CSCI-769 Quantum Computing Principles and Applications Homework 1

1. Quantum States and Qubits (10 pts)

Summary: This problem introduces the fundamental representation of a qubit and its geometric interpretation using the Bloch sphere. Understanding how the parameters θ and ϕ define a qubit provides insight into its probabilistic and phase properties. Calculating measurement probabilities and Bloch sphere coordinates highlights how qubits relate to classical measurements and spatial representations.

- Write the general form of a qubit state in terms of Bloch sphere parameters θ and ϕ. Derive expressions for the probabilities of measuring $|0\rangle$ and $|1\rangle$ in terms of θ, and show the sum of these probabilities is one.

- For the state:

  $$|\psi\rangle = \frac{\sqrt{3}}{2}|0\rangle + \frac{1}{2}e^{i\pi/4}|1\rangle,$$

  calculate:

  - The probabilities of measuring the qubit in the computational basis $|0\rangle$ and $|1\rangle$.
  - The corresponding (x, y, z) coordinates on the Bloch sphere.

2. Entanglement in Multi-Qubit Systems (10 pts)

Summary: This problem explores entanglement in multi-qubit systems, focusing on analyzing and quantifying entanglement using mathematical tools and examples. Entanglement is a fundamental resource in quantum computing, as it enables phenomena that have no classical counterpart. By leveraging entanglement, quantum computers can perform complex computations more efficiently, enable secure quantum communication (e.g., quantum key distribution), and facilitate quantum error correction. Furthermore, entanglement is crucial for implementing quantum algorithms such as Grover's search algorithm and Shor's factorization algorithm, as it allows qubits to operate in a correlated state, exponentially increasing the information they can process simultaneously. Understanding and quantifying entanglement is essential for harnessing its power in practical quantum systems.

(a) Consider the two-qubit state:

$$|\psi\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle).$$

At measurement what is the probability of being in each state. Do all these probabilities add to one?
(b) For the three-qubit state:

$$|\psi\rangle = \frac{1}{2}|000\rangle + \frac{1}{2}|011\rangle + \frac{1}{2}|101\rangle + \frac{1}{2}|110\rangle,$$

what is the probability of being in each of the **8** states? Hint: some aren't explicitly written, but what are these probabilities still - that is to say if we don't write basis states within the creation of a quantum state, then what are their respective probabilities.?

3. Superposition (10 pts)

Summary: Superposition is a core concept in quantum mechanics. This problem distinguishes between coherent and incoherent superpositions, emphasizing the role of phase and interference in quantum systems. The second part investigates the entanglement properties of a multi-qubit state, teaching how to analyze and classify quantum correlations.

(a) For the state:

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

calculate the probability of being in each state after applying a Hadamard gate. Write the final state and verify whether it remains a superposition.
(b) Consider the three-qubit state:

$$|\psi\rangle = \frac{1}{\sqrt{3}}|000\rangle + \frac{1}{\sqrt{3}}|010\rangle + \frac{1}{\sqrt{3}}|111\rangle.$$

Determine the probability of being in each state, and also calculate the vector representation of the basis states which have nontrivial probability. Hint: for d qubits $|q_1q_2....q_d\rangle = |q_1\rangle \otimes |q_2\rangle \otimes .... \otimes |q_{d-1}\rangle \otimes |q_d\rangle$ e.g. $|000\rangle = |0\rangle \otimes |0\rangle \otimes |0\rangle$, so on and so forth.

4. Unitary Matrices and Their Role in Quantum Computing (10 pts)

Summary: This problem investigates the properties and significance of unitary matrices, which are fundamental in quantum mechanics and quantum computing. A matrix U is unitary if it satisfies $U^*U = I$, where U^* is the complex conjugate transpose of U. Using another notation, dagger notation, this is written as $U^\dagger U = I$. Importantly, for a matrix to be unitary, the order of multiplication does not matter, meaning $U^*U = UU^* = I$.

The complex conjugate transpose (or Hermitian conjugate) of a general $N \times N$ matrix $A = (a_{ij})$ is defined as the matrix A^*, where $(A^*)_{ij} = \overline{a_{ji}}$. Recall that if $a_{i,j} = c_{i,j} + d_{i,j}i$ then $\overline{a_{i,j}} = c_{i,j} - d_{i,j}i$. In other words, A^* is obtained by taking the transpose of A and then taking the complex conjugate of each element.

For example:

$$U = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}, \quad U^* = \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix}, \quad U^\dagger = U^*.$$

Understanding these concepts is vital in quantum computing for designing reversible quantum operations and implementing error correction protocols to preserve quantum information integrity.

(a) **Norm Preservation Under Unitary Transformation:** Prove that if a quantum state $|\psi\rangle$ has norm 1 e.g. $||\psi|| = |\langle\psi|\psi\rangle|^2 = 1$, then applying a unitary operator U to it results in a state $|\psi'\rangle = U|\psi\rangle$ that also has norm 1. Specifically, show that $||\psi'|| = |\langle\psi'|\psi'\rangle|^2 = 1$.

