README

Introduction:

Proof of Work is a type of cryptography used in Bitcoin, the main component in Bitcoin mining. It would be nice to make it compatible with quantum computers so that the billions currently spent on mining could be spent on quantum computer manufacturing and research. This would also have the additional benefit of decreasing the energy consumption of the system which, today, is greater than that of some European countries (https://ccaf.io/cbeci/index)[2].

Here we are building on previous work on Optical Proof of Work, a cryptographic PoW protocol developed for compatibility with classical optical analog computers. Please refer to Towards Optical Proof of Work [1] for a more detailed description/introduction to Bitcoin, Proof of Work, energy consumption, cryptography, and so on.

Our goal is to develop qPoW, a cryptographic protocol that can be accelerated with existing quantum computers while all the security properties needed to use it in place of the existing protocol used in Bitcoin.

Bitcoin PoW

Resources:

How Bitcoin works (please watch this if you are confused):

- 3Blue1Brown breakdown video [3]

Original Bitcoin Whitepaper (awesome paper, highly recommended)

- https://bitcoin.org/bitcoin.pdf [4]

Hashcash Diagram:

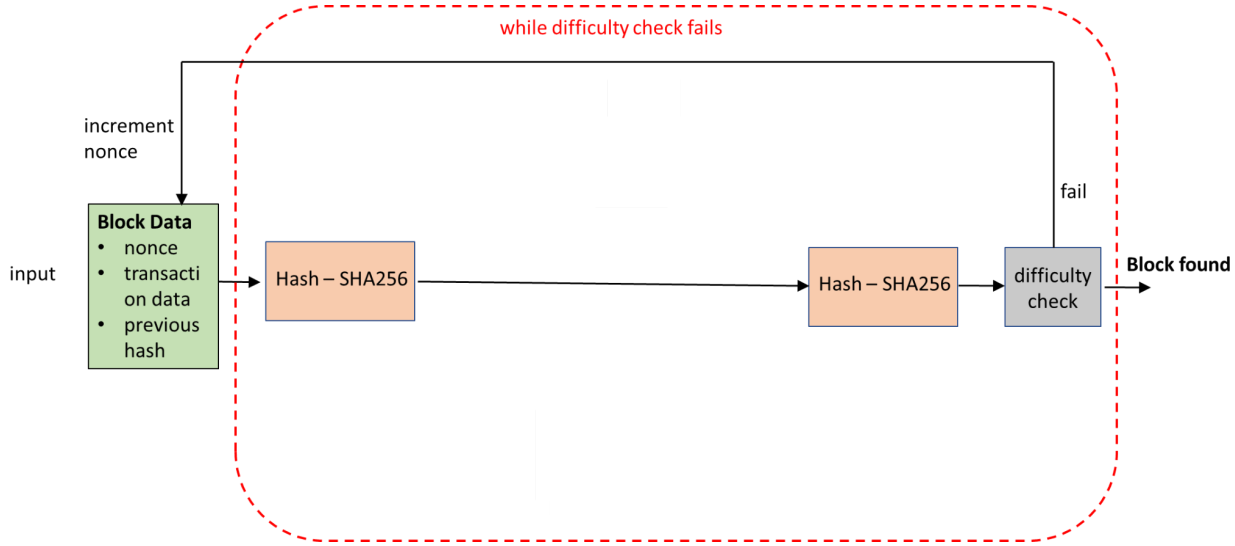


Fig 2

qPoW Diagram:

