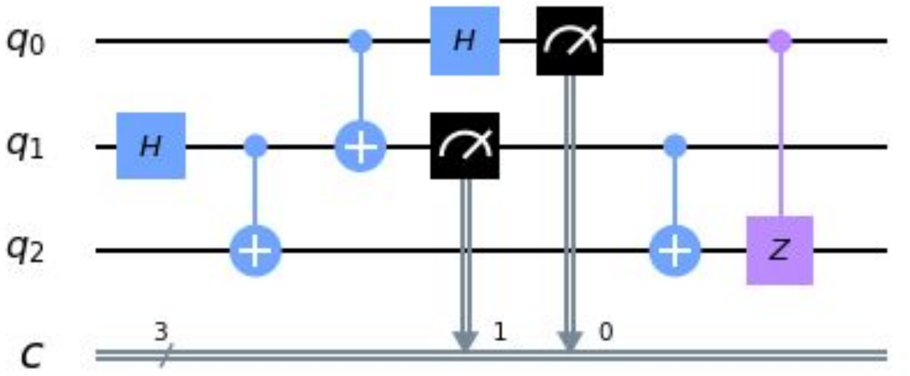
QTensor: Fast QAOA Tensor Network Quantum Simulator

Danylo Lykov

Argonne National Laboratory

Quantum Computing workshop,
June 17, 2021

What is a Quantum Circuit Simulator?

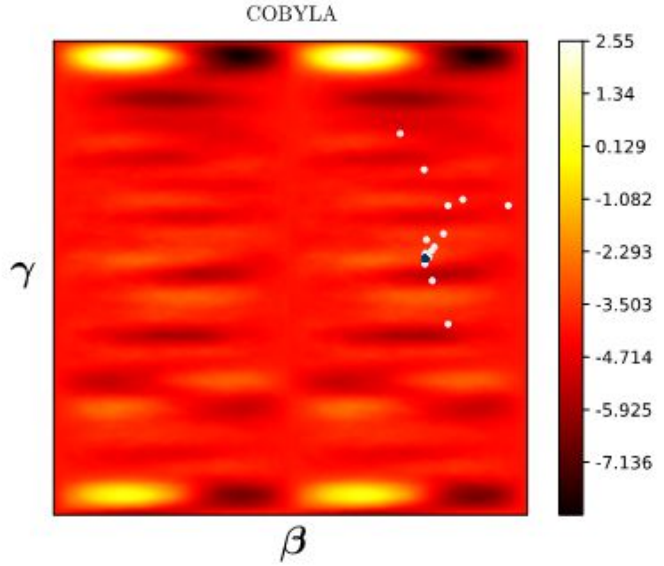


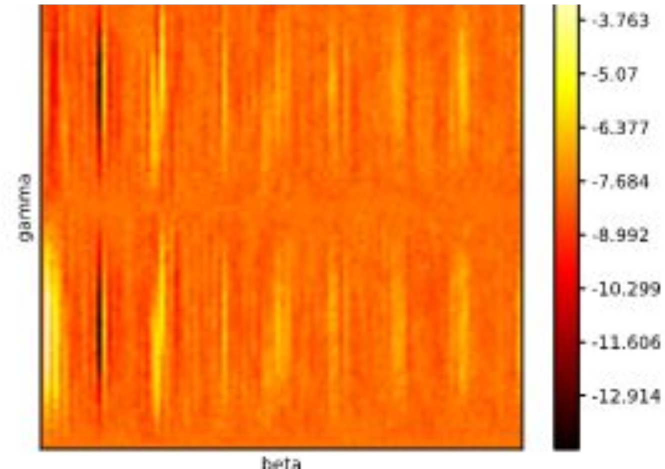
It is an universal quantum computer simulator which simulates the execution of quantum circuits with or without quantum noise

The input is a quantum circuit, which is a collection of gates applied to quibts.

