Viability of Raman coupling for realistic quantum computation

Marco Dall'Ara

September 11, 2023

Raman coupling allows to dynamically couple two long-lived atomic levels |0> and |1>, that do not have a direct dipole transition, by means of an intermediate level |ei strongly optically coupled (electric dipole) to both of them. By carefully choosing detunings and amplitudes of two lasers, it is possible to obtain full transition |0i $\rightarrow$ |1i without ever populating substantially the intermediate, short-lived level, albeit at second perturbative order. Consequently, the fast decay of the intermediate level does not really have an impact. But how much true is all of that? Let us find out in this exercise.

To begin with consider only the spontaneous emissions from level |ei into EACH of the two lower-energy levels. How do the spectral lines broaden? Is there interference between the two emission processes? Convert the spontaneous emission line-broadening into a set of Lindblad (jump) operators. How many do you need?

Consider a three-levels atom coupled with a bath. If the atom is in the excited state, there are several decay channels in the bath. The Hamiltonian for this system is made of H_A, H_{EM} and H_{AL1}, H_{AL2} which in the rotating-wave approximation are:

$$H_{AL_1} = -\vec{d}_{eg1}\sigma_{+1}\hat{\vec{E}}^{(+)} - \vec{d}^{*}_{eg1}\sigma_{-1}\hat{\vec{E}}^{(-)}$$

$$H_{AL_2} = -\vec{d}_{eg2}\sigma_{+2}\hat{\vec{E}}^{(+)} - \vec{d}^{*}_{eg2}\sigma_{-2}\hat{\vec{E}}^{(-)}$$

with

$$\hat{\vec{E}}^{(+)} = -i\sum_{\vec{k},\lambda}\sqrt{\frac{\hbar\omega_k}{2\varepsilon_0 V}}\vec{\epsilon}_\lambda\,\hat{a}_{\vec{k},\lambda},$$
$$\hat{\vec{E}}^{(-)} = i\sum_{\vec{k},\lambda}\sqrt{\frac{\hbar\omega_k}{2\varepsilon_0 V}}\vec{\epsilon}_\lambda\,\hat{a}^{\dagger}_{\vec{k},\lambda}.$$

RWA conditions :

$$|\omega_{eg1} + \omega_k| \gg |\omega_{eg1} - \omega_k|, \tilde{g}_1$$

$$|\omega_{eg2} + \omega_k| \gg |\omega_{eg2} - \omega_k|, \tilde{g}_2$$

with $\tilde{g}_i$ strength of H_{AL_i}.

Now consider the microscopic derivation of the Linblad master equation for the two-levels system :

$$\begin{aligned} H_{System} &= H_{System} \otimes \mathbb{I}_B, \\ H_{Bath} &= \mathbb{I}_S \otimes H_{Bath}, \\ H_{\text{int}} &= \sum_\alpha S_\alpha \otimes B_\alpha \end{aligned}$$

with

$$S_{\alpha=0} \equiv \sigma_+ \qquad \text{and} \qquad S_{\alpha=1} \equiv \sigma_-$$

$$\hat{B}_{\alpha=0} \equiv -i\vec{d}_{eg} \cdot \hat{\vec{E}}^{(+)}. \qquad \text{and} \qquad \hat{B}_{\alpha=1} \equiv -i\vec{d}^*_{eg} \cdot \hat{\vec{E}}^{(-)}.$$

We can adapt it to the three-levels system by just considering $H_{\text{int}} = H_{\text{int1}} + H_{\text{int2}}$, where in the i-th term we consider σ operators between the excited state and the i-th ground state.

Note that in this project we will alwais consider the case where the radiation from the two transitions is distinguishable, and so we neglet the off diagonal terms of the correlator matrix $G_{\alpha,\beta}(t) = \langle B^\dagger_\alpha(t) B_\beta(0) \rangle_{Bath}$.

The final master equation is the sum of Linblad superoperators.

$D[c]\rho := c\rho c^\dagger - \frac{1}{2}\left[c^\dagger c\rho + \rho c^\dagger c\right]$. Then:

$$\partial_t \rho = -\frac{i}{\hbar}\left[H_A, \rho\right] + \Gamma_{eg1}\mathcal{D}[\sigma_{-g1}]\rho + \Gamma_{eg2}\mathcal{D}[\sigma_{-g2}]\rho$$

with $\Gamma_{egi} = \frac{d^2_{eg_i}\omega^3_{eg_i}}{3\pi\epsilon_0\hbar c^3}$.

And finally:

$$\begin{aligned} \partial_t \rho_{ee} &= -(\Gamma_{eg1} + \Gamma_{eg2})\rho_{ee} \\ \partial_t \rho_{g_1 g_1} &= \Gamma_{eg1}\rho_{ee} \\ \partial_t \rho_{g_1 g_1} &= \Gamma_{eg2}\rho_{ee} \\ \partial_t \rho_{g_1 g_2} &= -i(\omega_{g1} - \omega_{g2})\rho_{g_1 g_2} \\ \partial_t \rho_{eg_1} &= i(\omega_{g1} - \omega_e)\rho_{eg_1} - \frac{\Gamma_{eg1} + \Gamma_{eg2}}{2}\rho_{eg_1} \\ \partial_t \rho_{eg_2} &= i(\omega_{g2} - \omega_e)\rho_{eg_2} - \frac{\Gamma_{eg1} + \Gamma_{eg2}}{2}\rho_{eg_2} \end{aligned}$$

By solving these equations with initial condition $\rho_{ee}(0) = 1$ we get:

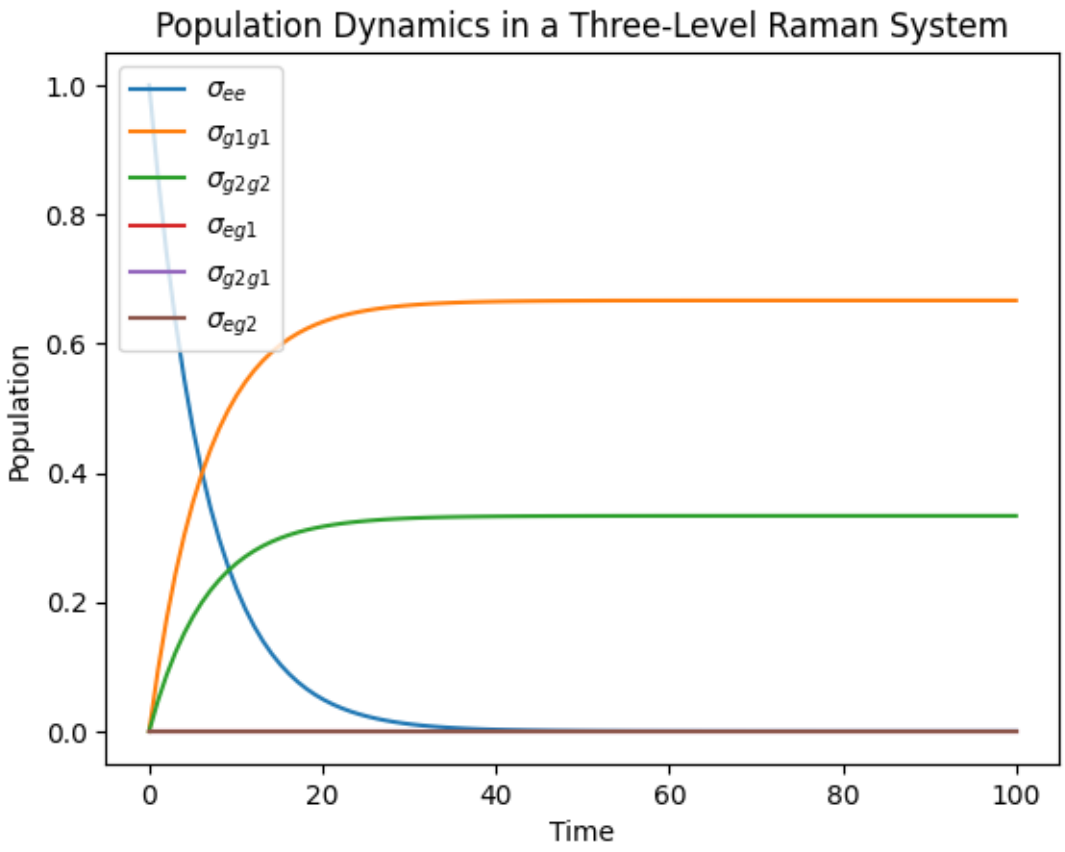
$$\rho_{ee} = e^{-(\Gamma_{eg1}+\Gamma_{eg2})t}$$
$$\rho_{g_1g_1} = \frac{\Gamma_{eg1}}{\Gamma_{eg1}+\Gamma_{eg2}}(1 - e^{-(\Gamma_{eg1}+\Gamma_{eg2})t})$$
$$\rho_{g_2g_2} = \frac{\Gamma_{eg2}}{\Gamma_{eg1}+\Gamma_{eg2}}(1 - e^{-(\Gamma_{eg1}+\Gamma_{eg2})t})$$

