Physics 245 Homework #8

Nathan Solomon

December 5, 2024

Problem 0.1.

(a)

$$T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} = e^{i\pi/8} \begin{bmatrix} e^{-i\pi/8} & 0 \\ 0 & e^{i\pi/8} \end{bmatrix} = e^{i\pi/8} R_z(\pi/4).$$

(b)

$$\begin{aligned}
HTH &= \left(\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\right) \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} \left(\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\right) \\
&= \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ e^{i\pi/4} & -e^{i\pi/4} \end{bmatrix} \\
&= \frac{1}{2} \begin{bmatrix} 1 + e^{i\pi/4} & 1 - e^{i\pi/4} \\ 1 - e^{i\pi/4} & 1 + e^{i\pi/4} \end{bmatrix} \\
&= \frac{e^{i\pi/8}}{2} \begin{bmatrix} e^{-i\pi/8} + e^{i\pi/8} & e^{-i\pi/8} - e^{i\pi/8} \\ e^{-i\pi/8} - e^{i\pi/8} & e^{-i\pi/8} + e^{i\pi/8} \end{bmatrix} \\
&= e^{i\pi/8} \begin{bmatrix} \cos(\pi/8) & -i\sin(\pi/8) \\ -i\sin(\pi/8) & \cos(\pi/8) \end{bmatrix} \\
&= e^{i\pi/8} R_x(\pi/4).
\end{aligned}$$

(c)

$$\begin{aligned}
THTH &= \left(e^{i\pi/8} R_z(\pi/4)\right) \left(e^{i\pi/8} R_x(\pi/4)\right) \\
&= e^{i\pi/4} \begin{bmatrix} e^{-i\pi/8} & 0 \\ 0 & e^{i\pi/8} \end{bmatrix} \begin{bmatrix} \cos(\pi/8) & -i\sin(\pi/8) \\ -i\sin(\pi/8) & \cos(\pi/8) \end{bmatrix} \\
&= e^{i\pi/4} \begin{bmatrix} e^{-i\pi/8}\cos(\pi/8) & -ie^{-i\pi/8}\sin(\pi/8) \\ -ie^{i\pi/8}\sin(\pi/8) & e^{i\pi/8}\cos(\pi/8) \end{bmatrix} \\
&= e^{i\pi/4} \left(\cos^2(\pi/8) I - i\sin(\pi/8)\cos(\pi/8)\sigma_z + \begin{bmatrix} 0 & -ie^{-i\pi/8}\sin(\pi/8) \\ -ie^{i\pi/8}\sin(\pi/8) & 0 \end{bmatrix}\right) \\
&= e^{i\pi/4} \left(\cos^2(\pi/8) I - i\sin(\pi/8)\cos(\pi/8)\sigma_z - i\sin(\pi/8)\cos(\pi/8)\sigma_x - i\sin^2(\pi/8)\sigma_y\right).
\end{aligned}$$

Ignoring the global phase factor of $e^{i\pi/4}$, I will guess that this can be written in the form

$$THTH = \cos(\theta/2) I - i\sin(\theta/2) n \cdot \sigma = e^{-i(\theta/2)(n\cdot\pi)}$$

for some $\theta \in \mathbb{R}, n \in \mathbb{R}^3$, which I will try to find. Also, $-i\sin(\theta/2)n$ must be normalized, meaning

$$\sin^2(\theta/2) = \|-i\sin(\theta/2)n\| = \left\| \begin{bmatrix} \sin(\pi/8)\cos(\pi/8) \\ \sin^2(\pi/8) \\ \sin(\pi/8)\cos(\pi/8) \end{bmatrix} \right\| = \sin^2(\pi/8)\left(1 + \cos^2(\pi/8)\right).$$