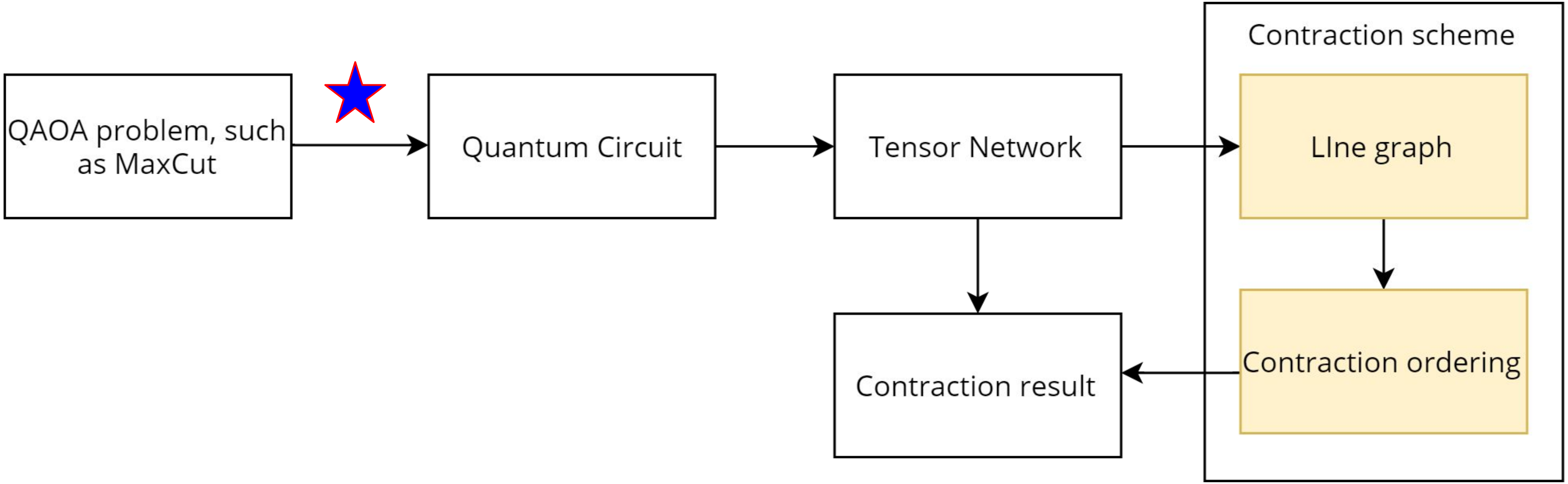
Quantum Simulator Use Cases

- Verification of quantum advantage and supremacy claims
- Verification of large quantum devices
- Co-design quantum computers
- Energy efficiency studies of quantum computers
- Design of new quantum algorithms
- Finding parameters for variational quantum algorithms





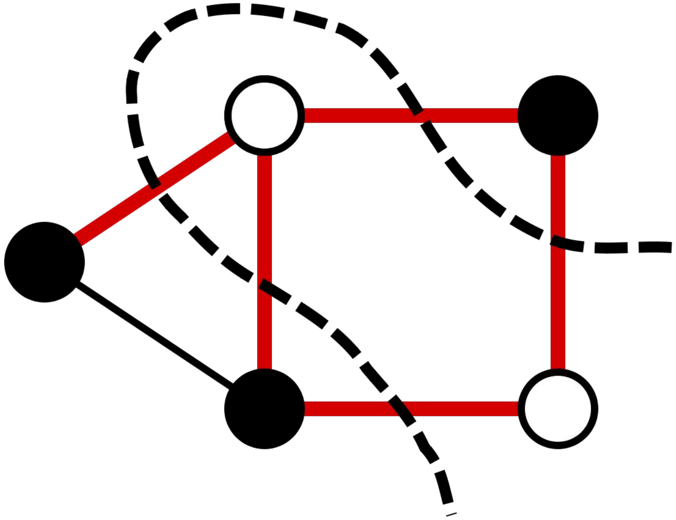
Simulation workflow



Max Cut

Assign +1 and -1 to vertices while minimizing the cost function

$$H_{\text{MaxCut}} = \frac{1}{2} \sum_{(u,v)\in E} (I - Z_u Z_v).$$



QAOA

$$|\gamma\beta\rangle = [e^{\imath\gamma_q \sum_{ij\in E} Z_i Z_j + \imath\beta_q \sum_{i\in V} X_i}]_{q=1..p}|+\rangle$$
$$= [\prod_{ij\in E,k\in V} e^{\imath\gamma_q Z_i Z_j} e^{\imath\beta_q X_k}]_{q=1..p}|+\rangle$$

1. Find variational parameters
2. Sample solution from the parameter-dependent quantum state

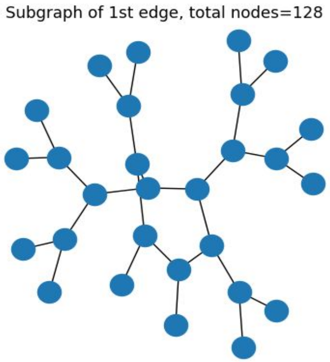
$$E = \mathbf{max}_{\gamma,\beta}\langle\gamma\beta|\hat{C}|\gamma\beta\rangle$$

Lightcones

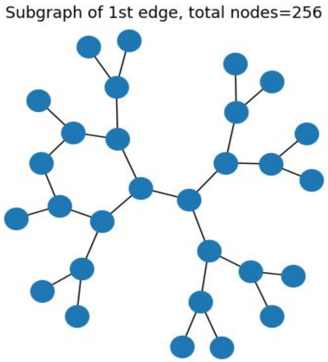
$$\langle\gamma\beta|\sum_{i,j\in E} Z_i Z_j|\gamma\beta\rangle = \sum_{i,j\in E}\langle\gamma\beta|Z_i Z_j|\gamma\beta\rangle$$

Most of gates in the parametric state cancel out!

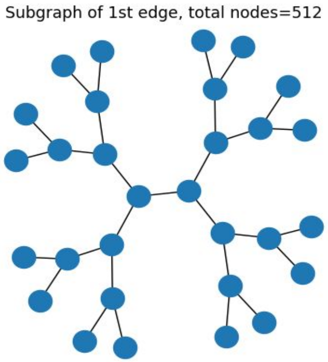
For graph with bound degree we get linear dependence of cost to calculate Energy - need to only consider **subgraphs**:



