Study on a numerical solution to the N-body problem for a quantum system

Salvador Alberto
Università degli Studi di Padova
Course: Quantum Information and Computing
Assignment 6

December 11, 2024

Contents

1 Introduction

2 Theoretical background

2.1 Density matrix formalism

Density matrix is a powerful tool in quantum mechanics used for describing the state of a system. Given the wavefunction representation $|\psi\rangle$, the density matrix is an operator defined as:

$$\rho = |\psi\rangle\langle\psi|. \tag{1}$$

Actually, density matrix extends the concept of quantum state. We say that a state as (1) is *pure*. There exist states representable in the density matrix formalism, but for which it is not possible to associate a wave function. We call them *mixed states*. A state is mixed if it corresponds to a "statistical mixture" of (pure) states $\{|\psi_i\rangle\}_{i=1,\dots,N}$. In such cases we have:

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|, \tag{2}$$

where p_i is the probability to be in state $|\psi_i\rangle\langle\psi_i|$.
Density matrices are characterized by some special properties:

- Their trace is always one: $\mathrm{Tr}(\rho) = 1$;
- Their eigenvalues are non-negative: $\rho \geq 0$;
- The expectation value of an operator $\hat{A}$ for a state ρ is given by $\mathrm{Tr}(\rho\hat{A})$.
- The expectation value of the density matrix itself is always positive and less than 1: $\mathrm{Tr}(\rho^2) \leq 1$. It is possible to demonstrate that for pure states $\mathrm{Tr}(\rho^2) = 1$, while for mixed states $\mathrm{Tr}(\rho^2) < 1$.

2.1.1 Qubit's state: Pauli Basis and the Bloch Sphere

Pauli matrices are a set of 2×2 Hermitian and unitary matrices defined as

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

Single qubit states can be expressed in the density matrix formalism using a basis of matrices composed by the identity matrix and the Pauli matrices, as:

$$\rho = \frac{1}{2}\left(\mathbb{1}_2 + r_x\hat{\sigma_x} + r_y\hat{\sigma_y} + r_z\hat{\sigma_z}\right). \tag{4}$$

The vector $\vec{r} = (r_x, r_y, r_z)$ is called *Bloch vector* and it encodes the state of the qubit. It is useful to represent the qubit's state on the *Bloch sphere*: a sphere in 3D space with a unitary radius. Pure states correspond to Bloch vectors on its surface ($\|\vec{r}\|^2 = 1$), while mixed states are represented by points inside the sphere ($\|\vec{r}\|^2 < 1$).

2.2 N-body quantum systems

Consider a quantum system composed by N sub-systems. The space where it is described is the *tensor product* between each Hilbert space for the subsystems:

$$\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \cdots \otimes \mathcal{H}_N. \tag{5}$$

To simplify the notation, suppose each subsystem is equivalent. Given $\{|n\rangle\}_{n=1,\dots,D}$ an orthonormal basis for $\mathcal{H}_i$, a complete basis for $\mathcal{H}$ is obtained through the tensor product:

$$|\vec{n}\rangle = |n_1 n_2 \cdots n_N\rangle = |n_1\rangle \otimes |n_2\rangle \cdots \otimes |n_N\rangle. \tag{6}$$

2.2.1 Separable states and entanglement

For the system's states, a relation analoguos to (6) is not always valid. The ones for which it holds are called *separable states*. For them:

$$|\vec{\psi}\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \cdots \otimes |\psi_N\rangle. \tag{7}$$

where $|\psi_i\rangle \in \mathcal{H}_i$ represents the state of the i-th subsystem. Separable states define non-interacting subsystems: each subsystem is independent from the states of the others.
In contrast, when interactions between subsystems are present, *entanglement* emerges. Due to entanglement, quantum systems become interconnected, with correlations between them that persists even when they are spatially separated. As a result, the state can no longer be expressed as a tensor product of single-subsystem states. Instead, the general state of a quantum entangled system takes the form of a superposition in the orthonormal basis $\{|\vec{n}\rangle\}$:

$$|\vec{\psi}\rangle = \sum_{|\vec{n}\rangle} c_{\vec{n}} |\vec{n}\rangle \tag{8}$$

where the coefficients $c_{\vec{n}}$ represents the contributions of each basis state.

