

Open Quantum Systems: from qubits to quantum biology

Open Quantum Systems: from qubits to quantum biology

Part 2:

Deriving approximate noise models: Lindblad master equation

Non-perturbative methods: HEOM, pseudo-modes and reaction coordinates

Outline:

Deriving approximate noise models: Lindblad master equation

- Recap Motivation: The need to model noise in quantum computing and quantum biology

Outline:

Deriving approximate noise models: Lindblad master equation

- Recap Motivation: The need to model noise in quantum computing and quantum biology
- Modelling noise: The Lindblad Master equation; a derivation, and how to evaluate using QuTiP
- FMO example

Non-perturbative methods: HEOM, pseudo-modes and reaction coordinates

- Beyond the Master Equation: Exact numerical methods for bosonic baths
 - The Hierarchy Equations of Motion
 - The reaction coordinate
 - Pseudo-modes
- Comparing different methods
- Evaluating efficiency
- Conclusions

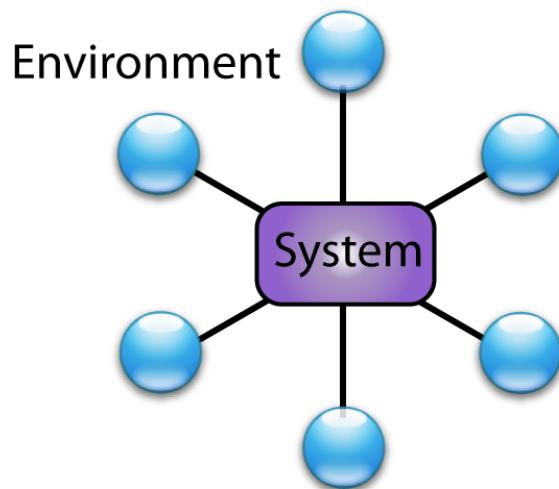
Motivation recap

Motivation recap

Every quantum system inevitably interacts with its environment.

In general, it is impossible for us to model the whole dynamics of the system and environment.

So the goal of the theory of open quantum systems is to construct minimal solvable models and methods which account for the influence of the environment on the system.



The motivation for the study of open quantum systems arises from:

1. **Describing real experiments** involving quantum degrees of freedom in contact with an environment, and real measurements, e.g.,
2. Understanding how the classical world arises from the quantum one.
3. Often we want to preserve quantum effects (quantum computation). Noise models are important for error correction, etc.

Quantum optics

Many of the tools we use in open-quantum systems were developed to describe interactions between atoms and light in cavities, as typified by the experiments of Serge Haroche (Nobel Prize in Physics 2012)

A B

Brune et al, 1996.

Quantum computing & quantum technologies

Artificial atoms are now used as the basis of quantum technologies, but are also subject to various noise sources.

A B

First observation of coherent oscillations: Nakamura, Pashkin, Tsai (NEC) 1999 Basis of modern quantum computing schemes (Photo from IBM)

Photosynthetic light-harvesting

10 years ago similar apparent "coherent oscillations" were observed in experiments on the Fenna-Matthews-Olsen complex from green sulphur bacteria (Nature Physics, Engel et al. 2007)

A

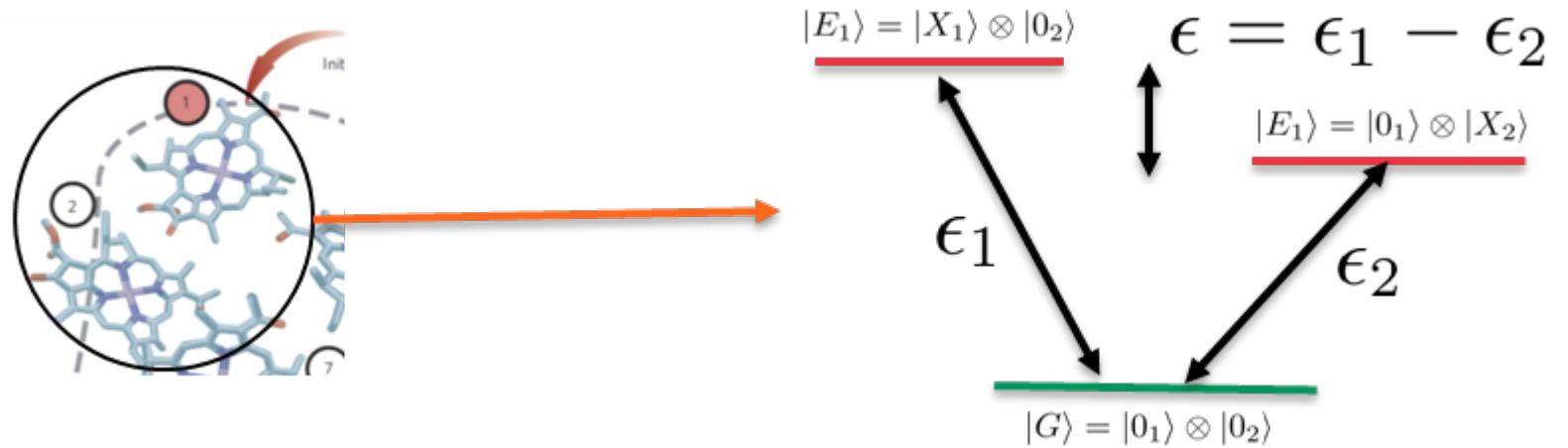
B

Panitchayangkoon et al. PNAS (2010).

Does this mean quantum coherence is important for light-harvesting?

A basic model of photosynthetic light-harvesting

How do we make a (very) minimal model of a single excitation being transported through the complex? Lets focus on just two sites (chromophores)



We write the states as either both chromophores unexcited, or an electronic excitation on either chromophore.

We ignore double excitation.

If we focus on a single excitation subspace (let us ignore the ground-state), we can just describe the two molecules as an effective spin:

$$\begin{array}{c} |E_1\rangle = |X_1\rangle \otimes |0_2\rangle \\ \hline \updownarrow \quad \epsilon = \epsilon_1 - \epsilon_2 \\ \hline |E_1\rangle = |0_1\rangle \otimes |X_2\rangle \end{array}$$

We can write the energy of these two levels in the form of a Hamiltonian, using the Pauli spin matrix:

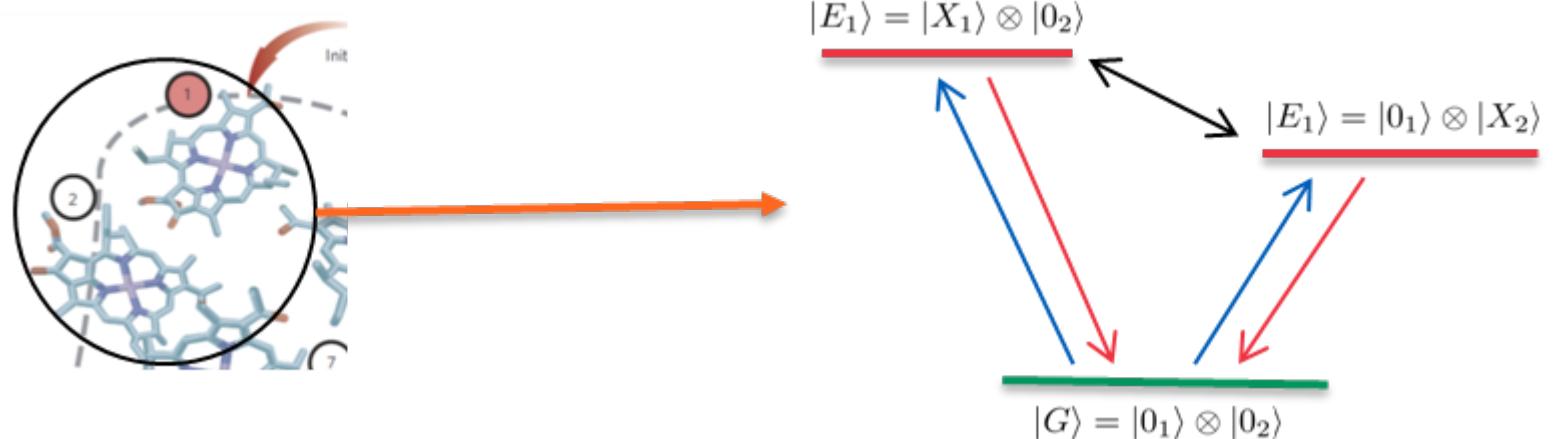
$$H_E = \frac{\epsilon}{2} \sigma_z, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$\sigma_z = |X_1, 0_2\rangle\langle X_1, 0_2| - |0_1, X_2\rangle\langle 0_1, X_2|$$

Of course, this is the same mathematical description we use for a qubit in quantum computing

What about coupling between molecules?

There is typically a dipole coupling between nearby chromophores. This is a dipole energy coupling:



In the single-excitation subspace, we can again ignore the ground state, and write

$$|E_1\rangle = |X_1\rangle \otimes |0_2\rangle$$



$$|E_1\rangle = |0_1\rangle \otimes |X_2\rangle$$

$$H_D = \frac{\Delta}{2} \sigma_x$$

$$\sigma_x = |X_1, 0_2\rangle\langle 0_1, X_2| + |0_1, X_2\rangle\langle X_1, 0_2|, \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Combining the two:

$$H = H_S + H_D = \frac{\epsilon}{2} \sigma_z + \frac{\Delta}{2} \sigma_x$$

Lets imagine we put an excitation on the first chromophore, and allow it to evolve under this simple Hamiltonian, what happens?

The initial condition is

$$|\psi(t=0)\rangle = |X_1, 0_2\rangle$$

The dynamics are given by the standard time-independant Schrodinger equation

$$\frac{d}{dt} |\psi(t)\rangle = \frac{-i}{\hbar} H |\psi(t)\rangle$$

Typical parameters for $\epsilon = -120 \text{ cm}^{-1}$, and $\Delta = 87 \text{ cm}^{-1}$.

Note, to convert $1 \text{ cm}^{-1} = 100 \times h \times c \text{ Joules}$, $c = \text{speed of light}$, $h = \text{Planck's constant}$.

In units of s^{-1} , 100 cm^{-1} corresponds to $10^4 \times h \times c / 2\pi\hbar = 3 \times 10^{12} s^{-1} = 3THz$, i.e., a frequency with a time-scale of ~ 333 femto-seconds.

Let us solve this example using QuTiP, and plot

$$P_1(t) = |\langle X_1, 0_2 | \psi(t) \rangle|^2$$

Let us solve this example using QuTiP, and plot

$$P_1(t) = |\langle X_1, 0_2 | \psi(t) \rangle|^2$$

In [2]:

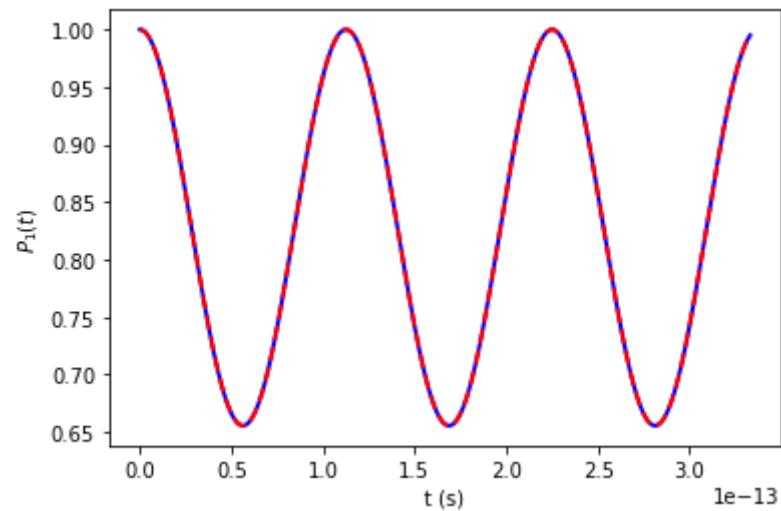
```
psi0 = basis(2,0)
tlist = linspace(0,0.333e-12,1000)
H = (-120 * sigmaz() + 87 * sigmax()) * 3e10 * 2 * pi
output = mesolve(H,psi0,tlist,[])
energies, states = H.eigenstates()

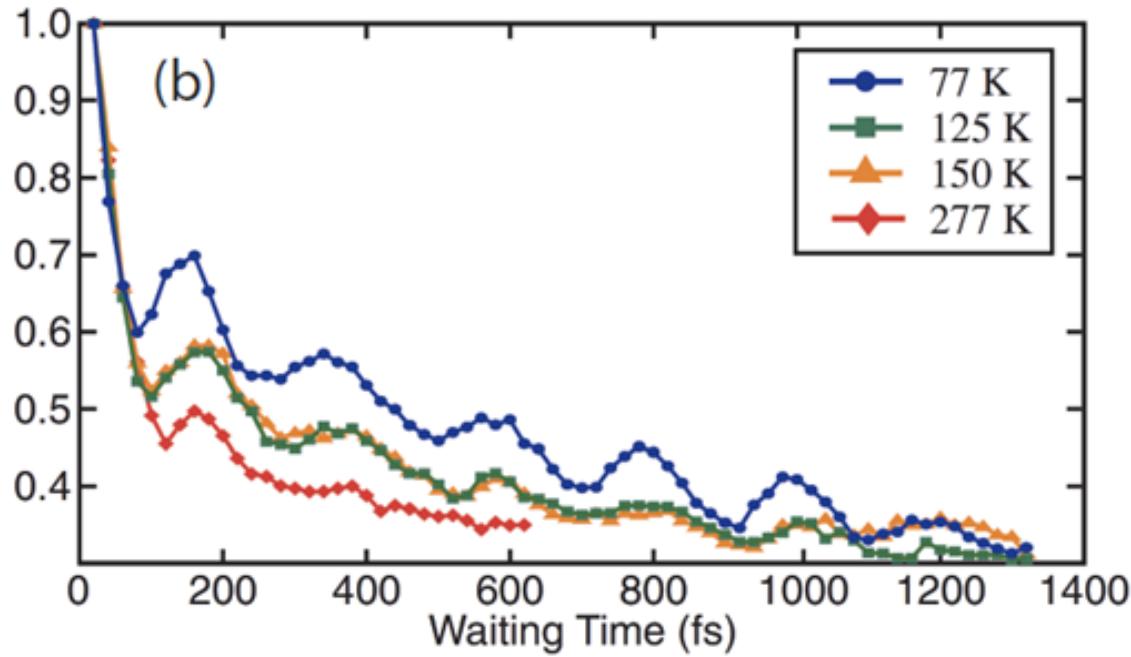
Pana = [abs((psi0*psi0.dag()).matrix_element(psi0,states[0])**2*exp(-1.0j*energies[0]*t)+  
           (psi0*psi0.dag()).matrix_element(psi0,states[1])**2*exp(-1.0j*energies[1]*t)))**2 for t in tlist]
```

In [3]:

```
fig, ax = plt.subplots(1, 1)
ax.plot(tlist,expect(psi0*psi0.dag(),output.states), 'b', lw=2)
ax.plot(tlist,Pana, 'r--', lw=2)
ax.set_xlabel('t (s)')
ax.set_ylabel('$P_1(t)$')

plt.show()
```





Some questions

How can we include noise in a 'quantum model'?

How accurate do we need to be?

Deriving approximate noise models: Lindblad master equation

- Modelling noise: The Lindblad Master equation; a quick derivation
- Introduction to QuTip
- FMO example

The Lindblad Master equation; a quick derivation

One approach we can take to include noise is to treat the environment as a quantum system, but include its affect on the system perturbatively. This is traditionally done with the Lindblad Master equation.

This result is the most widely used to study noise in quantum computation and quantum optics.

Let's sketch out its derivation first...

Steps:

- 1) A few definitions: density matrix, system and environment (bath) Hamiltonians.
- 2) Born - Markov approximations.
- 3) More definitions: The bath correlation function.
- 4) An explicit example: the spin-boson model.
- 5) The secular approximation.
- 6) And finally, the Lindblad Master Equation.

Definitions: The density operator

When a quantum system is effectively "closed" we describe its dynamics and evolution in terms of a pure state $|\psi(t)\rangle$ whose evolution is determined by a Hamiltonian and the Schrodinger equation (lets make life easier by neglecting \hbar)

$$\frac{\partial}{\partial t} |\psi(t)\rangle = -iH(t)|\psi(t)\rangle$$

More generally, for open systems it is convenient to use the density operator, which for a pure state, is given by

$$\rho = |\psi(t)\rangle\langle\psi(t)|$$

When describing one part of a larger composite system, as is the case for open systems **when we trace out the environment degrees of freedom**, the loss of knowledge about the influence of the other systems introduces **statistical uncertainty**.

This then gives rise to mixed states, in terms of a statistical ensemble of possible pure states, such that;

$$\rho = \sum_n p_n |\psi_n(t)\rangle\langle\psi_n(t)|$$

The evolution of the density operator obeys the Liouville-von Neumann equation of motion,

$$\frac{\partial}{\partial t} \rho(t) = -i[H(t), \rho(t)]$$

The density operator itself has some important properties;

- For pure states $\rho^2 = \rho$
- It is Hermitian $\rho^\dagger = \rho$.
- It has trace one $\text{Tr}\rho = 1$.
- Expectation values of system operators can be evaluated as $\langle A \rangle = \text{Tr}(A\rho)$.

The system and environment Hamiltonians

Ultimately the influence of a large environment on a systems evolution is to induce statistical uncertainty into its dynamical evolution.

This gives rise to dissipation (emission and absorbtion of energy) and dephasing (localization into particular states).

The philosophy we will describe here is to start with an **explicit Hamiltonian description of the system (**S**) and the full Environment (**E**)**, with the Hilbert space

$$\mathcal{H} = \mathcal{H}_S \otimes \mathcal{H}_E$$

The system and environment Hamiltonians

Ultimately the influence of a large environment on a systems evolution is to induce statistical uncertainty into its dynamical evolution.

This gives rise to dissipation (emission and absorbtion of energy) and dephasing (localization into particular states).

The philosophy we will describe here is to start with an **explicit Hamiltonian description of the system (**S**) and the full Environment (**E**)**, with the Hilbert space

$$\mathcal{H} = \mathcal{H}_S \otimes \mathcal{H}_E$$

and then **derive effective models, or equations of motion**, for the reduced density operator of the system alone, after tracing out the environment degrees of freedom, i.e.,

$$\rho_S(t) = \text{Tr}_E[\rho(t)]$$

since $\rho_S(t)$ describes all pertinent and accessible information about the system itself.

