Version 2021-07-07

Cohort Project 2021 Week 1 Report

Team 19: Michael Himmelfarb, Leo Belostotski, Johannes Zeiher, Sebastian Blatt

# Task 1: Simulate random circuit using matrix product states

State-vector probabilities for all randomly generated qubits can be found: Task1\_N\_Qubits.xlsx

Below is a sub-set of ‘speckle plots’ for N= 1,2,4,8.

Figure N = 1, Depth = 100, qubit 1 state-vector probability

Figure N = 2, Depth = 100, qubit 1 state-vector probability

Figure N = 4, Depth = 100, qubit 2 state-vector probability

Figure N = 8, Depth = 100, qubit 4 state-vector probability

Notes:

* By sweeping number of qubits in the random-circuit simulation we can see that the maximum number of states for any qubit increases with 2N.

State-vector probabilities for all randomly generated qubit depths can be found in: Task1\_Depth\_4\_Qubits.xlsx

Below is a sub-set of ‘speckle plots’ for N= 4 for depths = 1, 10, 100, 1000.

Figure N = 4, Depth = 1

Figure N = 4, Depth = 10

Figure N = 4, Depth = 100

Figure N = 4, Depth = 1000

Notes:

* when depth is 1, the maximum number of states can fall below 2^N for this random-noisy circuit.

# Task 2: Effect of a bit-flip error

State-vector probabilities for all randomly generated qubits can be found: Task2\_state\_flip\_4\_Qubits.xlsx

Below is a sub-set of ‘speckle plots’ for N= 4, with a X-gate applies to random gates (512, 324, 901) for a depth = 1000.

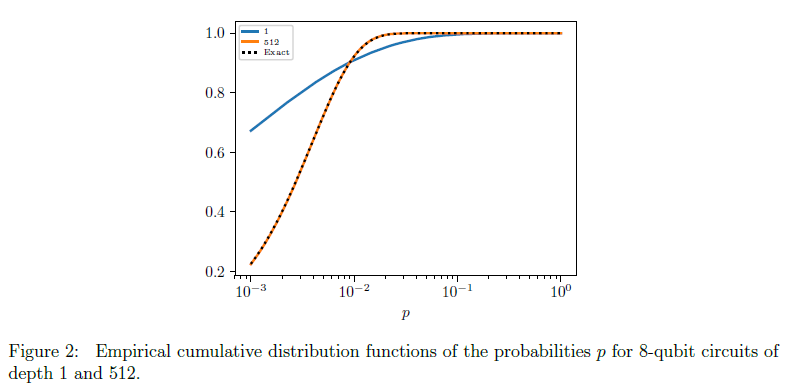
Figure qubit 3, state-vector probability collage comparison

Notes:

* The state probability of any qubit is strongly impacted by a single X-gate state-flip.
* States with relatively high probability to occur (ex: 1,6,12) are less affected by gate-flips (given the same random circuit with no error), but this may be due to the relatively low number of random flips introduced.

# Task 3: Convergence of Porter-Thomas distribution

Fig. 2 from the ‘instructions.pdf’ file, provides an example of the empirical CDF for depths of 1 and 512 compared to ideal:



We added another plot from the task 1 simulation using 8-bits and depth = 100. Original data can be found in: Task3\_N\_Qubits.xlsx

Figure N = 8, Depth = 100, qubit 4 Empirical CDF from simulated random-noise circuit

We found that our results with depth = 100 produced a good fit between 1 and 512 as shown in the example. There is a very high degree of agreement with the “Porter-Thomas distribution", but we will not go into how good of a ‘fit’ it is.

# Task 4: Effect of two-qubit gate errors

Not attempted.

# Task 5 (optional): Animate the speckle pattern

Not attempted.

# Task 6 (optional): Reproduce Google’s cross-entropy results

Not attempted.

# Task 7 (optional): Implement the circuit on trapped-ion hardware

Not attempted.

# Business Application

## Explain the technical problem to a layperson

The technical problem tackled in this exercise deals with showing that a quantum computer can be better at specific tasks than a classical computer. The argument builds on the comparison of measurement outcomes of a quantum computer with classically simulated results.

For this, the quantum computer is run many times to create a variety of measurement outcomes with their relative occurrences. To make the problem in some sense universal (i.e. to make sure no tricks can be used in the quantum-classical competition), the quantum computer is run with *random* gate sequences.

If we wanted to predict the exact output state given a certain input state on a classical computer, we would have to model the full quantum mechanical time evolution based on the Schrödinger equation. For the random circuits without simplifying assumptions, this becomes exceedingly difficult when trying to classically simulate more and more qubits, because simulating one extra qubit requires doubling the amount of memory required to store the full quantum state.

Using a specially chosen quantity to benchmark the quantum calculation, the so-called “cross-entropy”, Google claims that it’s significantly easier to do their randomized gate experiment on a quantum computer, rather than trying to predict the result on a classical computer. Easy in this case refers to the time it takes to run the calculation. This claim is referred to as “quantum supremacy” or “quantum advantage.”

What is interesting is that the quantum information disappears even more rapidly under the application of random gates when these gates are not perfect. Quantifying how quickly the information gets lost allows making statements about how perfectly “quantum” a given quantum computer is. Although there are better ways to compare different quantum computers, the randomized gate experiment is a simple way to rank these machines.

## Explain or provide examples of the types of real-world problems this solution solves

The significant advantage of the randomized gate experiment on a quantum computer over a classical computer is a bit hard to grasp because it’s a problem specifically designed to make the quantum computer look as good as possible, while making strong assumptions about how one could do the calculation classically. However, the Google experiment does demonstrate that a quantum computer can be used as a sampling device for statistical distributions and that it can be more efficient at this specific sampling task than classical hardware. If a real-world classical calculation problem can be mapped to such a sampling problem, then a quantum computer can have a significant advantage over a classical computer. Similarly to the argument about quantum supremacy, “more efficient” here implies faster and with fewer resources, which implies cost savings for business applications.

Sampling appears in many areas, for example in the guise of Monte-Carlo simulations. Real-world examples include

* The evaluation of complicated integrals in inverse problems, for example in medical imaging, where using quantum hardware could result in a direct speedup and an increased image quality.
* The evaluation of stochastic differential equations (differential equations with noise terms) for example in finance. Specifically in high-frequency trading, any speedup will result in a direct competitive advantage for the user. Here is another finance application promoted by Zapata computing: <https://www.zapatacomputing.com/publications/quantum-algorithm-cva/>
* Tolerancing in advanced manufacturing, for example when trying to predict the performance of electronic circuits taking component manufacturing variability into account.
* Prediction of flight trajectories and their uncertainty for defense applications.

## Identify at least one potential customer for this solution

If Monte-Carlo simulations are a low-hanging fruit for quantum computers, then there are very many potential customers, ranging from aerospace to materials to medical companies who have large computational needs. Basically any large company does Monte Carlo simulations of some kind. However, one would have to look very carefully at the potential speedups for each individual application, and in our opinion, sweeping claims of “Quantum is good for everything” without a careful analysis must be avoided.

## Prepare a 90 second video explaining the value proposition of your innovation to this customer using non-technical language

Not attempted.