2.2.2 Reduced density matrix

Given the density matrix ρ describing an N-body system, the states of a subsystem composing it (or a set of subsystems) is given by the *reduced density matrix*. To simplify ideas we consider a bi-partite system, i.e., made by two subsystems. We call them A and B. From ρ describing we get the *reduced density matrix* ρ_A describing the subsystem A's state by tracing away the degrees of freedom of the subsystem B:

$$\rho_A = \mathrm{Tr}_B \rho = \sum_{|\vec{n}^*\rangle} \langle \vec{n}^* | \rho | \vec{n}^* \rangle \quad \text{where} \quad |\vec{n}^*\rangle \in \mathcal{H}_B, \tag{9}$$

i.e., we trace the whole density matrix over the basis of the subsystem we are not interested in.
Reduced density matrices are essential for understanding how entanglement manifest in quantum systems. If the whole system is in an entangled pure state, the reduced density matrix of a subsystem generally describes a mixed state: the information regarding the entaglement is lost and so the reduced density matrix cannot describe a pure state of the subsystem. In contrast, a separable pure state for the whole system will always result in subsystems described by pure states.

3 Numerical Methods

3.1 N-body numerical wavefunctions

To represent the state of an N-body quantum system one needs to specify the coefficients of its expansion in the orthonormal basis $\{|\vec{n}\rangle\}$ (see (8)). If each subsystem is defined in a D-dimensional Hilbert space, this operation requires $\mathcal{O}(D^N)$ complex coefficients (or equivalently, $\mathcal{O}(2D^N)$ real coefficients). In contrast, if the state is separable as in (7), to represent it is sufficient to define only the states of the subsystem. This is because, in the absence of interactions between subsystems, one can apply the operators defining the state evolution separately to each one of the subsystem states. For this reason it is unnecessary to compute the tensor product to get the expression for the complete state. In conclusion, to represent a separable state, we need D complex coefficients for each state of the subsystem, and a total of ND coefficients for the state of the entire system.

3.2 N-body numerical density matrices

Once the state of the complete system is computed, the density matrix is easily obtainable by exploiting the `numpy.outer` function. This takes as input the array describing the wave function, and its complex conjugate, obtainable for example through `numpy.conj`. It returns the outer product between these two, effectively implementing the definition of the density matrix (1).

3.2.1 Numerical reduced density matrix

Obtaining the reduced density matrix requires a little more effort. The straightforward method of tracing the density matrix following (9) may be inefficient: as the number of sites N increases, so the dimension of ρ, exponentially. This could be limiting due to finite computational power. Luckily, it is possible to write a code able to return the reduced density matrix without computing the one for the entire system. We now discuss how this is possible. Consider the wavefunction $|\vec{\psi}\rangle$. This corresponds to a D^N-dimensional vector. The idea is to reshape it in N-dimensional tensor: each dimension will correspond to one of the sub-systems:

$$\psi_\alpha \in \mathcal{H}^{D^N} \longrightarrow \psi_{\alpha_1,\alpha_2,...,\alpha_N} \quad \text{where} \quad \alpha_i \in \{0,..,d-1\} \ \ \forall i \tag{10}$$

We then order its indices to group the ones associated with the subsystems the reduced density matrix wants to describe. We call the other group of subsystems *environment*. We denote as N_{env} the number of indices related to the environment and N_{sys} the number of indices for the group of subsystems we are interested in. This grouping allows us to reshape the tensor in a matrix with dimension $D^{N_{sys}} \times D^{N_{env}}$. We denote this matrix as $\psi_{partitioned}$. From it, it is possible to obtain the reduced density matrix by a contraction with its adjoint over the indices related to the environment:

$$\rho_{sys,ij} = \sum_k \psi_{partitioned,ik} \cdot \psi^*_{partitioned,jk}. \tag{11}$$

Implementing this numerically is possible exploiting the numpy function `tensor.dot`. What this function does is exactly returning the contraction of two tensor over specific indices. To sum up, instead of building the full $D^N \times D^N$ density matrix and explicitly performing the trace, this procedure only requires the computation of a $D^{N_{sys}} \times D^{N_{env}}$ matrix. This latter happens to be much smaller and more computationally affordable.

4 Results

4.1 Benchmarking the creation of separable VS. generic states

As discussed in Sec.3.1, computing numerically a separable state for an N-body system requires way fewer computational resources than a generic state. In the following, I want to verify this. For the relevance of systems of qubits simulations, I consider $D = 2$. In the separable case, $2N$ complex coefficients are needed to properly define the state, while 2^N are required for the general case. I generate states for these two cases using random coefficients. I define one function for building a separable state, named `Nbody_separable_wfc`, and one for a generic state, named `generate_Nbody_wfc`. Both functions generate the complex coefficients sampling from a uniform distribution through the `numpy.random.random` function. I benchmark their performances by measuring the computational time required to generate the state with respect to N. For the separable case, I use a grid for N from 2 to 40 with steps of length 1. For the generic case instead I need to stop at $N = 25$ as after it the computational power of my laptop is not enough and the program crashes. The results are reported in figure Fig.1. For each N I take 5 measures and compute the average value and standard deviation. I fit each case with the proper expected law: an exponential function for the generic case and a line for the separable case. Results are displayed in a logarithmic scale for the y-axis. As the plot shows, the theoretical expectations correctly match the data acquired. Simulating a separable state for an N-body system is much more affordable as the system's size increases.