The full dynamics of the system and environment would in principle be described by the Hamiltonian

$$H = H_S + H_E + H_I$$

The full dynamics of the system and environment would in principle be described by the Hamiltonian

$$H = H_S + H_E + H_I$$

In general, one can describe the evolution of the whole system and environment, **in the interaction picture**, as

$$\frac{\partial}{\partial t} \tilde{\rho}(t) = -i[\tilde{H}_I(t), \tilde{\rho}(t)]$$

where the interaction pictures moves the free evolution of both the system and environment into the operators themselves, such that

$$\tilde{H}_I(t) = e^{i(H_S+H_E)t} H_I e^{-i(H_S+H_E)t}$$

Formally we can solve the equation of motion with

$$\tilde{\rho}(t) = \rho(0) - i \int_0^t ds [\tilde{H}_I(s), \tilde{\rho}(s)]$$

Substituting this into the RHS of the original equation of motion, and formally **tracing out the environment degrees of freedom** , gives,

$$\frac{\partial}{\partial t} \tilde{\rho}_S(t) = -i \text{Tr}_E[\tilde{H}_I(t), \tilde{\rho}(0)] - \int_0^t ds \text{Tr}_E[\tilde{H}_I(t), [\tilde{H}_I(s), \tilde{\rho}(s)]]$$

The Born Approximation

This is still an intractable equation of motion. To make progress we must make a series of assumptions:

The Born Approximation

This is still an intractable equation of motion. To make progress we must make a series of **assumptions**:

-- ****Factorizable initial conditions**** : We assume that at $t = 0$ the system and environment are separable, such that

$$\rho(0) = \rho_S(0) \otimes \rho_E(0)$$

The Born Approximation

This is still an intractable equation of motion. To make progress we must make a series of **assumptions**:

-- ****Factorizable initial conditions**** : We assume that at $t = 0$ the system and environment are separable, such that

$$\rho(0) = \rho_S(0) \otimes \rho_E(0)$$

-- ****The Born approximation****: In addition, we assume that the density matrix **factorises** at all times, and that the **environment state is time-independent**,

$$\rho(t) = \rho_S(t)\rho_E(0)$$

This assumes that the environment is mostly unaffected by the interaction with the system.
(Any excitations emitted by the system quickly decay in the environment)

-- Note: We also remove the first term on the RHS (for many examples it is zero, or it can be cancelled with a c-number term in the system Hamiltonian).

Under these assumptions,

$$\frac{\partial}{\partial t} \tilde{\rho}_S(t) = - \int_0^t ds \text{Tr}_E[\tilde{H}_I(t), [\tilde{H}_I(s), \tilde{\rho}_S(s)\rho_E]]$$

This equation of motion assumes weak coupling, as the system and environment remain separable;

... but it still includes some **time non-local behavior** as the state of the system depends on its history (via the integral over s), and thus its past interaction with the bath.

The Markov Approximation

We can now make what is known as the ****Markov**** assumption, to eliminate this history.

The Markov Approximation

We can now make what is known as the ****Markov**** assumption, to eliminate this history.

We assume

$$\tilde{\rho}_S(s) \rightarrow \tilde{\rho}_S(t)$$

in the integral.

This assumptions implies there is a seperation of time scales between environment (τ_E) and system relaxation time (τ_R) and any intrinsic system time scales (τ_S),

$$\tau_E \ll \tau_R, \tau_S$$

The Markov Approximation

We can now make what is known as the ****Markov**** assumption, to eliminate this history.

We assume

$$\tilde{\rho}_S(s) \rightarrow \tilde{\rho}_S(t)$$

in the integral.

This assumptions implies there is a seperation of time scales between environment (τ_E) and system relaxation time (τ_R) and any intrinsic system time scales (τ_S),

$$\tau_E \ll \tau_R, \tau_S$$

I.e., if the environment were to be perturbed by the system, it relaxes back to equilibrium faster than the typical time scale of the evolution (**relaxation** and coherent evolution) of the system. This seems like a consistent assumption to make alongside the **Born approximation**, since that also implies the environment remains in equilibrium.

Now we have, after substituting $\tilde{\rho}_S(s) \rightarrow \tilde{\rho}_S(t)$

$$\frac{\partial}{\partial t} \tilde{\rho}_S(t) = - \int_0^t ds \text{Tr}_E [\tilde{H}_I(t), [\tilde{H}_I(s), \tilde{\rho}_S(t) \rho_E]]$$

This is sometimes called the Redfield equation; it local in time, but **not quite completely Markovian** (we still have t in the integral limits).

Now we have, after substituting $\tilde{\rho}_S(s) \rightarrow \tilde{\rho}_S(t)$

$$\frac{\partial}{\partial t} \tilde{\rho}_S(t) = - \int_0^t ds \text{Tr}_E [\tilde{H}_I(t), [\tilde{H}_I(s), \tilde{\rho}_S(t) \rho_E]]$$

This is sometimes called the Redfield equation; it local in time, but **not quite completely Markovian** (we still have t in the integral limits).

We can simplifiy this further by removing the reference to t in the limits of the integral, by choosing $s = t - \tau$, and letting the upper limit go to infinity;

In other words, we assume that the integrand disappears fast compared to the system time scales.

Now we have, after substituting $\tilde{\rho}_S(s) \rightarrow \tilde{\rho}_S(t)$

$$\frac{\partial}{\partial t} \tilde{\rho}_S(t) = - \int_0^t ds \text{Tr}_E [\tilde{H}_I(t), [\tilde{H}_I(s), \tilde{\rho}_S(t) \rho_E]]$$

This is sometimes called the Redfield equation; it local in time, but **not quite completely Markovian** (we still have t in the integral limits).

We can simplifiy this further by removing the reference to t in the limits of the integral, by choosing $s = t - \tau$, and letting the upper limit go to infinity;

In other words, we assume that the integrand disappears fast compared to the system time scales.

Thus we arrive at the **fully Markovian master equation**:

$$\frac{\partial}{\partial t} \tilde{\rho}_S(t) = - \int_0^\infty d\tau \text{Tr}_E [\tilde{H}_I(t), [\tilde{H}_I(t - \tau), \tilde{\rho}_S(t) \rho_E]]$$

Describing the bath with a correlation function

The system depends on the environment properties via the two-time bath correlation functions. For example, lets explicitly define the system-environment interaction term as,

$$H_I = A \otimes B$$

where A is a Hermitian system operator, and B a Hermitian bath operator. (An explicit example will soon make this a little clearer).

Describing the bath with a correlation function

The system depends on the environment properties via the two-time bath correlation functions. For example, lets explicitly define the system-environment interaction term as,

$$H_I = A \otimes B$$

where A is a Hermitian system operator, and B a Hermitian bath operator. (An explicit example will soon make this a little clearer).

In the interaction picture

$$\tilde{H} = A(t) \otimes B(t)$$

Our Markovian master equation contains terms like: $\text{Tr}_E[B(t)B(t - \tau)\rho_E]$

Describing the bath with a correlation function

The system depends on the environment properties via the two-time bath correlation functions. For example, lets explicitly define the system-environment interaction term as,

$$H_I = A \otimes B$$

where A is a Hermitian system operator, and B a Hermitian bath operator. (An explicit example will soon make this a little clearer).

In the interaction picture

$$\tilde{H} = A(t) \otimes B(t)$$

Our Markovian master equation contains terms like: $\text{Tr}_E[B(t)B(t - \tau)\rho_E]$

If we assume the initial state of the environment is a steady state, i.e., it does not evolve under H_E , these terms are "stationary", $C(\tau) = \text{Tr}_E[B(\tau)B\rho_E]$

Inserting these various terms into our Markovian master equation, and moving back to the Schrodinger picture, one obtains

$$\frac{\partial}{\partial t} \rho_S(t) = -i[H_S, \rho(t)] - \int_0^{\infty} d\tau ([A, [A(-\tau)\rho_S(t)]C(\tau) + [\rho_S(t)A(-\tau), A]C(-\tau)])$$

This equation in itself is useful, and can be developed and applied to certain problems.

However, it is still not in the famous ****Lindblad**** form, and does not guarantee positivity of the density matrix as it is evolved.

Inserting these various terms into our Markovian master equation, and moving back to the Schrodinger picture, one obtains

$$\frac{\partial}{\partial t} \rho_S(t) = -i[H_S, \rho(t)] - \int_0^\infty d\tau ([A, [A(-\tau)\rho_S(t)]C(\tau) + [\rho_S(t)A(-\tau), A]C(-\tau)])$$

This equation in itself is useful, and can be developed and applied to certain problems.

However, it is still not in the famous **Lindblad** form, and does not guarantee positivity of the density matrix as it is evolved.

Note: In general there may be a sum of interactions in the interaction Hamiltonian, in which case the integral contains a sum of terms for each interaction part, and the correlation functions are generalized $\tilde{H} = \sum_\alpha A_\alpha(t) \otimes B_\alpha(t)$

$$\begin{aligned} \frac{\partial}{\partial t} \rho_S(t) &= -i[H_S, \rho(t)] - \sum_{\alpha, \beta} \\ &\int_0^\infty d\tau ([A_\alpha, [A_\beta(-\tau)\rho_S(t)]C_{\alpha\beta}(\tau) + [\rho_S(t)A_\beta(-\tau), A_\alpha]C_{\beta\alpha}(-\tau)]) \end{aligned}$$

Interestingly, there is still not yet any assumption about the timescales τ_S , due to H_S , and the relaxation τ_R , induced by the bath.

If we perform an additional approximation, and **neglect terms due to system evolution which are very fast compared to the relaxation time scale τ_R** , we can obtain a **Secular master equation** which is in **Lindblad form**.

In other words, we will assume, $\tau_S \ll \tau_R$, so that now, the Born-Markov-Secular master equation will be valid when

$$\tau_E \ll \tau_S \ll \tau_R$$

Interestingly, there is still not yet any assumption about the timescales τ_S , due to H_S , and the relaxation τ_R , induced by the bath.

If we perform an additional approximation, and **neglect terms due to system evolution which are very fast compared to the relaxation time scale τ_R** , we can obtain a **Secular master equation** which is in **Lindblad form**.

In other words, we will assume, $\tau_S \ll \tau_R$, so that now, the Born-Markov-Secular master equation will be valid when

$$\tau_E \ll \tau_S \ll \tau_R$$

It is a little cumbersome to the general derivation of this step, so let us move onto an explicit example. However, one note; the above **non-secular** form of the master equation is **may be** applicable even if the relaxation time scale and the system time scale are comparable, and thus it is useful in and of itself. (See Chapter 3.6 in Breuer and Petruccione).

The spin-boson model

The archetypical example of an open quantum system is where the "system" is a two-level system, a spin, described by:

The spin-boson model

The archetypical example of an open quantum system is where the "system" is a two-level system, a spin, described by:

Pauli spin matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (2)$$

The spin-boson model

The archetypical example of an open quantum system is where the "system" is a two-level system, a spin, described by:

Pauli spin matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (3)$$

While the bath is a continuum of **boson modes**, or harmonic oscillators, described by creation and annihilation operators b_k , obeying the commutation relations $[b_k, b'_{k'}] = \delta_{k,k'}$

Returning to our example from earlier, our 'spin' was the position of the electronic excitation, and the bosons are the nuclear motion of the chromophores that modulate the energy of the electronic excitations

A

B

The total Hamiltonian is then,

$$H = H_S + H_E + H_I$$

where

The total Hamiltonian is then,

$$H = H_S + H_E + H_I$$

where

$$H_S = \frac{\epsilon}{2}\sigma_z + \frac{\Delta}{2}\sigma_x$$

The total Hamiltonian is then,

$$H = H_S + H_E + H_I$$

where

$$H_S = \frac{\epsilon}{2}\sigma_z + \frac{\Delta}{2}\sigma_x$$

$$H_E = \sum_k \omega_k b_k^\dagger b_k$$

The total Hamiltonian is then,

$$H = H_S + H_E + H_I$$

where

$$H_S = \frac{\epsilon}{2}\sigma_z + \frac{\Delta}{2}\sigma_x$$

$$H_E = \sum_k \omega_k b_k^\dagger b_k$$

$$H_I = \sigma_z \sum_k g_k (b_k^\dagger + b_k) = \sum_{j,l} c_{j,l} |\psi_j\rangle \langle \psi_l| \sum_k g_k (b_k^\dagger + b_k)$$

For eigenstates $|\psi_j\rangle$ of H_S , and $c_{j,l} = \langle \psi_j | \sigma_z | \psi_l \rangle$

In the **interaction picture** the interaction Hamiltonian is

$$\tilde{H}_I(t) = e^{i(H_S+H_E)t} H_I e^{-i(H_S+H_E)t}$$

$$\tilde{H}_I = \sum_{j,l} c_{j,l} |\psi_j\rangle\langle\psi_l| e^{i\Delta_{j,l}t} \sum_k g_k (b_k^\dagger e^{i\omega_k t} + b_k e^{-i\omega_k t})$$

$$\Delta_{j,l} = E_j - E_l$$

is the difference in eigenenergies.

In the **interaction picture** the interaction Hamiltonian is

$$\tilde{H}_I(t) = e^{i(H_S+H_E)t} H_I e^{-i(H_S+H_E)t}$$

$$\tilde{H}_I = \sum_{j,l} c_{j,l} |\psi_j\rangle\langle\psi_l| e^{i\Delta_{j,l}t} \sum_k g_k (b_k^\dagger e^{i\omega_k t} + b_k e^{-i\omega_k t})$$

$$\Delta_{j,l} = E_j - E_l$$

is the difference in eigenenergies.

Inserting this into our master equation, we find,

$$\frac{\partial}{\partial t} \rho_S(t) = - \int_0^\infty d\tau ([\sigma_z(t), [\sigma_z(t-\tau)\rho_S(t)]C(\tau) + [\rho_S(t)\sigma_z(t-\tau), \sigma_z(t)]C(-\tau)])$$

Recall that the bath correlation functions are defined as

$$C(\tau) = \text{Tr}_E[B(\tau)B\rho_E]$$

Where

$$B(t) = \sum_k g_k \{ b_k^\dagger e^{i\omega_k t} + b_k e^{-i\omega_k t} \}$$

$$C(t) = \text{Tr}_E[(\sum_k g_k \{ b_k^\dagger e^{i\omega_k t} + b_k e^{-i\omega_k t} \})(\sum_{k'} g_{k'} \{ b_{k'}^\dagger + b_{k'} \})\rho_E]$$

Assuming that the initial state of the environment is a thermal state

$$\rho_E = e^{-H_E/k_B T} / Z$$

$$Z = \text{Tr}(e^{-H_E/k_B T})$$

lets us evaluate this quite easily;

Assuming that the initial state of the environment is a thermal state

$$\rho_E = e^{-H_E/k_B T} / Z$$

$$Z = \text{Tr}(e^{-H_E/k_B T})$$

lets us evaluate this quite easily;

The only non-zero terms come from

$$\text{Tr}_E[b_k b_{k'}^\dagger \rho_E] = \delta_{kk'}[1 + n(\omega_k)]$$

and

$$\text{Tr}_E[b_{k'}^\dagger b_k \rho_E] = \delta_{kk'}n(\omega_k)$$

where $n(\omega) = (e^{\omega/k_B T} - 1)^{-1}$.

Is the Bose-Einstein occupation number.

Then

$$C(t) = \sum_k (g_k^2 e^{i\omega_k t} n(\omega_k) + g_k^2 e^{-i\omega_k t} [1 + n(\omega_k)])$$

Then

$$C(t) = \sum_k (g_k^2 e^{i\omega_k t} n(\omega_k) + g_k^2 e^{-i\omega_k t} [1 + n(\omega_k)])$$

We can then move to a continuum representation by turning the sum into an integral, and defining the **spectral density**

$$J(\omega) = \sum_k g_k^2 \delta(\omega - \omega_k)$$

Then

$$C(t) = \sum_k (g_k^2 e^{i\omega_k t} n(\omega_k) + g_k^2 e^{-i\omega_k t} [1 + n(\omega_k)])$$

We can then move to a continuum representation by turning the sum into an integral, and defining the **spectral density**

$$J(\omega) = \sum_k g_k^2 \delta(\omega - \omega_k)$$

$$C(t) = \int_0^\infty d\omega \{ J(\omega) e^{i\omega t} n(\omega) + J(\omega) e^{-i\omega t} [1 + n(\omega)] \}$$

**Note: sometimes it is useful to rewrite this in terms of real and imaginary parts. See part 3!

The secular approximaton

Let us return to the master equation, in the interaction picture;

$$\frac{\partial}{\partial t} \rho_S(t) = - \int_0^\infty d\tau ([\sigma_z(t), [\sigma_z(t-\tau)\rho_S(t)]C(\tau) + [\rho_S(t)\sigma_z(t-\tau), \sigma_z(t)]C(-\tau)])$$

The secular approximaton

Let us return to the master equation, in the interaction picture;

$$\frac{\partial}{\partial t} \rho_S(t) = - \int_0^\infty d\tau ([\sigma_z(t), [\sigma_z(t-\tau)\rho_S(t)]C(\tau) + [\rho_S(t)\sigma_z(t-\tau), \sigma_z(t)]C(-\tau)])$$

We are going to do two things:

- Neglect terms come from products of system operators that rotate quickly... this is the **secular approximation!**

Its equivalent in this case to a **rotating wave approximation**; and for it be valid, the relaxation rate has to be smaller than the system frequency.

- Evaluate the integral over τ and ω

This is a lot of algebra, so lets just do one term.

First, the secular approximation:

Recall that, in the interaction picture:

$$\sigma_z(t) = \sum_{j,l} c_{j,l} |\psi_j\rangle\langle\psi_l| e^{i\Delta_{j,l}t}$$

First, the secular approximation:

Recall that, in the interaction picture:

$$\sigma_z(t) = \sum_{j,l} c_{j,l} |\psi_j\rangle\langle\psi_l| e^{i\Delta_{j,l}t}$$

So the first term in the master equation contains products like

$$\sigma_z(t)\sigma_z(t-\tau) = \sum_{j,l} c_{j,l} |\psi_j\rangle\langle\psi_l| e^{i\Delta_{j,l}t} \sum_{n,m} c_{n,m} |\psi_n\rangle\langle\psi_m| e^{i\Delta_{n,m}(t-\tau)}$$

We neglect terms oscillating very quickly, like $2i\Delta_{j,l}$. (Note we also ignore diagonal terms ($j = l$ etc) just for simplicity, though these can induce additional dephasing contributions).

This is similar to a rotating wave approximation, and is valid if $\tau_S \ll \tau_R$ i.e., when the relaxation rate is smaller than the system frequencies. Now we have the separation of time scales

$$\tau_E \ll \tau_S \ll \tau_R$$

Second, some integrals

For the remaining terms, that do not rotate at this frequency, we have to evaluate integrals like,

$$|\psi_l\rangle\langle\psi_l|\rho_S(t)\int_0^\infty d\tau e^{i\Delta_{j,l}\tau}C(\tau)$$

Second, some integrals

For the remaining terms, that do not rotate at this frequency, we have to evaluate integrals like,

$$|\psi_l\rangle\langle\psi_l|\rho_S(t)\int_0^\infty d\tau e^{i\Delta_{j,l}\tau}C(\tau)$$

Using our earlier definition of the correlation function

$$C(t) = \int_0^\infty d\omega \{J(\omega)e^{i\omega t}n(\omega) + J(\omega)e^{-i\omega t}[1+n(\omega)]\}$$

We have to evaluate

$$\int_0^\infty d\tau e^{i\Delta_{j,l}\tau}C(\tau) = \int_0^\infty d\omega \int_0^\infty d\tau e^{i\Delta_{j,l}\tau} \{J(\omega)e^{i\omega\tau}n(\omega) + J(\omega)e^{-i\omega\tau}[1+n(\omega)]\}$$

We can do the τ integral first, using the formula,

$$\int_0^\infty d\tau e^{\pm i\epsilon\tau} = \pi\delta(\epsilon) \pm i\frac{\mathcal{P}}{\epsilon}$$

\mathcal{P} is the Cauchy principal value. The imaginary term leads to environment-induced energy shifts (sometimes called Lamb shifts. Here we neglect this term, for simplicity.