Now I can solve for θ and n (which I won't bother fully simplifying, because that would be painful):

$$\begin{aligned}
\sin^2(\theta/2) &= \left(\frac{2-\sqrt{2}}{4}\right)\left(1+\frac{2+\sqrt{2}}{4}\right) = \frac{2-\sqrt{2}}{4} + \frac{4-2}{16} = \frac{5}{8} - \frac{1}{2\sqrt{2}} \\
\frac{\theta}{2} &= \arcsin\sqrt{\frac{5}{8}-\frac{1}{2\sqrt{2}}} \approx 0.548 \\
\theta &\approx 1.096 \\
n &= \frac{1}{\sqrt{\frac{5}{8}-\frac{1}{2\sqrt{2}}}}\begin{bmatrix} \sin(\pi/8)\cos(\pi/8) \\ \sin^2(\pi/8) \\ \sin(\pi/8)\cos(\pi/8) \end{bmatrix} \approx \begin{bmatrix} 0.679 \\ 0.281 \\ 0.679 \end{bmatrix}.
\end{aligned}$$

(d) I think the question is worded wrong in two ways: (1) we need θ/π to be irrational, but don't need θ to be irrational, and (2) we can only use powers of $THTH$ to get arbitrarily close to qubit rotations about n.

I'm not sure how to prove θ/π is irrational, but I will assume it is. Let $f : \mathbb{Z} \to \mathbb{R}/(2\pi\mathbb{Z})$ be defined by $f(m) = m\theta$. Then applying the operator $(THTH)$ to a qubit m times is equivalent to a rotation (about the vector n) by an angle $f(m)$. Since θ/π is irrational, the image of f is dense in the codomain of f. That means that for any rotation about n, we can find some $m \in \mathbb{Z}$ such that $(TH)^{2m}$ is aribtrarily close to that rotation operator.

Problem 0.2.

(a) The initial state is

$$|\psi\rangle \otimes |\Phi^+\rangle = \frac{\alpha}{\sqrt{2}}|000\rangle + \frac{\alpha}{\sqrt{2}}|011\rangle + \frac{\beta}{\sqrt{2}}|100\rangle + \frac{\beta}{\sqrt{2}}|111\rangle .$$

Applying the CNOT gate turns state into

$$\frac{\alpha}{\sqrt{2}}|000\rangle + \frac{\alpha}{\sqrt{2}}|011\rangle + \frac{\beta}{\sqrt{2}}|110\rangle + \frac{\beta}{\sqrt{2}}|101\rangle ,$$

and then applying the Hadamard gate turns it into

$$\frac{\alpha}{2}(|000\rangle + |100\rangle + |011\rangle + |111\rangle) + \frac{\beta}{2}(|010\rangle - |110\rangle + |001\rangle - |101\rangle) .$$

which can be rewritten as

$$\frac{1}{2}(|00\rangle \otimes (\alpha|0\rangle + \beta|1\rangle) + |01\rangle \otimes (\alpha|1\rangle + \beta|0\rangle) + |10\rangle \otimes (\alpha|0\rangle - \beta|1\rangle) + |11\rangle \otimes (\alpha|1\rangle - \beta|0\rangle)) .$$

Now I will seperately consider the cases where the first two qubits are measured to be $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$:

Measurement	Qubit 3 before operation	Operator applied to qubit 3	Qubit 3 after operation
$\|00\rangle$	$\alpha\|0\rangle + \beta\|1\rangle$	I	$\alpha\|0\rangle + \beta\|1\rangle$
$\|01\rangle$	$\alpha\|1\rangle + \beta\|0\rangle$	X	$\alpha\|0\rangle + \beta\|1\rangle$
$\|10\rangle$	$\alpha\|0\rangle - \beta\|1\rangle$	Z	$\alpha\|0\rangle + \beta\|1\rangle$
$\|11\rangle$	$\alpha\|1\rangle - \beta\|0\rangle$	ZX	$\alpha\|0\rangle + \beta\|1\rangle$

In each case, the result is qubit 3 will be in the state $\alpha|0\rangle + \beta|1\rangle$.

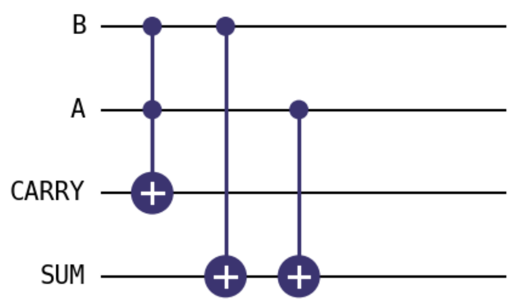
(b) See the Jupyter notebook at the end of this doc.

Problem 0.3.