Then the line broadening is $\Gamma_{eg1}+\Gamma_{eg2}$. Note that if we start the dynamics in the excited state then there are not non-zero terms off-diagonal in the denisty matrix, then there is no quantum coherence and then the two processes do not interefere. However, when working with a larger number of atoms, the photons emitted as a result of spontaneous emission can interact with each other, giving rise to a phenomenon known as either superradiance or subradiance.

In the code below, we verify computationally what just found.

[23]:
```
import numpy as np
import matplotlib.pyplot as plt
from qutip import *


# System parameters
Gamma_31 = 0.1    # Decay rate from |3⟩ to |1⟩
Gamma_32 = 0.05    # Decay rate from |3⟩ to |2⟩

#energies
w1 = 1
w2 = 1.5
w3 = 5

# Operators
sigma_21 = basis(3, 1) * basis(3, 0).dag()    # |2⟩ <-> |1⟩ transition
sigma_31 = basis(3, 2) * basis(3, 0).dag()    # |3⟩ <-> |1⟩ transition
sigma_32 = basis(3, 2) * basis(3, 1).dag()    # |3⟩ <-> |2⟩ transition
sigma33 = basis(3, 2) * basis(3, 2).dag()
sigma11 = basis(3, 0) * basis(3, 0).dag()
sigma22 = basis(3, 1) * basis(3, 1).dag()

# Hamiltonian
H = w3 * sigma33 + w2 * sigma22 + w1 * sigma11

# Collapse operators
c_ops = [np.sqrt(Gamma_31) * sigma_31.dag(), np.sqrt(Gamma_32) * sigma_32.dag()]

# Initial state
rho0 = basis(3, 2) * basis(3, 2).dag()    # Starting in ground state

# Time points
```

```
tlist = np.linspace(0, 100, 100)

# Solve master equation
result = mesolve(H, rho0, tlist, c_ops, [sigma33, sigma11, sigma22, sigma_31,␣
 ↪sigma_21, sigma_32])

# Plot population dynamics
plt.plot(tlist, result.expect[0], label=r'$\sigma_{ee}$')
plt.plot(tlist, result.expect[1], label=r'$\sigma_{g1g1}$')
plt.plot(tlist, result.expect[2], label=r'$\sigma_{g2g2}$')
plt.plot(tlist, result.expect[3], label=r'$\sigma_{eg1}$')
plt.plot(tlist, result.expect[4], label=r'$\sigma_{g2g1}$')
plt.plot(tlist, result.expect[5], label=r'$\sigma_{eg2}$')


#plt.plot(tlist, result.expect[0] + result.expect[1] + result.expect[2],␣
 ↪label='|sum⟩')

plt.xlabel('Time')
plt.ylabel('Population')
plt.legend()
plt.title('Population Dynamics in a Three-Level Raman System')
plt.show()
```



Write now the full master equation of a coupled three-level lambda / Raman system, plus the spontaneous emission terms. Is it reasonable to work in Rotating-Wave approximation? Highlight the RWA validity regime.

The hamiltonian H_A in the lab can be written as:

$$\begin{aligned} H_A^{lab} = & -\hbar\omega_{eg1}|g_1\rangle\langle g_1| - \hbar\omega_{eg2}|g_2\rangle\langle g_2| \\ & -\hbar\left[|e\rangle\langle g_1|\left(\Omega_1\cos(\omega_1 t) + \tilde{\Omega}_1\cos(\omega_2 t)\right) + \right. \\ & \left. +|e\rangle\langle g_2|\left(\Omega_2\cos(\omega_2 t) + \tilde{\Omega}_2\cos(\omega_1 t)\right) + \text{h.c.}\right] \end{aligned}$$

$$\begin{aligned} \Omega_1 &= \frac{\vec{d}_{eg_1}\cdot\vec{\mathcal{E}}_1}{\hbar} \quad \Omega_2 = \frac{\vec{d}_{eg_2}\cdot\vec{\mathcal{E}}_2}{\hbar} \\ \tilde{\Omega}_1 &= \frac{\vec{d}_{eg_2}\cdot\vec{\mathcal{E}}_1}{\hbar} \quad \tilde{\Omega}_2 = \frac{\vec{d}_{eg_1}\cdot\vec{\mathcal{E}}_2}{\hbar} \end{aligned}$$

In the RWA approximation, where we have the following conditions:

$$\Gamma_{eg1},\ \Gamma_{eg2},\ \Omega_1,\ \Omega_2,\ \tilde{\Omega}_1,\ \tilde{\Omega}_2, |\delta_1|,\ |\delta_2| \ \ll\ \omega_1,\ \omega_2,\ \omega_1+\omega_2,\ |\omega_1-\omega_2|,$$

where $\delta_1 \equiv \omega_{eg_1} - \omega_1$, and $\delta_2 \equiv \omega_{eg_2} - \omega_2$.

The Hamiltonian in the rotating frame after applying the RWA became:

$$H_A = \hbar \begin{pmatrix} 0 & \left|\frac{\Omega_1}{2}\right| & 0 \\ \left|\frac{\Omega_1}{2}\right| & \delta_1 & \left|\frac{\Omega_2}{2}\right| \\ 0 & \left|\frac{\Omega_2}{2}\right| & \Delta \end{pmatrix}$$

where $\Delta \equiv \delta_1 - \delta_2$ and we use as basis $|g_1\rangle, |e\rangle, |g_2\rangle$.

Note that since we have spontaneus emission we have to consider also that energy scale for applying the RWA. In the code below we verify that this condition is needed.

Here we simulate the system in rotating frame but without RWA, with large and small decay rates Γ_{eg1}, Γ_{eg2} copmpared to $|\omega_1 - \omega_2|$. However we still work with these conditions:

$$\Omega_1,\ \Omega_2,\ \tilde{\Omega}_1,\ \tilde{\Omega}_2, |\delta_1|,\ |\delta_2| \ \ll\ \omega_1,\ \omega_2,\ \omega_1 + \omega_2,\ |\omega_1 - \omega_2|.$$