(b) **Inverse of a Unitary Matrix:** Derive the formula for the inverse of U in terms of either or both notations U^* and $U^\dagger$. Verify that this formula satisfies the matrix inverse condition $U^{-1}U = I$.

(c) **Specific 2x2 Example and Inverse Calculation:**

(a) Consider matrix:

$$U = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{i}{\sqrt{2}} \\ \frac{i}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix};$$

verify this matrix is unitary.

(b) Calculate the inverse of the specific matrix U above and verify $UU^{-1} = U^{-1}U = I$. Hint: if $U = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$:

$$U^{-1} = \frac{1}{\det(U)} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}, \quad \text{where} \quad \det(U) = ad - bc.$$

To compute the determinant and inverse, recall:

- **Multiplying two complex numbers:** If $z_1 = x_1 + iy_1$ and $z_2 = x_2 + iy_2$, their product is:

$$z_1 z_2 = (x_1x_2 - y_1y_2) + i(x_1y_2 + y_1x_2).$$

- **Adding or subtracting complex numbers:** If $z_1 = x_1 + iy_1$ and $z_2 = x_2 + iy_2$:

$$z_1 + z_2 = (x_1 + x_2) + i(y_1 + y_2), \quad z_1 - z_2 = (x_1 - x_2) + i(y_1 - y_2).$$

- **Dividing two complex numbers:** If $z_1 = x_1 + iy_1$ and $z_2 = x_2 + iy_2$, then:

$$\frac{z_1}{z_2} = \frac{(x_1x_2 + y_1y_2) + i(y_1x_2 - x_1y_2)}{x_2^2 + y_2^2}.$$

In the special case where $z_1 = 1$, this becomes:

$$\frac{1}{z_2} = \frac{x_2}{x_2^2 + y_2^2} - i\frac{y_2}{x_2^2 + y_2^2}.$$

(c) Compute the inverse of U using the special property you found from "inverse of unitary matrix".

(d) Compare both methods to confirm that they yield the inverse.

(d) **Scaling to Higher Dimensions:** Discuss the computational complexity of finding the inverse of a general $N \times N$ complex matrix. Explain why the special inverse formula for a unitary matrix simplifies this task in terms of complexity. Feel free to do some research for this question ☺!

5. Bell States (10 pts)

Summary: Bell states are maximally entangled two-qubit states, essential for quantum computing, and quantum communication (even though we won't cover quantum communication or quantum key distribution in the course ☺). This problem demonstrates how to construct these states using gates.

The four Bell states are defined as follows:

$$
\begin{aligned}
|\Phi^+\rangle &= \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle),\\
|\Phi^-\rangle &= \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle),\\
|\Psi^+\rangle &= \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle),\\
|\Psi^-\rangle &= \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle).
\end{aligned}
$$

(a) For each of the Bell States, produces the circuit that yields the state starting from the $|00\rangle$ state (3 qubits all initially in the zero states). You can either build each circuit in Qiskit and plot it using the draw command in Qiskit:

```python
from qiskit import QuantumCircuit
from qiskit_ibm_runtime import QiskitRuntimeService, Session, Sampler
from qiskit.visualization import plot_histogram

# Step 1: Create a simple quantum circuit with one qubit and one classical bit.
qc = QuantumCircuit(1, 1)  # Initialize a quantum circuit with 1 qubit and 1 classical bit.
qc.h(0)  # Apply a Hadamard gate to the qubit (creates a superposition state).
qc.measure(0, 0)  # Measure the state of the qubit and store the result in the classical bit.

# Visualize the circuit to understand its structure.
print("Quantum␣Circuit:")
print(qc)
qc.draw('mpl')  # Use matplotlib to render the circuit diagram.
```

you can also use the latex package quantikz to make pretty circuits like this:

```latex
\documentclass{article}
\usepackage{quantikz}

\begin{document}

\begin{quantikz}
\lstick{$\ket{q_0}$} & \gate{H} & \ctrl{1} & \ctrl{2} & \qw & \qw \\
\lstick{$\ket{q_1}$} & \qw      & \gate{R_2} & \qw      & \ctrl{1} & \qw \\
\lstick{$\ket{q_2}$} & \qw      & \qw      & \gate{R_3} & \gate{R_2} & \qw
\end{quantikz}

\end{document}
```

- Install the `quantikz` package if it's not included in your LaTeX distribution.
- Replace gate names or add/remove wires as needed to create custom circuits.
- Use `\qw` for wires and `\meter{}` for measurement symbols.

Hint: one of the bell state circuits might be on the Wikipedia page for Bell states if you need some inspiration!

(b) Also for each circuit you create for the corresponding bell state, apply the circuit to the $|00\rangle$ state to verify the correct state is produced.

6. Measurement (10 pts)

Summary: Measurement collapses a quantum state into a classical outcome, central to understanding quantum mechanics. In the case of quantum computing, we are interested in measuring qubits to know if they are in the $|0\rangle$ or $|1\rangle$ state, respectively. This problem explores measurements in arbitrary bases and calculates probabilities for specific cases, demonstrating how quantum states relate to experimental results. Many quantum algorithms, including Shor's and Grover's algorithms, rely on measurements in specific bases.