We can do the τ integral first, using the formula,

$$\int_0^\infty d\tau e^{\pm i\epsilon\tau} = \pi\delta(\epsilon) \pm i\frac{\mathcal{P}}{\epsilon}$$

\mathcal{P} is the Cauchy principal value. The imaginary term leads to environment-induced energy shifts (sometimes called Lamb shifts). Here we neglect this term, for simplicity.

We see that the delta function lets us then evaluate the frequency integral, and thus,

$$\int_0^\infty d\tau e^{i\Delta_{j,l}\tau} C(\tau) = \pi J(\Delta_{j,l})(n(\Delta_{j,l}) + 1)$$

Thus for this term, the right hand side of the master equation becomes

$$|\psi_l\rangle\langle\psi_l|\rho_S(t) \left[\int_0^\infty d\tau e^{i\Delta_{j,l}\tau} c(\tau) \right] = \pi J(\Delta_{j,l})(n(\Delta_{j,l}) + 1) |\psi_l\rangle\langle\psi_l|\rho_S(t)$$

The Lindblad master equation

We are almost there! If we repeat this for each term in the master equation, and collect terms, and move back to the schrodinger picture, we find an equation of motion describing the density matrix of the system alone, which is known as the Lindblad master equation (Gorini–Kossakowski–Sudarshan–Lindblad)

The Lindblad master equation

We are almost there! If we repeat this for each term in the master equation, and collect terms, and move back to the schrodinger picture, we find an equation of motion describing the density matrix of the system alone, which is known as the Lindblad master equation (Gorini–Kossakowski–Sudarshan–Lindblad)

$$\begin{aligned} \frac{\partial}{\partial t} \rho_S(t) &= -i[H_S, \rho(t)] \\ &+ \sum_{j>l,l} \pi J(\Delta_{j,l}) |c_{j,l}|^2 (n(\Delta_{j,l}) + 1) [2|\psi_l\rangle\langle\psi_j|\rho_S(t)|\psi_j\rangle\langle\psi_l| - \{|\psi_j\rangle\langle\psi_j|, \rho_S\}] \\ &+ \sum_{j>l,l} \pi J(\Delta_{j,l}) |c_{j,l}|^2 n(\Delta_{j,l}) [2|\psi_j\rangle\langle\psi_l|\rho_S(t)|\psi_l\rangle\langle\psi_j| - \{|\psi_l\rangle\langle\psi_l|, \rho_S\}] \end{aligned}$$

The first part is the **coherent system evolution**, the second part describes **spontaneous and stimulated emission**, and the third part describes **absorption**. $n(\omega) = (e^{\omega/T} - 1)^{-1}$

Why is this so great?

- This Lindblad form guarantees positivity: we will always have a physical density matrix.
- It is easy to evaluate, and can be derived for many types of systems.

(see part 2 for more examples).

To recap, it is valid when:

- The **Born approximation** applies: the coupling is weak such that we can assume $\rho(t) = \rho_S(t) \otimes \rho_E$
- The **Markov approximation** applies: the bath returns to equilibrium on a time scale τ_E which is faster than the relaxation of the system induced by that bath τ_R .
- The **secular approximation** applies: the system time scales are small (i.e., higher frequency) than the relaxation time scales. In other words, $\epsilon > \pi J(\epsilon)(n(\epsilon) + 1)$.

$$\tau_E \ll \tau_S \ll \tau_R$$

To recap, it is valid when:

- The **Born approximation** applies: the coupling is weak such that we can assume $\rho(t) = \rho_S(t) \otimes \rho_E$
- The **Markov approximation** applies: the bath returns to equilibrium on a time scale τ_E which is faster than the relaxation of the system induced by that bath τ_R .
- The **secular approximation** applies: the system time scales are small (i.e., higher frequency) than the relaxation time scales. In other words, $\epsilon > \pi J(\epsilon)(n(\epsilon) + 1)$.

$$\tau_E \ll \tau_S \ll \tau_R$$

Two questions :

- Can we use this model to describe the environment affecting energy transport in FMO?
- Does noise slow down transport, or speed it up?

Solution to the Lindblad master equation

In some cases we can get analytical solutions. Using our earlier example of a two-level system, this is easily done by writing out equations of motion for the matrix elements of the master equation, and assuming the bath spectrum is flat, so that $\pi J(\Delta_{j,l})|c_{j,l}|^2 = \gamma$.

$$\begin{aligned} \frac{\partial}{\partial t} \rho_S(t) &= -i[H_S, \rho(t)] \\ &+ \sum_{j>l,l} \gamma(n+1) [2|\psi_l\rangle\langle\psi_j|\rho_S(t)|\psi_j\rangle\langle\psi_l| - \{|\psi_j\rangle\langle\psi_j|, \rho_S\}] \\ &+ \sum_{j>l,l} \gamma n [2|\psi_j\rangle\langle\psi_l|\rho_S(t)|\psi_l\rangle\langle\psi_j| - \{|\psi_l\rangle\langle\psi_l|, \rho_S\}] \end{aligned}$$

which give, for e = excited state, g = ground state, and defining matrix elements like $\rho_{ee} = \langle e | \rho | e \rangle$,

$$\dot{\rho}_{ee}(t) = -2\gamma(n+1)\rho_{ee}(t) + 2\gamma n \rho_{gg}(t)$$

$$\dot{\rho}_{gg}(t) = -2\gamma(n)\rho_{gg}(t) + 2\gamma(n+1)\rho_{ee}(t)$$

$$\dot{\rho}_{eg}(t) = -[\gamma(2n+1) + i\Delta_{e,g}]\rho_{eg}(t)$$

$$\dot{\rho}_{ge}(t) = -[\gamma(2n+1) - i\Delta_{e,g}]\rho_{ge}(t)$$

We can see that the populations are coupled with rates corresponding to the emission and absorption. The off-diagonal matrix elements decay proportional to the coupling the environment. This is decoherence!

The solutions to these equations are very simple for zero temperature ($n = 0$), where we see:

$$\rho_{ee}(t) = e^{-2\gamma t} \rho_{ee}(0)$$

$$\rho_{gg}(t) = 1 - e^{-2\gamma t} \rho_{ee}(0)$$

$$\rho_{eg}(t) = e^{-\gamma t} e^{-i\Delta_{e,g} t} \rho_{eg}(0)$$

$$\rho_{ge}(t) = e^{-\gamma t} e^{i\Delta_{e,g} t} \rho_{ge}(0)$$

The off diagonal terms decay at half the speed of the relaxation terms, leading to the infamous $\textcolor{red}{T_2 = 2T_1}$ relationship.

In the general case, it can be tedious to construct and solve such equations of motion.

So... lets use QuTiP to play with some examples, and see what noise does to our previous example of pure evolution.

QuTiP: A brief introduction

The Quantum Toolbox in Python was developed in our group, primarily by Robert Johansson (now working Rakuten) and Paul Nation (now working in IBM-Q).

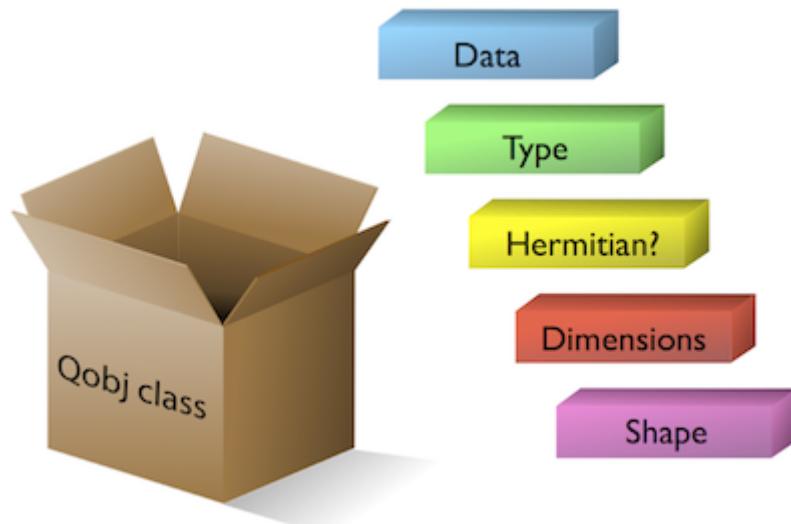
<http://qutip.org/devs.html>

It has a variety of built-in solvers for solving the dynamics via:

- the Schrodinger equation,
- master equations,
- Floquet formalism,
- Monte-Carlo quantum trajectories,
- stochastic Schrodinger/master equations,
- the Hierarchy method, the reaction coordinate method, etc.
- supports time-dependance in Hamiltonians, Lindblads, etc.

It is based around a quantum object class which can be used to represent state vectors, operators, and superoperators.

It is composed of a numerical matrix, and various properties that define its dimensionality, etc.



Other reasons to use QuTiP

- Open source science
- Promotes reproducibility
- Has even more functionality, including
 - Quantum optimal control library (CRAB, GOAT, and other animals)
 - QIP library
 - functions to calculate spectra, counting statistics, etc
 - Supports arbitrary time-dependance, in Hamiltonian and Lindblad collapse operators
 - Active development community.

Recalling our example of a single two-level system described by the Pauli spin matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (4)$$

In QuTiP we can define this manually, or with predefined operators:

Recalling our example of a single two-level system described by the Pauli spin matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (5)$$

In QuTiP we can define this manually, or with predefined operators:

```
In [2]: sz = Qobj([[1,0], [0,-1]]) # the sigma-z Pauli operator  
sz
```

```
Out[2]: Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True
```

$$\begin{pmatrix} 1.0 & 0.0 \\ 0.0 & -1.0 \end{pmatrix}$$

Recalling our example of a single two-level system described by the Pauli spin matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (6)$$

In QuTiP we can define this manually, or with predefined operators:

```
In [2]: sz = Qobj([[1,0], [0,-1]]) # the sigma-z Pauli operator  
sz
```

```
Out[2]: Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True
```

$$\begin{pmatrix} 1.0 & 0.0 \\ 0.0 & -1.0 \end{pmatrix}$$

```
In [ ]: sz = sigmaz() # the sigma-z Pauli operator  
sz
```

States in diagonal basis can be defined easily

States in diagonal basis can be defined easily

In [17]:

```
psi = basis(2, 0)
psi
```

Out[17]: Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket

$$\begin{pmatrix} 1.0 \\ 0.0 \end{pmatrix}$$

States in diagonal basis can be defined easily

In [17]:

```
psi = basis(2, 0)
psi
```

Out[17]: Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket

$$\begin{pmatrix} 1.0 \\ 0.0 \end{pmatrix}$$

The density matrix of this state is just the outer product

$$\rho = |\psi\rangle\langle\psi|$$

States in diagonal basis can be defined easily

In [17]:

```
psi = basis(2, 0)
psi
```

Out[17]: Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket

$$\begin{pmatrix} 1.0 \\ 0.0 \end{pmatrix}$$

The density matrix of this state is just the outer product

$$\rho = |\psi\rangle\langle\psi|$$

In [18]:

```
rho = psi*psi.dag()
rho
```

Out[18]: Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 0.0 \end{pmatrix}$$

Another simple example: equal superposition of both states

$$\psi = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

Another simple example: equal superposition of both states

$$\psi = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

```
In [19]: psi = (basis(2, 0)+basis(2, 1))/sqrt(2)  
psi
```

```
Out[19]: Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket
```

$$\begin{pmatrix} 0.707 \\ 0.707 \end{pmatrix}$$

Another simple example: equal superposition of both states

$$\psi = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

```
In [19]: psi = (basis(2, 0)+basis(2, 1))/sqrt(2)  
psi
```

Out[19]: Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket

$$\begin{pmatrix} 0.707 \\ 0.707 \end{pmatrix}$$

```
In [20]: rho = psi*psi.dag()  
rho
```

Out[20]: Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\begin{pmatrix} 0.500 & 0.500 \\ 0.500 & 0.500 \end{pmatrix}$$

Aside: Composite systems

If we have more than one system, like two coupled qubits, or a qubit coupled to cavity (or equivalently, a dimer coupled to a single vibrational mode), we construct operators in a tensor space.

For example, for two qubits, the σ_z operator acting on the first qubit $\sigma_z \otimes \mathbf{1}$

Aside: Composite systems

If we have more than one system, like two coupled qubits, or a qubit coupled to cavity (or equivalently, a dimer coupled to a single vibrational mode), we construct operators in a tensor space.

For example, for two qubits, the σ_z operator acting on the first qubit $\sigma_z \otimes \mathbf{1}$

In [22]:

```
sz1 = tensor(sigmax(), qeye(2))

sz1
```

Out[22]: Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = True

$$\begin{pmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & -1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & -1.0 \end{pmatrix}$$

Aside: Composite systems

If we have more than one system, like two coupled qubits, or a qubit coupled to cavity (or equivalently, a dimer coupled to a single vibrational mode), we construct operators in a tensor space.

For example, for two qubits, the σ_z operator acting on the first qubit $\sigma_z \otimes \mathbf{1}$

In [22]:

```
sz1 = tensor(sigmax(), qeye(2))

sz1
```

Out[22]: Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = True

$$\begin{pmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & -1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & -1.0 \end{pmatrix}$$

`qeye(2)` is the identity matrix, so in the larger space of two systems we are just making a tensor product to get this matrix

Similarly, for terms which are products of two different system operators (like the coupling in a Hamiltonian), we can define,

Similarly, for terms which are products of two different system operators (like the coupling in a Hamiltonian), we can define,

In [24]:

```
tensor(sigmax(), sigmax())
```

Out[24]: Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = True

$$\begin{pmatrix} 0.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 \end{pmatrix}$$

Similarly, for terms which are products of two different system operators (like the coupling in a Hamiltonian), we can define,

```
In [24]: tensor(sigmax(), sigmax())
```

```
Out[24]: Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = True
```

$$\begin{pmatrix} 0.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 \end{pmatrix}$$

```
In [25]: #equivalent to
sx1 = tensor(sigmax(), qeye(2))

sx2 = tensor(qeye(2), sigmax())

print(sx1 * sx2)
```

```
Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = True
Qobj data =
[[0. 0. 0. 1.]
 [0. 0. 1. 0.]]
```

We can define and solve the Lindblad Master equation we solved analytically earlier in few lines:

$$\frac{\partial}{\partial t} \rho_S(t) = -i[H_S, \rho(t)]$$
$$+ \sum_{j>l,l} \gamma(n+1) [2|\psi_l\rangle\langle\psi_j|\rho_S(t)|\psi_j\rangle\langle\psi_l| - \{|\psi_j\rangle\langle\psi_j|, \rho_S\}]$$
$$+ \sum_{j>l,l} \gamma n [2|\psi_j\rangle\langle\psi_l|\rho_S(t)|\psi_l\rangle\langle\psi_j| - \{|\psi_l\rangle\langle\psi_l|, \rho_S\}]$$

QuTiP has several options on how to solve this type of formula.

The default is "mesolve" which takes Qobj definitions of the Hamiltonian, and the Lindblad, and converts them into a set of coupled equations in matrix form, as we did earlier. It then directly solves this set of coupled ODEs.

Aside: Superoperators

The Lindblad contains terms like $X\rho_S(t)Y$

For numerical implementation it is convenient to change how we store the density matrix and apply operators by using the superoperator formulation. Here, we essentially convert the density matrix into a vector in a larger space, writing

$$\rho_S(t) = \sum_{n,m} c_n c_m |\psi_n\rangle\langle\psi_m| \rightarrow \sum_{n,m} c_n c_m |\psi_m\rangle \otimes |\psi_n\rangle$$

Aside: Superoperators

The Lindblad contains terms like $X\rho_S(t)Y$

For numerical implementation it is convenient to change how we store the density matrix and apply operators by using the superoperator formulation. Here, we essentially convert the density matrix into a vector in a larger space, writing

$$\rho_S(t) = \sum_{n,m} c_n c_m |\psi_n\rangle\langle\psi_m| \rightarrow \sum_{n,m} c_n c_m |\psi_m\rangle \otimes |\psi_n\rangle$$

In [14]:

```
rho = 0.5*(basis(2,0) * basis(2,0).dag() + basis(2,1) * basis(2,1).dag())
print(rho)
```

```
Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True
Qobj data =
[[0.5 0. ]
 [0. 0.5]]
```

Aside: Superoperators

The Lindblad contains terms like $X\rho_S(t)Y$

For numerical implementation it is convenient to change how we store the density matrix and apply operators by using the superoperator formulation. Here, we essentially convert the density matrix into a vector in a larger space, writing

$$\rho_S(t) = \sum_{n,m} c_n c_m |\psi_n\rangle\langle\psi_m| \rightarrow \sum_{n,m} c_n c_m |\psi_m\rangle \otimes |\psi_n\rangle$$

In [14]:

```
rho = 0.5*(basis(2,0) * basis(2,0).dag() + basis(2,1) * basis(2,1).dag())
print(rho)
```

```
Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True
Qobj data =
[[0.5 0.]
 [0. 0.5]]
```

In [15]:

```
vec_rho = operator_to_vector(rho)
print(vec_rho)
```

```
Quantum object: dims = [[[2], [2]], [1]], shape = (4, 1), type = operator-ket
Qobj data =
[[0.5]
 [0.]
 [0.]
 [0.5]]
```

Then the pre and post multiplication (linear map) becomes a matrix in this larger space. Essentially we are writing $X\rho_S(t)Y \equiv Y^\dagger \otimes X \sum_{n,m} c_n c_m |\psi_n\rangle \otimes |\psi_m\rangle$

Then the pre and post multiplication (linear map) becomes a matrix in this larger space.
Essentially we are writing $X\rho_S(t)Y \equiv Y^\dagger \otimes X \sum_{n,m} c_n c_m |\psi_n\rangle \otimes |\psi_m\rangle$

In [21]:

```
X = sigmax()
Y = sigmax()
print(sprepost(X,Y))
```

```
Quantum object: dims = [[[2], [2]], [[2], [2]]], shape = (4, 4), type = super, is
herm = True
Qobj data =
[[0. 0. 0. 1.]
 [0. 0. 1. 0.]
 [0. 1. 0. 0.]
 [1. 0. 0. 0.]]
```

Master equation example

First define the Hamiltonian $H_S = \frac{\epsilon}{2}\sigma_z + \frac{\Delta}{2}\sigma_x$

and initial condition $\psi(t = 0) = |1\rangle$, $\rho(0) = |1\rangle\langle 1|$

Master equation example

First define the Hamiltonian $H_S = \frac{\epsilon}{2}\sigma_z + \frac{\Delta}{2}\sigma_x$

and initial condition $\psi(t = 0) = |1\rangle$, $\rho(0) = |1\rangle\langle 1|$

In [142]:

```
eps = -120
Delta = 87

# Hamiltonian
H = (eps * sigmaz() + Delta * sigmax()) * 3e10 * 2 * pi

# initial state.
N = 2 # number of states in the Hilbert space
n = 0 # the state that will be occupied (python counts from zero)

psi0 = basis(N, n)
print(psi0)

print(psi0*psi0.dag())
```

```
Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket
Qobj data =
[[1.]
 [0.]]
```

```
Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True
Qobj data =
[[1. 0.]
 [0. 0.]]
```

In [143]:

```
H.eigenstates()
```

Out[143]:

```
(array([-2.79387047e+13,  2.79387047e+13]),
 array([Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket
Qobj data =
 [[-0.9512125 ]
 [ 0.30853651]],
      Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket
Qobj data =
 [[0.30853651]
 [0.9512125 ]]], dtype=object))
```

We can define our Lindblad master equation by just defining the rates and the operator which goes into master equation.

