Rogues Gallery - Quantum Computing

Alexander George-Kennedy, Girish Hari, Dylan Yost, Jack Sullivan, Stefan, Quon

Project Goals

The goal of this project for this semester was to expand upon the XACC quantum circuit optimizer, improving its performance as measured by frequency of producing the most optimal transpiled circuit. We wanted to improve on existing optimizers by creating a more general and extensible framework, for architectures with different connectivity and noise values. We added an additional CNOT optimization this semester, expanding the network size and allowing for much expanded optimization paths. We explored different neural network structures, and started investigating on-the-fly error correcting code additions.

Circuit Splitting Alternatives

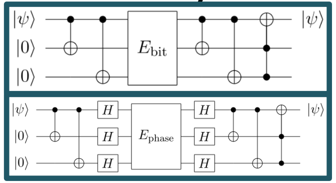
We wanted to look into a method for intelligently slicing circuits, since we currently arbitrarily slice large circuits. However, we opted to focus on researching alternate network layouts that support a continuous input space instead as it scales better and won't miss any potential optimizations.

Mirroring CNOTs

We added a new gate swap to our set of gate swaps. This new swap is the CNOT mirror. This swap uses multi-qubit gates, so we had to go back through and revise how we approach gate swaps in order to allow this swap to work properly. CNOT is a largely used gate in quantum computing, so adding CNOT gate swaps is very important https://arxiv.org/pdf/1110.2998.pdf.

Adding Error Correcting Codes on the fly

We wanted to increase the fidelity of circuits with our optimizer by using Ancilla bits to implement quantum error correcting codes. We identified the bit flip code (Top) and the phase flip code (Bottom) as candidates due to our current 5 qubit limit.

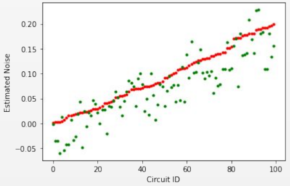


Alternate Network Layout

We explored the conversion of our circuit optimization to a continuous action space (CAS) architecture. One method was to transform our inputs in a way to take advantage of convolution layers to process our observation (input state) into a policy (action) to optimize circuits of arbitrary size due to properties of Fully Convolutional Neural Networks. Another option discussed to this end is the deep deterministic policy gradient (DDPG) algorithm. A circuit-altering action can be outputted in a continuous space, and Q-values computed for both the state of and action on the circuit (https://arxiv.org/pdf/2103.07585.pdf).

Next Steps and Future Challenges

- Consider pulse shapes instead of gates, this would allow for finer control and more generality at the cost of network complexity.
- Training and testing on multiple architectures (especially higher qubit count ones).
- Implementing the alternate network layouts and running tests on which optimizes better.
- Checks for correctness on noise function proof. (https://www.overleaf.com/read/bkppqqdgnxcs)

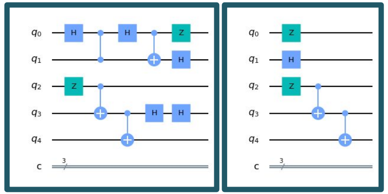


77/100 Circuits improved after network action

Updated Network Results

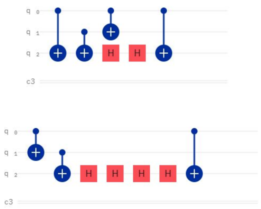
Sample optimizations:

One sample circuit same as Spring 2021

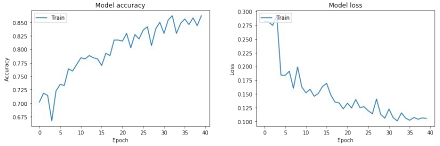


But the optimization path is different: the network tries to use the CNOT mirror, then undoes it.

Erroneous new sample example:



Large network issues:



("Accuracy": probability a move suggestion is legal) Informally, why this inability to converge as well as last semester? Increasing action heads by ~80 to add in the new CNOT mirror action greatly increased network complexity.