
Quantum Information and Quantum Computing, Problem set 11

Assistant : stefano.barison@epfl.ch, clemens.giuliani@epfl.ch

The goal of this problem set is to familiarize with the repetition quantum error correction code. Notice that in Qiskit repetition QECC are already part of the library. Here we want to use only our skills and avoid using the simplifications made available by Qiskit

Problem 1 : Repetition quantum error correction code

Consider a circuit with n qubits. We will start with $n = 3$. We will use the n qubits as a repetition code to represent the information of one logical qubit.

1. Design a quantum circuit that, by making use of ancilla qubits, can measure the error syndromes corresponding to the flip of one of the physical qubit of the code. Recall that for a repetition code, the error syndrome should detect if a flip occurred and on which of the physical qubits it happened.
2. Implement the circuit on the QASM simulator. The circuit should give as an output the measurement of the n physical qubits on the computational basis, and of the error syndrome. Recall that for $n = 3$ the error syndrome should essentially detect four possible cases, i.e. no flip, and one flip on each of the three qubits. Therefore, it should require two ancilla qubits.
3. Use a noise model and investigate how effective the QECC is to detect bit flip errors. You can carry out this analysis with initial states $|000\rangle$, and $|111\rangle$.
4. Now prepare the initial state into a logical state $|\psi\rangle$ of your choice (remember that we saw in class a circuit to encode an arbitrary state onto a repetition code). Repeat the error detection. Are there more errors than before?
5. Observe how the number of errors increases if, before the error syndrome measurement, one carries out a series of one-qubit operations on the logical qubits. You can start from the state $|000\rangle$, and execute a sequence of one-qubit gates, and then reverse the sequence, so that the overall effect would ideally be to revert to state $|000\rangle$.