GRAPE algorithm proposed in [8]. This would speed up the simulation since the information about the curvature of Φ_0 would be exploited. In addition, an algorithm that realizes unitary gates instead of the coherent state transfer should also be implemented.

Lastly, we want to point out that the red and blue sideband pulses generated with GRAPE might not be suitable for an experiment with trapped ions. The map of the bichromatic laser, with non trivial waveforms such as those considered in this project, could be very complicated to implement. Thus, one may consider a hamiltonian different from the SDF hamiltonian in (3.22). A nice candidate for this task could be a combination of one of the sideband hamiltonians in (2.46) and (2.43) and a tickling term, which is physically represented by a shake of the trapping potential by means of the electrodes of the trap. The hamiltonian takes the following form:

$$\begin{aligned} \hat{H}_{I,control} &= \hat{H}_{I,BSB/RSB} + (\varepsilon_{tickling}\hat{a} + h.c) \\ &= \hat{H}_{I,BSB/RSB} + (g_t e^{i\varphi_t}\hat{a} + h.c) \\ &= \hat{H}_{I,BSB/RSB} + (g_t \cos\varphi_t(\hat{a} + \hat{a}^\dagger) + g_t \sin\varphi_t(i(\hat{a} - \hat{a}^\dagger))) \end{aligned} \tag{4.1}$$

where the g_t and φ_t represent the coupling and the phase of the tickling hamiltonian. Other terms like a resonant carrier term are also possible candidates for the experimental implementation.

Acknowledgements

I want to thank Christa Flühmann for supervising my semester project, for her patience and useful insights during the whole work. I also want to thank my friend Paolo Colciaghi for stimulating discussions and for the numerous helpful inputs. Lastly I want to thank prof. Jonathan Home for having given me the opportunity to deepen my knowledge in physics.