Here we have two operators; one which causes emission,

$$\sum_{j>l,l} \gamma(n+1) [2|\psi_l\rangle\langle\psi_j|\rho_S(t)|\psi_j\rangle\langle\psi_l| - \{|\psi_j\rangle\langle\psi_j|, \rho_S\}]$$

the other absorption

$$\sum_{j>l,l} \gamma n [2|\psi_j\rangle\langle\psi_l|\rho_S(t)|\psi_l\rangle\langle\psi_j| - \{|\psi_l\rangle\langle\psi_l|, \rho_S\}]$$

In [230]:

```
gamma = 40 * 3e10 #this should be smaller than our system energies.
n_th = 0.0 #Start with zero temperure for simplicity

all_energy, all_state = H.eigenstates()
Nmax = 2
sz = sigmaz()
collapse_list = []
for j in range(Nmax):

    for k in range(j+1,Nmax):
        #FYI: QuTiPs definition of a Lindblad differs from our derivation by a factor of '2'
        rate = (np.absolute(sz.matrix_element(all_state[j].dag(),all_state[k]))**2 * 2 *
                gamma * (n_th+1))
        if rate > 0.0:
            collapse_list.append((np.sqrt(rate)*all_state[j]*all_state[k].dag())) #emission

        rate = (np.absolute(sz.matrix_element(all_state[k].dag(),all_state[j]))**2 * 2 *
                gamma * (n_th))
        if rate > 0.0:
            collapse_list.append((np.sqrt(rate)*all_state[k]*all_state[j].dag())) #absorption
```

In [230]:

```
gamma = 40 * 3e10 #this should be smaller than our system energies.
n_th = 0.0 #Start with zero temperure for simplicity

all_energy, all_state = H.eigenstates()
Nmax = 2
sz = sigmaz()
collapse_list = []
for j in range(Nmax):

    for k in range(j+1,Nmax):
        #FYI: QuTiPs definition of a Lindblad differs from our derivation by a factor of '2'
        rate = (np.absolute(sz.matrix_element(all_state[j].dag(),all_state[k]))**2 * 2 *
                gamma * (n_th+1))
        if rate > 0.0:
            collapse_list.append((np.sqrt(rate)*all_state[j]*all_state[k].dag())) #emission

        rate = (np.absolute(sz.matrix_element(all_state[k].dag(),all_state[j]))**2 * 2 *
                gamma * (n_th))
        if rate > 0.0:
            collapse_list.append((np.sqrt(rate)*all_state[k]*all_state[j].dag())) #absorption
```

Next define the time steps we want to solve the equations for, and run the "mesolve" function

In [145]:

```
# List of times for which the solver should store the state vector
tlist = linspace(0,0.333e-12,1000)
output = mesolve(H, psi0, tlist, collapse_list)
```

In [145]:

```
# List of times for which the solver should store the state vector
tlist = linspace(0,0.333e-12,1000)
output = mesolve(H, psi0, tlist, collapse_list)
```

Lets also run an example where we start in an eigenstate of the Hamiltonian

In [145]:

```
# List of times for which the solver should store the state vector
tlist = linspace(0,0.333e-12,1000)
output = mesolve(H, psi0, tlist, collapse_list)
```

Lets also run an example where we start in an eigenstate of the Hamiltonian

In [146]:

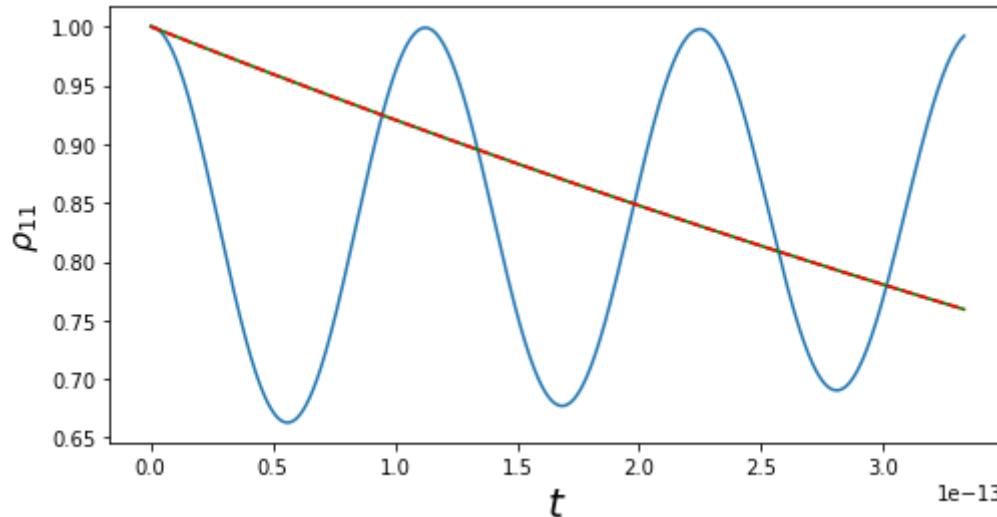
```
ee = all_state[1]
output2 = mesolve(H, ee, tlist, collapse_list)
```

Calculate some expectation values, and compare to analytical result

Calculate some expectation values, and compare to analytical result

In [148]:

```
#Since our initial state was the excited state, we can get the probability of being in the excited state at time t with:  
Pe = expect(output.states, psi0*psi0.dag())  
Pe2 = expect(output2.states, ee*ee.dag())  
  
#For the case where we start in eigenstate, we can compare this to our analytical result  
PeAna = [np.exp(-2 * sz.matrix_element(all_state[0],all_state[1])**2 * gamma * t) for t in tlist]  
  
fig, axes = plt.subplots(1,1, figsize=(8,4))  
axes.plot(tlist, Pe)  
axes.plot(tlist, Pe2, color = 'g')  
axes.plot(tlist, PeAna, color='r',ls='--')  
axes.set_xlabel(r'$t$', fontsize=20)  
axes.set_ylabel(r"$\rho_{11}$", fontsize=16);
```



If we assume non-zero temperature, the steady state (the long-time solution) of the master equation predicts a thermal state.

Analytically, setting the LHS of our equations to zero, and solving, gives the distribution:

$$\rho_{ee} = \frac{n}{(1 + 2n)}$$

$$\rho_{gg} = \frac{1 + n}{(1 + 2n)}$$

Which is equivalent to the thermal distribution:

$$\rho = e^{H_S/k_B T} / Z$$

Lets check this two ways in QuTiP. First with `steadystate()` function, then with the `thermal_dm()` function

Lets check this two ways in QuTiP. First with `steadystate()` function, then with the `thermal_dm()` function

In [156]:

```
#Set non-zero temperature and get new collapse_list first
n_th = 1.3

rhoss = steadystate(H,collapse_list)
print(expect(rhoss,all_state[1]*all_state[1].dag()))
print(expect(rhoss,all_state[0]*all_state[0].dag()))
```

```
0.361111111111127
0.6388888888888888
```

Lets check this two ways in QuTiP. First with `steadystate()` function, then with the `thermal_dm()` function

In [156]:

```
#Set non-zero temperature and get new collapse_list first
n_th = 1.3

rhoss = steadystate(H,collapse_list)
print(expect(rhoss,all_state[1]*all_state[1].dag()))
print(expect(rhoss,all_state[0]*all_state[0].dag()))
```

```
0.361111111111127
0.6388888888888888
```

In [157]:

```
#Compare to analytical prediction
print(n_th / (1+ 2 * n_th))
print((1+n_th) / (1+ 2 * n_th))
```

```
0.361111111111111
0.6388888888888888
```

Lets check this two ways in QuTiP. First with `steadystate()` function, then with the `thermal_dm()` function

In [156]:

```
#Set non-zero temperature and get new collapse_list first
n_th = 1.3

rhoss = steadystate(H,collapse_list)
print(expect(rhoss,all_state[1]*all_state[1].dag()))
print(expect(rhoss,all_state[0]*all_state[0].dag()))
```

```
0.361111111111127
0.6388888888888888
```

In [157]:

```
#Compare to analytical prediction
print(n_th / (1+ 2 * n_th))
print((1+n_th) / (1+ 2 * n_th))
```

```
0.361111111111111
0.6388888888888888
```

In [158]:

```
#Built in function to calculate thermal state with occupation n_th
print(thermal_dm(2, n_th))
```

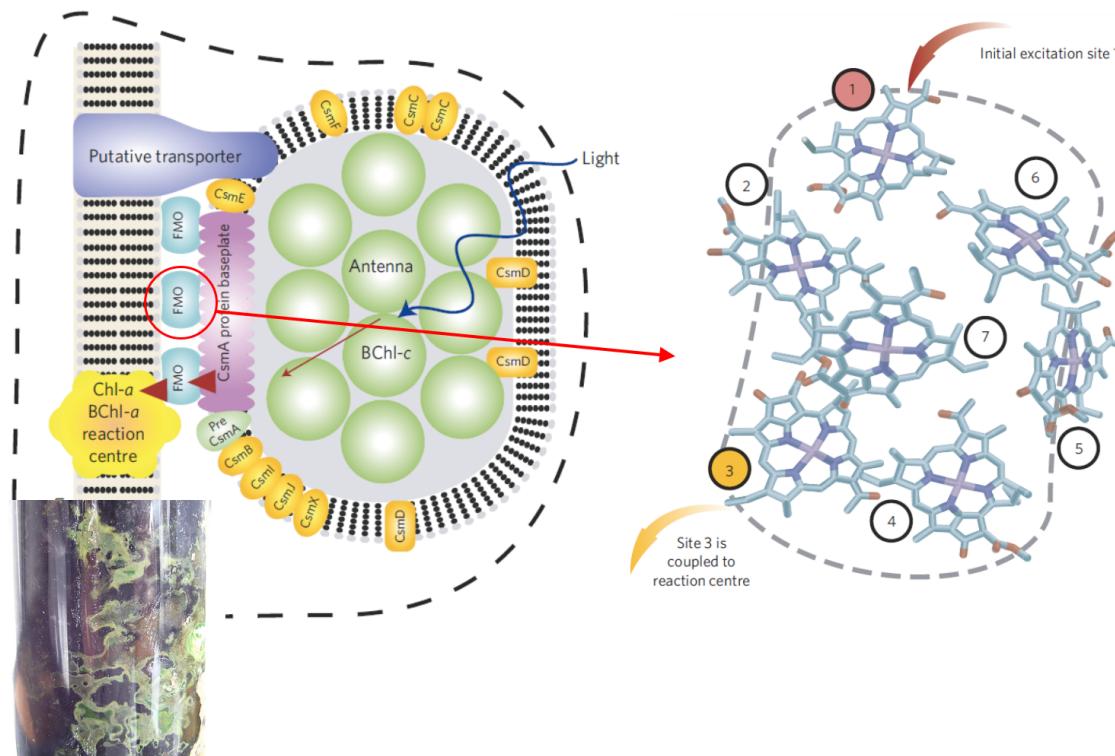
```
Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True
Qobj data =
[[0.63888889 0.          ]
 [0.          0.36111111]]
```

Returning back to our example, can we include more chromophores?

In the single-excitation limit, we wrote 2-sites as a 2-level system.

For 7-sites we can write a 7-level system with Hamiltonian

$$H = \sum_{j=1}^7 \epsilon_j |j\rangle\langle j| + \sum_{j,j'} J_{j,j'} (|j\rangle\langle j'| + c.c)$$



In [247]:

```
H = Qobj([[200, -87.7, 5.5, -5.9, 6.7, -13.7, -9.9],
           [-87.7, 320, 30.8, 8.2, 0.7, 11.8, 4.3],
           [5.5, 30.8, 0, -53.5, -2.2, -9.6, 6.0],
           [-5.9, 8.2, -53.5, 110, -70.7, -17.0, -63.3],
           [6.7, 0.7, -2.2, -70.7, 270, 81.1, -1.3],
           [-13.7, 11.8, -9.6, -17.0, 81.1, 420, 39.7],
           [-9.9, 4.3, 6.0, -63.3, -1.3, 39.7, 230]]))

print(H)

#Rescale H
H = 3e10 * 2 * pi * H
```

```
Quantum object: dims = [[7], [7]], shape = (7, 7), type = oper, isherm = True
Qobj data =
[[200. -87.7 5.5 -5.9 6.7 -13.7 -9.9]
 [-87.7 320. 30.8 8.2 0.7 11.8 4.3]
 [ 5.5 30.8 0. -53.5 -2.2 -9.6 6. ]
 [-5.9 8.2 -53.5 110. -70.7 -17. -63.3]
 [ 6.7 0.7 -2.2 -70.7 270. 81.1 -1.3]
 [-13.7 11.8 -9.6 -17. 81.1 420. 39.7]
 [ -9.9 4.3 6. -63.3 -1.3 39.7 230. ]]
```

In [243]:

```
# initial state.
N = 7 # number of states in the Hilbert space
n = 0 # the state that will be occupied (python counts from zero)

psi0 = basis(N, n)
print(psi0)
```

```
Quantum object: dims = [[7], [1]], shape = (7, 1), type = ket
Qobj data =
[[1.]
 [0.]
 [0.]
 [0.]
 [0.]
 [0.]
 [0.]]
```

The interaction with the bath becomes a little more complicated. Now each chromophore is assumed to interact with an independent bath. Each bath modulates the energy of that chromophore via coupling to the operator:

$$H_I = \sum_{m=1}^7 |m\rangle\langle m| \sum_k g_{k,m} (b_{k,m} + b_{k,m}^\dagger)$$

But, the derivation is done just as before, and we find we can easily adapt our earlier model.

Recall that in the Master equation emission and absorption depend on the environment temperature, and the environment spectral density:

$$J(\omega) = \sum_k g_k^2 \delta(\omega - \omega_k)$$

In many cases we just pretend this is not energy dependent. However, for FMO, it is known that this function has some structure. In Ishizaki and Fleming PNAS 2012 they argue that the environment has a decay time of between 50 fs and 200 fs. We can capture this with a Drude-Lorentz spectral density:

$$J(\omega) = \frac{2\lambda\gamma\omega}{\pi(\omega^2 + \gamma^2)}$$

λ tells us how strong the overall interaction with the environment is

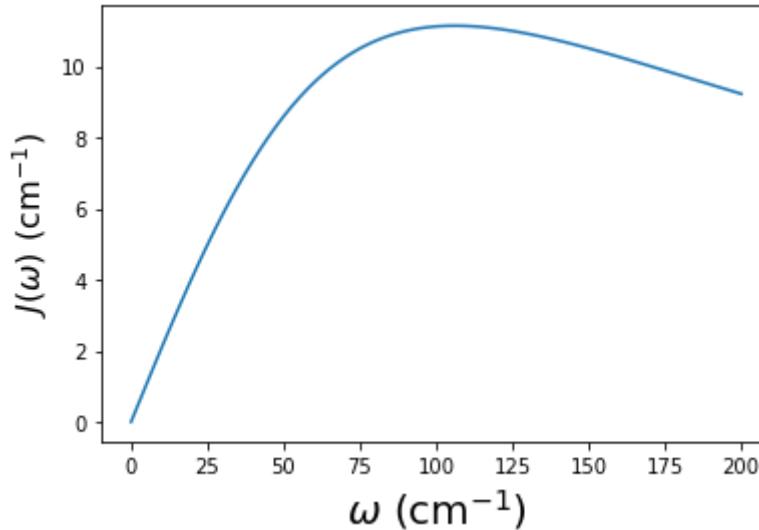
γ tell us the bath cut-off, or decay rate, so is inverse of the bath memory time.

