IBM **Quantum**

Simulating Larger Quantum Circuits with Circuit Cutting and Quantum Serverless

Caleb Johnson, Bryce Fuller, Jim Garrison, Jennifer Glick

Quantum Computational Science, IBM Quantum, IBM Thomas J. Watson Research Center

Introduction

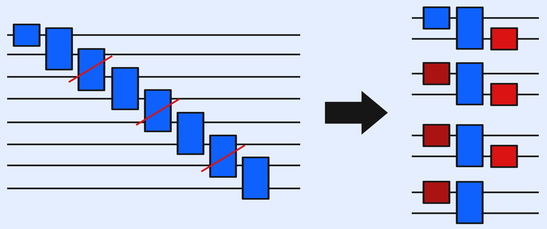
The limited number and quality of qubits poses a challenge for practical usage of near-term quantum computation. Circuit cutting is a technique to decrease the size of circuits at the cost of an additional sampling overhead [1]. This can enable executing problems larger in size and with higher-quality outcomes than what available quantum hardware would otherwise support.

Here, we use the **`Circuit Knitting Toolbox (CKT)`** [4] to demonstrate two applications of circuit cutting. The first is to split circuits into smaller partitions requiring fewer qubits. The second is to cut long-distance quantum gates to reduce circuit depth. We implement a new technique to find a set of cuts that minimizes circuit depth. To scale these workloads up to hundreds of qubits, we use **`Quantum Serverless`** [5] - a new framework for distributing computationally expensive workloads in the cloud.

Methods

Application 1: Gate cutting to reduce qubit overhead

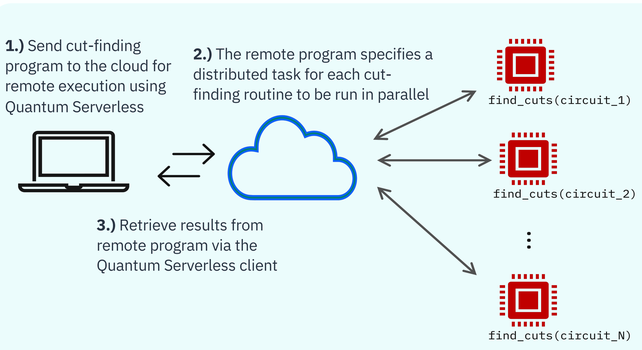
- GHZ circuits with 4-28 qubits
- Split each circuit into 4 parts using 3 gate cuts
- Qiskit Runtime Primitives to compute expvals of full and cut circuits
- Execute 28-qubit circuit on the 27-qubit quantum computer, IBM Hanoi



8-qubit GHZ state preparation circuit 4 subcircuits after cutting

Application 2: Gate cutting to reduce circuit depth with Quantum Serverless

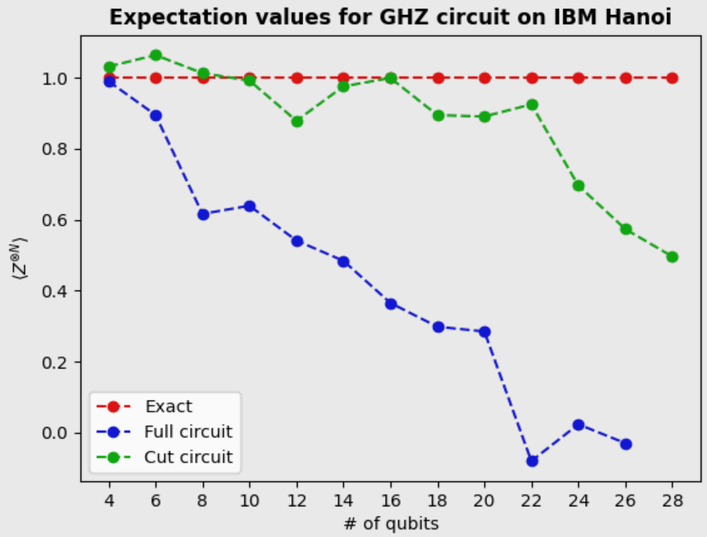
- Qiskit EfficientSU2 circuits with 100-400 qubits, circular entanglement
- Use cut-finding technique to find 4 gates that induce the highest SWAP overhead when transpiling to the 433-qubit Osprey quantum device
- Use Quantum Serverless to efficiently distribute workload in the cloud