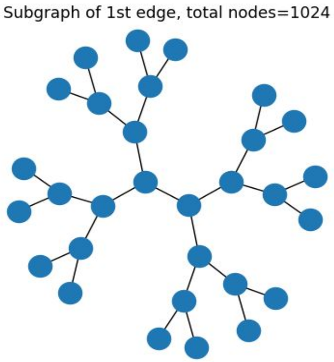
128



256



512



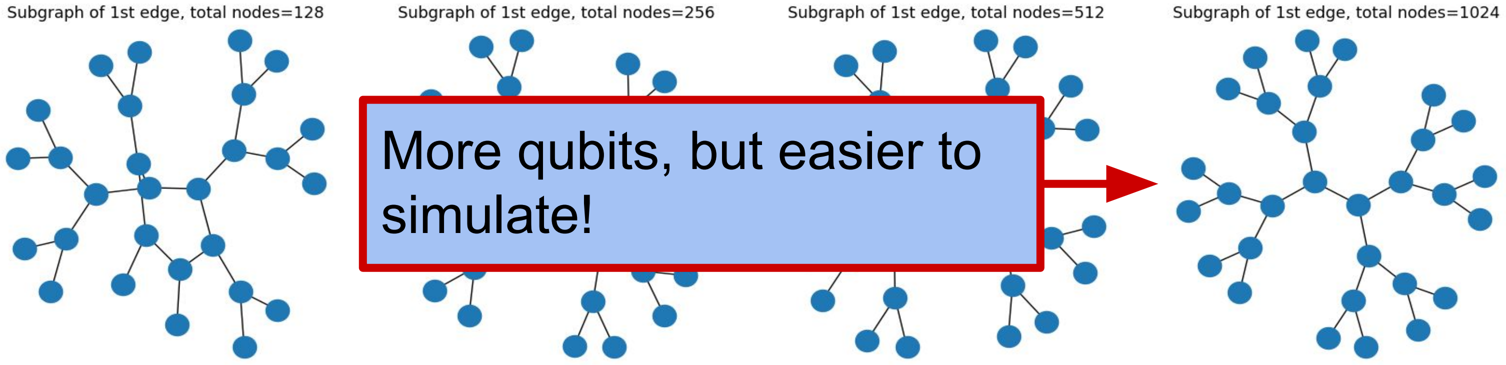
1024

Lightcones

$$\langle\gamma\beta|\sum_{i,j\in E} Z_i Z_j|\gamma\beta\rangle = \sum_{i,j\in E}\langle\gamma\beta|Z_i Z_j|\gamma\beta\rangle$$

Most of gates in the parametric state cancel out!

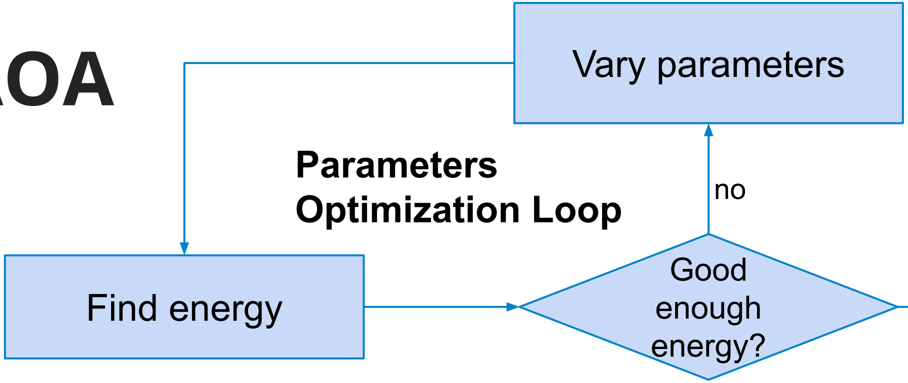
For graph with bound degree we get linear dependence of cost to calculate Energy - need to only consider **subgraphs**:



