February 5-9, 2025: Quantum Computing, Quantum Machine Learning and Quantum Information Theories

Morten Hjorth-Jensen[1,2]

Department of Physics, University of Oslo[1]

Department of Physics and Astronomy and Facility for Rare Isotope Beams, Michigan State University, USA[2]

February 7

Plans for the week of February 5-9

1. Reminder from last week
2. Entanglement and Schmidt decomposition
3. Entropy as a measurement of entanglement
4. Simple Hamiltonian systems and how to use the density matrix to estimate degrees of entanglement
5. Introduction to gates and calculations
6. Video of lecture
7. Whiteboard notes

Density matrices and traces

In order to study entanglement and why it is so important for quantum computing, we need to introduce some basic measures and useful quantities. For these endeavors, we will use our two-qubit system from the second lecture in order to introduce, through examples, density matrices and entropy. These two quantities, together with technicalities like the Schmidt decomposition define important quantities in analyzing quantum computing examples.

Reminder on density matrices and traces

We have the spectral decomposition of a given operator $\boldsymbol{A}$ given by

$$\boldsymbol{A} = \sum_i \lambda_i |i\rangle\langle i|,$$

with the ONB $|i\rangle$ being eigenvectors of $\boldsymbol{A}$ and λ_i being the eigenvalues.
The trace of a product of matrices is cyclic, that is

$$\mathrm{tr}[\boldsymbol{ABC}]) = \mathrm{tr}[\boldsymbol{BCA}]) = \mathrm{tr}[\boldsymbol{CBA}]),$$

and we have also

$$\mathrm{tr}[\boldsymbol{A}|\psi\rangle\langle\psi|]) = \langle\psi|\boldsymbol{A}|\psi\rangle.$$

Definition of density matrix

Using the spectral decomposition we defined also the density matrix as

$$\rho = \sum_i p_i |i\rangle\langle i|,$$

where the probability p_i are the eigenvalues of the density linked with the ONB $|i\rangle$.
The trace of the density matrix is $\mathrm{tr}\rho = 1$ and it is invariant under unitary transformations $|\psi_i'\rangle = \boldsymbol{U}|\psi_i\rangle$. The unitary transformation of the density matrix gives, with $\boldsymbol{U}^\dagger\boldsymbol{U} = \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$,

$$\boldsymbol{U}\rho\boldsymbol{U}^\dagger = \sum_i pi_i \boldsymbol{U}|\psi_i\rangle\langle\psi_i|\boldsymbol{U}^\dagger,$$

and with the unitary transformation it is easy to show that thet trace of the transformaed density matrix is equal to one,

$$\mathrm{tr}\left[\boldsymbol{U}\rho\boldsymbol{U}^\dagger\right] = \mathrm{tr}\left[\boldsymbol{U}^\dagger\boldsymbol{U}\rho\right] = \mathrm{tr}\left[\boldsymbol{U}\boldsymbol{U}^\dagger\rho\right] = 1.$$

From last week: First entanglement encounter, two qubit system

We define a system that can be thought of as composed of two subsystems A and B. Each subsystem has computational basis states

$$|0\rangle_{\mathrm{A,B}} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T \qquad |1\rangle_{\mathrm{A,B}} = \begin{bmatrix} 0 & 1 \end{bmatrix}^T.$$

The subsystems could represent single particles or composite many-particle systems of a given symmetry.

Computational basis

This leads to the many-body computational basis states

$$|00\rangle = |0\rangle_{\mathrm{A}} \otimes |0\rangle_{\mathrm{B}} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T,$$

and

$$|01\rangle = |0\rangle_{\mathrm{A}} \otimes |1\rangle_{\mathrm{B}} = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}^T,$$

and

$$|10\rangle = |1\rangle_{\mathrm{A}} \otimes |0\rangle_{\mathrm{B}} = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}^T,$$

and finally

$$|11\rangle = |1\rangle_{\mathrm{A}} \otimes |1\rangle_{\mathrm{B}} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T.$$

Bell states

The above computational basis states, which define an ONB, can in turn be used to define another ONB. As an example, consider the so-called Bell states

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}} \left[|00\rangle + |11\rangle \right] = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

$$|\Phi^-\rangle = \frac{1}{\sqrt{2}} \left[|00\rangle - |11\rangle \right] = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix},$$

The next two

$$|\Psi^{+}\rangle = \frac{1}{\sqrt{2}}\left[|10\rangle + |01\rangle\right] = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\1\\1\\0\end{bmatrix},$$

and

$$|\Psi^{-}\rangle = \frac{1}{\sqrt{2}}\left[|10\rangle - |01\rangle\right] = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\1\\-1\\0\end{bmatrix}.$$

It is easy to convince oneself that these states also form an orthonormal basis.