Results

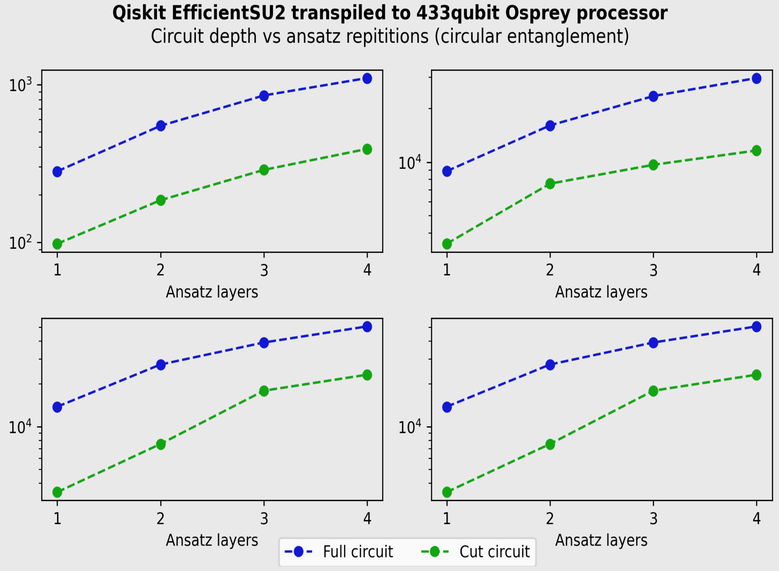
Application 1: Gate cutting to reduce qubit overhead

Without advanced error mitigation, the expectation value of the full circuit quickly deviates from the exact as circuit width is increased. Splitting the circuit into 4 pieces with 3 gate cuts, we obtain more accurate outcomes and can run wider circuits than overwise possible on the 27-qubit IBM Hanoi device.



Application 2: Gate cutting to reduce circuit depth with Quantum Serverless

The results show that for some circuit families, cutting relatively few long-distant gates can result in significant reduction in circuit depth.



Discussion

Application 1

Finding meaningful quantum experiments which can benefit from circuit cutting under this exponential sampling overhead is an ongoing research problem. This experiment was conducted on a very simple circuit and is meant only to demonstrate the validity of the gate cutting technique and propose the CKT as a tool for enabling circuit cutting research.
One recent application of circuit cutting involves investigating the effects of cutting a Quantum Approximate Optimization Algorithm (QAOA) circuit for the maximum cut (MaxCut) problem [2].

Application 2

While circuit cutting can reduce error for some experiments, finding the right places to cut a circuit can be computationally expensive. In this experiment we use the Quantum Serverless API to specify and run a remote program designed to find optimal cut locations for 16 large circuits in parallel.
As shown, the cut-finding routines produced significantly shallower circuits once transpiled onto the Osprey device. Additionally, this demonstrates one way Quantum Serverless can be used to enhance quantum computing workflows.

Outlook

Users of quantum computers during this era of quantum utility will have to deal with noisy quantum results, limitations in qubits, and a lack of local compute resources (e.g. RAM and CPU cores). Circuit cutting can reduce errors in quantum experiments by reducing the size of the circuits needed to run on the quantum processor. Quantum Serverless can be used to designate computationally expensive portions of a workflow to be run in parallel on the cloud or on some specific compute device, like a GPU or HPC cluster.