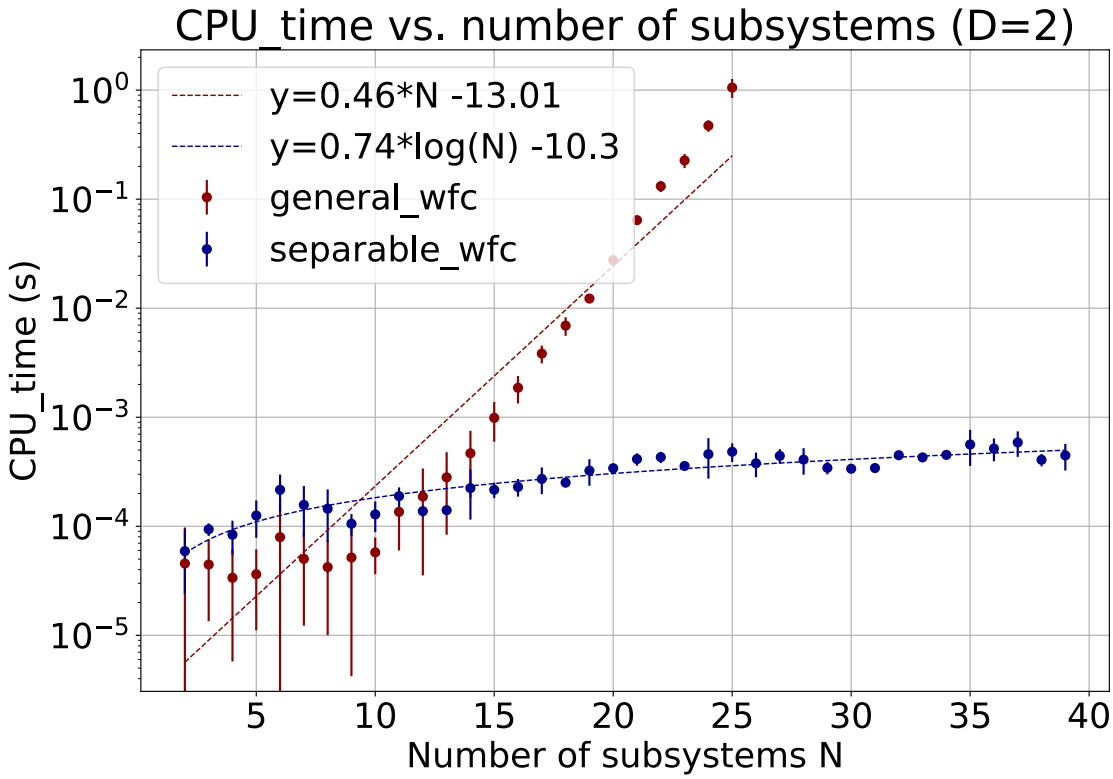

Figure 1: Benchmarking the time required for generating a separable state and a generic state for an N-body quantum system. Both states are generated from random coefficients sampled from a uniform distribution. The dashed lines represent the theoretical relation between the computational time and the size of the system. For the generic case I stop at $N = 25$ due to finite computational power.

4.2 Qubits simulations

One of the most interesting case in quantum information is a composite system made by *qubits*. For this reason, it is worth discussing the numerical implementation of such system following Sec.3. To keep it simple, I consider the $N = 2$ case. I develop a module named `qubit_module.py`, which contains functions dedicated to methods exclusively designed for working with qubits. As an example, this module contains the `density_matrix_pauli` function, which builds the density matrix of a qubit from the Bloch vector $\vec{r} = (r_x, r_y, r_z)$, exploiting the expansion in Pauli matrices (4). I also define a function implementing the inverse operation, named `qubit_bloch_vector`. This is fundamental for visualizing the quibit's state in the Bloch sphere through the `plot_bloch_vector` method from `qiskit.visualization`. Other useful functions here contained are: `get_reduced_density_matrix_from_density_matrix`, a function for computing the reduced density matrix from the density matrix describing the whole system of two qubits; `expectation_value`, a function for the computation of the expectation value of an operator given the state of the system as a density matrix or as a wavefunction; `state_ purity`, a function for computing the purity of a state described through the density matrix formalism.