Here was my first guess for how to approach the problem, which turned out to be incorrect:

> Call the two input bits A and B, and the output bits SUM and $CARRY$. The operator I want to design can be represented in the $(|00\rangle, |01\rangle, |10\rangle, |11\rangle)$ basis as
>
> $$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$
>
> where the input is $A \otimes B$ and the output is $CARRY \otimes SUM$.

However, that matrix is not unitary, so we can only implement a half-adder by adding at least one ancilla bit. It turns out if we use two ancilla bits, there's a very elegant and straightforward solution. Here, A and B are the bits to be added, and the SUM and $CARRY$ registers are both initialized to $|0\rangle$. See the Jupyter notebook for my code proving that this half-adder works.



Problem 0.4.

(a) The oracle is defined as the operator which maps $|s\rangle$ to

$$O|s\rangle := \cos\theta\,|s'\rangle - \sin\theta\,|q\rangle$$

and the diffuser is defined as the operator

$$D := 2|s\rangle\langle s| - I.$$

We can rewrite D as

$$\begin{aligned} D &= 2\left(\cos\theta\,|s'\rangle + \sin\theta\,|q\rangle\right)\left(\cos\theta\,\langle s'| + \sin\theta\,\langle q|\right) - \left(|s'\rangle\langle s'| + |q\rangle\langle q|\right) \\ &= 2\cos^2\theta\,|s'\rangle\langle s'| + 2\cos\theta\sin\theta\,|s'\rangle\langle q| + 2\cos\theta\sin\theta\,|q\rangle\langle s'| + 2\sin^2\theta\,|q\rangle\langle q| - |s'\rangle\langle s'| - |q\rangle\langle q|. \end{aligned}$$

When we apply that operator to $O|s\rangle$, we get

$$\begin{aligned} DO|s\rangle &= 2\cos^3\theta\,|s'\rangle - 2\cos\theta\sin^2\theta\,|s'\rangle + 2\cos^2\theta\sin\theta\,|q\rangle - 2\sin^3\theta\,|q\rangle - \cos\theta\,|s'\rangle + \sin\theta\,|q\rangle \\ &= \left(\cos^3\theta - 3\cos\theta\sin^2\theta\right)|s'\rangle + \left(3\cos^2\theta\sin\theta - \sin^3\theta\right)|q\rangle \\ &= \cos(3\theta)\,|s'\rangle + \sin(3\theta)\,|q\rangle. \end{aligned}$$

For that last step, I used the "triple-angle identity", which I will derive here:

$$\begin{aligned} e^{3i\theta} &= (\cos\theta + i\sin\theta)^3 = \cos^3\theta + 3i\cos^2\theta\sin\theta - 3\cos\theta\sin^2\theta - i\sin^3\theta \\ \cos(3\theta) &= \Re(e^{3i\theta}) = \cos^3\theta - 3\cos\theta\sin^2\theta \\ \sin(3\theta) &= \Im(e^{3i\theta}) = 3\cos^2\theta\sin\theta - \sin^3\theta. \end{aligned}$$

(b) Before applying Grover's step, the qubit is in an equal superposition of 4 states, meaning the coefficient of $|q\rangle$ is $1/2$ and the coefficient of $|s'\rangle$ is $\sqrt{3}/2$. That means we can say $\theta = \pi/6$, so

$$DO\,|s\rangle = \cos(\pi/2)\,|s'\rangle + \sin(\pi/2)\,|q\rangle = |q\rangle\,,$$

which means after one step of Grover's algorithm, the system has a 100% chance of being in the solution state.