[29]:
```
import numpy as np
import matplotlib.pyplot as plt
from qutip import *

# System parameters
Gamma_eg1 = 200   # Decay rate from |3⟩ to |1⟩
Gamma_eg2 = 200   # Decay rate from |3⟩ to |2⟩


#energies
weg1 = 200
weg2 = 100


Omega_1 = 1
Omega_1not = 0
Omega_2 = 1
Omega_2not = 0
omega_1 = 190
omega_2 = 90

# Operators
sigma_21 = basis(3, 1) * basis(3, 0).dag()   # |2⟩ <-> |1⟩ transition
sigma_31 = basis(3, 2) * basis(3, 0).dag()   # |3⟩ <-> |1⟩ transition
sigma_32 = basis(3, 2) * basis(3, 1).dag()   # |3⟩ <-> |2⟩ transition
sigma33 = basis(3, 2) * basis(3, 2).dag()
sigma11 = basis(3, 0) * basis(3, 0).dag()
sigma22 = basis(3, 1) * basis(3, 1).dag()
```

```
# Define time-dependent functions for the Hamiltonian coefficients
def H_eg1(t, args):
    return np.exp(1j*(omega_1*t))*(Omega_1 * np.cos(omega_1 * t) + Omega_1not *␣
 ↪np.cos(omega_2 * t ))

def H_g1e(t, args):
    return np.exp(-1j*(omega_1*t))*(Omega_1 * np.cos(omega_1 * t) + Omega_1not *␣
 ↪np.cos(omega_2 * t ))

def H_eg2(t, args):
    return np.exp(-1j*(omega_2*t))*(Omega_2 * np.cos(omega_2 * t) + Omega_2not *␣
 ↪np.cos(omega_1 * t ))

def H_g2e(t, args):
    return np.exp(1j*(omega_2*t))*(Omega_2 * np.cos(omega_2 * t) + Omega_2not *␣
 ↪np.cos(omega_1 * t ))


# Hamiltonian
H0 = (weg1-omega_1) * sigma11 + (weg2-omega_2) * sigma33


# Define the time-dependent Hamiltonian
H_t = [H0, [sigma_21, H_eg1],[sigma_21.dag(), H_g1e],[sigma_32.dag(), H_g2e],␣
 ↪[sigma_32, H_eg2]]


# Collapse operators
c_ops = [np.sqrt(Gamma_eg1) * sigma_21.dag(), np.sqrt(Gamma_eg2) * sigma_32]

# Initial state
rho0 = sigma11  # Starting in ground state

# Time points
tlist = np.linspace(0, 3000, 30000)

# Solve master equation
result_noRWA = mesolve(H_t, rho0, tlist, c_ops, [sigma33, sigma11, sigma22,␣
 ↪sigma_31, sigma_21, sigma_32])

Gamma_eg1 = 0.01   # Decay rate from |3⟩ to |1⟩
Gamma_eg2 = 0.01   # Decay rate from |3⟩ to |2⟩

tlist = np.linspace(0, 1000, 10000)

c_ops = [np.sqrt(Gamma_eg1) * sigma_21.dag(), np.sqrt(Gamma_eg2) * sigma_32]
```

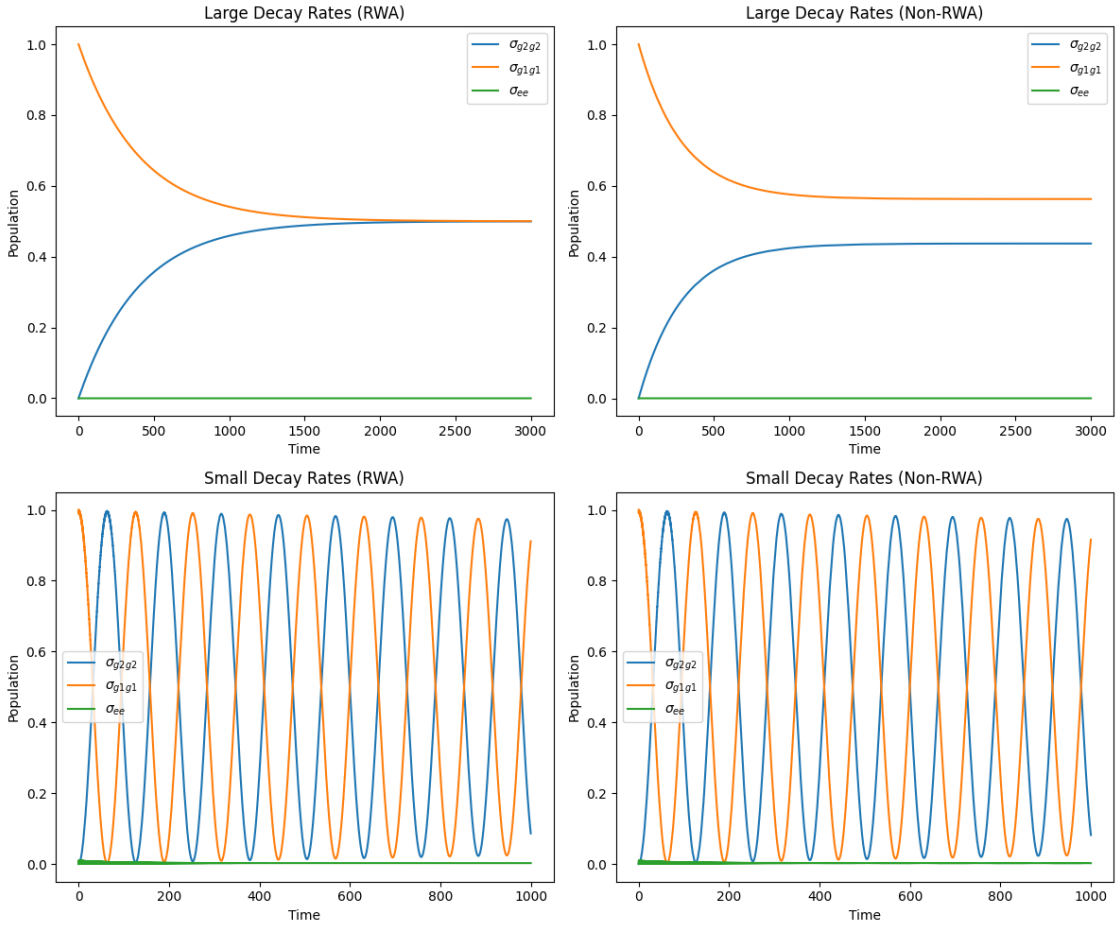
```
result2 = mesolve(H_t, rho0, tlist, c_ops, [sigma33, sigma11, sigma22, sigma_31,␣
↪sigma_21, sigma_32])
```

Here we simulate the system in rotating frame in RWA regime with large and small decay rates Γ_{eg1}, Γ_{eg2} copmpared to light $|\omega_1 - \omega_2|$.

[30]:
```
Gamma_eg1 = 200   # Decay rate from |3⟩ to |1⟩
Gamma_eg2 = 200   # Decay rate from |3⟩ to |2⟩


delta1 = 10
delta2 = 10
Omega_1 = 1
Omega_2 = 1

# Hamiltonian
H = delta1 * sigma11 + delta2 * sigma33  + Omega_1/2 * sigma_21 + Omega_2/2 *␣
↪sigma_32 + Omega_1/2 * sigma_21.dag() + Omega_2/2 * sigma_32.dag()

# Collapse operator
c_ops = [np.sqrt(Gamma_eg1) * sigma_21.dag(), np.sqrt(Gamma_eg2) * sigma_32]

# Initial state
rho0 = sigma11  # Starting in ground state

# Time points
tlist = np.linspace(0, 3000, 30000)

# Solve master equation
result = mesolve(H, rho0, tlist, c_ops, [sigma33, sigma11, sigma22, sigma_31,␣
↪sigma_21, sigma_32])

Gamma_eg1 = 0.01   # Decay rate from |3⟩ to |1⟩
Gamma_eg2 = 0.01   # Decay rate from |3⟩ to |2⟩

tlist = np.linspace(0, 1000, 10000)

c_ops = [np.sqrt(Gamma_eg1) * sigma_21.dag(), np.sqrt(Gamma_eg2) * sigma_32]

result1 = mesolve(H, rho0, tlist, c_ops, [sigma33, sigma11, sigma22, sigma_31,␣
↪sigma_21, sigma_32])
```