In [244]:

```
def J0(energy):
    #underdamped brownian oscillator
    lam = 35 * 3e10 * 2 * pi #35 cm^-1
    gamma = (1/(50e-15)) #1/50 fs
    return 2 * lam * gamma * (energy)/( pi * ((energy**2) + (gamma**2)))
wlist = linspace(0, 200*3e10*2*pi,100)

fig, axes = plt.subplots(1,1, figsize=(6,4))

axes.plot(wlist/(3e10*2*pi), [J0(w)/(3e10*2*pi) for w in wlist])
axes.set_xlabel(r'$\omega$ (cm$^{-1}$)', fontsize=20)
axes.set_ylabel(r"$J(\omega)$ (cm$^{-1}$)", fontsize=16);
```



In [246]:

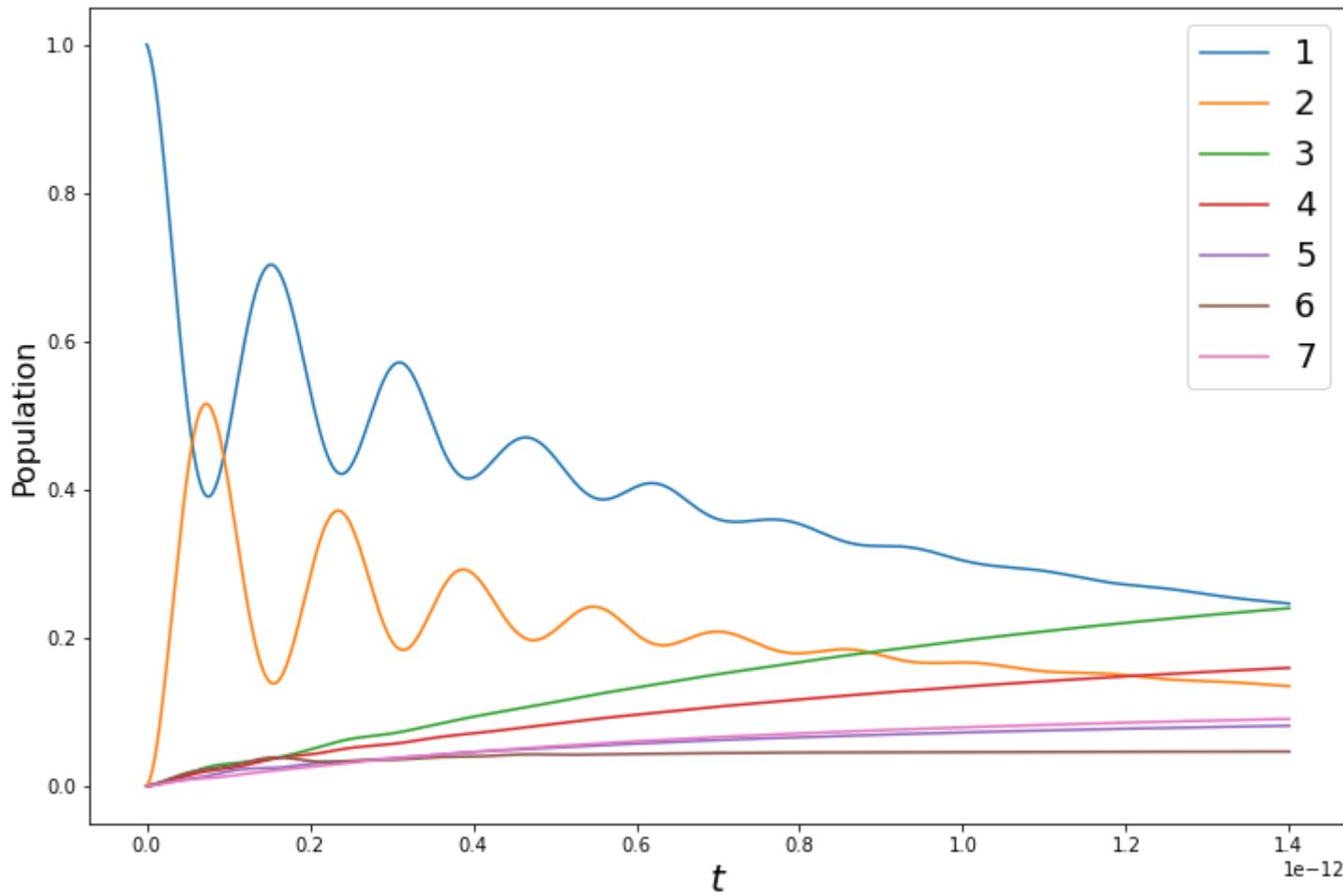
```
#To be more accurate let us use 300 Kelvin as the temperature
Temperature = 300 * 0.6949 * 3e10 * 2 * pi
lam = 35 * 3e10 * 2 * pi #35 cm^-1
gamma = (1/(50e-15)) #1/50 fs
tlist = linspace(0,1.4e-12,1000)
```

```
outputFMO = mesolve(H, psi0, tlist, collapse_list)
fig, axes = plt.subplots(1,1, figsize=(12,8))
for m,Q in enumerate(Q_list):
    axes.plot(tlist, expect(outputFMO.states,Q), label=m+1)
axes.set_xlabel(r'$t$', fontsize=20)
axes.set_ylabel(r"Population", fontsize=16);

axes.set_title('Lindblad solution ', fontsize=24)
axes.legend(loc=0, fontsize=18)
#fig.savefig("fmosec2.png")
```

Out[246]: <matplotlib.legend.Legend at 0x217c3ea88b0>

Lindblad solution



In [234]:

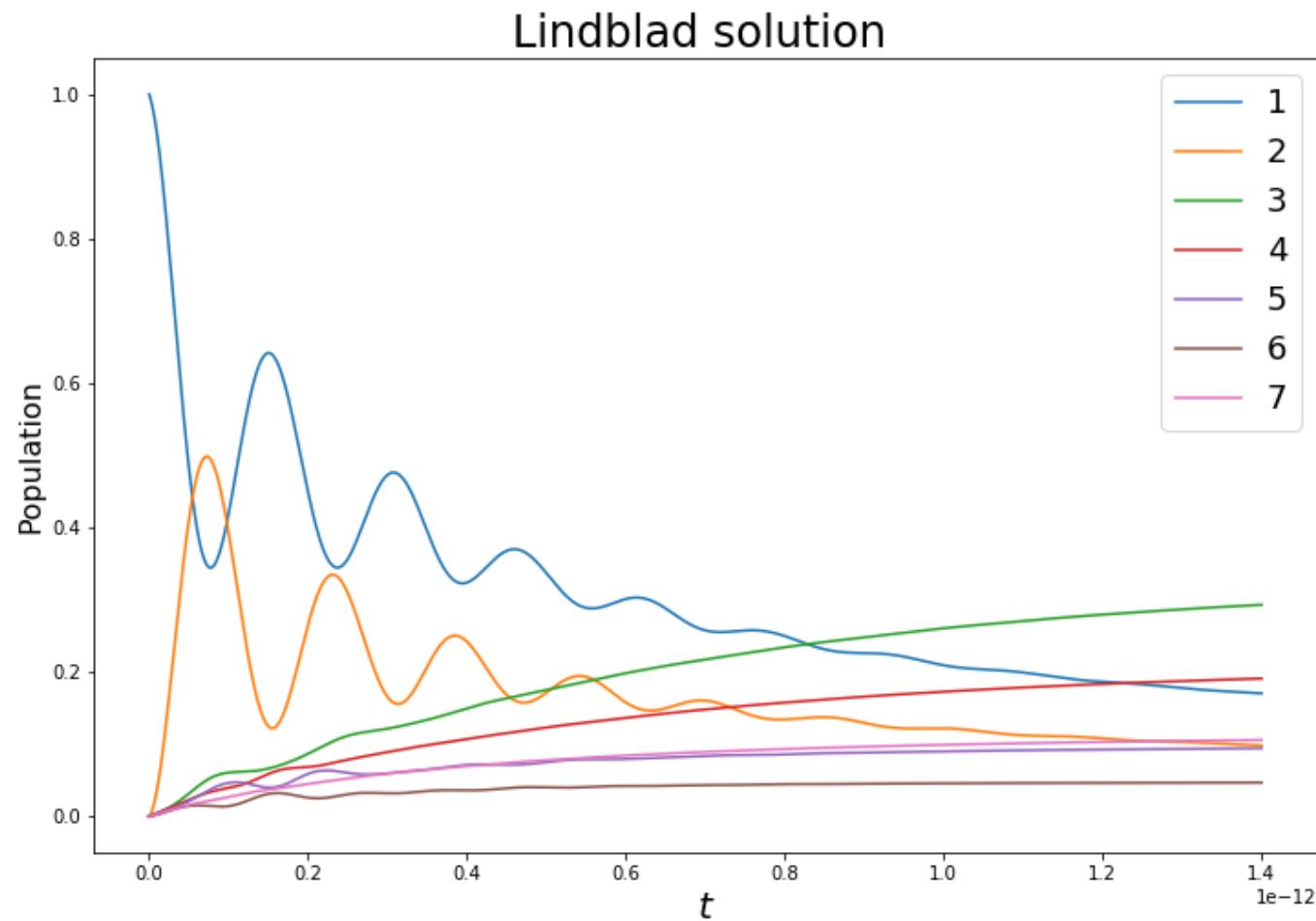
```
#To be more accurate let us use 300 Kelvin as the temperature
Temperature = 300 * 0.6949 * 3e10 * 2 * pi
lam = 35 * 3e10 * 2 * pi #35 cm^-1
gamma = (1/(50e-15)) #1/50 fs
tlist = linspace(0,1.4e-12,1000)
outputFMO = mesolve(H, psi0, tlist, collapse_list)
fig, axes = plt.subplots(1,1, figsize=(12,8))
```

```
for m,Q in enumerate(Q_list):
    axes.plot(tlist, expect(outputFMO.states,Q),label=m+1)
axes.set_xlabel(r'$t$', fontsize=20)
axes.set_ylabel(r"Population", fontsize=16);

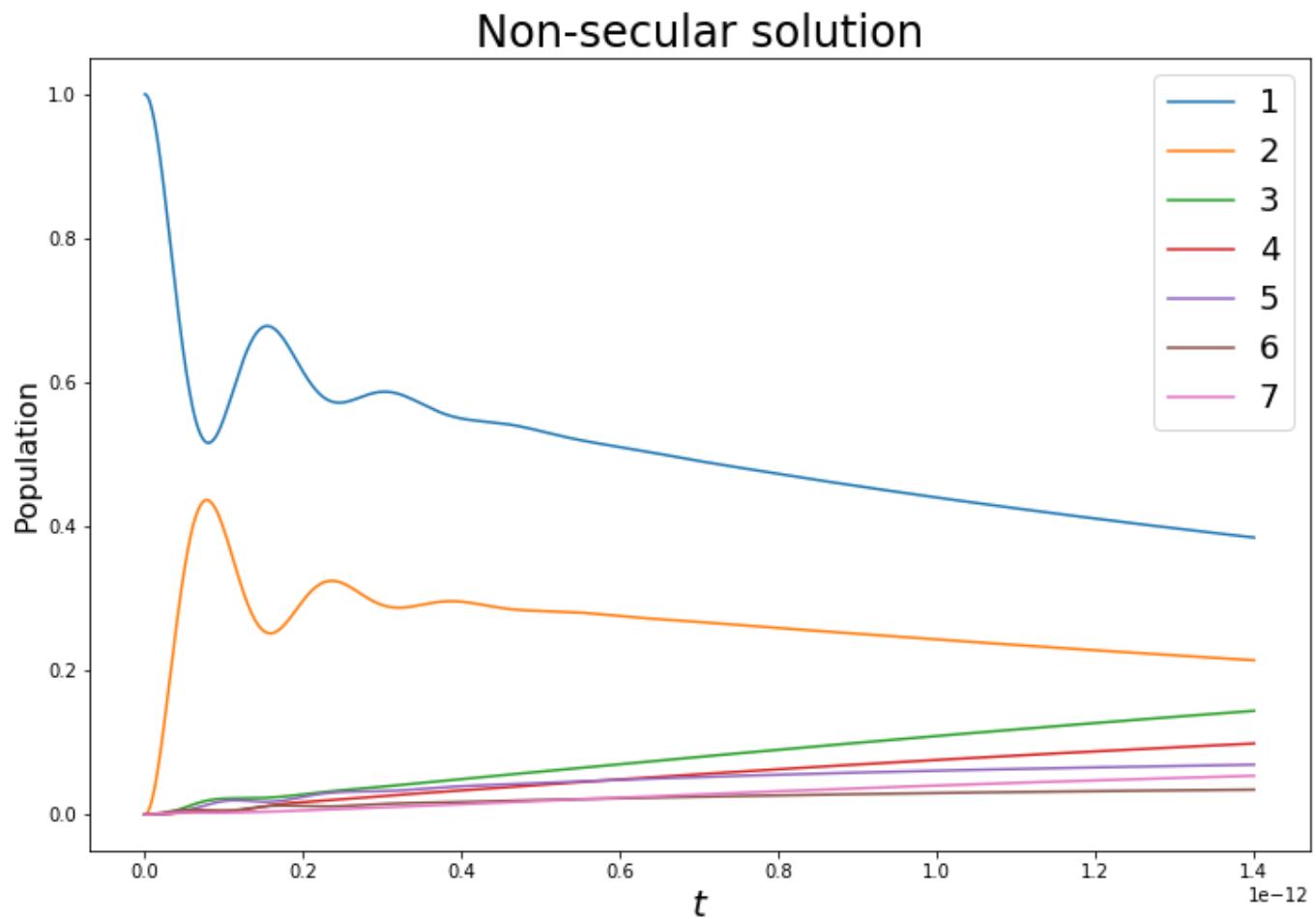
axes.set_title('Lindblad solution ', fontsize=24)
axes.legend(loc=0, fontsize=18)
#fig.savefig("fmosec2.png")
```

Out[234]: <matplotlib.legend.Legend at 0x217c3f8c040>

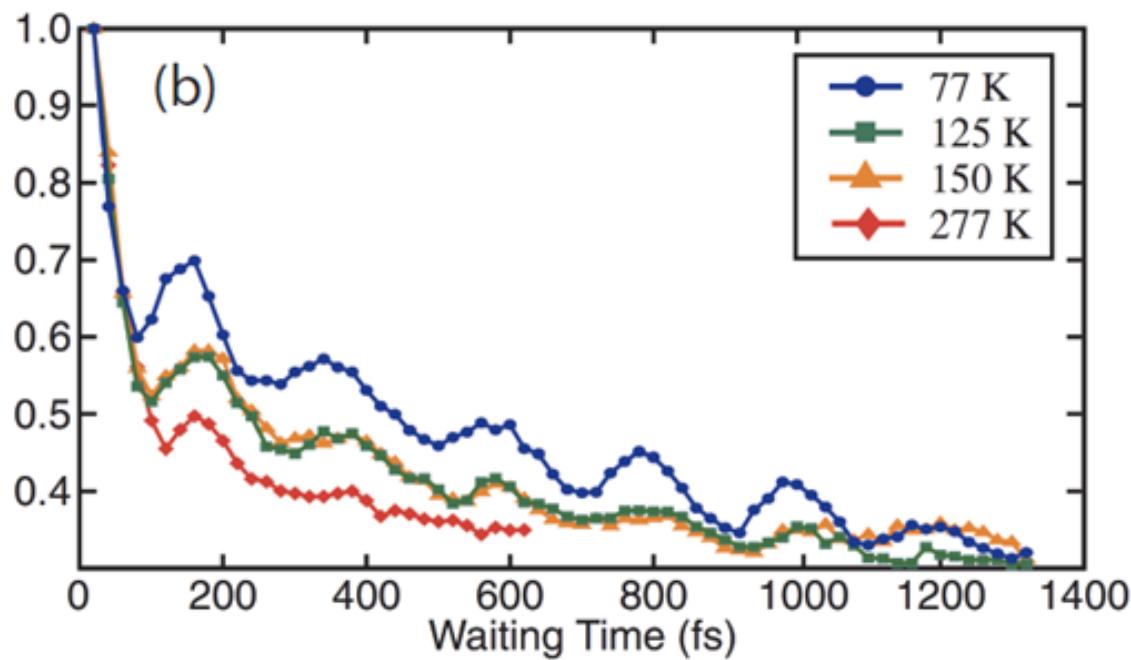
Using the same parameters as above, $\lambda = 35 \text{ cm}^{-1}$, $1/\gamma = 50 \text{ fs}$, $T = 300\text{K}$, we find...



For completeness, lets also look at the solution when we include the dephasing terms we ignored (as well as some non-secular contributions)...



Not too dissimilar time-scales appear... but is our approach valid here? If the memory time of the bath is between 50 and 200 fs, and the coupling strength is of the order of 35 cm^{-1} , both values that may violate the Born-Markov approximations.



Part 3

- Beyond the Master Equation: Exact numerical methods for bosonic baths
 - The Hierarchy Equations of Motion
 - The reaction coordinate method
 - Pseudo-modes
- Comparing different methods
- Evaluating efficiency
- Conclusions

Non-Markovian systems

In other situations the assumption that $\tau_E \ll \tau_S, \tau_R$ also breaks down. E.g.,

- Molecular systems strongly interacting with their environment, e.g., energy transfer in photosynthetic complexes strongly couples to protein "vibrations".
- Hyperfine Interaction between electron spins and nuclear spins in quantum dots
- Superconducting qubits strongly coupled to open transmission lines
- Recent evidence suggests significant non-Markovian noise in IBM-Q simulations

A recent review: "Dynamics of non-Markovian open quantum systems" Inés de Vega and Daniel Alonso Rev. Mod. Phys. 89, 015001

Bosonic environments

The choice of a bosonic environment, or in general, a Gaussian environment, simplifies the problem to some degree.

Using the fact that the cumulants of a **Gaussian** system truncate at second order, one can show (see Breur and Petruccione, 3.6.4, or Jian Ma et al, PRA 2012) that there is a formal solution (assuming that at $t = 0$, $\rho(0) = \rho_S(0) \otimes \rho_E(0)$),

Bosonic environments

The choice of a bosonic environment, or in general, a Gaussian environment, simplifies the problem to some degree.

Using the fact that the cumulants of a **Gaussian** system truncate at second order, one can show (see Breur and Petruccione, 3.6.4, or Jian Ma et al, PRA 2012) that there is a formal solution (assuming that at $t = 0$, $\rho(0) = \rho_S(0) \otimes \rho_E(0)$),

$$\tilde{\rho}_S(t) = \mathcal{T} \exp \left(- \int_0^t dt_2 \int_0^{t_2} A^\times(t_2) [C^R(t_2 - t_1) A(t_1)^\times + i C^I(t_2 - t_1) A(t_1)^\circ] \right) \rho_S(0)$$

Where, as before $A(t)$ is the system operator which couples to the environment, $C(t)$ is the correlation function of the environment, split into real and imaginary parts, $A^\times B \equiv [A, B]$ and $A^\circ B = \{A, B\}$.

\mathcal{T} is the time-ordering operator, which means that all terms have to reordered so that time arguments increase from right to left.

Directly solving or using this equation is intractable; however, by assuming that we can write the correlation functions as a sum of exponentials, and taking repeated time-derivatives, one can derive the **hierarchy equations of motion**.

Again assuming that the initial bath state is a thermal state, and recalling that

$$J(\omega) = \sum_k g_k^2 \delta(\omega - \omega_k)$$

$$C(t) = \int_0^\infty d\omega \{ J(\omega) e^{i\omega t} n(\omega) + J(\omega) e^{-i\omega t} [1 + n(\omega)] \}$$

we can rewrite this as

$$C(t) = \int_0^\infty d\omega J(\omega) \left[\coth\left(\frac{\beta\omega}{2}\right) \cos(\omega t) - i \sin(\omega t) \right]$$

$\beta = 1/k_B T$, the temperature of the initial environment state.

The fact that the environment is fully described by 2nd order correlation functions allows us to identify important degrees of freedom in the environment, and include them in the simulation. We then can have a model which captures quantum effects (coherence/entanglement between system and environment).

For some choices of spectral density, this does have the analytical form of a sum of exponentials, e.g. a Lorentzian. Traditionally, the Drude-Lorentz form is chosen,

$$J(\omega) = \frac{\omega}{\pi} \frac{2\lambda\gamma}{(\gamma^2 + \omega^2)}$$

Inserting this into $C(\tau)$ one finds,

The fact that the environment is fully described by 2nd order correlation functions allows us to identify important degrees of freedom in the environment, and include them in the simulation. We then can have a model which captures quantum effects (coherence/entanglement between system and environment).