(a) A qubit is in the state:

$$|\psi\rangle = \frac{1}{\sqrt{3}}|0\rangle + \sqrt{\frac{2}{3}}|1\rangle.$$

calculate the probabilities of being in the $|0\rangle$ and $|1\rangle$ state at measurement.

(b) If a measurement is performed in the Hadamard basis:

$$\{|+\rangle, |-\rangle\}, \quad \text{where } |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle),$$

calculate the probabilities of observing each outcome mathematically. Hint: write ψ in terms of this new basis e.g. $|\psi\rangle = c_0|+\rangle + c_1|-\rangle$, and then calculate the probabilities of observing each of these states using the coefficients on the states like usual.

7. Circuit Construction (10 pts)

Summary: This problem focuses on creating a three-qubit GHZ state, a maximally entangled state. Constructing this state highlights how quantum gates work together to generate complex, useful quantum states for applications in quantum computing and communication.

Design a quantum circuit to create the three-qubit GHZ state:

$$|GHZ\rangle = \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle),$$

starting from the initial state $|000\rangle$. Write down the mathematical expressions for the state after each gate is applied. Hint: if you came up with a valid circuit (there are many that could be correct), you should get the GHZ state! If you want to really push yourself, if after you find one that works, try to make a circuit that uses less gates but archives the same result as well!

8. Quantum Gate Operations (10 pts)

Summary: This problem demonstrates the use of the CNOT gate for generating entanglement and preserving it. Understanding the behavior of CNOT on various input states is crucial for analyzing entangling operations in quantum algorithms and protocols.

(a) Consider the two-qubit state:

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle).$$

Write the result of the CNOT gate to this state where the first qubit left to right is the control and the second one is the target
(b) From the original quantum state apply CNOT gate where the second qubit is the control and the first qubit is the target

9. SWAPs and CNOTs (10 pts)

Summary: The SWAP operator is not a native gate in most quantum hardware, so it is implemented using the identity

$$SWAP = CNOT_{01}CNOT_{10}CNOT_{01},$$

where $CNOT_{ij}$ denotes that i qubit is the control qubit and j is the target qubit. Why this relationship matters:

- The ability to decompose SWAP into CNOTs is crucial for quantum circuit optimization, as CNOT gates are universal for two-qubit operations.
- In many quantum algorithms and architectures (such as superconducting qubits), direct swapping is inefficient, and using a sequence of CNOTs provides flexibility.

Prove this identity. Also calculate $SWAP|01\rangle$ and $CNOT_{12}CNOT_{21}CNOT_{12}|01\rangle$, do these two results match, and what is that result? Hint for proving the idenity: Maybe try to verify it with these operations matrix representation... maybe you know everything except what $CNOT_{21}$ is as a matrix, but once you figure that out just work out the matrix multiplications to verify the matrices agree on both sides of the equation!

10. Flip-ity flop (10 pts)

Summary: When you learned to program classical computers, you were given a problem and asked to write a program that solved the problem. This problem will be the quantum version of that! ☺ Quantum computing isn't all "quantum". There are many pre and post-classical computing components that are required for a quantum computing platform to be useful. We will begin to appreciate this workflow with this problem.

(a) Write a program that takes a string, and produces a quantum circuit that will prepare the binary representation of that string from a quantum system in the ground state (all qubits are in $|0\rangle$) (b) Use the IBM AerSimulator to run the circuit, and read out the result for 1000 shots for each of the three string:

- RIT
- QIS
- GOS

(c) Convert the binary back to words, and count how many times you got the string you expected versus strings you did not want to produce.

d) On a real IBM machine run the circuit using 100 shots on the string "RIT" and tally the results. Compare the results to the "RIT" results when ran on the noisy simulator.

Bonus Problems: AKA getting a deeper understanding of quantum mechanics/computing (50 pts)

a) Examine the Schrödinger equation (a system of two ordinary differential equations) that varies with time for a two-level quantum system:

$$i\frac{d|\psi\rangle}{dt} = H(t)|\psi\rangle \tag{1}$$

usually us in the "biz" work with this form: $\frac{d|\psi\rangle}{dt} = -iH(t)|\psi\rangle$ since $i^2 = -1$ then $\frac{1}{i} = -i$.

Suppose in the instance where the initial quantum state, Hamiltonian, and time duration is:

$$|\psi(0)\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$
$$H(t) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$
$$t \in [0, 2\pi]$$

Find the analytic solution, and for a general time t, and also tell the the probability you are in the zero $P_0(t = 2\pi)$ and one state $P_1(t = 2\pi)$ respectively.Hint: feel free to do some research - some things that might be helpful is understanding what the general solution of this equation is, and Euler's identity.

b) Resolve the same problem in step (a), but instead of producing an analytic solution, produce a numerical solution with Qutip. Solve the equation using 2000 time points, and plot the numerical solution $P_0(t)$, $P_1(t)$ on one plot, and $P_0(t)+P_1(t)$ on the another plot. This might be a helpful link to start: https://qutip.org/docs/4.1/guide/dynamics/dynamics-master.html