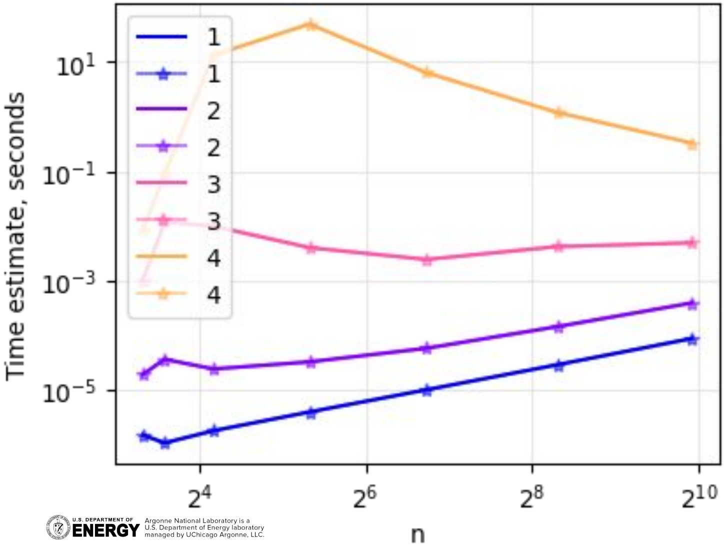
128 256 512 1024

QAOA

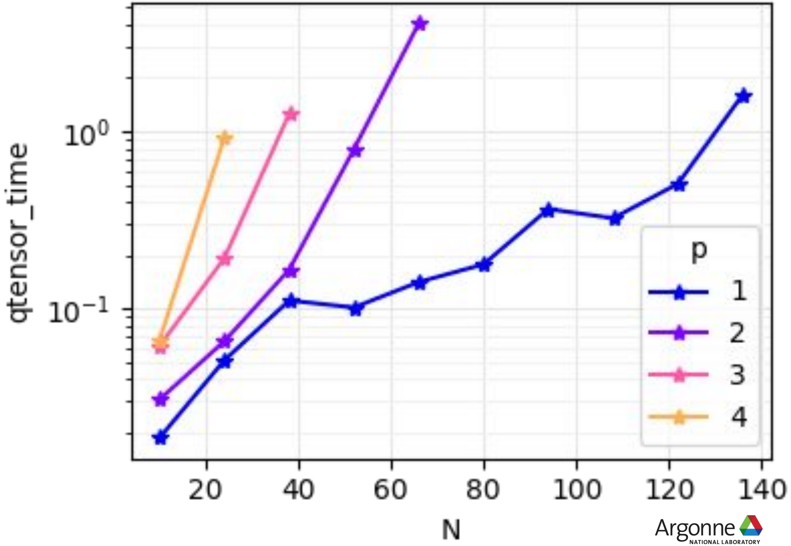
Co-design of Quantum Computers



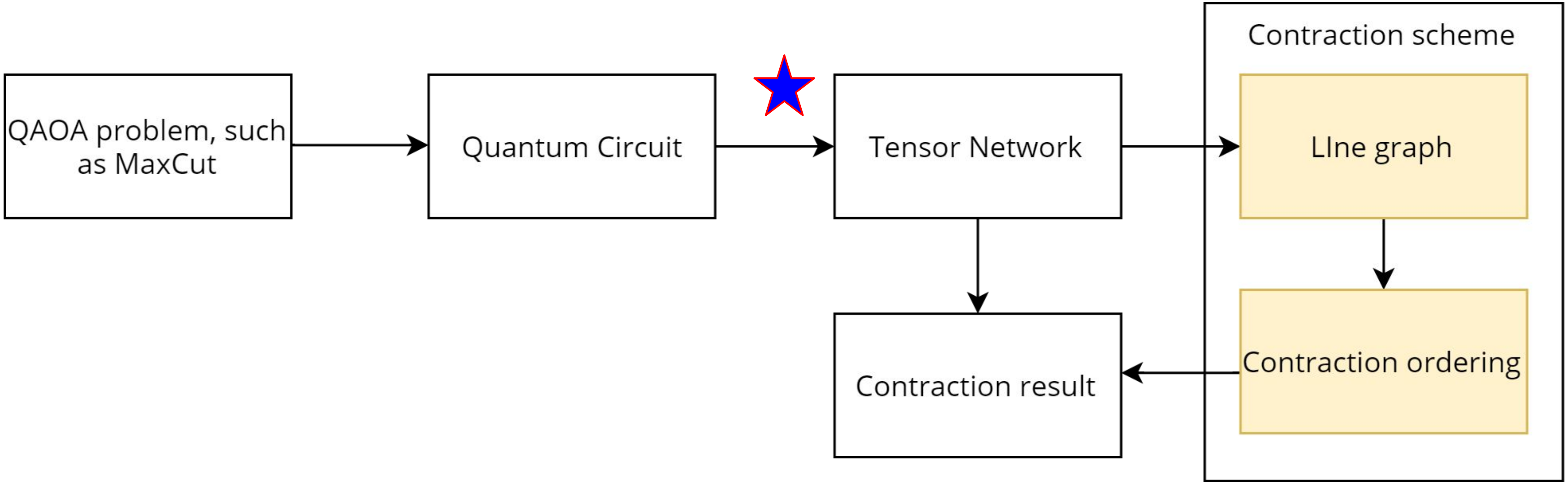
- Easy on classical



- Easy on quantum



Simulation workflow



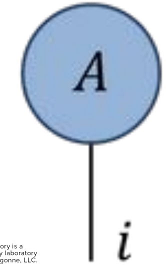
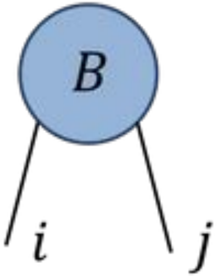
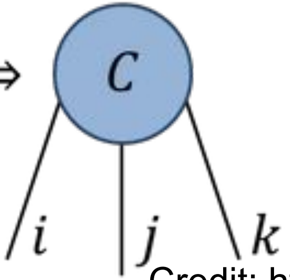
Tensor network introduction

- Tensors are just generalization of matrix representation
- The different order of tensors:

Diagrammatic notation

Tensors are represented as an object with a number of 'legs' that corresponds to the rank of the tensor:

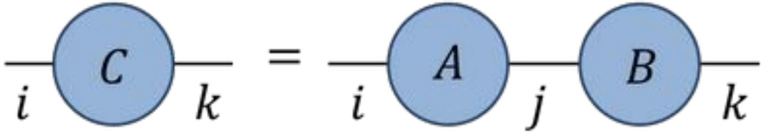
$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_m \end{bmatrix} \qquad B = \begin{bmatrix} B_{11} & \cdots & B_{1n} \\ \vdots & \ddots & \vdots \\ B_{m1} & \cdots & B_{mn} \end{bmatrix} \qquad C = \left[\left[\begin{bmatrix} C_{111} & \cdots & C_{1n1} \\ \vdots & \ddots & \vdots \\ C_{m11} & \cdots & C_{mn1} \end{bmatrix}_1 \cdots\right]_2 \cdots\right]_3$$

vector	matrix	3rd order tensor
$A_i \Leftrightarrow$ 	$B_{ij} \Leftrightarrow$ 	$C_{ijk} \Leftrightarrow$ 