[33]:
```
# Create a 2x2 grid of subplots
fig, axes = plt.subplots(2, 2, figsize=(12, 10))
tlist = np.linspace(0, 1000, 10000)
tlist1 = np.linspace(0, 3000, 30000)
```

```
# Left-top plot: Population dynamics with "result"
axes[0, 0].plot(tlist1, result.expect[0], label=r'$\sigma_{g2g2}$')
axes[0, 0].plot(tlist1, result.expect[1], label=r'$\sigma_{g1g1}$')
axes[0, 0].plot(tlist1, result.expect[2], label=r'$\sigma_{ee}$')
axes[0, 0].set_xlabel('Time')
axes[0, 0].set_ylabel('Population')
axes[0, 0].legend()
axes[0, 0].set_title('Large Decay Rates (RWA)')

# Right-top plot: Population dynamics with "result_noRWA"
axes[0, 1].plot(tlist1, result_noRWA.expect[0], label=r'$\sigma_{g2g2}$')
axes[0, 1].plot(tlist1, result_noRWA.expect[1], label=r'$\sigma_{g1g1}$')
axes[0, 1].plot(tlist1, result_noRWA.expect[2], label=r'$\sigma_{ee}$')
axes[0, 1].set_xlabel('Time')
axes[0, 1].set_ylabel('Population')
axes[0, 1].legend()
axes[0, 1].set_title('Large Decay Rates (Non-RWA)')

# Left-bottom plot: Population dynamics with "result1"
axes[1, 0].plot(tlist, result1.expect[0], label=r'$\sigma_{g2g2}$')
axes[1, 0].plot(tlist, result1.expect[1], label=r'$\sigma_{g1g1}$')
axes[1, 0].plot(tlist, result1.expect[2], label=r'$\sigma_{ee}$')
axes[1, 0].set_xlabel('Time')
axes[1, 0].set_ylabel('Population')
axes[1, 0].legend()
axes[1, 0].set_title('Small Decay Rates (RWA)')

# Right-bottom plot: Population dynamics with "result2"
axes[1, 1].plot(tlist, result2.expect[0], label=r'$\sigma_{g2g2}$')
axes[1, 1].plot(tlist, result2.expect[1], label=r'$\sigma_{g1g1}$')
axes[1, 1].plot(tlist, result2.expect[2], label=r'$\sigma_{ee}$')
axes[1, 1].set_xlabel('Time')
axes[1, 1].set_ylabel('Population')
axes[1, 1].legend()
axes[1, 1].set_title('Small Decay Rates (Non-RWA)')

# Adjust layout and display the plots
plt.tight_layout()
plt.show()
```



We can see that when $\Gamma_{eg1},\ \Gamma_{eg2} \approx |\omega_1 - \omega_2|$ the RWA approximation is not valid, since we have different dynamics with same parameters.

If $\Gamma_{eg1},\ \Gamma_{eg2} \ll |\omega_1 - \omega_2|$ the RWA approximation is valid and we have oscillations between the two ground states.

Integrate the master equation dynamics in time, and find the appropriate regimes where the emission has negligible impact. Compare with your intuition and interpret the result.

Now writing the master equation we get:

$$\partial_t \rho = -\frac{i}{\hbar}\left[H_A, \rho\right] + \Gamma_{eg1}\mathcal{D}[\sigma_{-g1}]\rho + \Gamma_{eg2}\mathcal{D}[\sigma_{-g2}]\rho$$

And writing it explicetely in the ordered basis $\{|g_1\rangle, |e\rangle, |g_2\rangle\}$:

$$\dot{\rho}_{11}(t) = -\frac{i}{2}\left(\Omega_1^*\rho_{21}(t) - \Omega_1\rho_{12}(t)\right) + \Gamma_{21}\rho_{22}(t)$$
$$\dot{\rho}_{33}(t) = -i\frac{\Omega_2}{2}\left(\rho_{23}(t) - \rho_{32}(t)\right) + \Gamma_{23}\rho_{22}(t)$$
$$\dot{\rho}_{22}(t) = -\frac{i}{2}\left(\Omega_1\rho_{12}(t) - \Omega_1^*\rho_{21}(t)\right) - \frac{i}{2}\left(\Omega_2\rho_{32}(t) - \Omega_2^*\rho_{23}(t)\right) - (\Gamma_{21} + \Gamma_{23})\rho_{22}(t)$$
$$\dot{\rho}_{13}(t) = i(\delta_1 - \Delta)\rho_{13} - i\frac{\Omega_1^*}{2}\rho_{23} + i\frac{\Omega_2}{2}\rho_{12}$$
$$\dot{\rho}_{12}(t) = -i\delta_1\rho_{12} - i\frac{\Omega_1^*}{2}(\rho_{22} - \rho_{11}) + i\frac{\Omega_2^*}{2}\rho_{13} - \frac{\Gamma_{21} + \Gamma_{23}}{2}\rho_{12}$$
$$\dot{\rho}_{23}(t) = -i\Delta\rho_{23} + i\frac{\Omega_2}{2}(\rho_{22} - \rho_{33}) - i\frac{\Omega_1}{2}\rho_{13} - \frac{\Gamma_{21} + \Gamma_{23}}{2}\rho_{23}$$

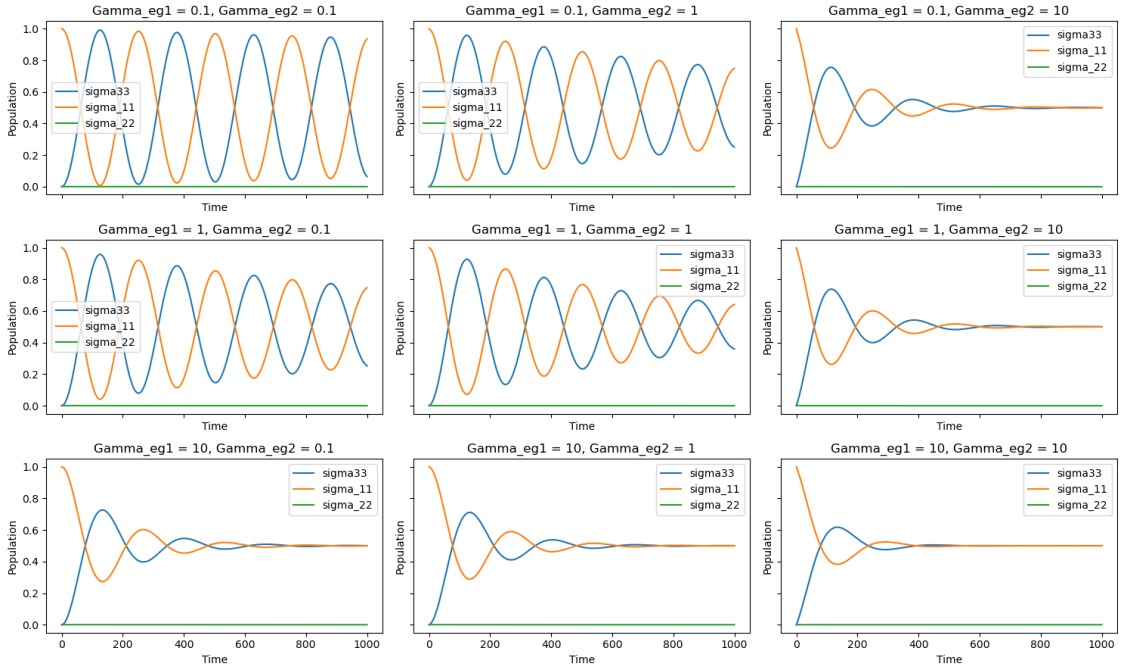
If we neglet for a moment the decays and consider $\Omega_1 = \Omega_2 = \Omega << \delta\_1$, $\Delta = 0$, the parameter that determine the frequency of the oscillation between the two ground states is $\propto \frac{\Omega^2}{\delta_1}$. Indeed this can be easly shown in the resolution of a Lambda system where we consider the terms proportional to Ω as perturbations.

