Stark-Effect

Qiskit Pulse

HTransistor

October 29, 2020

Contents

1 Quantum Machine Learning

2 Stark Effect

2.1 Problems

- hyperfine structure negligible (normally when $\vec{E}$ is strong enough)
- perpendicular and transverse (formulas?)
- perturbation theory issue: states without $\vec{E}$ are square-integrable, with not (resonances of finite width, which have decay time, but for lower states its approximately stable), watch out for high $\vec{E}$
- in semiconductor enhance Stark-Effect because in the bigger band gap materials the holes and electrons are pulled in different directions, and not compensated by those in smaller band gap material

2.2 Questions

- ion trap or superconductor?
- which atom are we looking at (degenerated, just lower states?)
- perturbation theory
- Runge-Lenz-Vector $\Rightarrow$ exactly solvable approximate model for an atom in strong oscillatory $\vec{E}$

3 Quantum-confined Stark Effect (QCSE)

3.1 Problems

- $\vec{E}_{ex}$ absorption spectrum or emission spectrum of a quantum well
- electron states shift to lower energies, while the hole states shift to higher energies $\Rightarrow$ reduces the permitted light absorption or emission frequencies
- Holes and electrons pulled to different sides $\Rightarrow$ decreasing overlap integral $\Rightarrow$ reduces recombination efficiency

- Quantum Objects (Wells, Dots or Discs, for instance) emit and absorb light generally with higher energies than the band gap of the material, the QCSE may shift the energy to values lower than the gap.
- perturbation theory
- redshift for optical transition and decreases magnitude of absorption coefficient
- effect of excitons: $h\nu > E_g - E_X$ with E_X as energy of exciton (like hydrogen atom)
- $\vec{E}$ applied to bulk semiconductor, redshift also trough Franz-Keldysh effect, this one is limited by the absence of exciton as to strong $\vec{E}$ pull holes and electrons apart, this doesn't happen for QCSE as e^- are confined in quantum wells
- quantum wells behave as 2D $\Rightarrow$ enhance excitonic effects (E_X 4 times larger than in 3D)

4 Qiskit Pulse

4.1 Calibrating Pulse

- what are measurement_map ?
- meas_level=0 returns raw data (an array of complex values per shot), meas_level=1 returns kerneled data (one complex value per shot), and meas_level=2 returns classified data (a 0 or 1 bit per shot) $\Rightarrow$ what is measured?
- why is there some hard-coded 8 and 16? fft?
- What are the different channels, what are they for? (drive, acquire, measure)
- why is $\omega_{rabi,rough} + \omega_{detune} - \omega_0$ helping experimentally over $\omega_{rabi,rough} - \omega_0$?
- Why is Dynamical decoupling removing static noise?

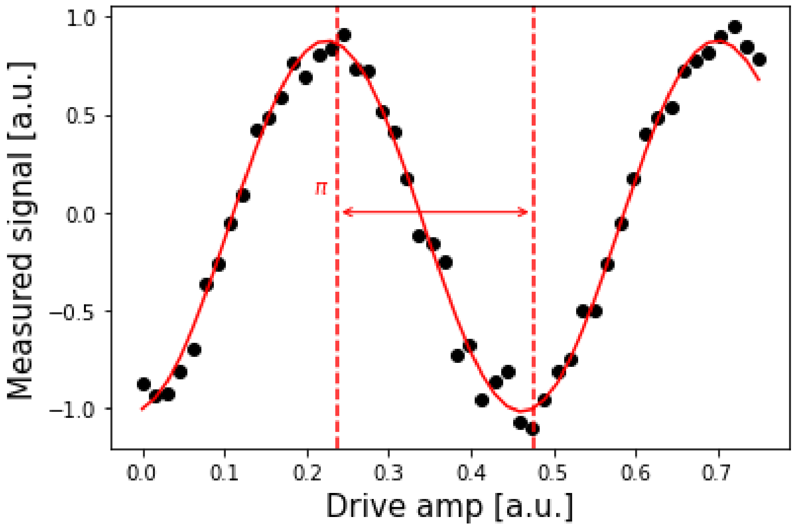


Figure 1: Normally rabi-oscillation are P_g population in ground state over t, while $\vec{E}$ is a constant wave (constant amplitude), but here we have pulses. QISKIT-TEXTBOOK, *Calibrating Qubits Pulse* (2019)