Appendix A

GRAPE Python code

Listing A.1: Python code for the creation of grid states

```
import qutip as q
import numpy as np

N = 50#number of Fock levels I want to consider
n=q.num(N)
g = q.basis(2,1)
e = q.basis(2,0)
#grid states as target
#1) create the squeezed state
sqzparam=0.89 #squeezing parameter
alpha= np.sqrt(2*np.pi)
sq_op =q.squeeze(N, sqzparam)
sq_state=q.squeeze(N, sqzparam)*coherent(N, 0)
rho0=(q.tensor(e,sq_state)*q.tensor(e,sq_state).dag()).unit()
#2) grid states
zero=((q.displace(N,-alpha)+2+q.displace(N,alpha))*sq_state).unit()

one=(q.displace(N,alpha/2)*zero).unit()

plus= (zero+one).unit()

minus1=(zero-one).unit()

minus=(q.displace(N,1j*alpha/2)*plus).unit()

plusy= (zero+1j*one).unit()

minusy1=(zero-1j*one).unit()
```

```
minusy=(q.displace(N,1j*alpha/2)*plusy).unit()
#here we can choose among different logical grid states,
#we chose as the target state the logical zero in this example
grid0=q.tensor(e,zero)
rhot=grid0*grid0.dag()

grid1=q.tensor(e,one)
grid0_dm=grid1*grid1.dag()

grid_plus=q.tensor(e,plus)
gridplus_dm=grid_plus*grid_plus.dag()


grid_minus=q.tensor(e,minus)
gridminus_dm=grid_minus*grid_minus.dag()
#def of pauli matrices
sx=q.sigmax()
sy=q.sigmay()
sz=q.sigmaz()

#define the hamiltonian in the hermitian form
sp=q.sigmap()
sm=q.sigmam()

H1=q.tensor(sm,q.destroy(N))
H1dag=H1.dag()


#associated to BSB
H_herm1a=(H1+H1dag)
H_herm1b=1.j*(H1-H1dag)




H2=q.tensor(sm,q.create(N))
H2dag=H2.dag()

#associated to RSB
H_herm2a=(H2+H2dag)
H_herm2b=1.j*(H2-H2dag)
```

```
def H_sdf_herm(coupb,coupr,p1,p2):
    H=coupr*np.cos(p2)*H_herm2a+coupr*np.sin(p2)*H_herm2b\
    +coupb*np.cos(p1)*H_herm1a+coupb*np.sin(p1)*H_herm1b
    return H
#define single hermitian hamiltonians

def H_1():
    H=H_herm1a
    return H

def H_2():
    H=H_herm1b
    return H

def H_3():
    H=H_herm2a
    return H

def H_4():
    H=H_herm2b
    return H

#define the forward time propagation
def timef(coupb,coupr,p1,p2,rho,dt):
    times = np.linspace(0.0,dt,100)
    result = q.mesolve(H_sdf_herm(coupb,coupr,p1,p2), rho,times, [], []
    ,options=q.Options(nsteps=1000))
    return result.states
#define the backward time propagation
def timeb(coupb,coupr,p1,p2,rho,dt):
    times = np.linspace(0.0,dt,100)
    result = q.mesolve(-H_sdf_herm(coupb,coupr,p1,p2), rho,times, [], []
    ,options=q.Options(nsteps=1000))
    return result.states

#forward propagated state
def rho_j(rho_init,n_steps,dt,coupb,coupr,p1,p2):
    phia=[]
    #n_steps=n_steps+1
    if n_steps==0: return rho_init
    elif n_steps>0:
        for i in range(n_steps):
            if i==0:
                a=timef(coupb[i],coupr[i],p1[i],p2[i],rho_init,dt)
                phia.append(a[-1].unit())
            if i>0:
                a=timef(coupb[i],coupr[i],p1[i],p2[i],phia[-1],dt)
                phia.append(a[-1].unit())
    return phia

#backward propagated state

def lambda_j(rho_target,n_steps,dt,coupb,coupr,p1,p2):
    phia=[]
    ph1=p1[::-1]
    ph2=p2[::-1]
    coupb1=coupb[::-1]
    coupr1=coupr[::-1]
    #n_steps=n_steps+2
    if n_steps==0: return rho_target
    elif n_steps>0:
        for i in range(n_steps):