For some choices of spectral density, this does have the analytical form of a sum of exponentials, e.g. a Lorentzian. Traditionally, the Drude-Lorentz form is chosen,

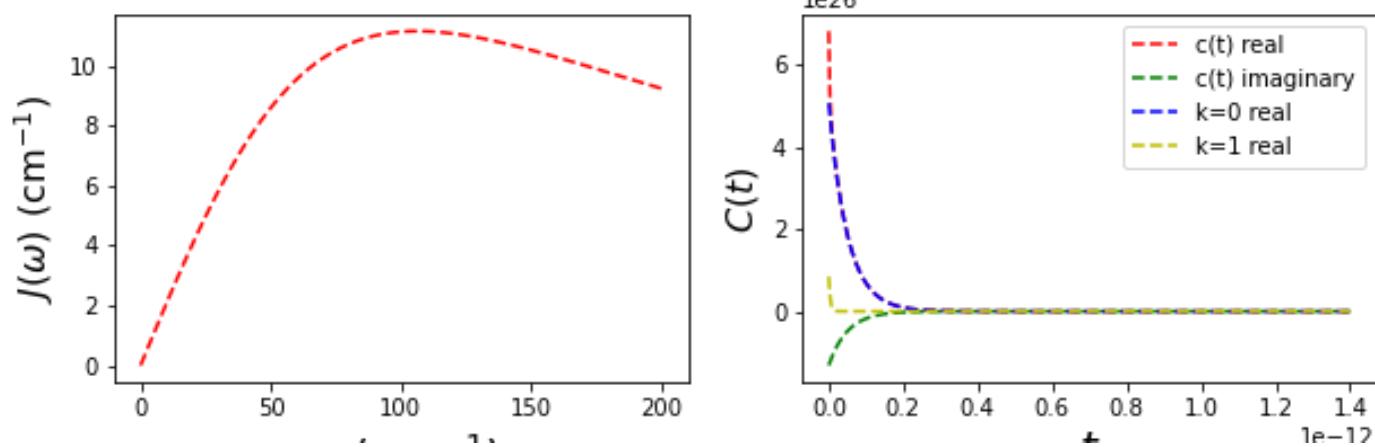
$$J(\omega) = \frac{\omega}{\pi} \frac{2\lambda\gamma}{(\gamma^2 + \omega^2)}$$

Inserting this into $C(\tau)$ one finds,

$$C(\tau) = \sum_{k=0}^{\infty} c_k e^{-v_k |\tau|}$$

Where $v_k = 2\pi k / \beta$ for $k \geq 1$, and $v_0 = \gamma$, and $c_0 = \lambda\gamma[-i + \cot(\beta\gamma/2)]$ and $c_k = (4\lambda\gamma/\beta)v_k/(v_k^2 - \gamma^2)$

E.g., for the parameters given before $\gamma = 1/50$ fs, $\lambda = 35$ cm $^{-1}$, $T = 300$ K



Given these assumptions (**seperable initial condition, bosonic environment initially in a thermal state**), one can derive the **Tanimura and Kubo HEOM**, which looks like the following

$$\dot{\rho}^n(t) = -(iH_S^\times + \sum_{k=0}^M n_k v_k) \rho^n(t)$$

$$-i \sum_{k=0}^M A^\times \rho^{n_k^+}(t)$$

$$-i \sum_{k=0}^M n_k (c_k A \rho^{n_k^-}(t) - c_k^* \rho^{n_k^-}(t) A)$$

This is a set of coupled equations for a hierarchy of "density matrices" labelled by an index n .

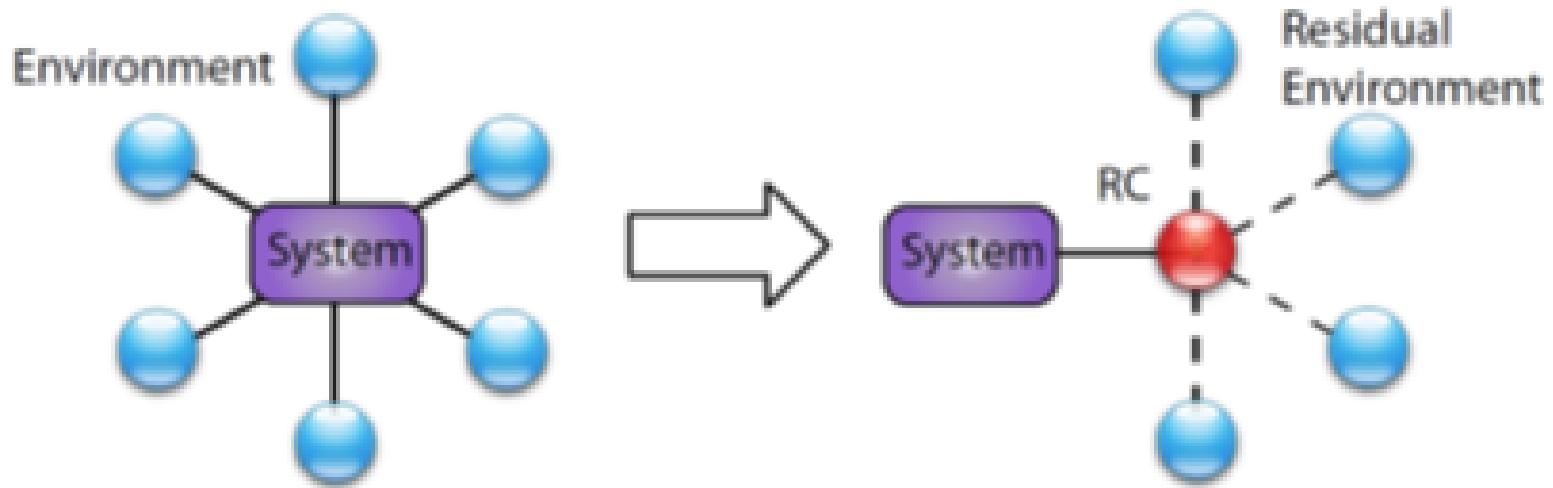
In this case $n = \{n_1, n_2, \dots, n_m, \dots, n_M\}$: it contains M indices, 0 to M , a cut-off in the number of exponentials.

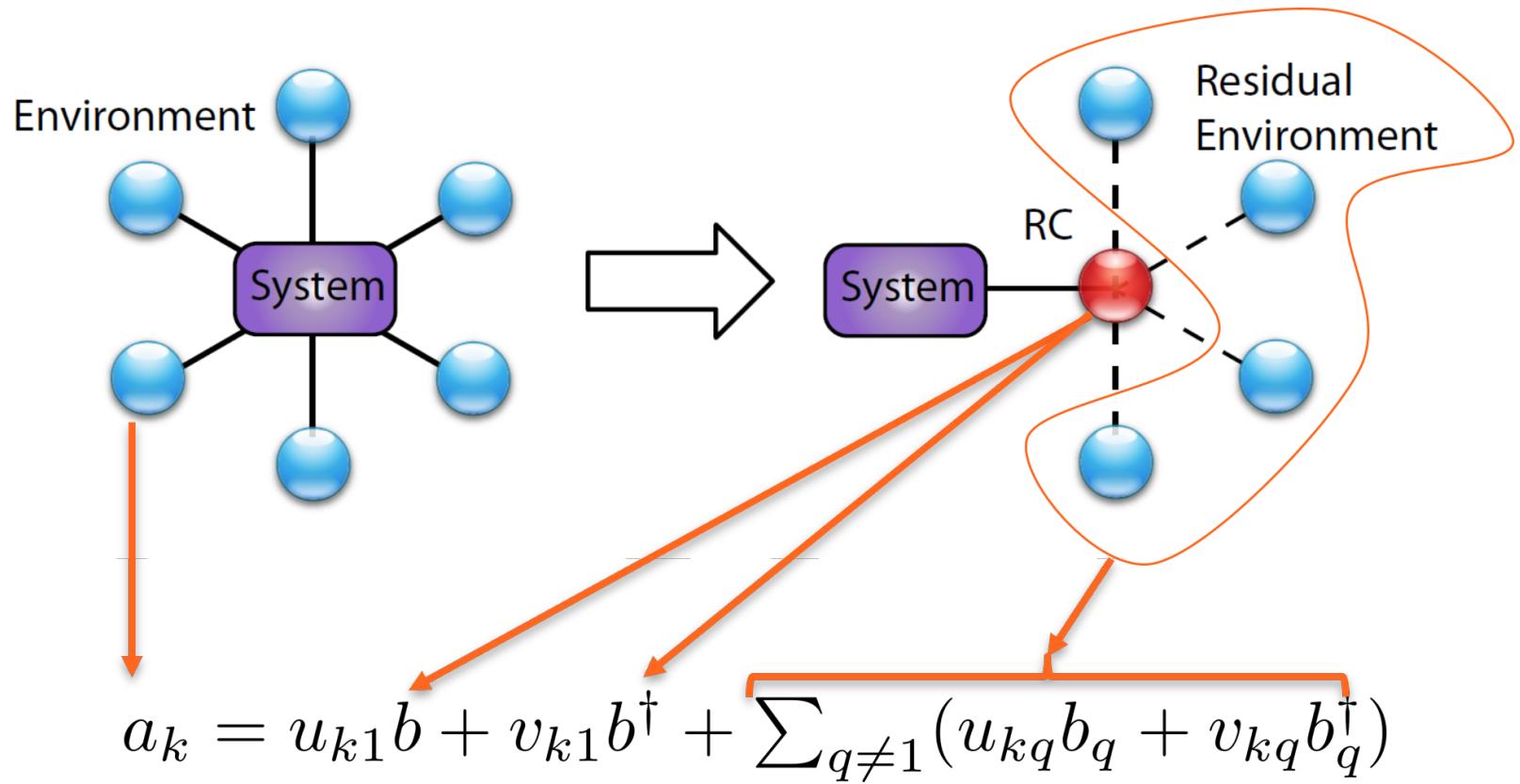
Each element runs from 0 to N_c , an additional cut-off term.

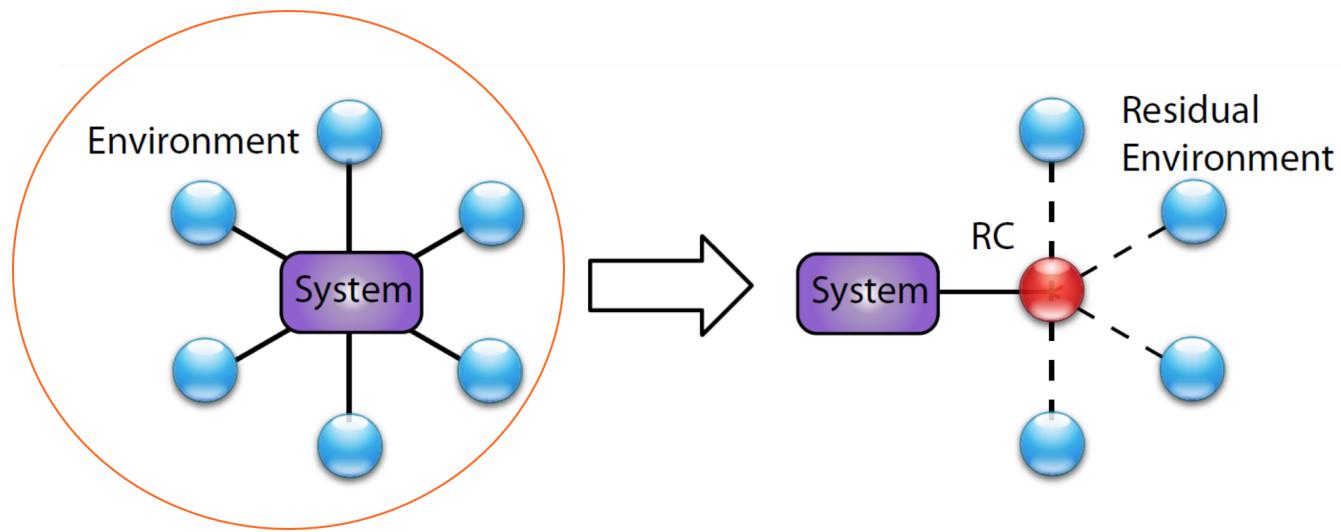
The reaction coordinate

The HEOM is rather complex and requires a custom solver (e.g., BoFiN). The reaction coordinate is a simpler approach (Illes-Smith, Lambert, Nazir, Phys. Rev. A (2014), and J. Chem. Phys. (2016), Strasberg et al. NJP (2017), Nazir and Schaller <https://arxiv.org/abs/1805.08307>).

Starting from our original spin-boson Hamiltonian, we rearrange the bath modes with a unitary transformation:





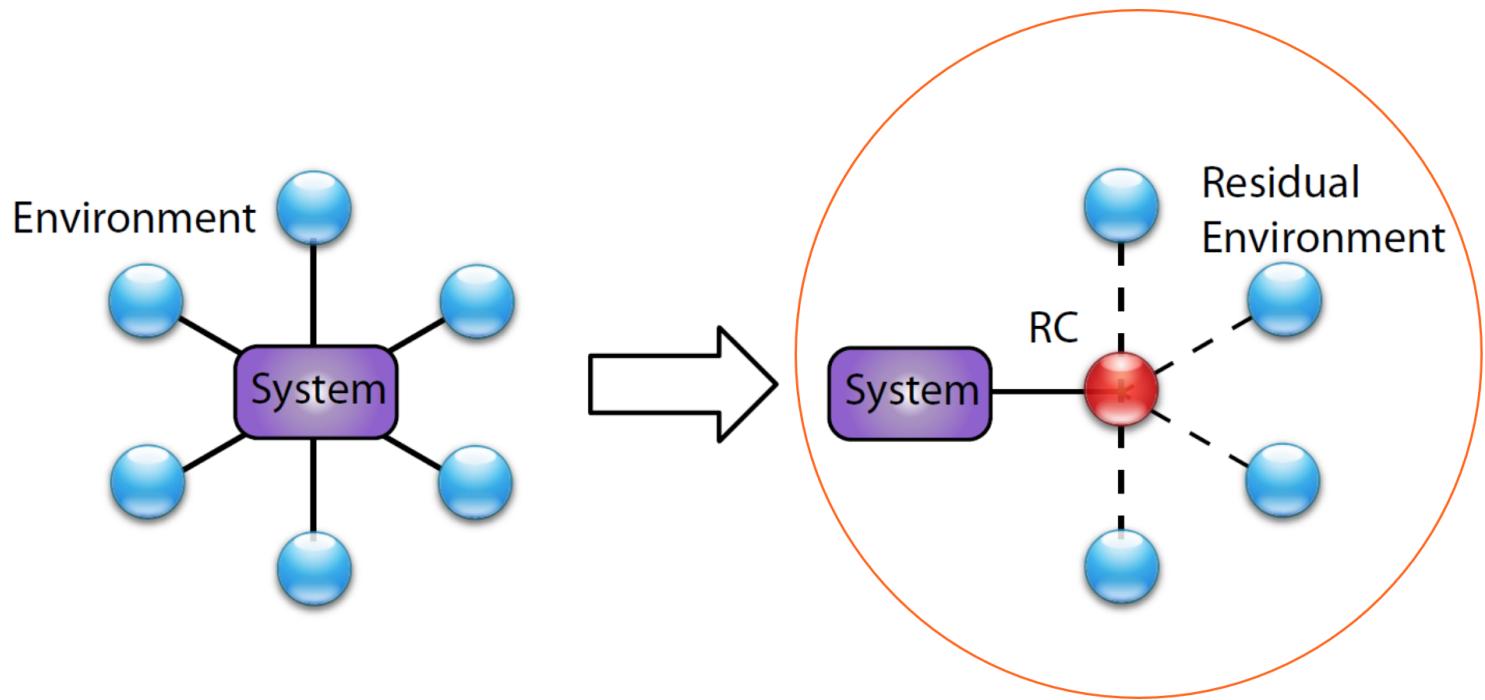


$$a_k = u_{k1}b + v_{k1}b^\dagger + \sum_{q \neq 1} (u_{kq}b_q + v_{kq}b_q^\dagger)$$

$$H_{\text{SB}} = \frac{\epsilon}{2}\sigma_z + \frac{\Delta}{2}\sigma_x + \sum_k \omega_k a_k^\dagger a_k + \sigma_z \sum_k \frac{g_k}{\sqrt{2\omega_k}} (a_k + a_k^\dagger)$$

$$J(\omega) = \pi \sum_k \frac{g_k^2}{2\omega_k} \delta(\omega - \omega_k)$$

One finds a new RC Hamiltonian whose parameters depend on the original one



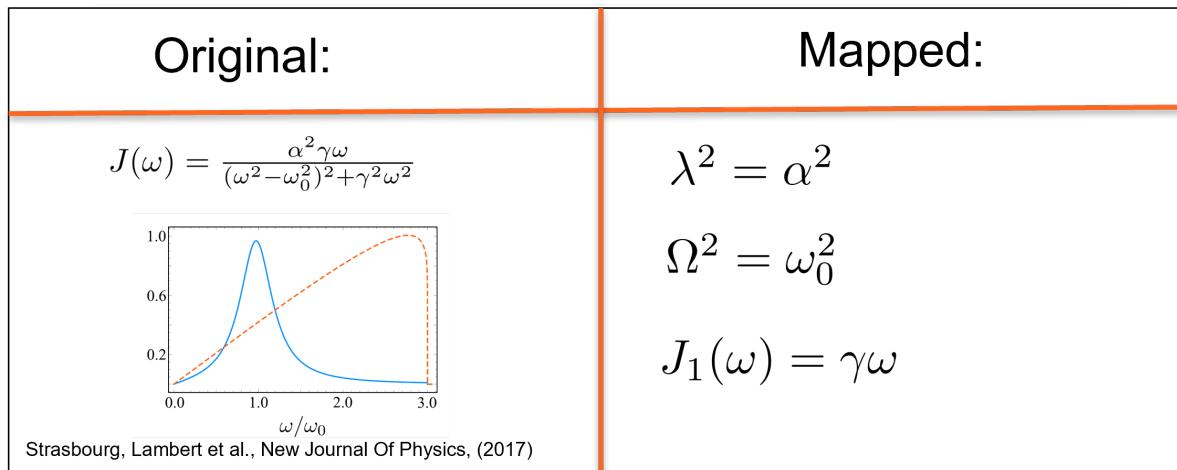
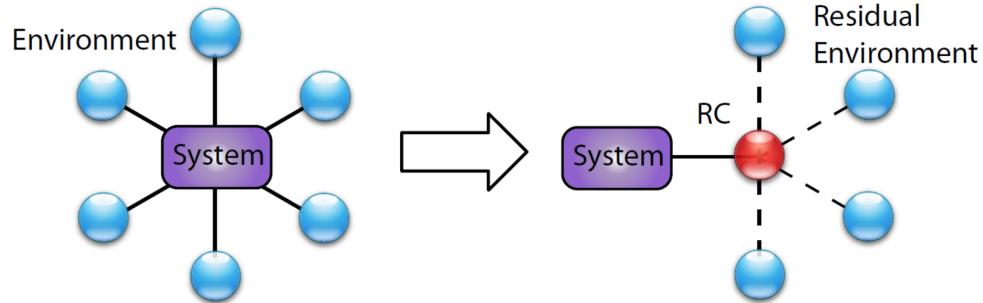
$$\begin{aligned} H_{\text{RC}} = & \frac{\epsilon}{2}\sigma_z + \frac{\Delta}{2}\sigma_x + \Omega b^\dagger b + \frac{\lambda}{\sqrt{2\Omega}}(b + b^\dagger)\sigma_z \\ & + \sum_q \mu_q b_q^\dagger b_q + \sum_q \frac{f_q}{\sqrt{2\mu_q}}(b_q + b_q^\dagger)(b + b^\dagger) \end{aligned}$$