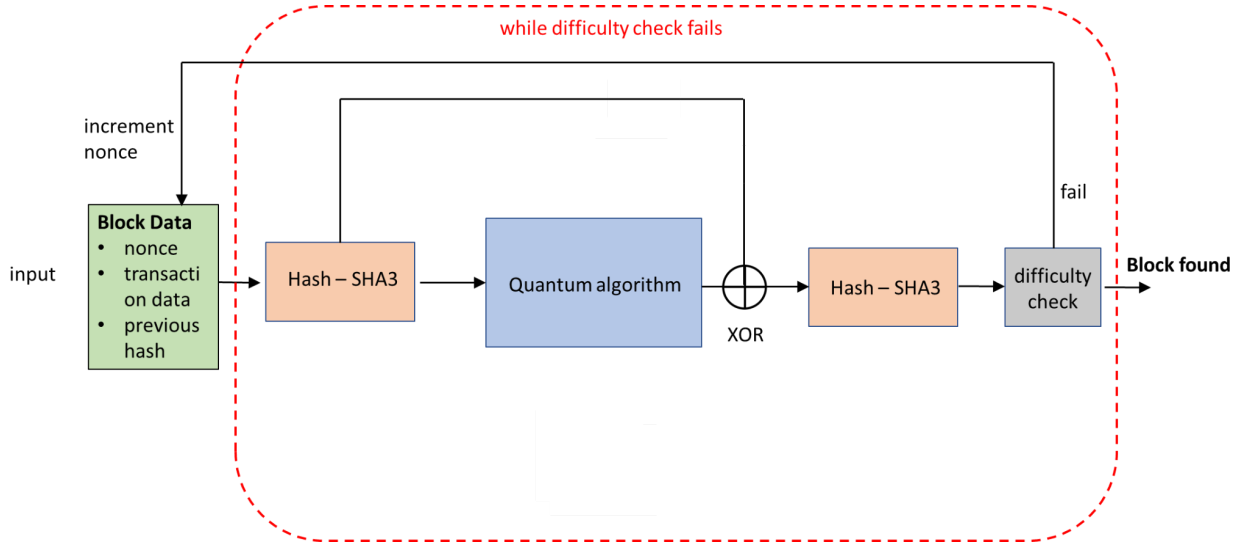


Fig 3

Necessary attributes of the Quantum Operation

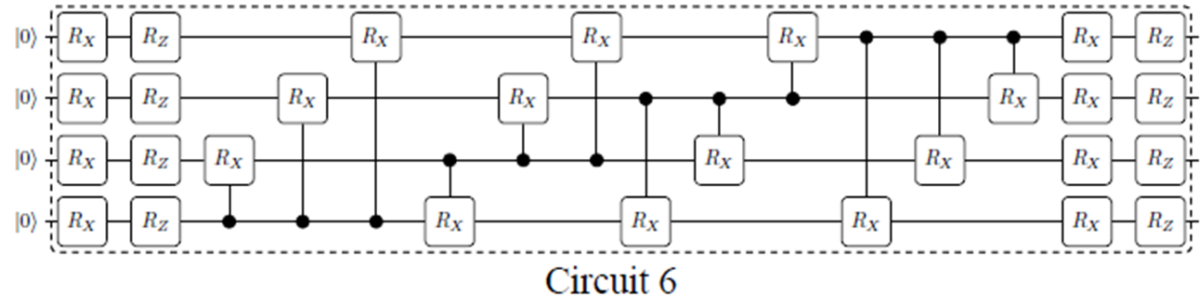
a. It must have the same cryptographic security as SHA256 from the perspective of Hashcash (the protocol that secures Bitcoin and is colloquially called “mining”). This is achieved by following the same cryptographic construction as oPoW[2], detailed in a diagram below, with a modification to the algorithm that replaced the optical computation block with an alternative and novel quantum computation block.
b. As a rule of thumb, if the ratio of the cost of a GPU to the Quantum Computer is X then the difficulty of performing the qPoW calculation with a GPU must be >X or ideally >10X
c. When a “miner” finds a solution or “block” by brute force it must be possible for others to verify the computation using a classical computer, meaning the output of the quantum computation must either agree with simulations or be verifiable to be “acceptable” in some other way.
d. This means that if you run qPoW on the QC its results must agree with the verification often enough to be useful. If the computational advantage over a GPU is very high the accuracy may be lower. There is no penalty for producing solutions that don’t verify besides wasted time on the QC.
e. Note that the qPoW operation will be performed many times to mine a block (continuously for 1-10 minutes) but to verify a mined block it only needs to be performed once. However, every node in the network performs the verification so the verification must be relatively inexpensive.

Options Considered for Quantum Operation

Randomly Initialized High Expressivity Quantum Circuit

Quantum Circuit Selection

We started the most expressive circuit from [5], number 6 in Fig 3 below, in order to increase our chances of simulation agreeing with hardware results, however, we found empirically that a modification of the position of one layer of Rz gates, seen in Fig 4; gave better agreement between simulation and experiment, see Fig 5 and 6. We hypothesize that this is related to the physical topology differences between the QC used in [5] and the IonQ and IBM QCs we tested.



