

Quantum trajectories for simulations

In this exercise we will implement a rudimentary version of a Quantum trajectory simulation. You are allowed to implement the code either in `python`, `rust`, or in `julia`, all other programming languages are not allowed. You can submit your solution either as pdf generated from jupyter notebook or submit a Latex file, that contains the code, figures, and answers to the questions you will find below. Make sure your figures contain all necessary information and are of high quality, i.e., they should be ready to go into your next paper / thesis.

The Wavefunction Monte Carlo (Counting Trajectories) algorithm

The algorithm you are supposed to implement includes the following steps:

- Choose a time interval τ short on the time scale of the relaxation, but long compared to the reservoir correlation time, so to ensure that the probability of a jump $p_\mu(t) = \tau \langle \psi(t) | \hat{\Gamma}_\mu^\dagger \hat{\Gamma}_\mu | \psi(t) \rangle \ll 1$ for any μ and any t . Moreover, τ should be small enough to ensure a smooth integration of the deterministic part of the master equation.
- Initialize the state to the chosen $|\psi(t=0)\rangle$
- For each time interval τ over the total evolution time, evolve the quantum state $|\psi(t)\rangle$ according to the rules:
 - Compute $p_\mu = \tau \langle \psi(t) | \hat{\Gamma}_\mu^\dagger \hat{\Gamma}_\mu | \psi(t) \rangle$ for every $\mu = 1 \dots n$ and $p_0 = 1 - \sum_{\mu \neq 0} p_\mu$.
 - Construct the variables $P_{-1} = 0, P_0 = p_0, P_1 = p_0 + p_1, \dots, P_n = 1$.
 - Extract a random number $r \in [0, 1]$.
 - Find the element of $\bar{\mu}$ such that $P_{\bar{\mu}-1} < r < P_{\bar{\mu}}$.
 - If $\bar{\mu} = 0$, evolve

$$|\psi(t+\tau)\rangle = \frac{1 - i\hat{H}\tau - \tau \sum_{\mu} \hat{\Gamma}_\mu^\dagger \hat{\Gamma}_\mu / 2}{\sqrt{p_0}} |\psi(t)\rangle$$

- Else,

$$|\psi(t+\tau)\rangle = \frac{\hat{\Gamma}_{\bar{\mu}}}{\sqrt{\langle \hat{\Gamma}_{\bar{\mu}}^\dagger \hat{\Gamma}_{\bar{\mu}} \rangle}} |\psi(t)\rangle$$

- Create a new trajectory, and repeat until the wanted number of trajectories has been computed.

While more efficient versions of quantum trajectory simulations exists, this one is sufficient to teach you about the fundamentals and does not require more advanced numerical integration techniques such as adaptive time-stepping.

Task 1 [8 points]

We will restrict in this exercises to states and operators in \mathbb{C}^2 and $\mathbb{C}^{2 \times 2}$, respectively. It is therefore sufficient if your code works only on this predefined vector space.

- For future tasks we will need to implement states and operators. First, implement a function that allows you create an arbitrary single qubit state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. Make sure that the state the function creates is properly normalized. Second, define the Pauli operators X and Z as well as the Hadamard operator Had , together with annihilation and creation operators σ_- and σ_+ , respectively.
- Steps 1 to 5 in the Wavefunction Monte Carlo (WFMC) algorithm evolve a pure state by a single time step. Choosing one of the programming languages mentioned above, write a function that implements the desired functionality.
- Evolving a state ψ for multiple time-steps τ results in a quantum trajectory. Implement a function that creates such a trajectory and evolves an initial state ψ_{in} up to user-specified time t_{max} . Having in mind later tasks in this exercise, the function should return a list (or equivalently an array) of values $[|\langle 1|\psi(0)\rangle|^2, |\langle 1|\psi(\tau)\rangle|^2, \dots, |\langle 1|\psi(t_{max})\rangle|^2]$.

Task 2 - Qubit decay [6 points]

Having implemented the barebones for WFMC, we can use the algorithm to look at physics problems. The canonical example of this course has been qubit decay. We now want to use the stochastic Schrödinger equation to visualize the decay of a single qubit described by the following master equation

$$\partial_t \hat{\rho}(t) = \gamma \mathcal{D}[\hat{\sigma}_-] \hat{\rho}.$$

We might choose $\gamma = 1$ for this task.

- Simulate a single trajectory up to time $t_{max} = 10/\gamma$ for the initial state $|\psi\rangle = |1\rangle$ and plot $|\langle 1|\psi(t)\rangle|^2$ on the y-axis and the time t on the axis. Explain your choice of τ . What should the relation between τ and γ be? What is happening in the physical system?
- Now simulate multiple trajectories, you will have to find out yourself how many are necessary. In a new figure, plot each trajectory individually in the same figure, as well as their average. What kind of functional dependence do you expect for the average value $|\langle 1|\psi(t)\rangle|^2$. Plot this function as well. How well does it agree with the average? How many trajectories were necessary for to find good agreement between the analytical expectation and numerical average?

[Hint: If you are using `python` and you want to optimize the performance of your code, you can take a look at library `numba`, i.e., <https://numba.readthedocs.io/en/stable/user/5minguide.html>]

Task 2 - Qubit dephasing [6 points]

We now want to investigate qubit dephasing using the quantum trajectories approach. To this end, consider the master equation

$$\partial_t \hat{\rho}(t) = -i [\Delta Z, \hat{\rho}] + \gamma_\phi \mathcal{D}[Z] \hat{\rho},$$

where Δ is a detuning parameter we will use later. We will simulate now the measurement of the T_ϕ time.

- Assuming the state $|\psi\rangle$ will evolve subject to the above master equation, how should you initialize $|\psi\rangle$ and the expectation value of which observable should you calculate in order to estimate T_ϕ ?
- Adjust the code you have written above for a single trajectory, such that you can use it to determine T_ϕ of the system.
- Consider $\gamma_\phi = 0.25$ for now and set $\Delta = 0$. Simulate the time evolution of a single trajectory up to $t_{max} = 10$ using an appropriate value for τ and plot the result. What do you observe? Describe the qualitatively using the Bloch sphere picture.
- Consider now $\Delta = 0$ and again $\gamma_\phi = 0.25$ and simulate multiple trajectories for a suitable choice of τ until the results converge with analytic form that you derived in Hand-in 1. Again, plot each trajectory individually in the same figure, as well as their average.

Note that the results won't converge as nicely as in the case of qubit decay. Give an intuitive explanation for this observation.

- Now consider non-zero Δ , i.e., $\Delta = 1.5$ and repeat the task above for this parameter set. Additionally, either plot an individual trajectory in another figure or highlight one in the plot with multiple trajectories. Explain what is happening. What is the role of Δ ? What has changed in comparison to the case of $\Delta = 0$.

Determining T_2 in a qubit. [4 points]

We now want to combine qubit dephasing and qubit decay in our simulations, i.e, we want to simulate with quantum trajectories the master equation

$$\partial_t \hat{\rho}(t) = -i [\Delta Z, \hat{\rho}] + \gamma_\phi \mathcal{D}[Z] \hat{\rho} + \gamma \mathcal{D}[\sigma_-] \hat{\rho},$$

and determine the relaxation time T_2 for this model.

- Choose parameters $\gamma_\phi = 0.25$, $\gamma = 1.$ and $\Delta = 2.$. Simulate a couple of individual trajectories until you find one in which one can observe both quantum jumps due to σ_- and Z . Plot such a trajectory and describe at which time point which kind of jump occurred.
- As usual simulate multiple trajectories and plot them individually, as well as their average and the theoretical model you expect to observe.