Now we study the dynimics for different decay rates:

[332]:
```
# Define different values for Gamma_eg1 and Gamma_eg2
gamma_eg1_values = [0.1, 1, 10]
gamma_eg2_values = [0.1, 1, 10]

delta1 = 10
Delta = 0
Omega_1 = 1
Omega_2 = 1

# Operators
sigma_21 = basis(3, 1) * basis(3, 0).dag()    # |2⟩ <-> |1⟩ transition
sigma_31 = basis(3, 2) * basis(3, 0).dag()    # |3⟩ <-> |1⟩ transition
sigma_32 = basis(3, 2) * basis(3, 1).dag()    # |3⟩ <-> |2⟩ transition
sigma33 = basis(3, 2) * basis(3, 2).dag()
sigma11 = basis(3, 0) * basis(3, 0).dag()
sigma22 = basis(3, 1) * basis(3, 1).dag()

# Create a grid of plots
fig, axs = plt.subplots(len(gamma_eg1_values), len(gamma_eg2_values),␣
↪figsize=(15, 10), sharex=True, sharey=True)

# Iterate through different parameter values and create plots
for i, Gamma_eg1 in enumerate(gamma_eg1_values):
    for j, Gamma_eg2 in enumerate(gamma_eg2_values):
        # Your simulation and result calculation here
```

```
        # Hamiltonian
        H = delta1 * sigma22 + Delta * sigma33  + Omega_1/2 * sigma_21 + Omega_2/
↪2 * sigma_32
        H = H + H.dag()

        # Collapse operator
        c_ops = [np.sqrt(Gamma_eg1) * sigma_21.dag(), np.sqrt(Gamma_eg2) *␣
↪sigma_32]

        # Initial state
        rho0 = basis(3, 0)  # Starting in ground state
        #psi0 = basis(3,0)

        # Time points
        tlist = np.linspace(0, 1000, 1000)

        #result = sesolve(H, psi0, tlist, [sigma33, sigma11, sigma22])

        # Solve master equation
        result = mesolve(H, rho0, tlist, c_ops, [sigma33, sigma11, sigma22,␣
↪sigma_31, sigma_21, sigma_32])

        # Plot population dynamics on the current subplot
        axs[i, j].plot(tlist, result.expect[0], label='sigma33')
        axs[i, j].plot(tlist, result.expect[1], label='sigma_11')
        axs[i, j].plot(tlist, result.expect[2], label='sigma_22')
        # ...

        # Set labels and title for the subplot
        axs[i, j].set_xlabel('Time')
        axs[i, j].set_ylabel('Population')
        axs[i, j].set_title(f'Gamma_eg1 = {Gamma_eg1}, Gamma_eg2 = {Gamma_eg2}')
        axs[i, j].legend()

# Adjust layout and show the plot
plt.suptitle(fr"$\frac{{\Omega_1^2}}{{\delta_1}} = {Omega_1**2 / delta1},␣
 ↪\delta_1 = {delta1}, \Omega_1 = {Omega_1}$", fontsize=32)
plt.tight_layout()
plt.show()
```

$$\frac{\Omega_1^2}{\delta_1} = 0.1, \delta_1 = 10, \Omega_1 = 1$$



As we can see above, the emission has negligible impact when $\Gamma_{eg1}, \Gamma_{eg2} \leq \frac{\Omega_1^2}{\delta_1}$. This is reasonable because $cost * \frac{\Omega_1^2}{\delta_1}$ is approximately the frequency of the oscillations between the two population and then the energy scale without spontaneus emission.

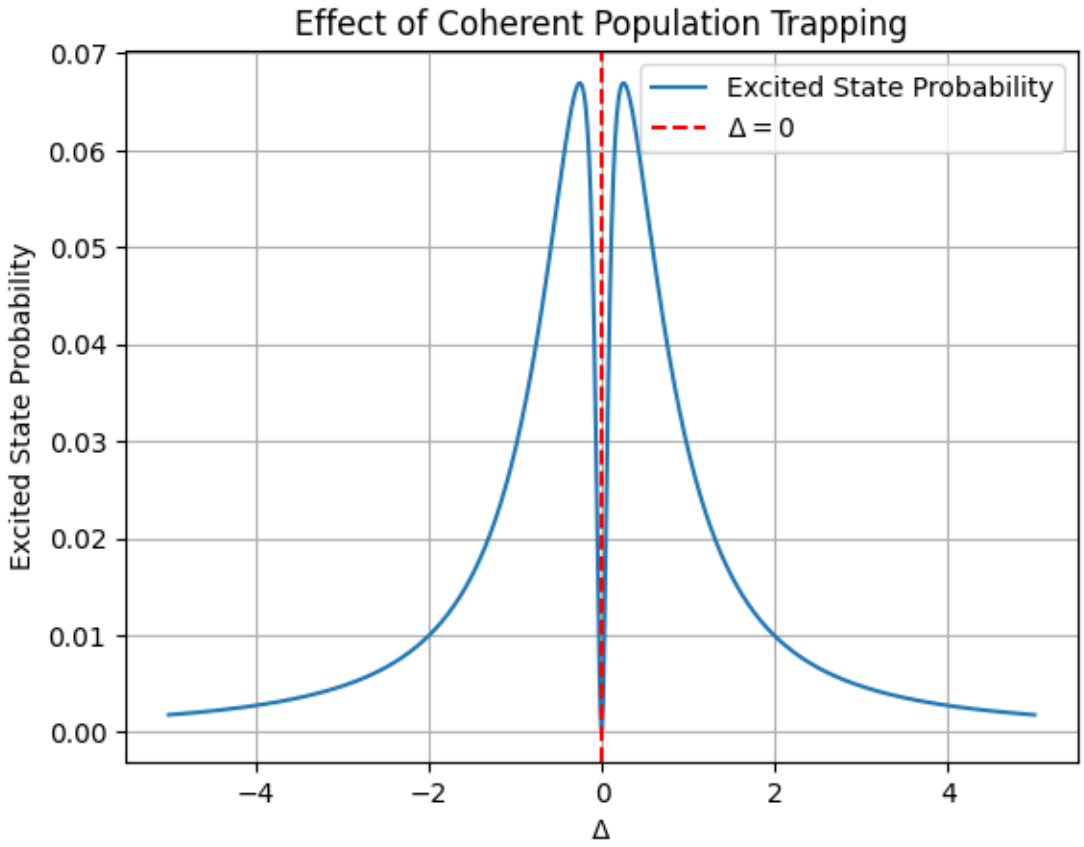
One interesting case of this system is the phenomenon known as coherent population trapping (CPT). CPT is a quantum interference effect that arises in multi-level atomic systems, leading to the suppression of absorption or fluorescence under specific conditions. The atomic system becomes trapped in a coherent superposition of states, resulting in a dark state that is decoupled from the laser fields. This phenomenon occurs due to the interaction of dipole moments associated with the two transitions, leading to either constructive or destructive interference. When destructive interference takes place, the atom does not emit light, essentially becoming disconnected from the surrounding field. Conversely, if there's constructive interference, the spontaneous emission disrupts the dipole phase alignment until the interference becomes entirely destructive.

In our context, coherent population trapping occurs when $\Delta = 0$, $\delta_1 = 0$ and $\Gamma_{eg1} = \Gamma_{eg2}$. We now simulate this to show that the probability of the excited state collapse when both laser are have the same detuning.

[2]:
```
import numpy as np
import matplotlib.pyplot as plt
from qutip import *

# System parameters
Gamma_eg1 = 1    # Decay rate from |3⟩ to |1⟩
```

```
Gamma_eg2 = 1  # Decay rate from |3⟩ to |2⟩

#energies
delta1 = 0

Omega_1 = 0.6
Omega_2 = 0.6

# Operators
sigma_21 = basis(3, 1) * basis(3, 0).dag()    # |2⟩ <-> |1⟩ transition
sigma_31 = basis(3, 2) * basis(3, 0).dag()    # |3⟩ <-> |1⟩ transition
sigma_32 = basis(3, 2) * basis(3, 1).dag()    # |3⟩ <-> |2⟩ transition
sigma33 = basis(3, 2) * basis(3, 2).dag()
sigma11 = basis(3, 0) * basis(3, 0).dag()
sigma22 = basis(3, 1) * basis(3, 1).dag()

p = []
d2 = np.linspace(-5, 5, 1000)
for Delta in d2:
    # Hamiltonian
    H = delta1 * sigma22 + Delta * sigma33  + Omega_1/2 * sigma_21 + Omega_2/2 *␣
 ↪sigma_32
    H = H + H.dag()

    # Collapse operator
    c_ops = [np.sqrt(Gamma_eg1) * sigma_21.dag(), np.sqrt(Gamma_eg2) * sigma_32]

    # Initial state
    rho0 = basis(3, 0)  # Starting in ground state
    #psi0 = basis(3,0)

    # Time points
    tlist = np.linspace(0, 1000, 1000)

    #result = sesolve(H, psi0, tlist, [sigma33, sigma11, sigma22])

    # Solve master equation
    result = mesolve(H, rho0, tlist, c_ops, [sigma33, sigma11, sigma22,␣
 ↪sigma_31, sigma_21, sigma_32])

    p.append(result.expect[2][999])

# Plotting
plt.plot(d2, p, label='Excited State Probability')
plt.axvline(x=0, color='r', linestyle='--', label=r'$\Delta = 0$')
plt.xlabel(r'$\Delta$')
plt.ylabel('Excited State Probability')
```

```
plt.title('Effect of Coherent Population Trapping')
plt.legend()
plt.grid()

plt.show()
```



Let us now go beyond simple spontaneous emission. Repeat the calculation of line broadening for a two-level atom (energy separation ωeg), but this time, instead of using a photon vacuum (zero temperature) we use a laser state (coherent state α at a given mode k = ωL/c, λ, vacuum elsewhere). Interpret your result in terms of stimulated emission/absorption.