            if i==0:
                a=timeb(coupb1[i],coupr1[i],ph1[i],ph2[i]
                ,rho_target,dt)
                phia.append(a[-1].unit())
            if i>0:
                a=timeb(coupb1[i],coupr1[i],ph1[i],ph2[i]
                ,phia[-1],dt)
                phia.append(a[-1].unit())
    return phia
#gradient
def gradient(l_j,p_j,H_i,dt):
#mult1,mult2 are the lagrange multipliers
    f=1.j*dt*commutator(H_i,p_j)
    #gradient of the ith control amplitude
    #at time j with lagrange multipliers
    grad_i=(-(l_j.dag()*f).tr()).real
    return grad_i
def grad1(mult1,amp,amp_max):#amplitude gradient
    grada=mult1*(2*np.sign(amp)*(abs(amp)\
    -amp_max)*np.heaviside((abs(amp)-amp_max),1))
    return grada

def grad2(mult2,amplate,amp,ampbefore):#derivative gradient
    gradd=mult2*((-2*amplate+4*amp-2*ampbefore))
    return gradd
```

```
def main_calculation(time_steps=18,threshold=5285,rho0,rhot\
,coup=310000,t_tot=(2/coup)*4)
    m=numpy.zeros((4,time_steps)) #1st index is the column index,
    #2nd index is the row index
    #m will be the matrix of coefficients

    #define the time for which we want to apply the SDF
    sdf_t1=t_tot
    #arrays for forward propagation, randomly generated
    phi_b=np.random.rand(1,time_steps)*2*np.pi
    phi_r=np.random.rand(1,time_steps)*2*np.pi
    coupb=np.random.rand(1,time_steps)*1*coup
    coupr=np.random.rand(1,time_steps)*1*coup
    #initialize the lagrange multipliers
    #4 multipliers for blue 4 multipliers for red
    mult=[1,0.1,1,0.1,1,0.1,1,0.1]
    epsilon=1e10
    fidelity=[]
    #matrix of coefficients
    for j in range(0,(time_steps),1):
        m[0][j]=coupb[0][j]*np.cos(phi_b[0][j])
        m[1][j]=coupb[0][j]*np.sin(phi_b[0][j])
        m[2][j]=coupr[0][j]*np.cos(phi_r[0][j])
        m[3][j]=coupr[0][j]*np.sin(phi_r[0][j])
    f1=0
    flag=0
    while(flag<=threshold):
        a=lambda_j(rhot,time_steps,sdf_t1/time_steps,coupb[0][:]
        ,coupr[0][:],phi_b[0][:],phi_r[0][:])
        b=rho_j(rho0,time_steps,sdf_t1/time_steps,coupb[0][:]
        ,coupr[0][:],phi_b[0][:],phi_r[0][:])
        for j in range(0,(time_steps),1):
            if j==0:
                dummie0=(gradient(a[time_steps-1],rho0,H_1()
                ,sdf_t1/(time_steps)))
                dummie1=(gradient(a[time_steps-1],rho0,H_2()
                ,sdf_t1/(time_steps)))
                dummie2=(gradient(a[time_steps-1],rho0,H_3()
                ,sdf_t1/(time_steps)))
                dummie3=(gradient(a[time_steps-1],rho0,H_4()
                ,sdf_t1/(time_steps)))
            if j>0:
                dummie0=(gradient(a[time_steps-1-j],b[j-1],H_1()
                    ,sdf_t1/(time_steps)))
                    dummie1=(gradient(a[time_steps-1-j],b[j-1],H_2()
                    ,sdf_t1/(time_steps)))
                    dummie2=(gradient(a[time_steps-1-j],b[j-1],H_3()
                    ,sdf_t1/(time_steps)))
                    dummie3=(gradient(a[time_steps-1-j],b[j-1],H_4()
                    ,sdf_t1/(time_steps)))
                #these are the gradients of the penalties
                if j==0:
                    a_first=2*m[0][0]-m[0][1]# epsilon0(n-1) blue1 Pb
                    a1_first=2*m[1][0]-m[1][1]# epsilon0(n-1) blue2 Qb
                    b_first=2*m[2][0]-m[2][1]# epsilon0(n-1) red1 Pr
                    b1_first=2*m[3][0]-m[3][1]# epsilon0(n-1) red2 Qr
                    #gradient for Pb
                    dummieba_Pb=grad1(-mult[0],m[0][j],coup)
                    dummiebd_Pb=grad2(-mult[1],m[0][j+1],m[0][j]
                    ,a_first)
                    #gradient for Qb
                    dummieba_Qb=grad1(-mult[2],m[1][j],coup)
                    dummiebd_Qb=grad2(-mult[3],m[1][j+1],m[1][j]
                    ,a1_first)
                    #gradient for Pr
                    dummiera_Pr=grad1(-mult[4],m[2][j],coup)
                    dummierd_Pr=grad2(-mult[5],m[2][j+1],m[2][j]
                    ,b_first)
                    #gradient for Qr
                    dummiera_Qr=grad1(-mult[6],m[3][j],coup)
                    dummierd_Qr=grad2(-mult[7],m[3][j+1],m[3][j]
                    ,b1_first)
                elif j>0 and j<(time_steps-1):
                    #gradient for Pb
                    dummieba_Pb=grad1(-mult[0],m[0][j],coup)
                    dummiebd_Pb=grad2(-mult[1],m[0][j+1],m[0][j]
                    ,m[0][j-1])
                    #gradient for Qb
                    dummieba_Qb=grad1(-mult[2],m[1][j],coup)
                    dummiebd_Qb=grad2(-mult[3],m[1][j+1],m[1][j]
                    ,m[1][j-1])
                    #gradient for Pr
                    dummiera_Pr=grad1(-mult[4],m[2][j],coup)
                    dummierd_Pr=grad2(-mult[5],m[2][j+1],m[2][j]
                    ,m[2][j-1])
                    #gradient for Qr
                    dummiera_Qr=grad1(-mult[6],m[3][j],coup)
```

```
    dummierd_Qr=grad2(-mult[7],m[3][j+1],m[3][j]
    ,m[3][j-1])
elif j==time_steps-1:
   # epsilon0(n+1) blue1 Pb
   a_last=2*m[0][time_steps-1]-m[0][time_steps-2]
   # epsilon0(n+1) blue2 Qb
   a1_last=2*m[1][time_steps-1]-m[1][time_steps-2]
   # epsilon0(n+1) red1 Pr
   b_last=2*m[2][time_steps-1]-m[2][time_steps-2]
   # epsilon0(n+1) red2 Qr
   b1_last=2*m[3][time_steps-1]-m[3][time_steps-2]
   #gradient for Pb
   dummieba_Pb=grad1(-mult[0],m[0][j],coup)
   dummiebd_Pb=grad2(-mult[1],a_last,m[0][j],m[0][j-1])
   #gradient for Qb
   dummieba_Qb=grad1(-mult[2],m[1][j],coup)
   dummiebd_Qb=grad2(-mult[3],a1_last,m[1][j],m[1][j-1])
   #gradient for Pr
   dummiera_Pr=grad1(-mult[4],m[2][j],coup)
   dummierd_Pr=grad2(-mult[5],b_last,m[2][j],m[2][j-1])
   #gradient for Qr
   dummiera_Qr=grad1(-mult[6],m[3][j],coup)
   dummierd_Qr=grad2(-mult[7],b1_last,m[3][j],m[3][j-1])
#update of matrix of coefficients
m[0][j]=m[0][j]+epsilon*dummie0+dummieba_Pb+dummiebd_Pb\
+dummieba_Qb+dummiebd_Qb+dummiera_Pr+dummierd_Pr+dummiera_Qr\
+dummierd_Qr
m[1][j]=m[1][j]+epsilon*dummie1+dummieba_Pb+dummiebd_Pb\
+dummieba_Qb+dummiebd_Qb+dummiera_Pr+dummierd_Pr+dummiera_Qr\
+dummierd_Qr
m[2][j]=m[2][j]+epsilon*dummie2+dummieba_Pb+dummiebd_Pb\
+dummieba_Qb+dummiebd_Qb+dummiera_Pr+dummierd_Pr+dummiera_Qr\
+dummierd_Qr
m[3][j]=m[3][j]+epsilon*dummie3+dummieba_Pb+dummiebd_Pb\