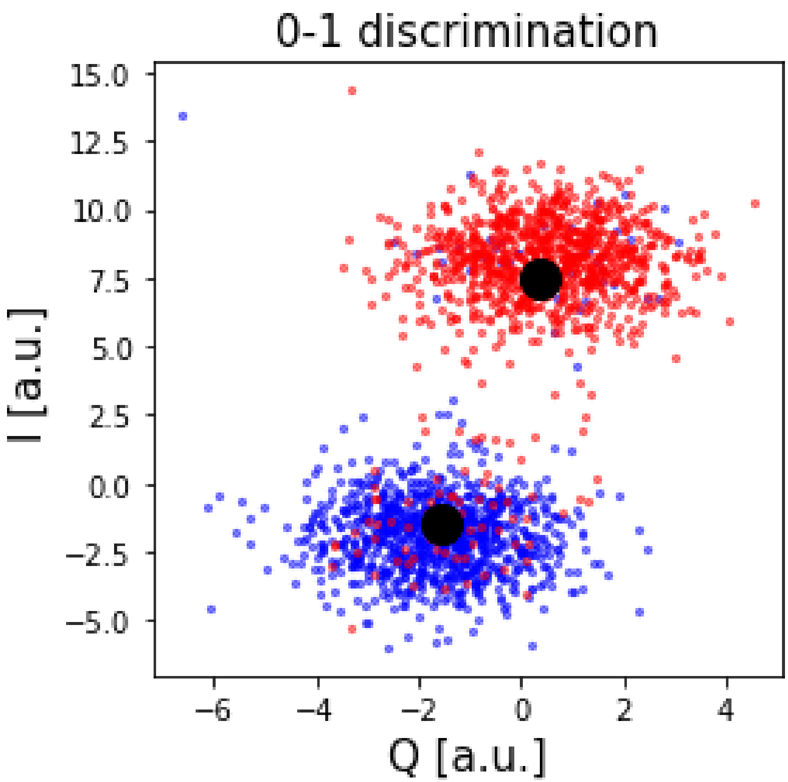
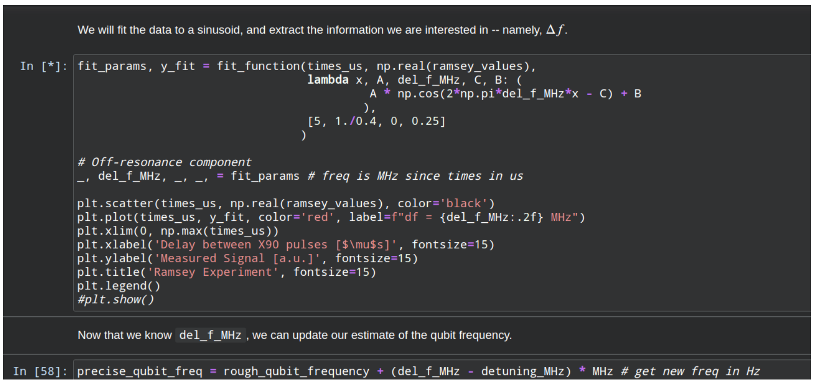


Figure 2: The blue cluster is $|0\rangle$ and red cluster is $|1\rangle$. But which physical quantity is measured (complex value of what?)
QISKIT-TEXTBOOK, *Calibrating Qubits Pulse* (2019)