Credit: https://www.tensors.net

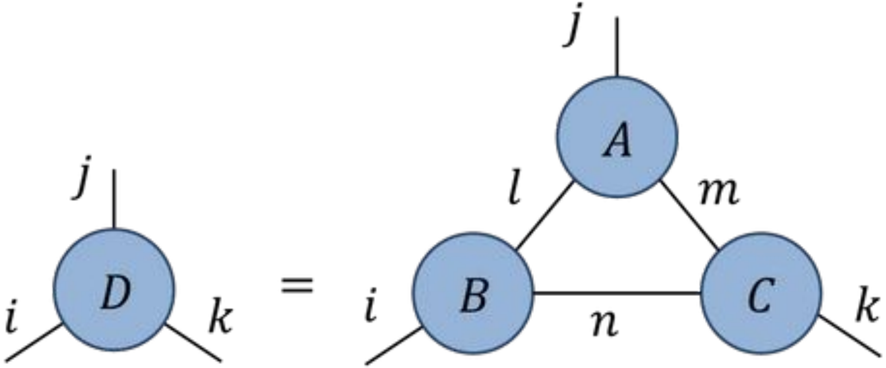
Tensor operations

Diagrammatic tensor notation is especially useful for describing networks comprised of multiple tensors:



$$\Updownarrow$$

$$C_{ik} = \sum_j A_{ij} B_{jk}$$



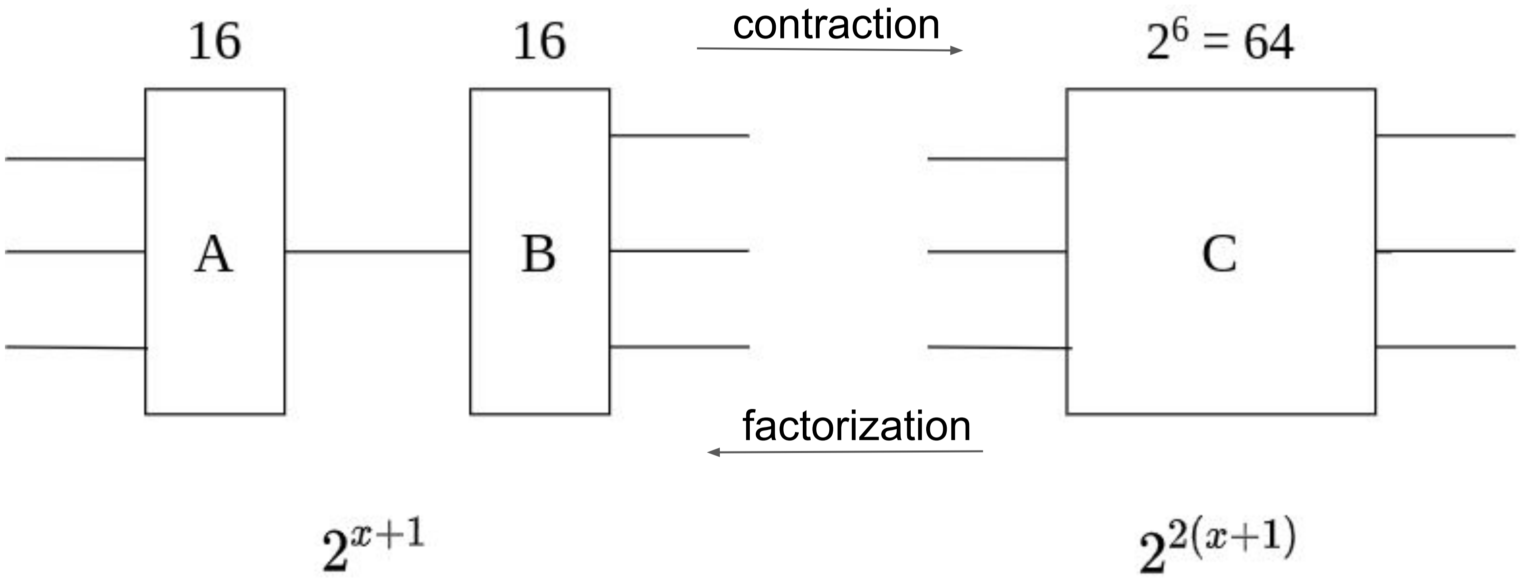
$$\Updownarrow$$

$$D_{ijk} = \sum_{lmn} A_{ljm} B_{iln} C_{nmk}$$

Credit: https://www.tensors.net

Contraction increases tensor size

For tensors with dimension sizes = 2

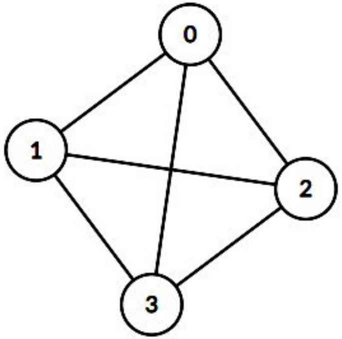


x = tensor rank

Gates as tensors

Operator	Gate(s)		Matrix
Pauli-X (X)	$\boxed{X}$	$\oplus$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y (Y)	$\boxed{Y}$		$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z (Z)	$\boxed{Z}$		$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Hadamard (H)	$\boxed{H}$		$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Phase (S, P)	$\boxed{S}$		$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
$\pi/8$ (T)	$\boxed{T}$		$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$

QAOA circuit



Fully connected graph with 4 vertices and 6 edges. The corresponding circuit to solve MaxCut problem is below