Measurement

Measuring one of the qubits of one of the above Bell states, automatically determines, as we will see below, the state of the second qubit. To convince ourselves about this, let us assume we perform a measurement on the qubit in system A by introducing the projections with outcomes 0 or 1 as

$$\boldsymbol{P}_0 = |0\rangle\langle 0|_A \otimes \boldsymbol{I}_B = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

for the projection of the $|0\rangle$ state in system A and similarly

$$\boldsymbol{P}_1 = |1\rangle\langle 1|_A \otimes \boldsymbol{I}_B = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

for the projection of the $|1\rangle$ state in system A.

Probability of outcome

We can then calculate the probability for the various outcomes by computing for example the probability for measuring qubit 0

$$\langle\Phi^+|\boldsymbol{P}_0|\Phi^+\rangle = \frac{1}{2}\left[\langle 00| + \langle 11|\right]\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\left[|00\rangle + |11\rangle\right] = \frac{1}{2}.$$

Similarly, we obtain

$$\langle\Phi^+|\boldsymbol{P}_1|\Phi^+\rangle = \frac{1}{2}\left[\langle 00| + \langle 11|\right]\begin{bmatrix}0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\left[|00\rangle + |11\rangle\right] = \frac{1}{2}.$$

States after measurement

After the above measurements the system is in the states

$$|\Phi_0'\rangle = \sqrt{2}\,[|0\rangle\langle 0|_A \otimes \boldsymbol{I}_B]\,|\Phi^+\rangle = |00\rangle,$$

and

$$|\Phi_1'\rangle = \sqrt{2}\,[|1\rangle\langle 1|_A \otimes \boldsymbol{I}_B]\,|\Phi^+\rangle = |11\rangle.$$

We see from the last two equations that the state of the second qubit is determined even though the measurement has only taken place locally on system A.

Other states

If we on the other hand consider a state like

$$|00\rangle = |0\rangle_A \otimes |0\rangle_B,$$

this is a pure **product** state of the single-qubit, or single-particle states, of two qubits (particles) in system A and system B, respectively. We call such a state for a **pure product state**. Quantum states that cannot be written as a mixture of other states are called pure quantum states or just product states, while all other states are called mixed quantum states.

More on Bell states

A state like one of the Bell states (where we introduce the subscript AB to indicate that the state is composed of single states from two subsystem)

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}\left[|00\rangle_{AB} + |11\rangle_{AB}\right],$$

is on the other hand a mixed state and we cannot determine whether system A is in a state 0 or 1. The above state is a superposition of the states $|00\rangle_{AB}$ and $|11\rangle_{AB}$ and it is not possible to determine individual states of systems A and B, respectively.

Entanglement

We say that the state is entangled. This yields the following definition of entangled states: a pure bipartite state $|\psi\rangle_{AB}$ is entangled if it cannot be written as a product state $|\psi\rangle_A \otimes |\phi\rangle_B$ for any choice of the states $|\psi\rangle_A$ and $|\phi\rangle_B$. Otherwise we say the state is separable.

Maximally entangled

A so-called maximally entangled state for a bipartite system has equal probability amplitudes

$$|\Psi\rangle = \frac{1}{\sqrt{d}}\sum_{i=0}^{d-1}|ii\rangle.$$

We call a bipartite state composed of systems A and B (these systems can be single-particle systems, or single-qubit systems representing low-lying states of complicated many-body systems) for separable if its density matrix ρ_{AB} can be written out as the tensor product of the individual density matrices ρ_A and ρ_B, that is we have for a given probability distribution p_i

$$\rho_{AB} = \sum_i p_i \rho_A(i) \otimes \rho_B(i).$$

Schmidt decomposition

If we cannot write the density matrix in this form, we say the system AB is entangled. In order to see this, we can use the so-called Schmidt decomposition, which is essentially an application of the singular-value decomposition.