We will fit the data to a sinusoid, and extract the information we are interested in -- namely, Δf.

```
In [*]: fit_params, y_fit = fit_function(times_us, np.real(ramsey_values),
                                 lambda x, A, del_f_MHz, C, B: (
                                          A * np.cos(2*np.pi*del_f_MHz*x - C) + B
                                         ),
                                 [5, 1./0.4, 0, 0.25]
                                )

        # Off-resonance component
        _, del_f_MHz, _, _, = fit_params # freq is MHz since times in us

        plt.scatter(times_us, np.real(ramsey_values), color='black')
        plt.plot(times_us, y_fit, color='red', label=f"df = {del_f_MHz:.2f} MHz")
        plt.xlim(0, np.max(times_us))
        plt.xlabel('Delay between X90 pulses [$\mu$s]', fontsize=15)
        plt.ylabel('Measured Signal [a.u.]', fontsize=15)
        plt.title('Ramsey Experiment', fontsize=15)
        plt.legend()
        #plt.show()
```

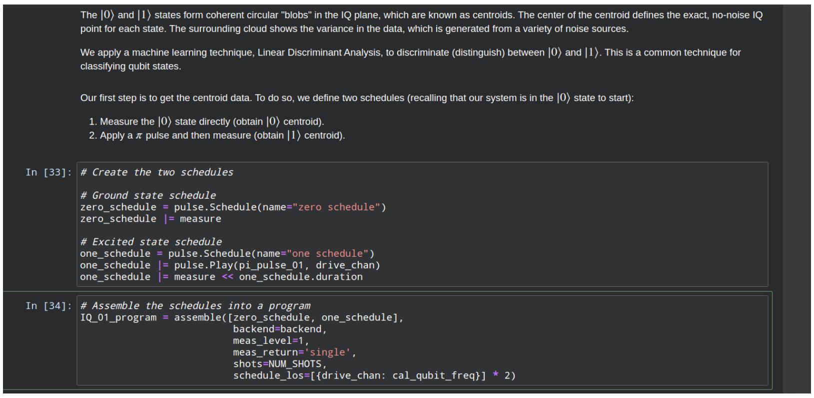
Now that we know `del_f_MHz`, we can update our estimate of the qubit frequency.

```
In [58]: precise_qubit_freq = rough_qubit_frequency + (del_f_MHz - detuning_MHz) * MHz # get new freq in Hz
```

Figure 3: Probability for excited state is $P_g = \cos^2\left(\Delta t \cdot \frac{(\omega - \omega_0)}{2}\right)$
QISKIT-TEXTBOOK, *Calibrating Qubits Pulse* (2019)

4.2 Accessing Higher Energy States

- What does the following mean? The Pulse specification requires a single local oscillator frequency per schedule.
- The pulses to get to higher levels need to be shorter than the half time of the state? In state $|1\rangle$ the population is going down fast, so measuring the next energy jump is much harder. Where are the limits?



The $|0\rangle$ and $|1\rangle$ states form coherent circular "blobs" in the IQ plane, which are known as centroids. The center of the centroid defines the exact, no-noise IQ point for each state. The surrounding cloud shows the variance in the data, which is generated from a variety of noise sources.

We apply a machine learning technique, Linear Discriminant Analysis, to discriminate (distinguish) between $|0\rangle$ and $|1\rangle$. This is a common technique for classifying qubit states.

Our first step is to get the centroid data. To do so, we define two schedules (recalling that our system is in the $|0\rangle$ state to start):

1. Measure the $|0\rangle$ state directly (obtain $|0\rangle$ centroid).
2. Apply a π pulse and then measure (obtain $|1\rangle$ centroid).