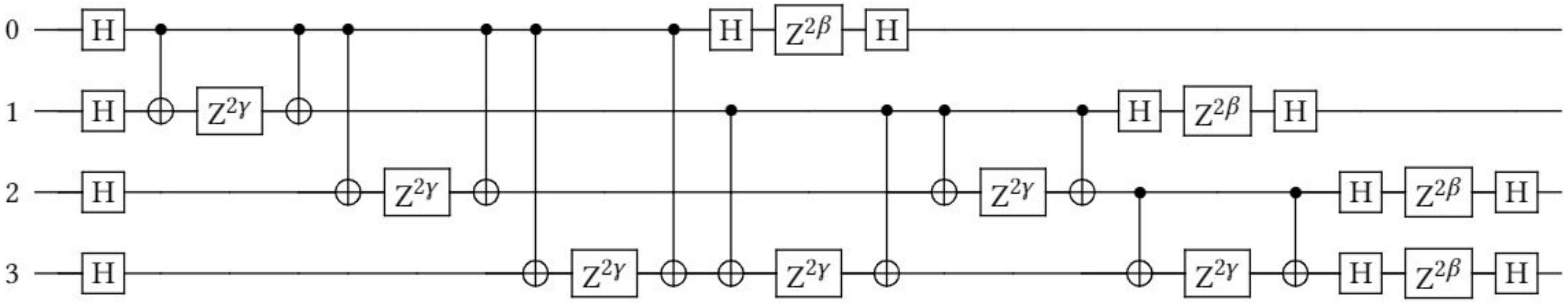
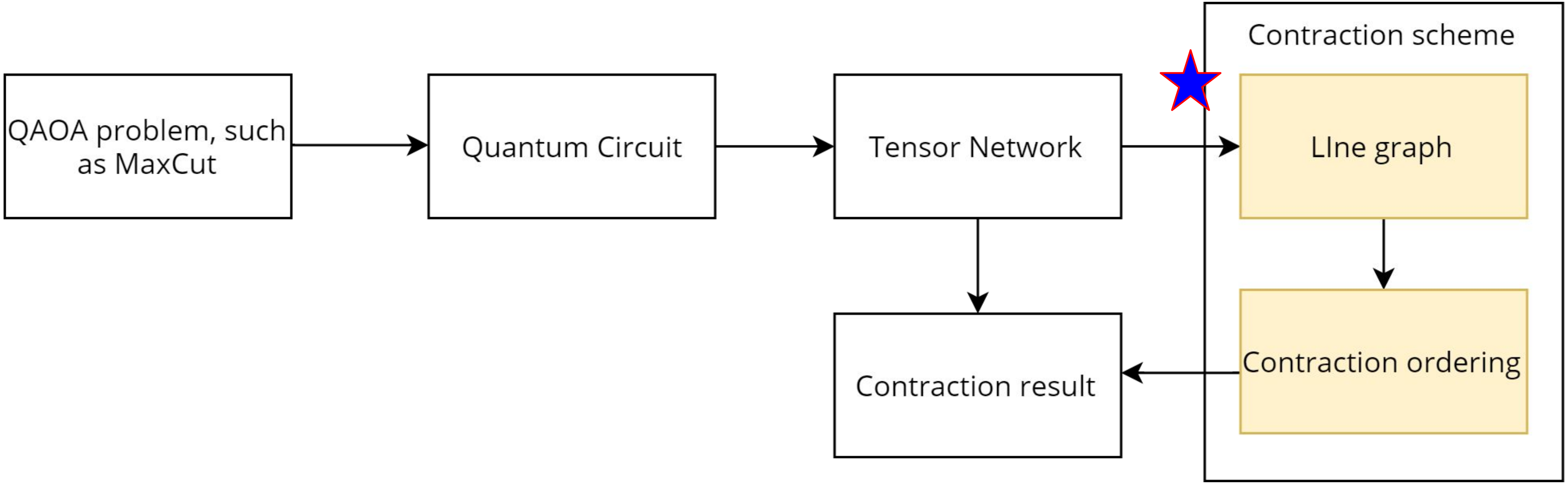


Figure 1: p=1 depth QAOA circuit for a fully connected graph with 4 nodes.