Although CKT and Quantum Serverless attempt to address limitations in compute resources, more work needs to be done in order to maximize the potential of these tools. Some areas for future consideration are:

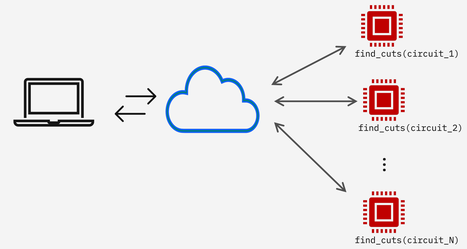
- Leveraging real-time communication during circuit cutting [3]
- Understanding resource requirements of typical quantum workflows

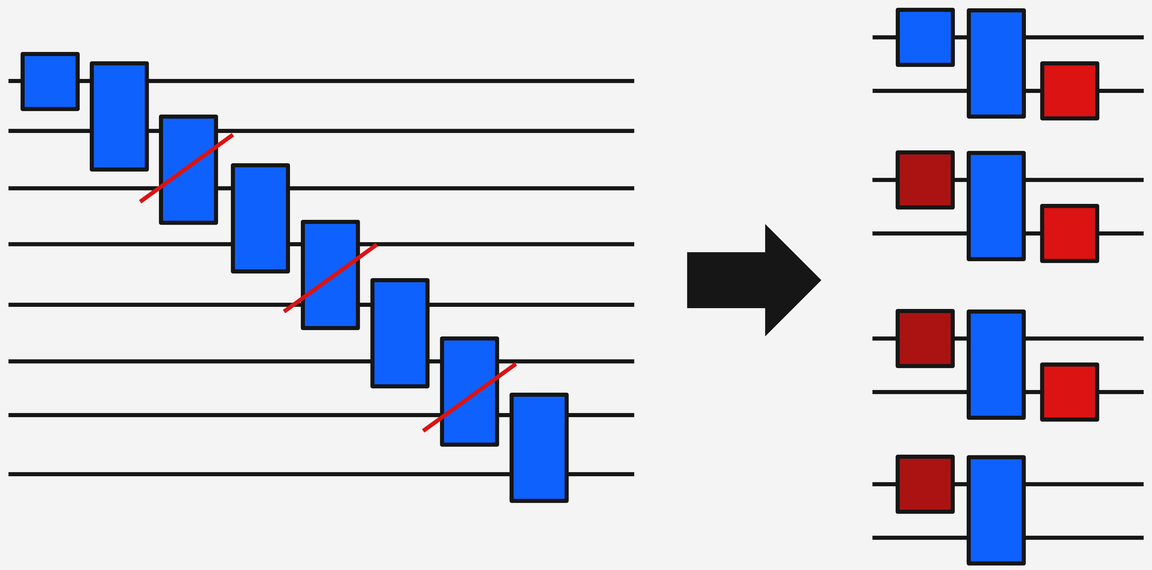
References

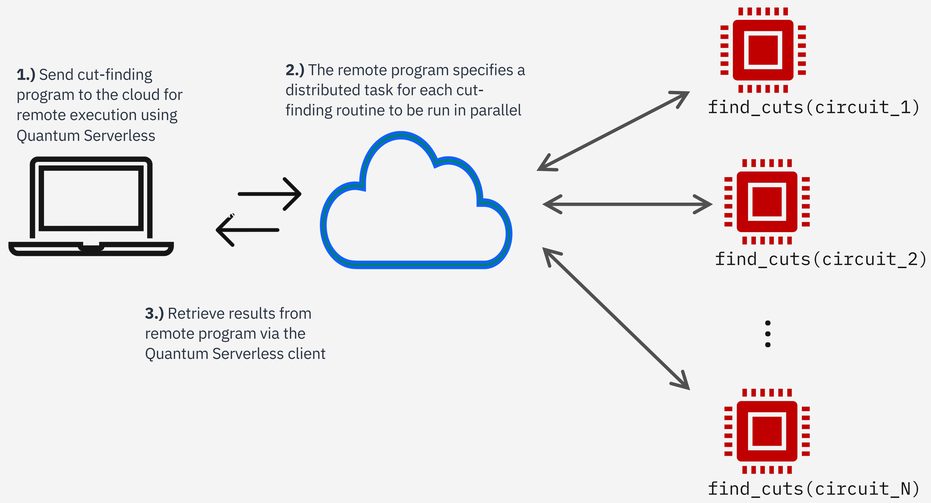
1. Kosuke Mitarai, Keisuke Fujii, “Constructing a virtual two-qubit gate by sampling single-qubit operations,” *New J. Phys.* 23 023021.
2. Marvin Bechtold et al., “Investigating the effect of circuit cutting in QAOA for the MaxCut problem on NISQ devices,” *arXiv:2302.01792* [quant-ph].
3. Christophe Piveteau, David Sutter, “Circuit knitting with classical communication,” *arXiv:2205.00016* [quant-ph].