Using these tools I then investigate some notable 2-body states and the results for the single qubits connected to them. The first case I consider is the one where the system's state is initialized to the maximally entangled state (also called "Bell state"):

$$|\psi\rangle = \frac{1}{2}\left(|00\rangle + |11\rangle\right). \tag{12}$$

As the name suggests, in this state the subsystems are completely correlated through entanglement. By computing the single-qubit states via the reduced density matrix I obtain the states represented in Fig.2a. Each qubit is in a *maximally mixed state*, represented by a Bloch vector at the origin of the Bloch sphere. This is a well-known state, and it represents complete uncertainty about the system: there are equal probabilities of being in either state $|0\rangle$ or $|1\rangle$. This is consistent with the theory discussed in Sec.2.2.2: tracing out one qubit of an entangled system leads to a loss of information. In contrast, Fig.2b shows the single-qubit states obtained when starting from a separable two-qubit pure state initialized with random entries, generated using the function `Nbody_separable_wfc`. Here, the reduced states exhibit non-zero Bloch vectors pointing in some specific directions. By applying `state_purity` I obtain that they are both pure states. indicating that the single-qubit states are pure. This behavior is expected for separable states, where the subsystems are independent.

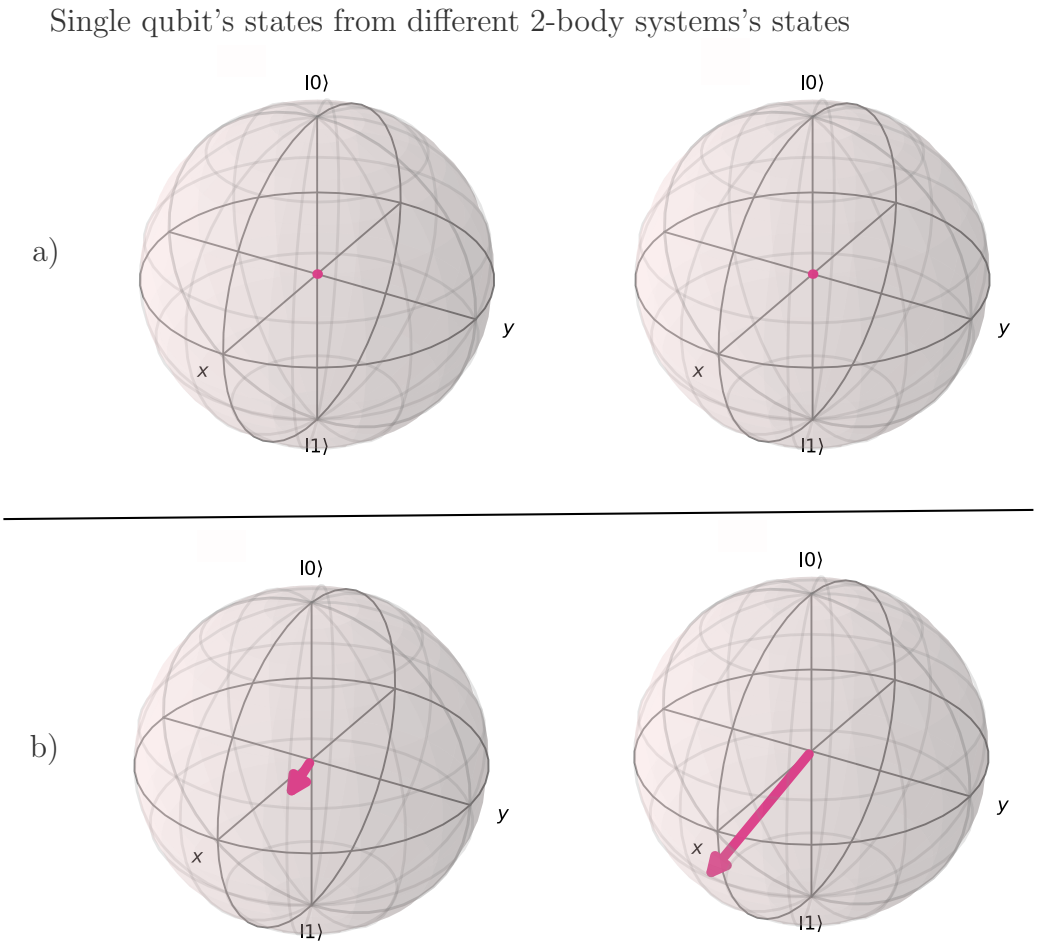

Figure 2: Bloch sphere representations of single-qubit states from two-qubit systems. (a) The states come from a maximally entangled Bell state (b) The states come from a separable two-qubit pure state initialized with random entries.

5 Conclusions

Here are the key points I have learned through this study on N-body quantum systems:

- Separable states require far less computational power to be simulated on a classical computer with respect to general states.
- Tracing away part of the system for obtaining the state of a sub-system causes loss of information in the non-separable case
- To compute the reduced density matrix it is not required to have the complete density matrix. Reshaping it as a tensor and exploiting the structure of the system leads to great benefits in terms of the computational power required to accomplish this task.

References

[1] D. J. Griffiths, *Introduction to Quantum Mechanics*, 3rd Edition, Pearson, 2018.

[2] S. Montangero, *Introduction to Tensor Network Methods*, Springer, 2018.