Pure states and Schmidt decomposition

The Schmidt decomposition allows us to define a pure state in a bipartite Hilbert space composed of systems A and B as

$$|\psi\rangle = \sum_{i=0}^{d-1} \sigma_i |i\rangle_A |i\rangle_B,$$

where the amplitudes σ_i are real and positive and their squared values sum up to one, $\sum_i \sigma_i^2 = 1$. The states $|i\rangle_A$ and $|i\rangle_B$ form orthornormal bases for systems A and B respectively, the amplitudes λ_i are the so-called Schmidt coefficients and the Schmidt rank d is equal to the number of Schmidt coefficients and is smaller or equal to the minimum dimensionality of system A and system B, that is $d \leq \min(\dim(A), \dim(B))$.

Proof of Schmidt decomposition

The proof for the above decomposition is based on the singular-value decomposition. To see this, assume that we have two orthonormal bases sets for systems A and B, respectively. That is we have two ONBs $|i\rangle_A$ and $|j\rangle_B$. We can always construct a product state (a pure state) as

$$|\psi\rangle = \sum_{ij} c_{ij} |i\rangle_A |j\rangle_B,$$

where the coefficients c_{ij} are the overlap coefficients which belong to a matrix $\boldsymbol{C}$.

Further parts of proof

If we now assume that the dimensionalities of the two subsystems A and B are the same d, we can always rewrite the matrix $\boldsymbol{C}$ in terms of a singular-value decomposition with unitary/orthogonal matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ of dimension $d \times d$ and a matrix $\boldsymbol{\Sigma}$ which contains the (diagonal) singular values $\sigma_0 \le \sigma_1 \le \dots 0$ as

$$\boldsymbol{C} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\dagger}.$$

SVD parts in proof

This means we can rewrite the coefficients c_{ij} in terms of the singular-value decomposition

$$c_{ij} = \sum_k u_{ik}\sigma_k v_{kj},$$

and inserting this in the definition of the pure state $|\psi\rangle$ we have

$$|\psi\rangle = \sum_{ij}\left(\sum_k u_{ik}\sigma_k v_{kj}\right)|i\rangle_A|j\rangle_B.$$

Slight rewrite

We rewrite the last equation as

$$|\psi\rangle = \sum_k \sigma_k \left(\sum_i u_{ik} |i\rangle_A \right) \otimes \left(\sum_j v_{kj} |j\rangle_B \right),$$

which we identify simply as, since the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ represent unitary transformations,

$$|\psi\rangle = \sum_k \sigma_k |k\rangle_A |k\rangle_B.$$

Different dimensionalities

It is straight forward to prove this relation in case systems A and B have different dimensionalities. Once we know the Schmidt decomposition of a state, we can immmediately say whether it is entangled or not. If a state ψ has is entangled, then its Schmidt decomposition has more than one term. Stated differently, the state is entangled if the so-called Schmidt rank is greater than one. There is another important property of the Schmidt decomposition which is related to the properties of the density matrices and their trace operations and the entropies. In order to introduce these concepts we will look look at the two-qubit Hamiltonian described after the discussion of entropy.

Entropies and density matrices

Note: more details on whiteboard from week 3.

Shannon information entropy

We start our discussions with the classical information entropy, or just Shannon entropy, before we move over to a quantum mechanical way to define the entropy based on the density matrices discussed earlier.
We define a set of random variables $X = \{x_0, x_1, \dots, x_{n-1}\}$ with probability for an outcome $x \in X$ given by $p_X(x)$, the information entropy is defined as

$$S = -\sum_{x \in X} p_X(x) \log_2 p_X(x).$$

Von Neumann entropy

$$S = -\mathrm{Tr}[\rho \log_2 \rho].$$

Two-qubit system and calculation of density matrices and exercise

This part is best seen using the jupyter-notebook.

The system we discuss here is a continuation of the two qubit example from week 2.