Circuit 6

Fig 3

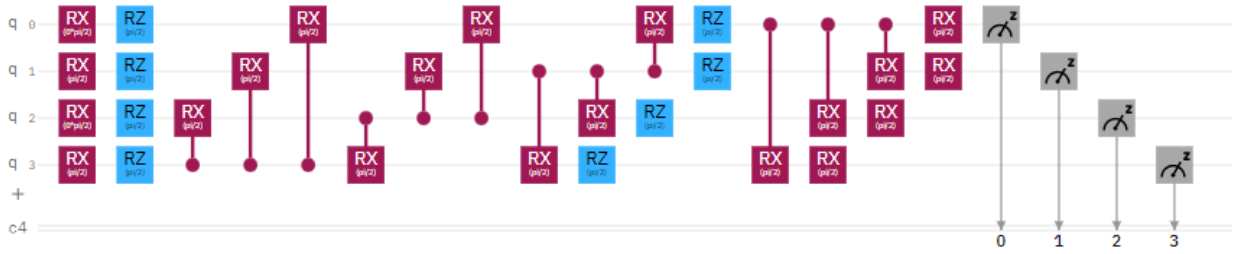
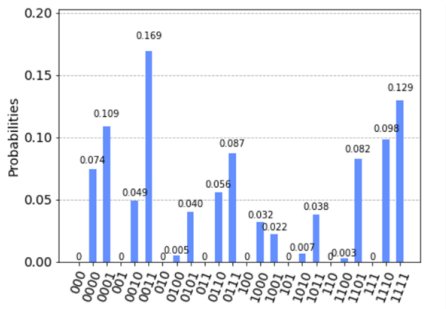
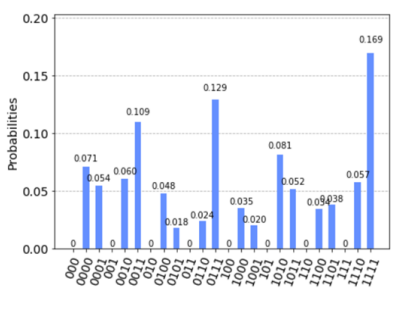


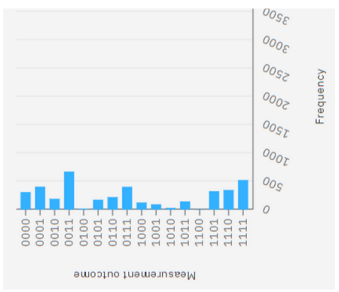
Fig 4



IonQ simulator

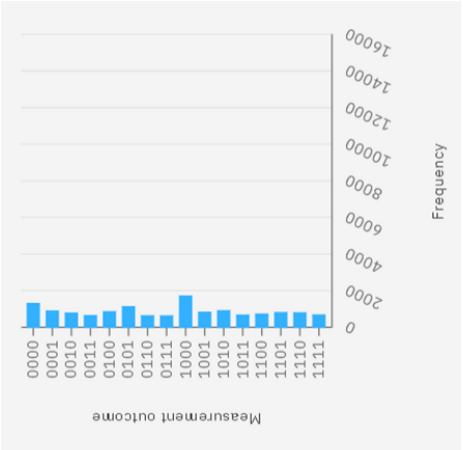


IonQ qpu

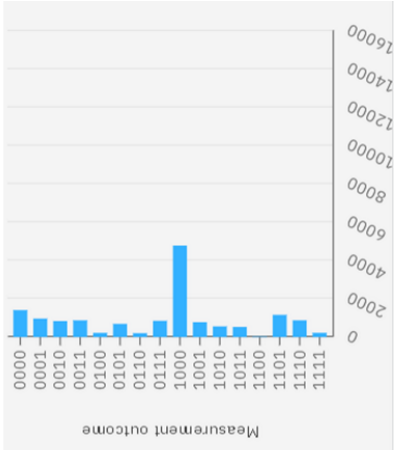


IBM simulator

Fig 5: Original Circuit from Fig 3



IBM manila



IBM Qasm

Fig 6: Modified Circuit from Fig 4 results on IBM

This approach has the weakness that verification is completely dependent on the highest probability result of simulation completely agreeing with the highest probability result of the experiment. An approach based on solving optimization problems (detailed below) may offer some advantages from the perspective of verification.

Initialization

Each “nonce” that is iterated through the qPoW construction must produce a unique Quantum Operation. This is accomplished by using the output of the first SHA3 hash to initialize the phases of the gates. Please see the code for more detail.

Implementation

We implemented a 4-qubit version of this construction, which can be easily extended to N-qubits.

Randomly Initialized Impressive Cut

This approach was conceptualized but not fully implemented.