Going back to the two level case we have :

$$\begin{aligned} H_{System} &= H_{System} \otimes \mathbb{I}_B, \\ H_{Bath} &= \mathbb{I}_S \otimes H_{Bath}, \\ H_{\text{int}} &= \sum_{\alpha} S_\alpha \otimes B_\alpha \end{aligned}$$

with

$$S_{\alpha=0} \equiv \sigma_+ \qquad \text{and} \qquad S_{\alpha=1} \equiv \sigma_-$$

$$\hat{B}_{\alpha=0} \equiv -i\vec{d}_{eg} \cdot \hat{\vec{E}}^{(+)}. \qquad \text{and} \qquad \hat{B}_{\alpha=1} \equiv -i\vec{d}^{*}_{eg} \cdot \hat{\vec{E}}^{(-)}.$$

After the application of the Born approximation we can write the reduced matrix density of the system as:

$$\begin{aligned}\dot{\rho}'_S =& \frac{1}{\hbar^2}\int_0^t dt' \sum_{\alpha,\,\beta}\Big(G_{\alpha\beta}(t-t')\left(S_\alpha(t')\rho'_S(t')S^\dagger_\beta(t) - S^\dagger_\beta(t)S_\alpha(t')\rho'_S(t')\right) + \\ &+ G_{\beta\alpha}(t'-t)\left(S^\dagger_\beta(t)\rho'_S(t')S_\alpha(t') - \rho'_S(t')S_\alpha(t')S^\dagger_\beta(t)\right)\Big),\end{aligned}$$

where

$$G_{\alpha\beta}(t-t') \equiv \langle {B'}^\dagger_\beta(t-t')B_\alpha\rangle_B.$$

Taking $\tau = t - t'$ and considering the indexes α and β in $\{+,-\}$:

$$\begin{aligned}G_{--}(\tau) &= \langle\left(\vec{d}_{eg}\cdot\hat{\vec{E}}^{(+)}(\tau)\right)\left(\vec{d}^{*}_{eg}\cdot\hat{\vec{E}}^{(-)}(0)\right)\rangle_B \\ &\propto \sum_{\vec{k},\lambda}\langle\hat{a}_{\vec{k},\lambda}(\tau)\hat{a}^\dagger_{\vec{k},\lambda}(0)\rangle_B\end{aligned}$$

$$\begin{aligned}G_{++}(\tau) &= \langle\left(\vec{d}_{eg}\cdot\hat{\vec{E}}^{(-)}(\tau)\right)\left(\vec{d}^{*}_{eg}\cdot\hat{\vec{E}}^{(+)}(0)\right)\rangle_B \\ &\propto \sum_{\vec{k},\lambda}\langle\hat{a}^\dagger_{\vec{k},\lambda}(\tau)\hat{a}_{\vec{k},\lambda}(0)\rangle_B\end{aligned}$$

$$\begin{aligned}G_{+-}(\tau) &= \langle\left(\vec{d}_{eg}\cdot\hat{\vec{E}}^{(-)}(\tau)\right)\left(\vec{d}^{*}_{eg}\cdot\hat{\vec{E}}^{(-)}(0)\right)\rangle_B \\ &\propto \sum_{\vec{k},\lambda}\langle\hat{a}^\dagger_{\vec{k},\lambda}(\tau)\hat{a}^\dagger_{\vec{k},\lambda}(0)\rangle_B\end{aligned}$$

$$\begin{aligned}G_{-+}(\tau) &= \langle\left(\vec{d}_{eg}\cdot\hat{\vec{E}}^{(+)}(\tau)\right)\left(\vec{d}^{*}_{eg}\cdot\hat{\vec{E}}^{(+)}(0)\right)\rangle_B \\ &\propto \sum_{\vec{k},\lambda}\langle\hat{a}_{\vec{k},\lambda}(\tau)\hat{a}_{\vec{k},\lambda}(0)\rangle_B\end{aligned}$$

We can neglet the time dependence of ladder operators since we are in the Born approximation.

Averging over a bath at thermodynamic equilibrium G_{-+} and G_{+-} cancel outs. The second term G_{++} result $\propto \sum_{\vec{k},\lambda}(n_{\vec{k},\lambda})$ while the first one $\propto \sum_{\vec{k},\lambda}(1+n_{\vec{k},\lambda})$, thanks to commutation relations. Note that in a photon vacuum at 0 temperture only G_{--} will be different from zero and this lead to spontaneus emission. Consider now the case with a coherent state in the bath for only one wavevector $\vec{k_L}$. Then we can write the average number of photon in the bath for a specific wavevector

as $n_{\vec{k},\lambda}$ and it is different from 0 only when $\vec{k} = \vec{k_L}$. When $\vec{k} = \vec{k_L}$ then $n_{\vec{k_L},\lambda} = |\alpha|^2$, with α index of coherent state. From now on the calculation are very similar to spontaneus emission case. When intergrating on time, $|\alpha|^2$ will survive only if $w_L \equiv c|\vec{k_L}| = \omega_{eg}$ where ω_{eg} is the frequency of the transition in the two level atom.

In the end the G_{--} give rise to $\Gamma(1 + |\alpha|^2)\mathcal{D}[\sigma_-]\rho$, which correspond to the spontaneus emission term added to the stimulated emission term. Furthermore G_{++} give rise to $\Gamma|\alpha|^2\mathcal{D}[\sigma_+]\rho$, which can be interpreted as the absorbtion term.

The final master equation then is:

$$\partial_t \rho = -\frac{i}{\hbar}[H_A, \rho] + \Gamma(1 + |\alpha|^2)\mathcal{D}[\sigma_-]\rho + \Gamma|\alpha|^2\mathcal{D}[\sigma_+]\rho$$

Note that if we turn off the laser in the bath considering $\alpha = 0$ we go back to spontaneus emission case.

(Open ended) Try once again to address the open-system 3-level problem. But this time attempt to add all the contributions (spontaneous emission, stimulated emission and absorption) to the Master Equation.