This system can be thought of as composed of two subsystems A and B. Each subsystem has computational basis states

$$|0\rangle_{\mathrm{A,B}} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T \qquad |1\rangle_{\mathrm{A,B}} = \begin{bmatrix} 0 & 1 \end{bmatrix}^T .$$

The subsystems could represent single particles or composite many-particle systems of a given symmetry. This leads to the many-body computational basis states

$$|00\rangle = |0\rangle_{\mathrm{A}} \otimes |0\rangle_{\mathrm{B}} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T ,$$

and

$$|01\rangle = |0\rangle_{\mathrm{A}} \otimes |1\rangle_{\mathrm{B}} = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}^T ,$$

and

$$|10\rangle = |1\rangle_{\mathrm{A}} \otimes |0\rangle_{\mathrm{B}} = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}^T ,$$

and finally

Exercise from last week: Two-qubit Hamiltonian

Use the Hamiltonian for the two-qubit example to find the eigenpairs as functions of the interaction strength λ and study the final eigenvectors as functions of the admixture of the original basis states. Discuss the results as functions of the parameter λ and compute the von Neumann entropy and discuss the results. You will need to calculate the entropy of the subsystems A or B.

Quantum gates, circuits and simple algorithms

Quantum gates are physical actions that are applied to the physical system representing the qubits. Mathematically, they are complex-valued, unitary matrices which act on the complex-values normalized vectors that represent qubits. As the quantum analog of classical logic gates (such as AND and OR), there is a corresponding quantum gate for every classical gate; however, there are quantum gates that have no classical counter-part. They act on a set of qubits and, changing their state. That is, if U is a quantum gate and $|q\rangle$ is a qubit, then acting the gate U on the qubit $|q\rangle$ transforms the qubit as follows:

$$|q\rangle \overset{U}{\to} U|q\rangle. \tag{1}$$

Quantum circuits

Quantum circuits are diagrammatic representations of quantum algorithms. The horizontal dimension corresponds to time; moving left to right corresponds to forward motion in time. They consist of a set of qubits $|q_n\rangle$ which are stacked vertically on the left-hand side of the diagram. Lines, called quantum wires, extend horizontally to the right from each qubit, representing its state moving forward in time. Additionally, they contain a set of quantum gates that are applied to the quantum wires. Gates are applied chronologically, left to right.

Single-Qubit Gates

A single-qubit gate is a physical action that is applied to one qubit. It can be represented by a matrix U from the group SU(2). Any single-qubit gate can be parameterized by three angles: θ, ϕ, and λ as follows

$$U(\theta,\phi,\lambda) = \begin{bmatrix} \cos\frac{\theta}{2} & -e^{i\lambda}\sin\frac{\theta}{2} \\ e^{i\phi}\sin\frac{\theta}{2} & e^{i(\phi+\lambda)}\cos\frac{\theta}{2} \end{bmatrix}.$$

Widely used gates

There are several widely used quantum gates. Perhaps the most famous are the Pauli gates correspond to the Pauli matrices

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix},$$

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Algebra basis

These gates form a basis for the algebra $\mathfrak{su}(2)$. Exponentiating them will thus give us a basis for SU(2), the group within which all single-qubit gates live.

Exponentiated Pauli gates

These exponentiated Pauli gates are called rotation gates $R_\sigma(\theta)$ because they rotate the quantum state around the axis $\sigma = X, Y, Z$ of the Bloch sphere by an angle θ. They are defined as

$$R_X(\theta) = e^{-i\frac{\theta}{2}X} = \begin{bmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix},$$

$$R_Y(\theta) = e^{-i\frac{\theta}{2}Y} = \begin{bmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix},$$

$$R_Z(\theta) = e^{-i\frac{\theta}{2}Z} = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}.$$

Basis for $\mathrm{SU}(2)$

Because they form a basis for $\mathrm{SU}(2)$, any single-qubit gate can be decomposed into three rotation gates. Indeed

$$R_z(\phi)R_y(\theta)R_z(\lambda) = \begin{bmatrix} e^{-i\phi/2} & 0 \\ 0 & e^{i\phi/2} \end{bmatrix} \begin{bmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix} \begin{bmatrix} e^{-i\lambda/2} & 0 \\ 0 & e^{i\lambda/2} \end{bmatrix}$$

which we can rewite as

$$e^{-i(\phi+\lambda)/2} \begin{bmatrix} \cos\frac{\theta}{2} & -e^{i\lambda}\sin\frac{\theta}{2} \\ e^{i\phi}\sin\frac{\theta}{2} & e^{i(\phi+\lambda)}\cos\frac{\theta}{2} \end{bmatrix},$$

which is, up to a global phase, equal to the expression for an arbitrary single-qubit gate.