Working Principle

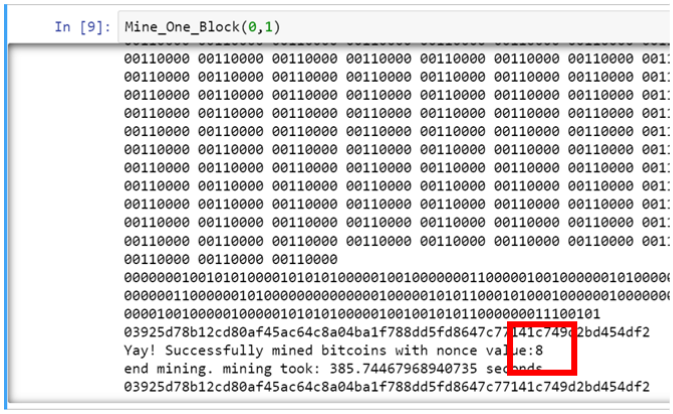
1. Using the output of the first SHA3 hash, a random graph would be initialized
2. QAOA or a similar algorithm would be used to find a candidate for a max cut solution
3. This would not need to be the best solution, it simply needs to be a hard-to-find solution
4. Some criteria would need to be implemented for an “Impressive Cut”
    a. This may be straightforward since the probability of finding a cut higher than N/2 drops to 0 as you approach N/2 + sqrt(N)
5. The quantum miner would use the output of the Impressive Cut as the input to the XOR
6. When reporting a block, the quantum miner would report both the nonce and the output of the Impressive Cut
    a. It is hard to find an Impressive Cut but easy to verify that a cut is impressive, meaning a CPU would be able to both verify that the graph the nonce initializes is indeed compatible with the Impressive Cut solution found by the quantum miner

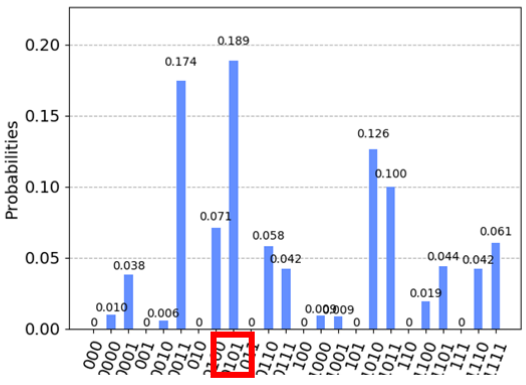
We hope to implement this in the near future.

Results

Successfully Mining Blocks!

We were able to mine qPoW blocks successfully using the IonQ simulator. With a difficulty of 1, the mining time was ~386 seconds.





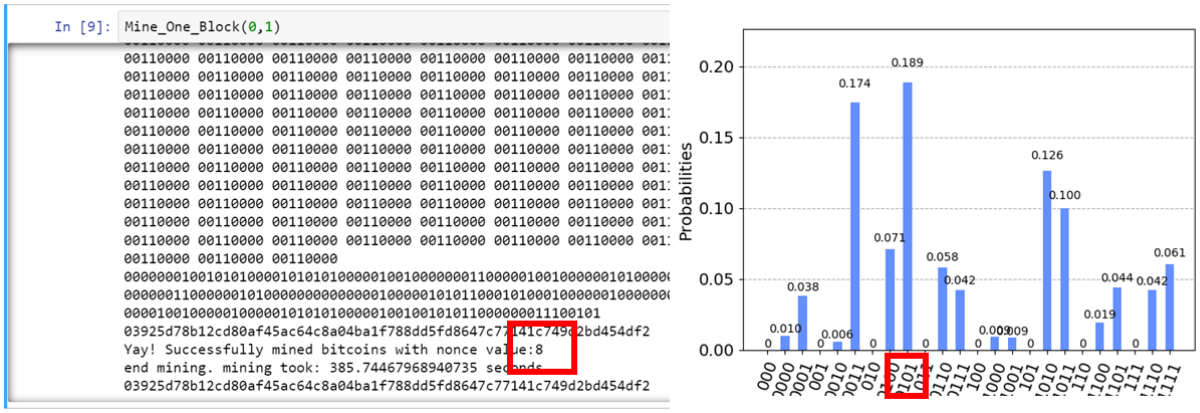


Fig 7 Mined block, showing the nonce and the histogram of the quantum computation

Next Steps

We wrote a script for collecting statistics on the agreement between qPoW run using the simulator vs. the real hardware, however, due to very long queues to use the IonQ hardware we were not able to collect any statistics before the submission deadline.

References

[1] https://assets.pubpub.org/xi9h9rps/01581688887859.pdf
[2] https://ccaf.io/cbeci/index
[3] https://www.youtube.com/watch?v=bBC-nXj3Ng4&ab_channel=3Blue1Brown
[4] https://bitcoin.org/bitcoin.pdf
[5] https://arxiv.org/abs/1905.10876