Simulation workflow



Line graph

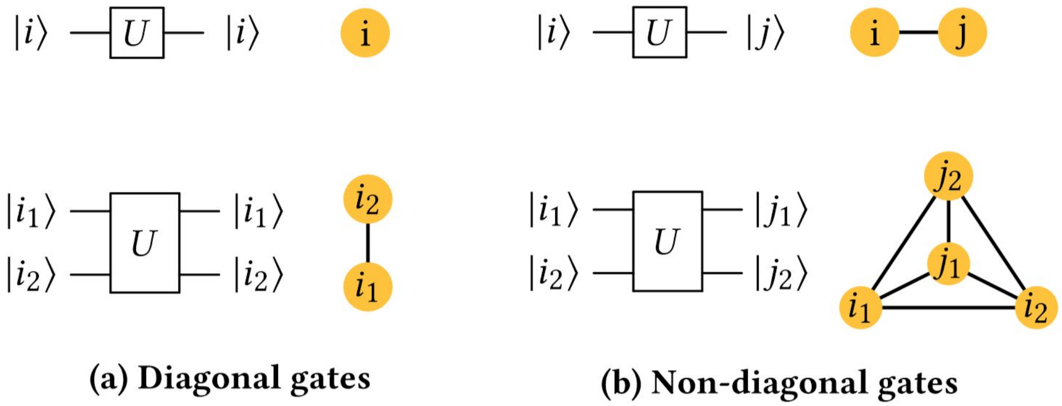
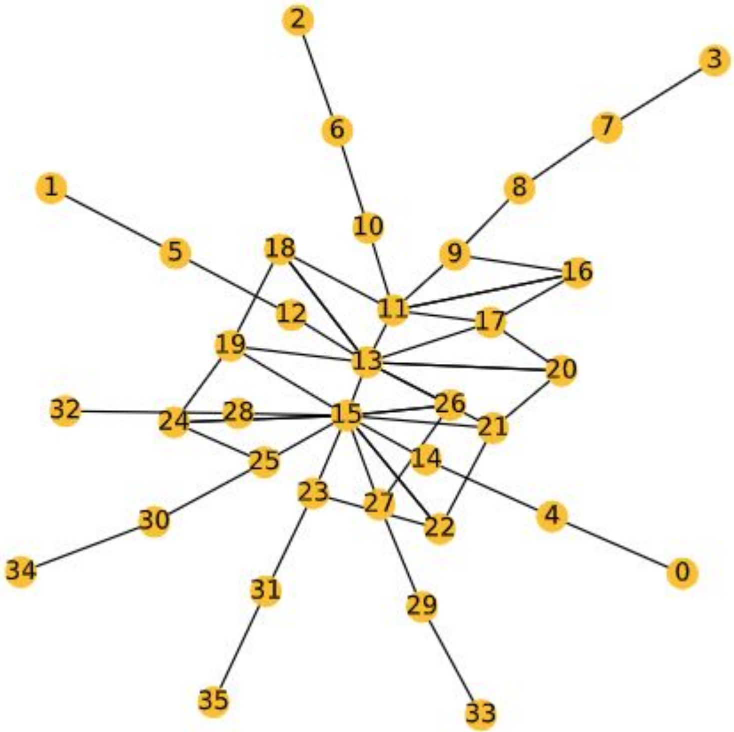


Figure 2: Correspondence of quantum gates and graphical representation.

The only 2-qubit gate in the circuit is diagonal!

$$e^{\alpha Z_i Z_j} = \mathrm{diag}(e^{-\alpha/2}, e^{\alpha/2}, e^{\alpha/2}, e^{-\alpha/2})$$

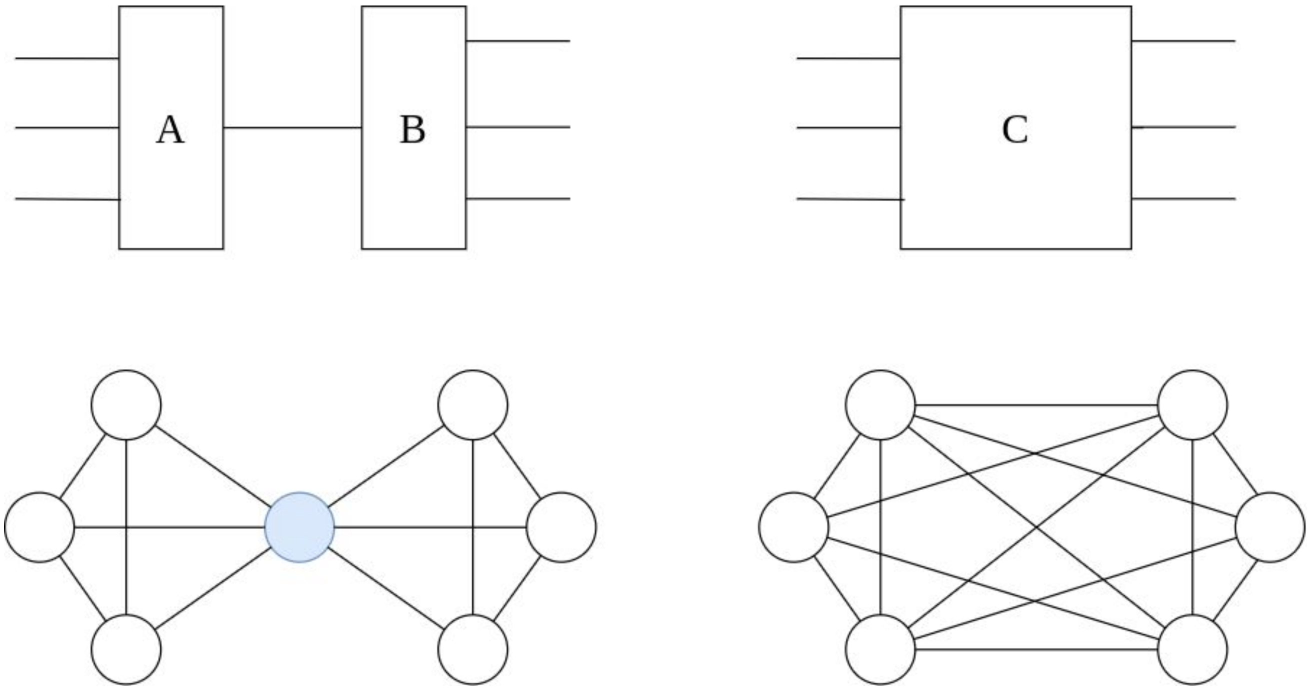
QAOA Tensor Network



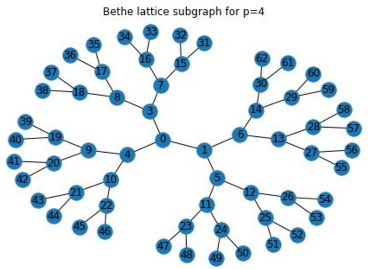
Graph representation of tensor expression of the circuit from previous slide. Every vertex corresponds to a tensor index of a quantum gate