+dummieba_Qb+dummiebd_Qb+dummiera_Pr+dummierd_Pr+dummiera_Qr\
+dummierd_Qr
#update of parameters used in the propagation
phi_b[0][j]=np.arctan2((m[1][j]-(epsilon*dummie1+dummieba_Pb\
+dummiebd_Pb+dummieba_Qb+dummiebd_Qb+dummiera_Pr+dummierd_Pr\
+dummiera_Qr+dummierd_Qr))
,(m[0][j]-(epsilon*dummie0+dummieba_Pb+dummiebd_Pb\
+dummieba_Qb+dummiebd_Qb+dummiera_Pr+dummierd_Pr+dummiera_Qr\
+dummierd_Qr)))
coupb[0][j]=np.sqrt((m[0][j]-(epsilon*dummie0\
    +dummieba_Pb+dummiebd_Pb+dummieba_Qb+dummiebd_Qb\
    +dummiera_Pr+dummierd_Pr+dummiera_Qr+dummierd_Qr))**2\
    +(m[1][j]-(epsilon*dummie1+dummieba_Pb+dummiebd_Pb\
    +dummieba_Qb+dummiebd_Qb+dummiera_Pr+dummierd_Pr\
    +dummiera_Qr+dummierd_Qr))**2)
    phi_r[0][j]=np.arctan2((m[3][j]-(epsilon*dummie3\
    +dummieba_Pb+dummiebd_Pb+dummieba_Qb+dummiebd_Qb\
    +dummiera_Pr+dummierd_Pr+dummiera_Qr+dummierd_Qr))
    ,(m[2][j]-(epsilon*dummie2+dummieba_Pb+dummiebd_Pb\
    +dummieba_Qb+dummiebd_Qb+dummiera_Pr+dummierd_Pr\
    +dummiera_Qr+dummierd_Qr)))
    coupr[0][j]=np.sqrt((m[2][j]-(epsilon*dummie2\
    +dummieba_Pb+dummiebd_Pb+dummieba_Qb+dummiebd_Qb\
    +dummiera_Pr+dummierd_Pr+dummiera_Qr+dummierd_Qr))**2\
    +(m[3][j]-(epsilon*dummie3+dummieba_Pb+dummiebd_Pb\
    +dummieba_Qb+dummiebd_Qb+dummiera_Pr+dummierd_Pr\
    +dummiera_Qr+dummierd_Qr))**2)
    #if the amplitude exceeds the bounds
    #assign the max possible value
    if m[0][j]>coup or m[0][j]<(-coup):
        if m[0][j]>0:
            m[0][j]=coup
        elif m[0][j]<0:
            m[0][j]=-coup
    if m[1][j]>coup or m[1][j]<(-coup):
        if m[1][j]>0:
            m[1][j]=coup
        elif m[1][j]<0:
            m[1][j]=-coup
    if m[2][j]>coup or m[2][j]<(-coup):
        if m[2][j]>0:
            m[2][j]=coup
        elif m[2][j]<0:
            m[2][j]=-coup
    if m[3][j]>coup or m[3][j]<(-coup):
        if m[3][j]>0:
            m[3][j]=coup
        elif m[3][j]<0:
            m[3][j]=-coup
answer3=rhot
answer4=rho_j(rho0,time_steps,sdf_t1/time_steps,
coupb[0][:],coupr[0][:],phi_b[0][:],phi_r[0][:])
#f1 is the trace distance after 1 cycle of the algorithm
f1=((answer3.dag()*answer4[time_steps-1]).tr()).real
```

```
        flag=flag+1
        if (f1>0.62):
            epsilon=1e9
            mult=[1,0.09,1,0.09,1,0.09,1,0.09]
        fidelity.append(f1)
    return (fidelity,answer4,coupb[0][:],coupr[0][:],
    phi_b[0][:],phi_r[0][:])
```

Bibliography

[1] C.Flühmann *et al.*, Physical Review X, 8, 021001 (2018).

[2] D. Gottesman, A. Kitaev, J. Preskill, Phys. Rev. A, 64, 012310 (2001).

[3] Haroche, Raimond, Exploring the quantum, Oxford Graduate Text

[4] N.Khaneja *et al.*, jmr, 172, 2, (2005)

[5] Mark IJspeert , bachelor thesis,(2015)

[6] R. F. Snider ,Journal of Mathematical Physics, 5, 1580, (1964)

[7] RW Heeres *et al.*, Nature volume 8, 94 (2017).

[8] P. de Fouquieres *et al.*, jmr, 212, 2, (2011).

[9] Jonathan Home, Lecture Notes of Cavity QED and ion trapping course

[10] D. Leibfried *et al.*, Nature volume 422, 412–415, (2003)

[11] Daniel Kienzler *et al.*, Science 347, 6217 (2015)

[12] WP Schleich,Quantum Optics in Phase Space, (2001)