In [33]:
```
# Create the two schedules

# Ground state schedule
zero_schedule = pulse.Schedule(name="zero schedule")
zero_schedule |= measure

# Excited state schedule
one_schedule = pulse.Schedule(name="one schedule")
one_schedule |= pulse.Play(pi_pulse_01, drive_chan)
one_schedule |= measure << one_schedule.duration
```

In [34]:
```
# Assemble the schedules into a program
IQ_01_program = assemble([zero_schedule, one_schedule],
                         backend=backend,
                         meas_level=1,
                         meas_return='single',
                         shots=NUM_SHOTS,
                         schedule_los=[{drive_chan: cal_qubit_freq}] * 2)
```

Figure 4: Why is the *meas_return* single here and before always average? QISKIT-TEXTBOOK, *Accessing Higher Energy States* (2019)

```
# Compute accuracy
score_012 = LDA_012.score(IQ_012_test, state_012_test)
print(score_012)
```

0.7356770833333334

The last step is to plot the separatrix.

```
IQ_012_plot(x_min, x_max, y_min, y_max)
separatrixPlot(LDA_012, x_min, x_max, y_min, y_max, NUM_SHOTS)
```

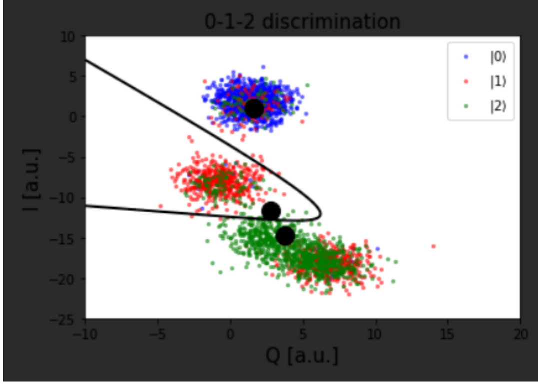


Figure 5: What does the overlapping of the regions mean in terms of the system/calibration?
QISKIT-TEXTBOOK, *Accessing Higher Energy States* (2019)

4.3 Circuit Quantum Electrodynamics

- regimes where the resonator acts as a perturbation to the qubit, perturbation theory (Schrieffer-Wolff transformation)
- Why is the drive part of the Hamiltonian block off-diagonal?

$$H_{\text{eff}} = \left(\omega_r + \frac{g^2}{\Delta}\sigma_z\right) a^\dagger a - \frac{1}{2}\left(\omega_q - \frac{g^2}{\Delta}\right)\sigma_z$$

which shows a state-dependent shift by $\chi \equiv g^2/\Delta$ of the resonator frequency called the *ac Stark shift* and a shift in qubit frequency due to quantum vacuum fluctuations called the *Lamb shift*.

Figure 6: James-Cumming Model
QISKIT-TEXTBOOK, *Circuit Quantum Electrodynamics* (2019)

$$H^{\text{tr}} = \omega_r a^\dagger a + \sum_j \omega_j |j\rangle\langle j| + g\left(a^\dagger c + a c^\dagger\right),$$

where now $c = \sum_j \sqrt{j+1}|j\rangle\langle j+1|$ is the transmon lowering operator. Similarly, taking the weakly interacting subsets A as the even-numbered transmon modes and B as the odd-numbered transmon modes. Using the ansatz

$$S^{(1)} = \sum_j \alpha_j a^\dagger \sqrt{j+1}|j\rangle\langle j+1| - \alpha_j^* a \sqrt{j+1}|j+1\rangle\langle j|,$$

one may proceed along a messier version of the Jaynes-Cummings Hamiltonian. With some effort one can show the second order effective Hamiltonian is

$$H_{\text{eff}}^{\text{tr}} = \left(\omega_r + \sum_j \frac{g^2(\omega_r - \omega + \delta)}{(\omega_r - \omega - \delta j)(\omega_r - \omega - \delta(j-1))}|j\rangle\langle j|\right) a^\dagger a + \sum_j \left[j\omega + \frac{\delta}{2}(j-1)j + \frac{jg^2}{\omega - \omega_r + (j-1)\delta}\right]|j\rangle\langle j|.$$

Figure 7: Transmon qubit
QISKIT-TEXTBOOK, *Circuit Quantum Electrodynamics* (2019)