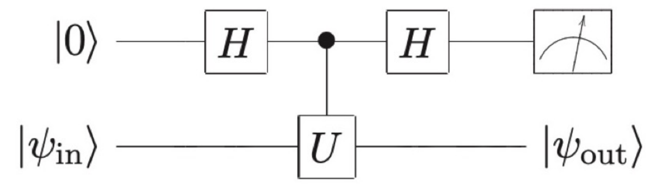
Phys 523B: Fault tolerant quantum commputation
Homework 5

Part I (Short answers)

Q1. Given the following Hadamard test circuit:



where U is unitary and Hermitian and has eigenvalues ± 1, show that:

a). depending on the measurement outcome of the ancilla qubit $|0\rangle$, $|\psi_{out}\rangle$ will be one of corresponding eigenvector of U;

b). through repeated measurements, this circuit can be used to evaluate the expectation value $\langle U \rangle$. Or if U is non-Hermitian, it can evaluate the real part of $\langle U \rangle$.

c). What is the variance (statistical errror) $\epsilon^2 = Var[\langle U \rangle]$ of the expectation value estimator in b)?

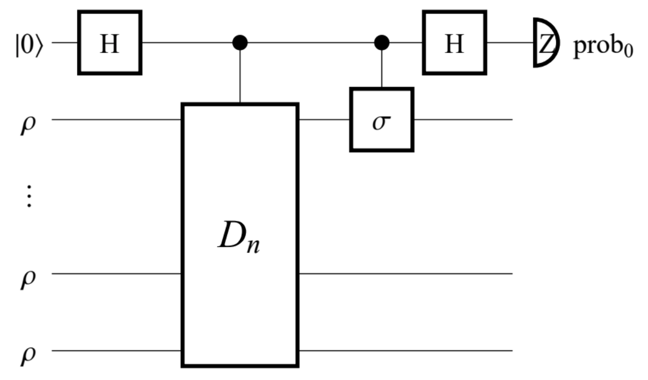
Part II

Q2. In this question we will use Qiskit to simulate the error suppression by derangement, based on this [reference (https://arxiv.org/pdf/2011.05942.pdf)](https://arxiv.org/pdf/2011.05942.pdf). If you are new to Qiskit, you might find this [tutorial (https://qiskit.org/documentation/intro_tutorial1.html)](https://qiskit.org/documentation/intro_tutorial1.html) helpful.

Consider the erroneous ansatz state of interest given by:

$$\rho = \lambda|\psi\rangle\langle\psi| + (1-\lambda)\sum_{k=2}^{2^N} p_k|\psi_k\rangle\langle\psi_k|,$$

where the $|\psi\rangle$ is the ideal (or dominant) input state and we assume that all $|\psi_k\rangle$ and $|\psi\rangle$ are orthogonal. The circuit for mitigating the observable measurement error is:



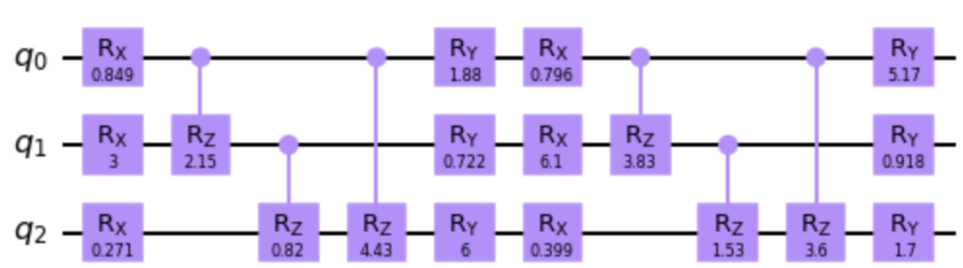
There are multiple copies of the input state ρ, the σ is the obervable to be measured, and the D_n is the circuit that permutes the copies of ρ. D_n provides a full cyclic permutation: $D_n|\psi_1, \psi_2, \ldots \psi_n\rangle = |\psi_2, \psi_3, \ldots \psi_1\rangle$.

The estimator for the observable measurement is ('Method A' in the reference paper):

$$\frac{2prob_0 - 1}{2prob_0' - 1} = \langle\psi|\sigma|\psi\rangle + \epsilon,$$

Note: The code below is merely for reference. Please feel free to code however you like.

a). The input state ρ is usually generated by an parameterized ansatz circuit. Here is an example circuit with randomly parameterized single and two- qubit gates (see Appendix E of the reference paper):



Construct a similar circuit and evaluate the ideal expectation value of the observable Pauli string "$\sigma = X_0Y_1Z_2$". (You can do this with 'aer_simulator' backend)

In [ ]:
```
import numpy as np
from qiskit import QuantumCircuit, Aer, transpile
import qiskit.quantum_info as qi

qc = QuantumCircuit(3)

### Create the ansatz circuit

qc.draw('mpl')
```

In [ ]:
```
# The following code uses aer_simulator backend
simulator = Aer.get_backend('aer_simulator')
qc = transpile(qc, simulator)

Pauli_obs = qi.Pauli('ZYX') # Reversed string to match Qiskit convention, see qiskit.quantum_info.Pauli
qubit_list = range(3)
qc.save_expectation_value(Pauli_obs,qubit_list)
result = simulator.run(qc).result()

exp = result.data()['expectation_value']
print("The ideal expectation value is ",exp)
```

b). Add the ancilla qubit and construct the Hadamard test circuit as in Q1 and evalutate the error-free 'experimental' expectation value in a). Then apply a 5% and 10% of depolarizing error associcated with the single and two- qubit gates of the ansatz circuit and evaluate the noisy 'experimental' expectation value. What are the errors ϵ for both cases? (Use the 'qasm_simulator' backend to simulate the sampling)

In [ ]:
```
import qiskit.providers.aer.noise as noise

qc = QuantumCircuit(4,1)

### Create the full circuit

# Error model
prob_1 = 0.05  # 1-qubit gate
prob_2 = 0.1  # 2-qubit gate

error_1 = noise.depolarizing_error(prob_1, 1)
error_2 = noise.depolarizing_error(prob_2, 2)

noise_model = noise.NoiseModel()
noise_model.add_all_qubit_quantum_error(error_1, ['rx','ry','rz'])
noise_model.add_all_qubit_quantum_error(error_2, ['crz'])

### Execute the circuit
simulator = Aer.get_backend('qasm_simulator')
```

c). Add a second copy of ρ, and implement the controlled-D_n for the two copies. Under the same error model in b), pick 20 random Pauli strings as observables, plot a histogram (log-scale) of their absolute errors $abs(\epsilon)$ versus the number of copies $n = 1, 2$.