notebook

December 5, 2024

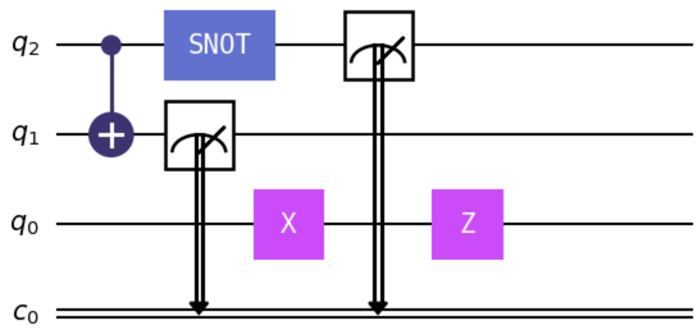
[1]:
```
!pip install qutip-qip
```

```
Requirement already satisfied: qutip-qip in
/home/nathan/anaconda3/lib/python3.9/site-packages (0.4.0)
Requirement already satisfied: qutip>=4.6 in
/home/nathan/anaconda3/lib/python3.9/site-packages (from qutip-qip) (5.0.4)
Requirement already satisfied: numpy>=1.16.6 in
/home/nathan/anaconda3/lib/python3.9/site-packages (from qutip-qip) (1.26.4)
Requirement already satisfied: scipy>=1.0 in
/home/nathan/anaconda3/lib/python3.9/site-packages (from qutip-qip) (1.13.1)
Requirement already satisfied: packaging in
/home/nathan/anaconda3/lib/python3.9/site-packages (from qutip-qip) (21.0)
Requirement already satisfied: pyparsing>=2.0.2 in
/home/nathan/anaconda3/lib/python3.9/site-packages (from packaging->qutip-qip)
(3.0.4)
```

[2]:
```
import qutip as qt
from qutip_qip.circuit import QubitCircuit
import matplotlib.pyplot as plt
import numpy as np
```

[7]:
```
# Problem 2(b)
qc = QubitCircuit(3, num_cbits=1)
qc.add_gate("CNOT", controls=2, targets=1)
qc.add_gate("SNOT", targets=2)
qc.add_measurement("M0", targets=1, classical_store=0)
qc.add_gate("X", targets=0, classical_controls=0)
qc.add_measurement("M0", targets=2, classical_store=0)
qc.add_gate("Z", targets=0, classical_controls=0)

# This throws an error if I try to draw in the LaTeX style (which
# is supposed to look a lot nicer than the matplotlib style)
qc.draw("matplotlib", bulge=False)

for i in range(100):
    # I didn't wanna do symbolic computation with alpha and beta,
    # so instead, just check that teleportation works for a ton
    # of randomly generated qubits (this is super janky)
```

```
    Psi = (np.random.uniform(-10, 10) + np.random.uniform(-10, 10) * 1j) * qt.
↪ket("0") + \
        + (np.random.uniform(-10, 10) + np.random.uniform(-10, 10) * 1j) * qt.
↪ket("1")
    Psi = Psi.unit()
    initial_state = qt.tensor(Psi, qt.bell_state("00"))
    final_state = qc.run(initial_state)
    teleported_state = final_state.ptrace(0)
    # Using ptrace turns the state into a density matrix, so now I
    # need to compare it to the density matrix of the original state
    assert teleported_state == Psi * Psi.dag()

print("It doesn't show up here, but the magenta X and Z gates are both " \
      "controlled by the c_0 register.\nq_2 is the qubit to be teleported, " \
      "and q_0 \otimes q_1 is the bell state \Phi^+")
```

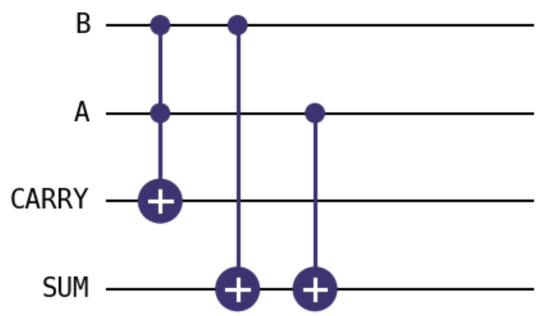



```
It doesn't show up here, but the magenta X and Z gates are both controlled by
the c_0 register.
q_2 is the qubit to be teleported, and q_0 \otimes q_1 is the bell state \Phi^+
```

[4]:
```
# Problem 3
from qutip_qip.operations import Measurement

qc = QubitCircuit(4)
qc.add_gate("TOFFOLI", controls=[3, 2], targets=[1])
qc.add_gate("CNOT", controls=[3], targets=[0])
qc.add_gate("CNOT", controls=[2], targets=[0])

for input_bits in ["00", "01", "10", "11"]:
    initial_state = qt.ket("00" + input_bits)
    final_state = qc.run(initial_state)
```

```
    sum_bit   = Measurement("sum",   targets=[0]).
↪measurement_comp_basis(final_state)[1][1]
    carry_bit = Measurement("carry", targets=[1]).
↪measurement_comp_basis(final_state)[1][1]
    A = input_bits[0]
    B = input_bits[1]
    print(f"{A} + {B} = {int(carry_bit)}{int(sum_bit)}")

qc.draw("matplotlib", bulge=False, wire_label=["SUM", "CARRY", "A", "B"])
```

```
0 + 0 = 00
0 + 1 = 01
1 + 0 = 01
1 + 1 = 10
```