The parameters of new model can be obtained from the original spectral density

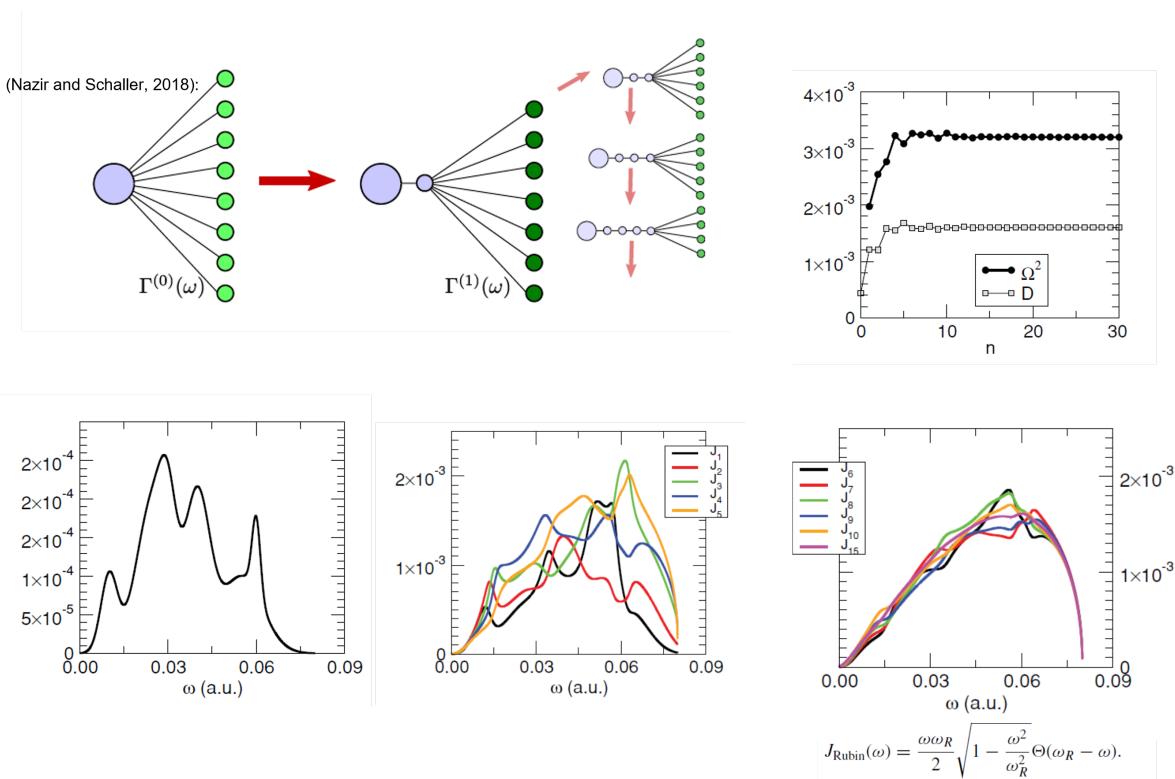
$$\begin{aligned}
 H_{\text{RC}} &= \frac{\epsilon}{2}\sigma_z + \frac{\Delta}{2}\sigma_x + \Omega b^\dagger b + \frac{\lambda}{\sqrt{2\Omega}}(b + b^\dagger)\sigma_z \\
 &+ \sum_q \mu_q b_q^\dagger b_q + \sum_q \frac{f_q}{\sqrt{2\mu_q}}(b_q + b_q^\dagger)(b + b^\dagger)
 \end{aligned}$$

Original:	Mapped:
$J(\omega) = \pi \sum_k \frac{g_k^2}{2\omega_k} \delta(\omega - \omega_k)$	$\lambda^2 = \frac{2}{\pi} \int_0^\infty d\omega \omega J(\omega)$ $\Omega^2 = \frac{2}{\pi \lambda_0^2} \int_0^\infty d\omega \omega^3 J(\omega)$ $J_1(\omega) = \lambda^2 J(\omega) / W_0^+(\omega) ^2$ $W_0(z) = \frac{1}{\pi} \int_{-\infty}^\infty d\omega \frac{J(\omega)}{\omega - z}$

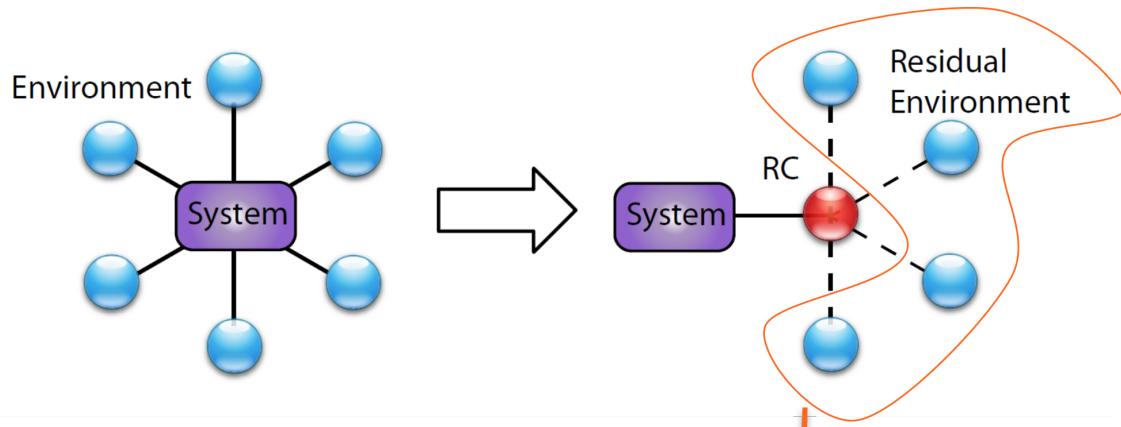
We can use an underdamped Drude-Lorentze SD as a specific example:



Aside: In principle one can repeat this process following Martinazzo et al. J. Chem. Phys. 2011:



For practical application, we need to treat the residual bath:



$$\begin{aligned}
 H_{\text{RC}} = & \frac{\epsilon}{2} \sigma_z + \frac{\Delta}{2} \sigma_x + \Omega b^\dagger b + \frac{\lambda}{\sqrt{2\Omega}} (b + b^\dagger) \sigma_z \\
 & + \sum_q \mu_q b_q^\dagger b_q + \sum_q \frac{f_q}{\sqrt{2\mu_q}} (b_q + b_q^\dagger)(b + b^\dagger)
 \end{aligned}$$

Next step: master equation description of residual environment. (Illes-Smith, Lambert, Nazir, Phys. Rev. A (2014), and J. Chem. Phys. (2016), Strasberg et al. NJP (2017)).

Born-Markov Assumptions: coupling to residual bath is weak, and memory-less (Markovian).

When is this ok? Residual bath spectral density is $J_1(\omega) = \gamma\omega$, implies $\gamma \ll \lambda, \Omega, \omega_q$

Making standard Born-Markov (but non-secular) approximations:

$$\begin{aligned}\frac{d}{dt}\rho(t) &= -i[H_S, \rho(t)] - [s, [\chi, \rho(t)]] + [s, \{\theta, \rho(t)\}] \\ s &= a + a^\dagger\end{aligned}$$

$$\chi = \frac{1}{2} \sum_{kl} J_1(\omega) \coth \frac{\beta\omega_{kl}}{2} s_{kl} |k\rangle\langle l| \quad \theta = \frac{1}{2} \sum_{kl} J_1(\omega) s_{kl} |k\rangle\langle l|$$

Comparing BMS master equation, HEOM and RC

Before looking again at the FMO example, lets try and see the breakdown of the BMS (Born-Markov-Secular) master equation as we increase the bath correlation time with some simpler (unitless) parameters, and with just a 2-level system.

First try lets try some parameters where BMS should work (in units of system energy ϵ):

$\gamma = 10\epsilon$ Very large: short memory time, Markovian

$\lambda = 0.05\epsilon$ Very weak: Born approximation should be valid.

$\beta = \epsilon$ Overall dissipation rate not too large because of temperature? Secular approximation should be ok.

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

(recall everything here is in units of $\epsilon = 1$)

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

(recall everything here is in units of $\epsilon = 1$)

In [36]:

```
print(1/(np.pi * j0(1.0) * nbs(1.0, temperature)))
```

173.54646467436356

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

(recall everything here is in units of $\epsilon = 1$)

In [36]:

```
print(1/(np.pi * j0(1.0) * nbs(1.0, temperature)))
```

173.54646467436356

The largest environment correlation times are $\tau_E \approx 1/\gamma, 1/2\pi k_B T$

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

(recall everything here is in units of $\epsilon = 1$)

In [36]:

```
print(1/(np.pi * j0(1.0) * nbs(1.0,temperature)))
```

173.54646467436356

The largest environment correlation times are $\tau_E \approx 1/\gamma, 1/2\pi k_B T$

In [37]:

```
print(1/gamma)
print(1/(2 * np.pi * temperature))
```

0.1

0.15915494309189535

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

(recall everything here is in units of $\epsilon = 1$)

In [36]:

```
print(1/(np.pi * j0(1.0) * nbs(1.0,temperature)))
```

173.54646467436356

The largest environment correlation times are $\tau_E \approx 1/\gamma, 1/2\pi k_B T$

In [37]:

```
print(1/gamma)
print(1/(2 * np.pi * temperature))
```

0.1

0.15915494309189535

The system time scale is $\tau_S \approx 1/\epsilon$, so $\tau_E \ll \tau_S \ll \tau_R$ for the Born-Markov-Secular approximation seems valid. Let's check..

In [207]:

```
# Configure the solver
from qutip.nonmarkov.heom import HSolverDL
hsolver = HSolverDL(Hsys, Q, lam, temperature, Ncut, Nk, gamma, stats=True, )
resultHEOM = hsolver.run(rho0, tlist)
```

In [207]:

```
# Configure the solver
from qutip.nonmarkov.heom import HSolverDL
hsolver = HSolverDL(Hsys, Q, lam, temperature, Ncut, Nk, gamma, stats=True, )
resultHEOM = hsolver.run(rho0, tlist)
```

In [209]:

```
#lets compare to the BMS solution,
plus = (basis(2,0)+basis(2,1))/np.sqrt(2.)
minus = (basis(2,0)-basis(2,1))/np.sqrt(2.)
smt = minus*plus.dag()
resultMEsec = mesolve(Hsys, rho0, tlist, [np.sqrt(2*np.pi)*J0(1.0)*(nbs(1.0,temperature)+1))*smt,np.s
resultMEnonsec = Lnonsec(Hsys, Q, tlist, rho0, temperature, [])
```

2

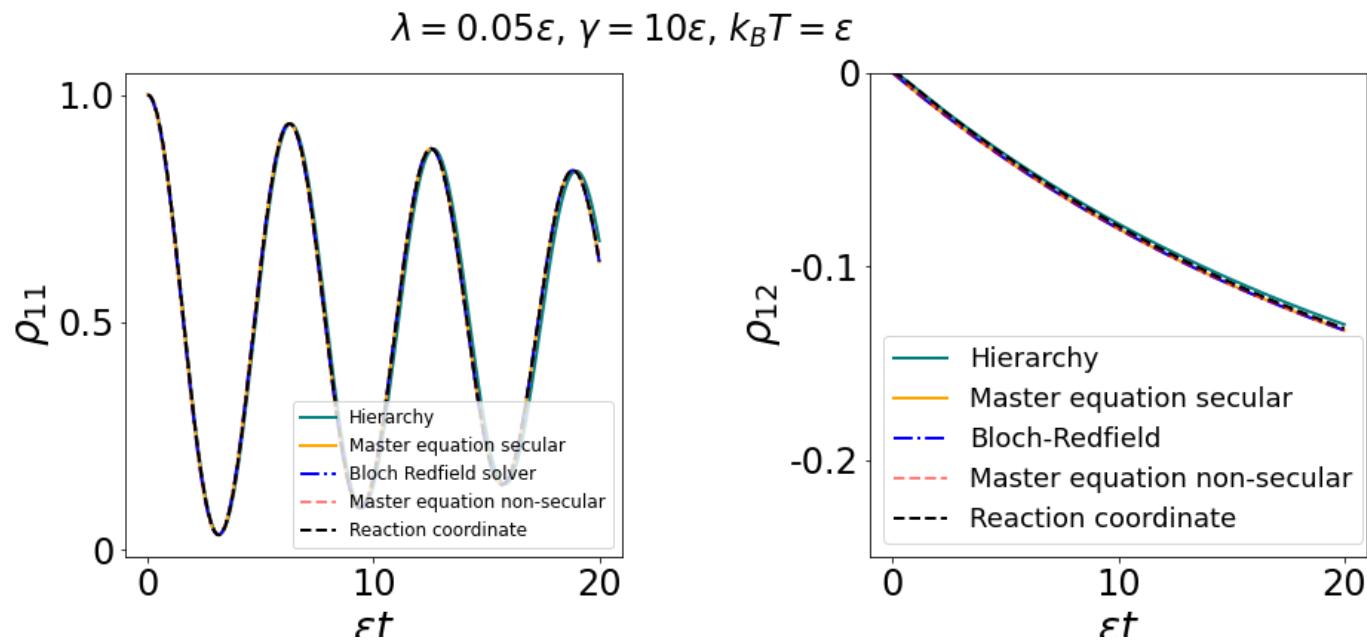
In [207]:

```
# Configure the solver
from qutip.nonmarkov.heom import HSolverDL
hsolver = HSolverDL(Hsys, Q, lam, temperature, Ncut, Nk, gamma, stats=True, )
resultHEOM = hsolver.run(rho0, tlist)
```

In [209]:

```
#lets compare to the BMS solution,
plus = (basis(2,0)+basis(2,1))/np.sqrt(2.)
minus = (basis(2,0)-basis(2,1))/np.sqrt(2.)
smt = minus*plus.dag()
resultMEsec = mesolve(Hsys, rho0, tlist, [np.sqrt(2*np.pi)*J0(1.0)*(nbs(1.0,temperature)+1))*smt,np.s
resultMEnonsec = Lnonsec(Hsys, Q, tlist, rho0, temperature, [])
```

2



Next try lets try some parameters where BMS should break,

$\gamma = 0.1\epsilon$ Narrow environment, long memory time... Non-Markovian...

$\lambda = 0.2\epsilon$ Large coupling, Born-approximation breaks.

$\beta = \epsilon$ Same as before...

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

In [165]:

```
print(1./(np.pi*nbs(1.0,temperature)))
```

43.386616168590876

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

In [165]:

```
print(1./(np.pi*nbs(1.0,temperature)))
```

43.386616168590876

The largest environment correlation times are $\tau_E \approx 1/\gamma, 1/2\pi k_B T$

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

In [165]:

```
print(1./(np.pi*nbs(1.0,temperature)))
```

43.386616168590876

The largest environment correlation times are $\tau_E \approx 1/\gamma, 1/2\pi k_B T$

In [166]:

```
print(1/gamma)
print(1/(2 * np.pi * temperature))
```

10.0

0.15915494309189535

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

In [165]:

```
print(1./(np.pi*nbs(1.0,temperature)))
```

43.386616168590876

The largest environment correlation times are $\tau_E \approx 1/\gamma, 1/2\pi k_B T$

In [166]:

```
print(1/gamma)
print(1/(2 * np.pi * temperature))
```

10.0
0.15915494309189535

The system time scale is $\tau_S \approx 1/\epsilon$, so $\tau_S \ll \tau_R$ seems ok, but now $\tau_E > \tau_S$, so Markov approximation should break

In [216]:

```
# Configure the solver
from qutip.nonmarkov.heom import HSolverDL
hsolver = HSolverDL(Hsys, Q, lam, temperature, Ncut, Nk, gamma, stats=True)
resultHEOM = hsolver.run(rho0, tlist)
```

In [216]:

```
# Configure the solver
from qutip.nonmarkov.heom import HSolverDL
hsolver = HSolverDL(Hsys, Q, lam, temperature, Ncut, Nk, gamma, stats=True)
resultHEOM = hsolver.run(rho0, tlist)
```

In [217]:

#lets compare to the BMS solution,

```
resultMEsec = mesolve(Hsys, rho0, tlist, [np.sqrt(2*np.pi)*J0(1.0)*(nbs(1.0,temperature)+1.))*smt,np.
resultMEnonsec = Lnonsec(Hsys, Q, tlist, rho0, temperature, [])
```

2

In [216]:

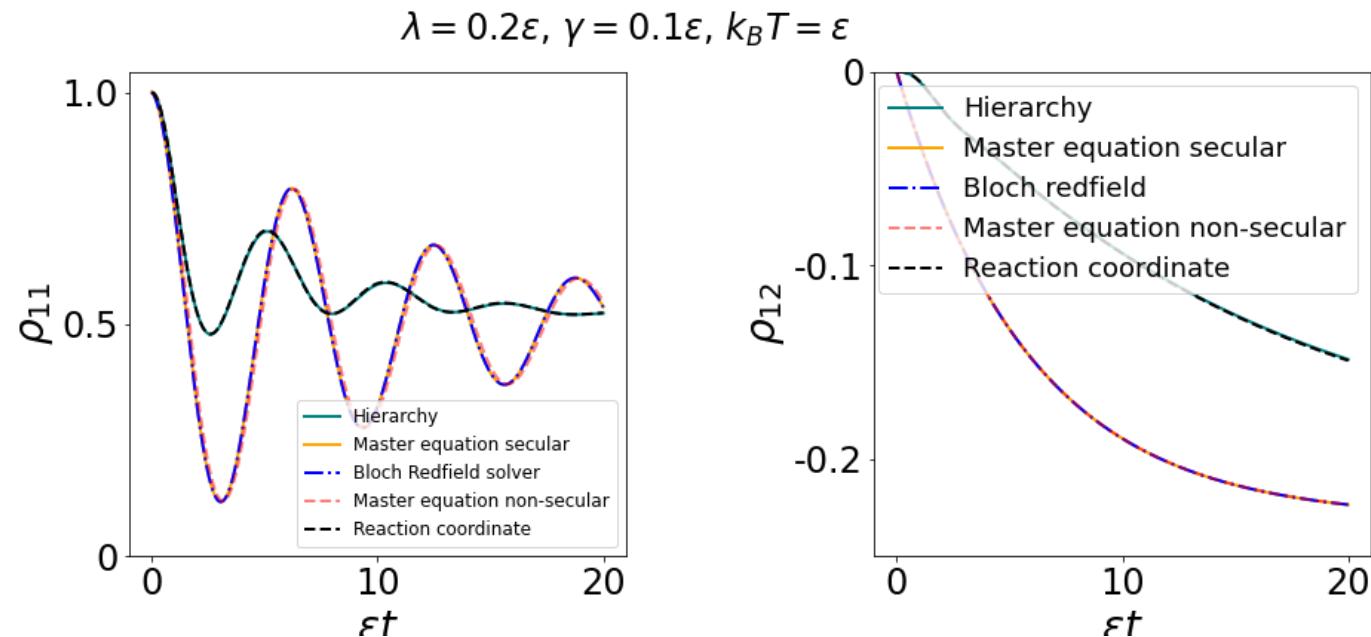
```
# Configure the solver
from qutip.nonmarkov.heom import HSolverDL
hsolver = HSolverDL(Hsys, Q, lam, temperature, Ncut, Nk, gamma, stats=True)
resultHEOM = hsolver.run(rho0, tlist)
```

In [217]:

#lets compare to the BMS solution,

```
resultMEsec = mesolve(Hsys, rho0, tlist, [np.sqrt(2*np.pi)*J0(1.0)*(nbs(1.0,temperature)+1.))*smt,np.
resultMEnonsec = Lnonsec(Hsys, Q, tlist, rho0, temperature, [])
```

2



Next try lets try some parameters where secular approximation should break, but Born-Markov is ok

$\gamma = 10\epsilon$ As before (Markovian)

$\lambda = 0.5\epsilon$ Strong coupling..