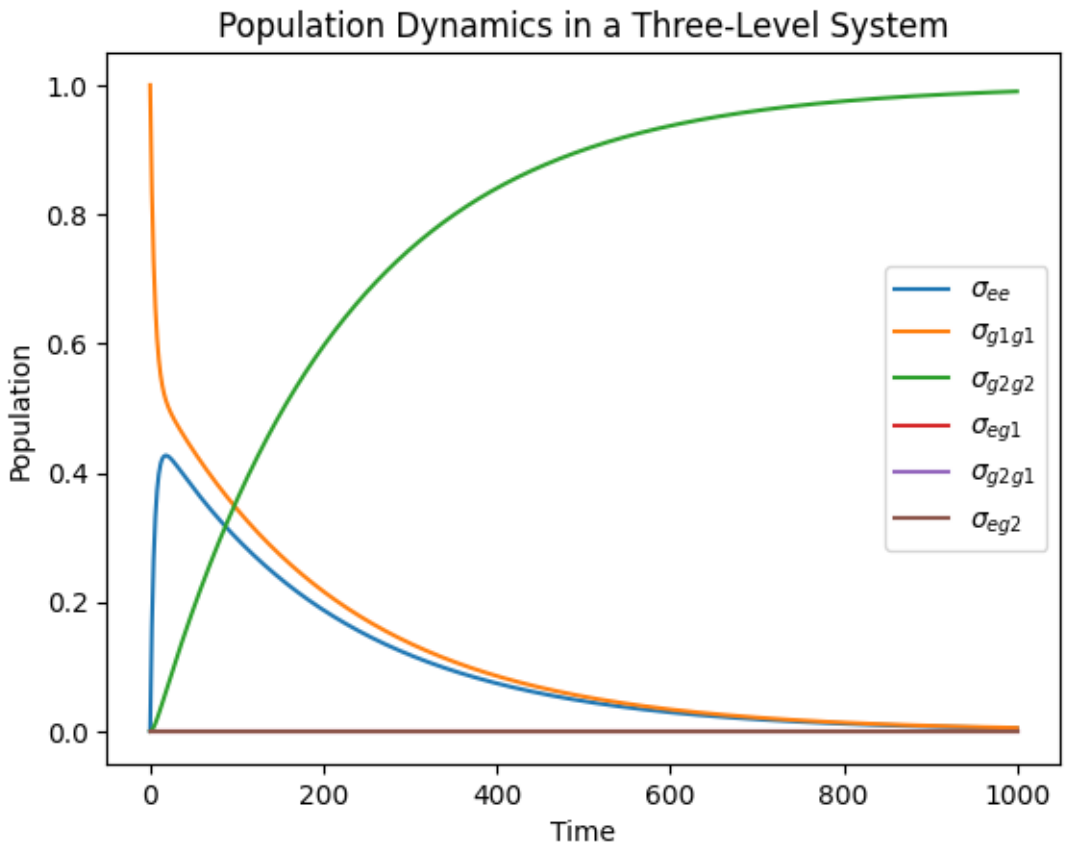
Now it is easy to conciliate this with the three level system we have seen before. We have to consider $H_{\text{int}} = H_{\text{int1}} + H_{\text{int2}}$, where in the i-th term we consider σ operators between the excited state and the i-th ground state, and then for each of the term the derivation is the same of the one of the two level system.

First of all we notice that stimulated emission and absorption give a non-zero contribute only if the coherent state in the bath is at resonance with one of the two transitions.

Lets now take the three level system without lasers coupling and but with a coherent state with 10 photons in the bath resonant with $|g_1\rangle$:

[118]:
```
import numpy as np
import matplotlib.pyplot as plt
from qutip import *


# System parameters
Gamma_eg1 = 0.01 # Decay rate from |3⟩ to |1⟩
Gamma_eg2 = 0.01 # Decay rate from |3⟩ to |2⟩
n = 10 # N average number of photons
Gamma_eg1_ems = Gamma_eg1*(1+n)
Gamma_eg1_abs = Gamma_eg1*(n)



#energies
w1 = 1
w2 = 1.5
w3 = 5
```

```
# Operators
sigma_21 = basis(3, 1) * basis(3, 0).dag()    # |2⟩ <-> |1⟩ transition
sigma_31 = basis(3, 2) * basis(3, 0).dag()    # |3⟩ <-> |1⟩ transition
sigma_32 = basis(3, 2) * basis(3, 1).dag()    # |3⟩ <-> |2⟩ transition
sigma33 = basis(3, 2) * basis(3, 2).dag()
sigma11 = basis(3, 0) * basis(3, 0).dag()
sigma22 = basis(3, 1) * basis(3, 1).dag()

# Hamiltonian
H = w3 * sigma33 + w2 * sigma22 + w1 * sigma11

# Collapse operators
c_ops = [np.sqrt(Gamma_eg1_ems) * sigma_31.dag(), np.sqrt(Gamma_eg2) * sigma_32.
 ↪dag(),np.sqrt(Gamma_eg1_abs) * sigma_31]

# Initial state
#rho0 = basis(3, 2) * basis(3, 2).dag()
rho0 = basis(3, 0) * basis(3, 0).dag()

# Time points
tlist = np.linspace(0, 1000, 1000)

# Solve master equation
result = mesolve(H, rho0, tlist, c_ops, [sigma33, sigma11, sigma22, sigma_31,␣
 ↪sigma_21, sigma_32])

# Plot population dynamics
plt.plot(tlist, result.expect[0], label=r'$\sigma_{ee}$')
plt.plot(tlist, result.expect[1], label=r'$\sigma_{g1g1}$')
plt.plot(tlist, result.expect[2], label=r'$\sigma_{g2g2}$')
plt.plot(tlist, result.expect[3], label=r'$\sigma_{eg1}$')
plt.plot(tlist, result.expect[4], label=r'$\sigma_{g2g1}$')
plt.plot(tlist, result.expect[5], label=r'$\sigma_{eg2}$')


#plt.plot(tlist, result.expect[0] + result.expect[1] + result.expect[2],␣
 ↪label='|sum⟩')

plt.xlabel('Time')
plt.ylabel('Population')
plt.legend()
plt.title('Population Dynamics in a Three-Level System')
plt.show()
```

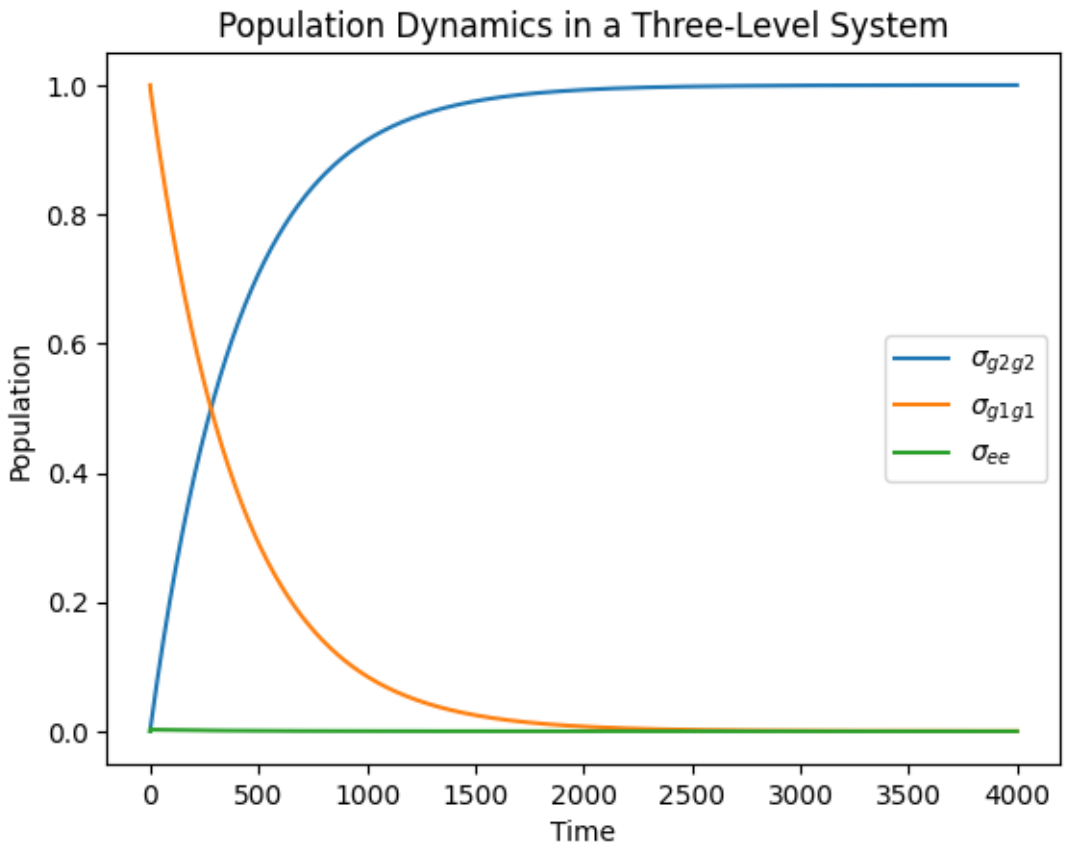


As we can see we found a population inversion.