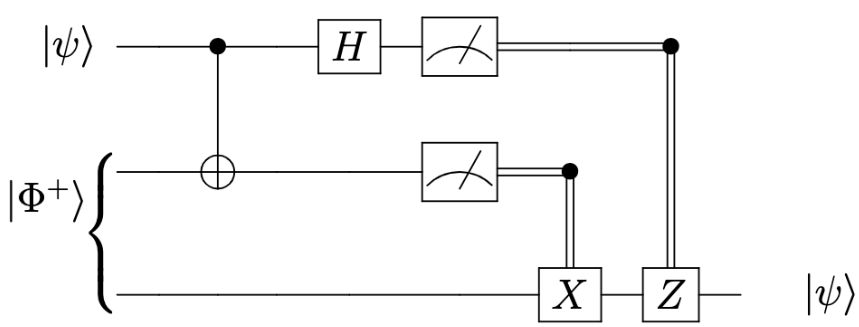
[ ]:

```
```

Phys 245 Quantum Computation
Homework 8

1. *[50] Universal Single Qubit Gates.* In this problem, we'll explore how the $\hat{T}$ and $\hat{H}$ gate can be used to effect an arbitrary single qubit gate.
    a. [5] Show that the up to a global phase, the $\hat{T}$ gate can be written as a rotation about the $\hat{z}$ axis and find the angle of rotation.
    b. [15] Show that the up to a global phase, the combination of $\hat{H}\hat{T}\hat{H}$ gates can be written as a rotation about the $\hat{x}$ axis and find the angle of rotation.
    c. [25] Find the evolution due to the combination of gates $\hat{T}\hat{H}\hat{T}\hat{H}$ and show that it can be written as $e^{-\iota\frac{\theta}{2}(\hat{n}\cdot\vec{\sigma})}$ and determine $\hat{n}$ – remember $\hat{n}$ needs to be normalized -- and θ.
    d. [5] Given that the θ you just found is irrational, how can this be used to produce an arbitrary single qubit rotation?

2. *[50] Beam me up, Scotty!* The usual set up for quantum teleportation is to imagine two folks, Alice and Bob. Alice and Bob share a pair of qubits, with one qubit each, that are in the $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(\,|00\rangle + |11\rangle)$ Bell state. The details of how they got in this state aren't important for us, but you could imagine they were entangled then Bob put his in his suitcase and flew off to the Moon or wherever. Alice also has an additional qubit in some state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. The goal of quantum teleportation is to put Bob's qubit into the state $|\psi\rangle$ using only classical communication. A quantum circuit that accomplishes this teleportation is:



    a.) [30] Work this out by hand and show that the teleportation is indeed accomplished by this circuit.
    b.) [20] Work this in QuTip using an instance of QubitCircuit() and show that the teleportation is indeed accomplished by this circuit.

3. [20] *1+1 = ?* A [two-bit half adder circuit] is a Boolean logic circuit that adds two bits and outputs their sum. Design a quantum version of this circuit and implement it in QuTip to show it works.
4. [20] *Grover loves math.* Assume we are using Grover's algorithm to solve an unstructured search problem with only one solution $|q\rangle$.

a. [15] Show that when starting in an equal superposition state $|s\rangle = \cos\theta\,|s'\rangle + \sin\theta\,|q\rangle$, after applying the oracle $\hat{O}$ and diffuser $\hat{D}$ operators the system is in the state: $\hat{D}\hat{O}|s\rangle = \cos 3\theta\,|s'\rangle + \sin 3\theta\,|q\rangle$. Here $|s'\rangle$ is an equal superposition of all states except the solution $|q\rangle$.

b. [5] After one application of the Grover's step, i.e. $\hat{G} = \hat{D}\hat{O}$, what is the probability of finding the system in the solution state if a measurement is performed for a total state space dimension of N = 4?