$\beta = 0.25\epsilon$ Large temperature...

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

In [48]:

```
print(1./(np.pi*j0(1.0)*(nbs(1.0,temperature)+1.)))
```

2.2341120909788104

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

In [48]:

```
print(1./(np.pi*nbs(1.0,temperature)+1.)))
```

2.2341120909788104

The largest environment correlation times are $\tau_E \approx 1/\gamma, 1/2\pi k_B T$

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

In [48]:

```
print(1./(np.pi*j0(1.0)*(nbs(1.0,temperature)+1.)))
```

2.2341120909788104

The largest environment correlation times are $\tau_E \approx 1/\gamma, 1/2\pi k_B T$

In [49]:

```
print(1/gamma)
print(1/(2 * np.pi * temperature))
```

0.1

0.039788735772973836

The relaxation time is roughly $\tau_R \approx 1/\pi J(\epsilon)n(\epsilon)$

In [48]:

```
print(1./(np.pi*j0(1.0)*(nbs(1.0,temperature)+1.)))
```

2.2341120909788104

The largest environment correlation times are $\tau_E \approx 1/\gamma, 1/2\pi k_B T$

In [49]:

```
print(1/gamma)
print(1/(2 * np.pi * temperature))
```

0.1

0.039788735772973836

The system time scale is $\tau_S \approx 1/\epsilon$, so now $\tau_R \approx \tau_S$, so Secular appoximation should break

In [223]:

```
# Configure the solver
from qutip.nonmarkov.heom import HSolverDL
hsolver = HSolverDL(Hsys, Q, lam, temperature, Ncut, Nk, gamma, stats=True)
resultHEOM = hsolver.run(rho0, tlist)
```

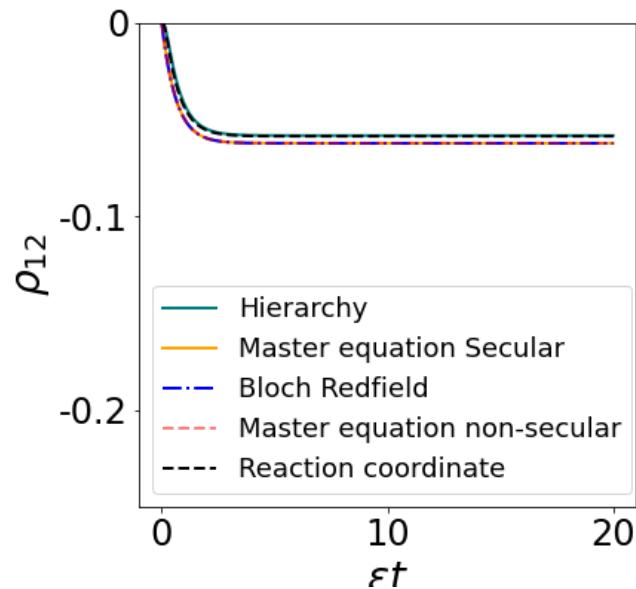
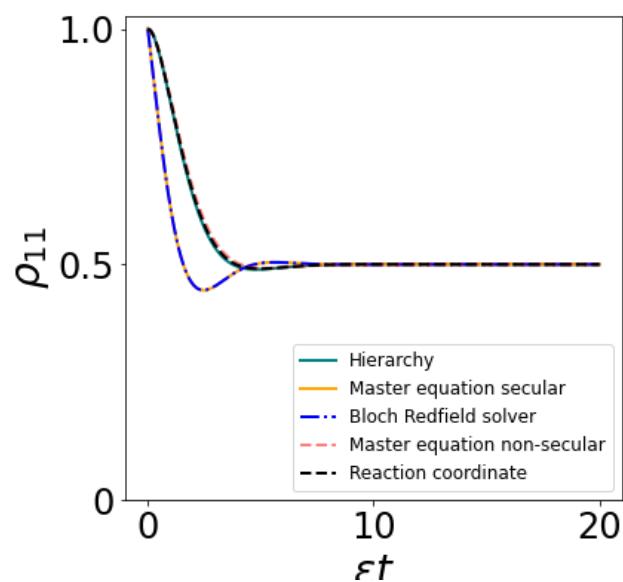
In [223]:

```
# Configure the solver
from qutip.nonmarkov.heom import HSolverDL
hsolver = HSolverDL(Hsys, Q, lam, temperature, Ncut, Nk, gamma, stats=True)
resultHEOM = hsolver.run(rho0, tlist)
```

In [224]:

```
#lets compare to the BMS solution,
resultMEsec = mesolve(Hsys, rho0, tlist, [np.sqrt(2*np.pi)*J0(1.0)*(nbs(1.0,temperature)+1.))*smt,np.
resultMEnonsec = Lnonsec(Hsys, Q, tlist, rho0, temperature, [])
```

$$\lambda = 0.5\epsilon, \gamma = 10\epsilon, k_B T = 4\epsilon$$



The HEOM and FMO

Let us return back to the FMO result. We saw the simulation from our Lindblad model already. Similarly, lets use the same spectral density and temperature in the HEOM model.

The HEOM and FMO

Let us return back to the FMO result. We saw the simulation from our Lindblad model already. Similarly, lets use the same spectral density and temperature in the HEOM model.

$$\lambda = 35 \text{ cm}^{-1}, 1/\gamma = 50 \text{ fs}, T = 300\text{K},$$

A

B

C

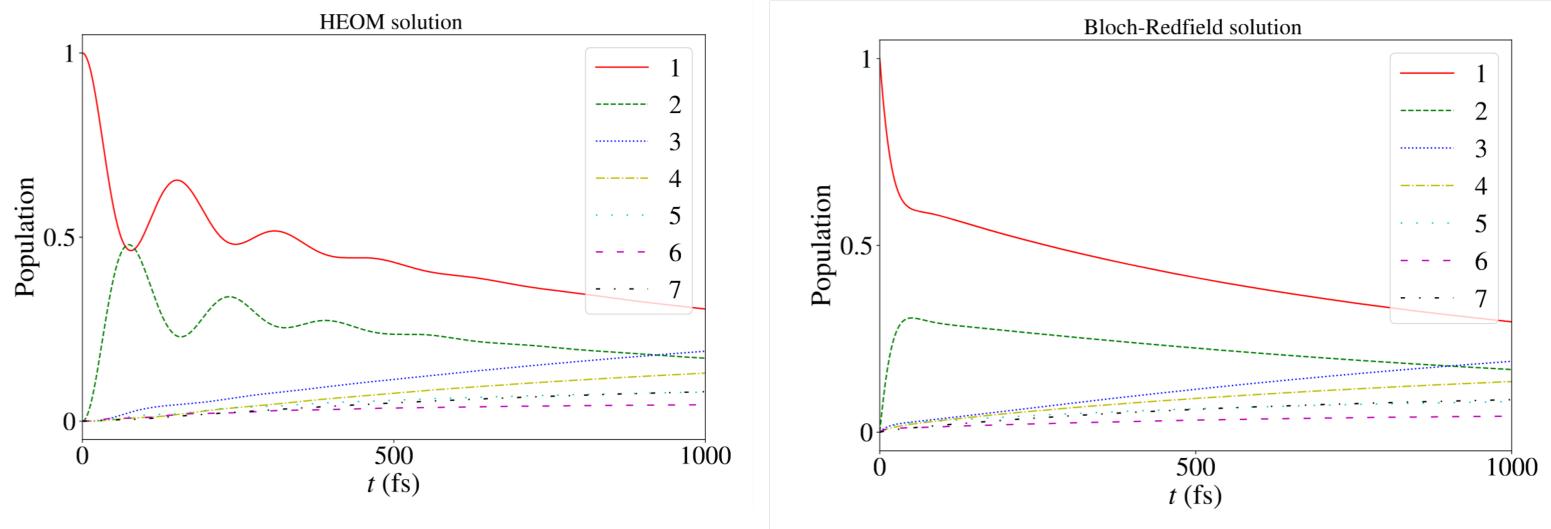
Bloch-redfield does a little better (== Lindblad+dephasing but Secular)

A

B

The difference here is not so clear... when do perturbative methods clearly fail?

Making the environment more 'non-Markovian' makes the coherence last longer



$$\lambda = 35 \text{ cm}^{-1}, \gamma^{-1} = 166 \text{ fs} \quad T = 300 \text{ K.}$$

What about changing the coupling strength? How can we capture which model predicts a better long-time efficiency of energy transport?

Let's return back to our two-level example (these results are from J. Chem. Phys. (2016), Illes-Smith, Dijkstra, Lambert and Nazir).

$$H_S = \frac{\epsilon}{2}\sigma_z + \frac{\Delta}{2}\sigma_x$$

where the eigenstates of σ_z represent the position of the excitation on the two chromophores, $\sigma_z = |X_1, 0_2\rangle\langle X_1, 0_2| - |0_1, X_2\rangle\langle 0_1, X_2|$

$$\begin{array}{c} |E_1\rangle = |X_1\rangle \otimes |0_2\rangle \\ \hline \updownarrow \quad \epsilon = \epsilon_1 - \epsilon_2 \\ \hline |E_1\rangle = |0_1\rangle \otimes |X_2\rangle \end{array}$$

Efficiency relies on how quickly the excitation is moved from one chromophore to another (ultimately so that it can be moved to a reaction center).

We can evaluate the dynamics of different models, and fit the long-time behavior to a rate:

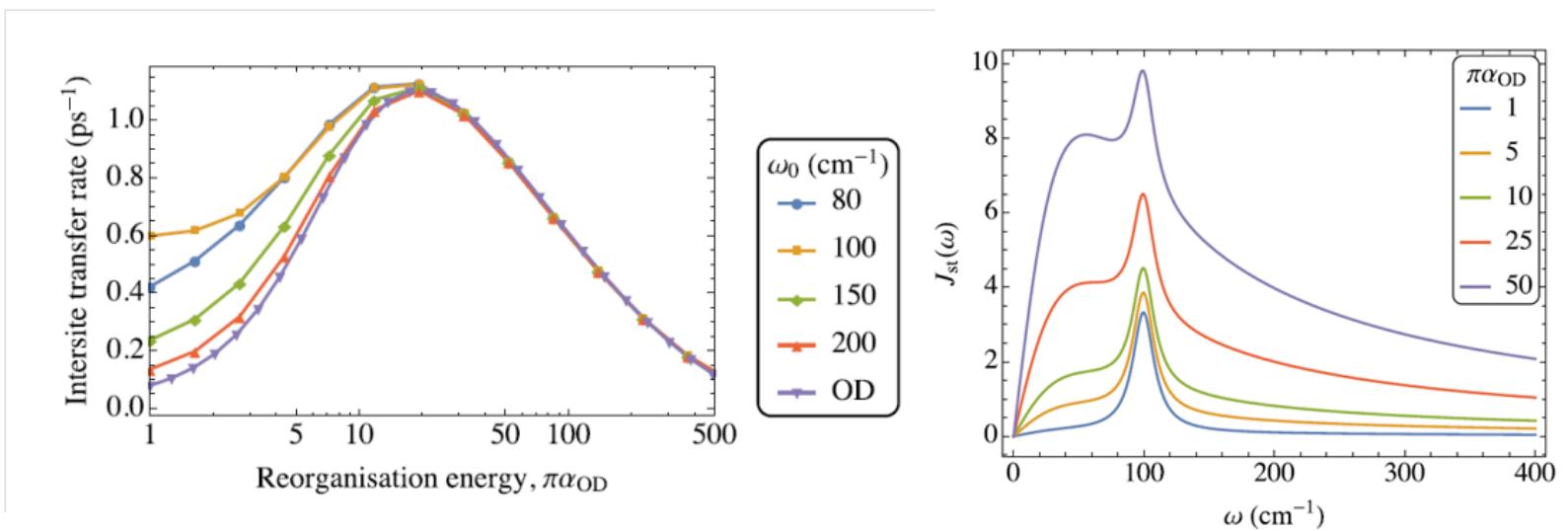
$$\frac{dP_1(t)}{dt} = -k_{1 \rightarrow 2} P_1 + k_{2 \rightarrow 1} P_2$$

$$\frac{dP_2(t)}{dt} = k_{1 \rightarrow 2} P_1 - k_{2 \rightarrow 1} P_2$$

A

B

We can also evaluate this for spectral densities with broad background and resonant structures:



Conclusions

The study of open quantum systems lets us model how a system interacts with its environment in various limits:

- Weak coupling, no bath memory (Markov), and weak damping gives us the Lindblad Born-Markov-Secular master equation

$$\tau_E \ll \tau_S \ll \tau_R$$

- Exact methods (HEOM, reaction coordinate, pseudo-modes (Lambert et al, Nat. Comm. 2019) give insight into strongly correlated system-bath problems when this approximation no longer holds, and everything occurs on comparable time-scales

$$\tau_E \approx \tau_S \approx \tau_R$$

How does this help us study light-harvesting?

- Comparing exact models, approximate models, and experimental data gives us insight more than each one alone.
- E.g., in this example we saw that while the noise suppresses coherent oscillations, non-Markovian (resonant) environments can speed-up the transfer time of electronic energy across the whole complex, which may be important for improving efficiency.

All of these tools are useful for studying in noise in any type of quantum system:

1- quantum computers, understanding and mitigating noise

2- quantum optics, measurement and control

3- condensed matter systems, electron transport, Fermionic environments

nwlambert@gmail.com

Generalizing the spectral density

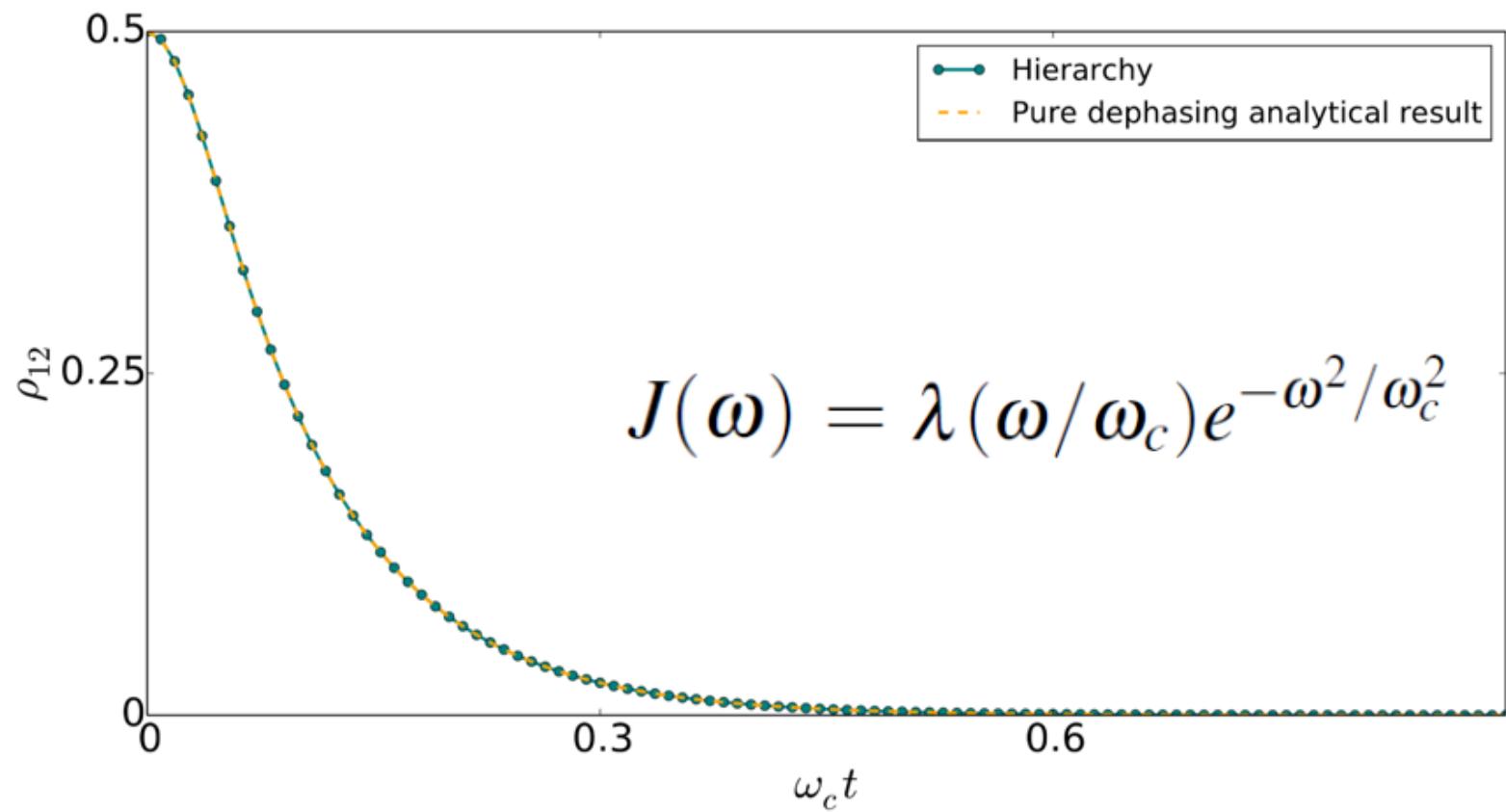
The Drude-Lorentz spectral density is convenient because we can **analytically expand it as exponentials**.

In other cases, we can **numerically calculate** the correlation functions, and then **numerically fit** them to a sum of exponentials.

As a simple example, we can bench-mark against the analytically solvable "pure dephasing model", i.e., we set the coupling operator equal to the system Hamiltonian

(see A. fruchtman, N. Lambert, E. Gauger, Scientific Reports 2016 for more interesting examples)

Ohmic bath, $\lambda = 0.015 \text{ ps}^{-1}$, $\omega_c = 2.2 \text{ ps}^{-1}$, $T = 50 \text{ K}$



Super-Ohmic bath, $\lambda=0.015 \text{ ps}^{-1}$, $\omega_c=2.2 \text{ ps}^{-1}$, $T=50 \text{ K}$