4. Circuit Knitting Toolbox

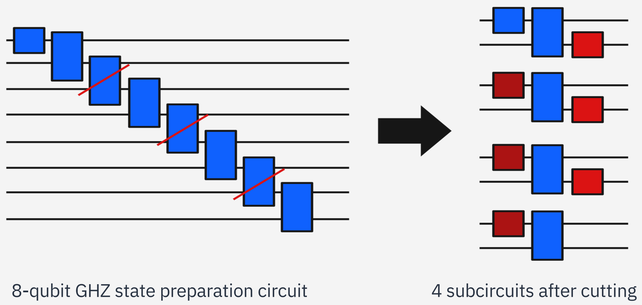
5. Quantum Serverless


find_cuts(circuit_1)
find_cuts(circuit_2)
find_cuts(circuit_N)




1.) Send cut-finding program to the cloud for remote execution using Quantum Serverless
2.) The remote program specifies a distributed task for each cut-finding routine to be run in parallel
3.) Retrieve results from remote program via the Quantum Serverless client
find_cuts(circuit_1)
find_cuts(circuit_2)
find_cuts(circuit_N)

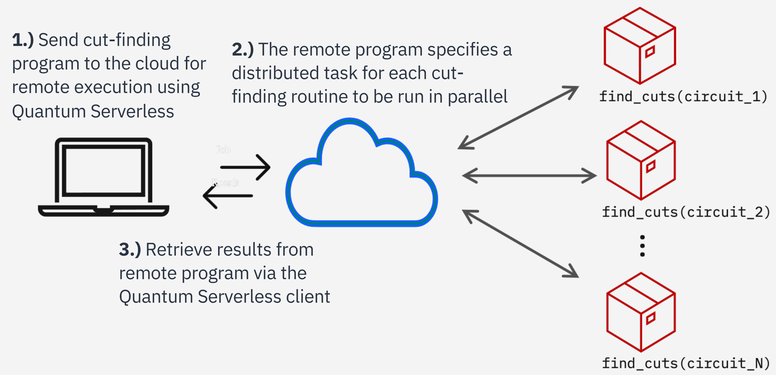
(a)



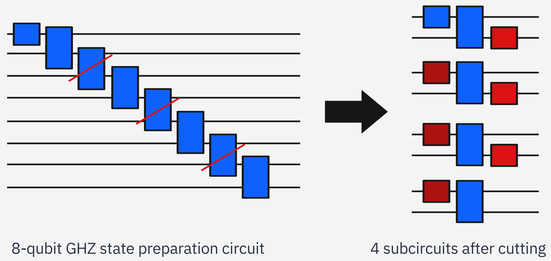
8-qubit GHZ state preparation circuit

4 subcircuits after cutting

(b)



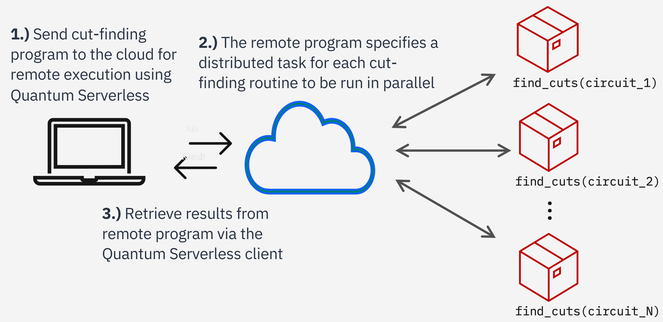
(a)



8-qubit GHZ state preparation circuit

4 subcircuits after cutting

(b)



1.) Send cut-finding program to the cloud for remote execution using Quantum Serverless