Two-Qubit Gates

A two-qubit gate is a physical action that is applied to two qubits. It can be represented by a matrix U from the group SU(4). One important type of two-qubit gates are controlled gates, which work as follows: Suppose U is a single-qubit gate. A controlled-U gate (CU) acts on two qubits: a control qubit $|x\rangle$ and a target qubit $|y\rangle$. The controlled-U gate applies the identity I or the single-qubit gate U to the target qubit if the control gate is in the zero state $|0\rangle$ or the one state $|1\rangle$, respectively.

Control qubit

The control qubit is not acted upon. This can be represented as follows if

$$CU|xy\rangle = |xy\rangle \text{ if } |x\rangle = |0\rangle.$$

In matrix form

It is easier to see in a matrix form. It can be written in matrix form by writing it as a superposition of the two possible cases, each written as a simple tensor product

$$CU = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes U = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{bmatrix}.$$

CNOT gate

One of the most fundamental controlled gates is the CNOT gate. It is defined as the controlled-X gate CX and thus flips the state of the target qubit if the control qubit is in the zero state $|0\rangle$. It can be written in matrix form as follows:

$$\mathrm{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

Swap gate

A widely used two-qubit gate that goes beyond the simple controlled function is the SWAP gate. It swaps the states of the two qubits it acts upon

$$\mathrm{SWAP}|xy\rangle = |yx\rangle.$$

and has the following matrix form

$$\mathrm{SWAP} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Introduction to Qiskit

This part is best seen using the jupyter-notebook.

For the installation

```
pip install qiskit
pip install qiskit-ibm-runtime
```

Documentation can be found at

https://docs.quantum.ibm.com/api/qiskit

Simple code

```
#!/usr/bin/env python
# coding: utf-8
import numpy as np
import qiskit as qk
from scipy.optimize import minimize


# # Initialize registers and circuit

n_qubits = 1 #Number of qubits
n_cbits = 1 #Number of classical bits (the number of qubits you want t
qreg = qk.QuantumRegister(n_qubits) #Create a quantum register
creg = qk.ClassicalRegister(n_cbits) #Create a classical register
circuit = qk.QuantumCircuit(qreg,creg) #Create your quantum circuit


circuit.draw() #Draw circuit. It is empty
```

Thereafter we perform operations on qubit

```
circuit.x(qreg[0]) #Applies a Pauli X gate to the first qubit in the q
circuit.draw()
```

and select a qubit to measure and encode the results to a classical bit

```
#Measure the first qubit in the quantum register
#and encode the results to the first qubit in the classical register
```

Exercises for next week and Bell states, developed by Keran

```
import qiskit as qk
from qiskit import QuantumRegister, ClassicalRegister, QuantumCircuit
from qiskit.visualization import plot_histogram
from qiskit_aer import AerSimulator
import matplotlib.pyplot as plt
```

Setting up a circuit

```
qreg_q = QuantumRegister(2) # 2 qubits
creg_c = ClassicalRegister(2) # 2 classical bits

qc = QuantumCircuit(qreg_q, creg_c) # alternatively, circuit = Quantum

qc.h(qreg_q[0]) # apply the hadamard gate to the first qubit (to the r

# apply the CNOT gate with the first qubit being the control qubit and
qc.cx(qreg_q[0], qreg_q[1])


# measure the qubits, specify which qubit you want to measure
qc.measure(qreg_q[0], creg_c[0])
qc.measure(qreg_q[1], creg_c[1])
```

Next we will use an ideal(noiseless) simulator (AerSimulator) to run our circuit. We will use the qasm_simulator backend. This simulator returns a result object which contains the counts, or

Plans next week

Lecture next week we discuss simple algorithms and quantum circuits

1. More on one-, two- and three-qubit gates
2. Setting up quantum circuits and simple algorithms