Rogues Gallery - Quantum Computing

Alexander George-Kennedy, Girish Hari, Dylan Yost

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 built on (overhauled) the network made in Fall 2020 and set up a framework for different gate representations.

Project Overview and Methods:

We wanted to produce a proof-of-concept neural network that considers the problem of optimizing a quantum circuit on an arbitrary architecture as a single-player game. The game state is the current quantum circuit the network takes as input, as well as the noise values for each component gate in the architecture. The network gives a recommendation on how to alter the circuit to reduce overall noise, through action heads as output. The circuit is then altered as recommended, noise on the circuit evaluated, and these parameters are fed back into the network. We include a memory component to the network, to allow it to remember combinations of past recommendations and the impact "move combinations" have on noise.

Process and Results

Handling Arbitrary Length Circuits:

In order to work with a finite, discrete action space, we slice a given circuit into depth 5 segments.

More Gate Swaps:

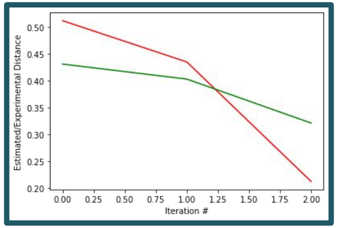
Added CNOT reversal. This allows for some non-obvious optimizations.

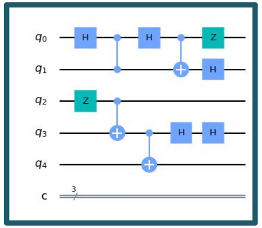
Updated Data Structure:

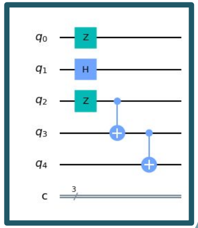
We want to experiment with different representations of gates in our network, and perhaps even move on to working directly with pulses. Our data structure for each gate now contains connectivity and architecture noise information and is extensible for other representations.

Network Updates:

With a static circuit size, we can dedicate one action head to every possible gate swap action: 76 total action heads. Model is now entirely deterministic in choosing gate swaps, and always optimizes trivial (identity) circuits to the identity. We tested 10 randomly generated circuits, and achieved better-than-stock fidelity on 6.







Next Steps and Future Challenges

- Adding more possible gate swaps
- Improving the legal swap constraint efficiency
- Improving the structure of the game model network, possibilities include:
  - Duplicate OpenAI game networks since the action space in our game model should actually be continuous
- Consider pulse shapes instead of gates, this would allow for finer control and more generality at the cost of network complexity.
- Improve upon the circuit-splitting scheme to split circuits "smartly", at locations where it is unlike the network will be able to make any optimizations.
- Training and testing on multiple architectures (especially higher qubit count ones).
- Explore different representations for each gate in the network.