Now we remove the coherent state in the bath by setting $n = 0$ and we study the evolution with only one laser coupled to $|g_1\rangle$ (in the hamiltonian) :

[119]:
```
#System parameters
Gamma_eg1 = 1 # Decay rate from |3⟩ to |1⟩
Gamma_eg2 = 1 # Decay rate from |3⟩ to |2⟩
n = 0 # N average number of photons
Gamma_eg1_ems = Gamma_eg1*(1+n)
Gamma_eg1_abs = Gamma_eg1*(n)

delta1 = 10
delta2 = 0
Omega_1 = 1
Omega_2 = 0

# Hamiltonian
H = delta1 * sigma11 + delta2 * sigma33  + Omega_1/2 * sigma_21 + Omega_2/2 *␣
 ↪sigma_32 + Omega_1/2 * sigma_21.dag() + Omega_2/2 * sigma_32.dag()
```

```
# Collapse operator
c_ops = [np.sqrt(Gamma_eg1_ems) * sigma_21.dag(), np.sqrt(Gamma_eg2) * sigma_32,␣
 ↪np.sqrt(Gamma_eg1_abs) * sigma_21]

# Initial state
rho0 = sigma11  # Starting in ground state

# Time points
tlist = np.linspace(0, 4000, 4000)

# Solve master equation
result = mesolve(H, rho0, tlist, c_ops, [sigma33, sigma11, sigma22, sigma_31,␣
 ↪sigma_21, sigma_32])

# Plot population dynamics
plt.plot(tlist, result.expect[0], label=r'$\sigma_{g2g2}$')
plt.plot(tlist, result.expect[1], label=r'$\sigma_{g1g1}$')
plt.plot(tlist, result.expect[2], label=r'$\sigma_{ee}$')
#plt.plot(tlist, result.expect[3], label=r'$\sigma_{eg1}$')
#plt.plot(tlist, result.expect[4], label=r'$\sigma_{g2g1}$')
#plt.plot(tlist, result.expect[5], label=r'$\sigma_{eg2}$')


#plt.plot(tlist, result.expect[0] + result.expect[1] + result.expect[2],␣
 ↪label='|sum⟩')

plt.xlabel('Time')
plt.ylabel('Population')
plt.legend()
plt.title('Population Dynamics in a Three-Level System')
plt.show()
```



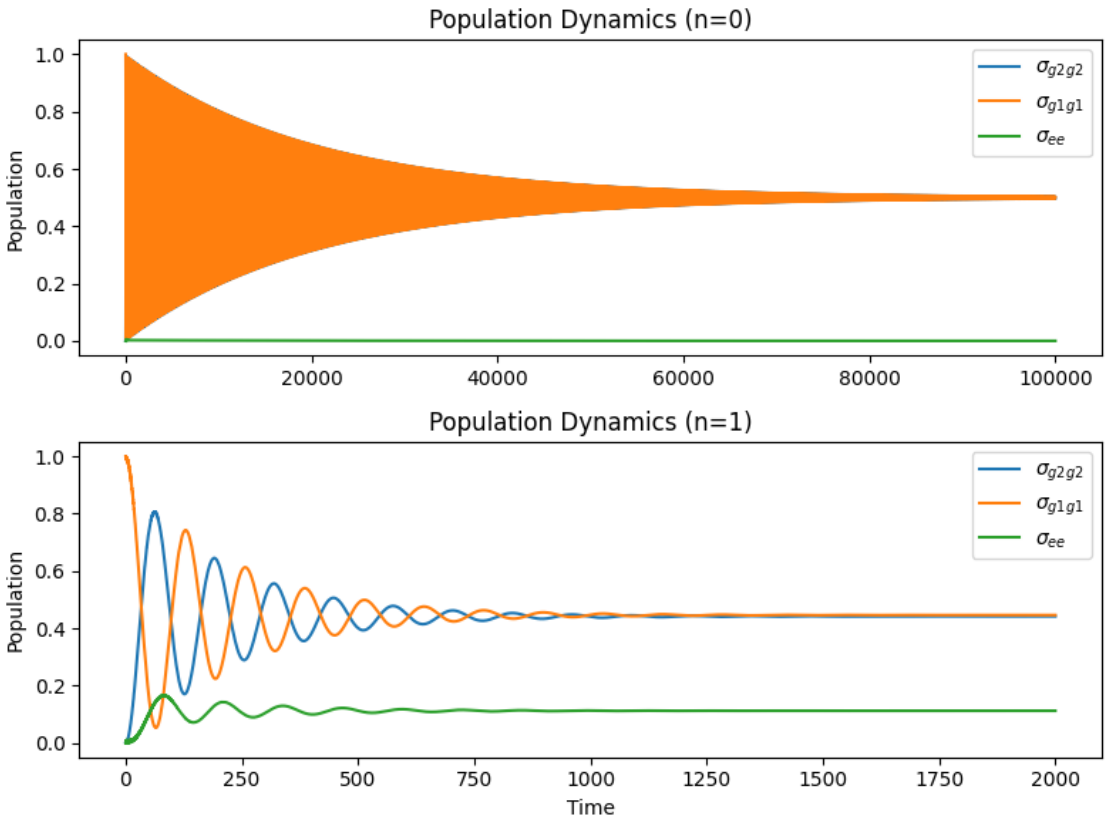
As we can see the final steady state is the same of the one above with a coherent state in the bath, indeed we are describing the same physical process.

Population inversion can be understood intuitively by considering that the excited state decays through both channels, yet only one of the ground states can be re-excited. Over time, the system tends to settle into the other ground state due to this preferential re-excitation process.

Lets now consider a Raman Coupled system with also a coherent state in the bath coupled with only a ground state:

[138]:
```
# System parameters
Gamma_eg1 = 0.01  # Decay rate from |3⟩ to |1⟩
Gamma_eg2 = 0.01  # Decay rate from |3⟩ to |2⟩
n_values = [0, 1]  # N average number of photons

delta1 = 10
delta2 = 10
Omega_1 = 1
Omega_2 = 1
```

```
# Time points for different plots
tlist_n0 = np.linspace(0, 100000, 10000)
tlist_n1 = np.linspace(0, 2000, 10000)  # Different time range for n=1 plot


# Initialize the figure and subplots
fig, axes = plt.subplots(len(n_values), 1, figsize=(8, 6))

# Create the collapse operators
c_ops = [np.sqrt(Gamma_eg1) * sigma_21.dag()]

for i, n in enumerate(n_values):
    # Update collapse operators for different n
    Gamma_eg2_ems = Gamma_eg2 * (1 + n)
    Gamma_eg2_abs = Gamma_eg2 * n
    c_ops.append(np.sqrt(Gamma_eg2_ems) * sigma_32)
    c_ops.append(np.sqrt(Gamma_eg2_abs) * sigma_32.dag())

    if n == 0:
        tlist = tlist_n0
    else:
        tlist = tlist_n1

    # Hamiltonian
    H = delta1 * sigma11 + delta2 * sigma33 + Omega_1 / 2 * sigma_21 + Omega_2 /␣
 ↪2 * sigma_32 + Omega_1 / 2 * sigma_21.dag() + Omega_2 / 2 * sigma_32.dag()

    # Initial state
    rho0 = sigma11  # Starting in ground state


    # Solve master equation
    result = mesolve(H, rho0, tlist, c_ops, [sigma33, sigma11, sigma22,␣
 ↪sigma_21, sigma_32])

    # Plot population dynamics
    axes[i].plot(tlist, result.expect[0], label=r'$\sigma_{g2g2}$')
    axes[i].plot(tlist, result.expect[1], label=r'$\sigma_{g1g1}$')
    axes[i].plot(tlist, result.expect[2], label=r'$\sigma_{ee}$')
    axes[i].set_ylabel('Population')
    axes[i].set_title(f'Population Dynamics (n={n})')
    axes[i].legend()


axes[-1].set_xlabel('Time')
plt.tight_layout()
plt.show()
```



If $n = 0$ we have only spontaneus emission and we find what already discussed. However if we change n we found that the stability is reached much faster and that the population of the excited state is not zero.