The simulator contracts tensors in the optimal order

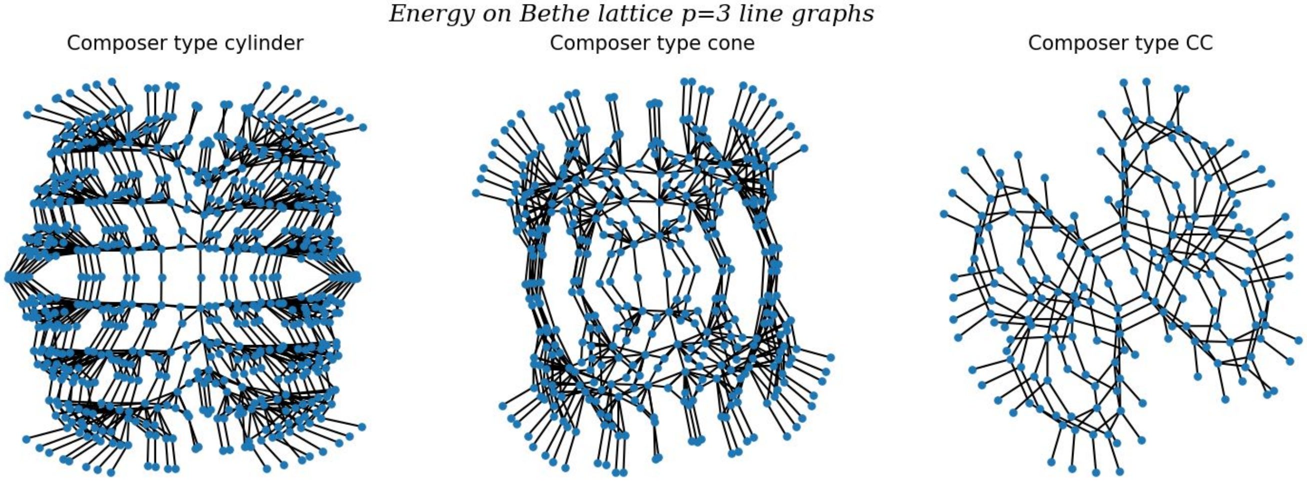
Line graph: contract two tensors

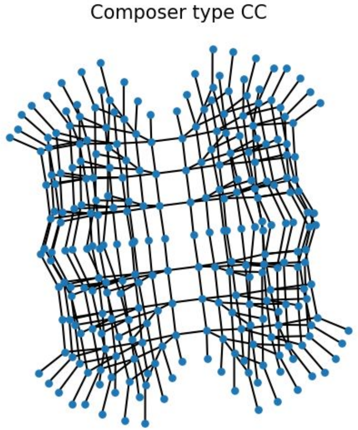


Bethe lattice, line graphs



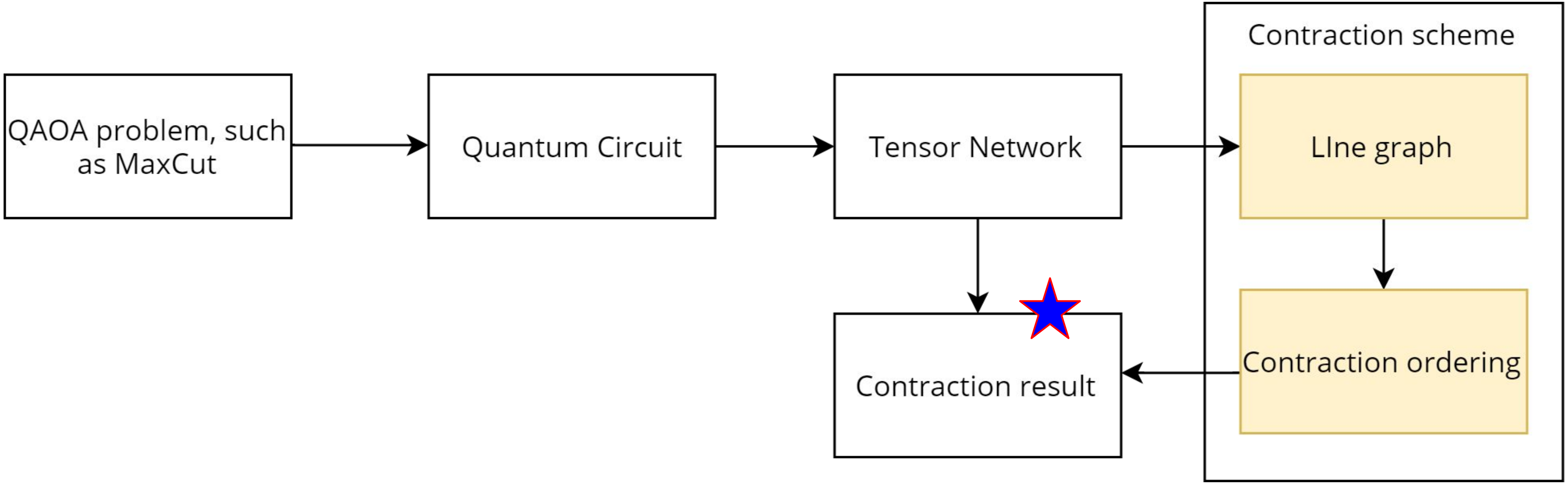
‘cylinder’





$tw > 2p$

Simulation workflow



QTensor: Energy Calculations

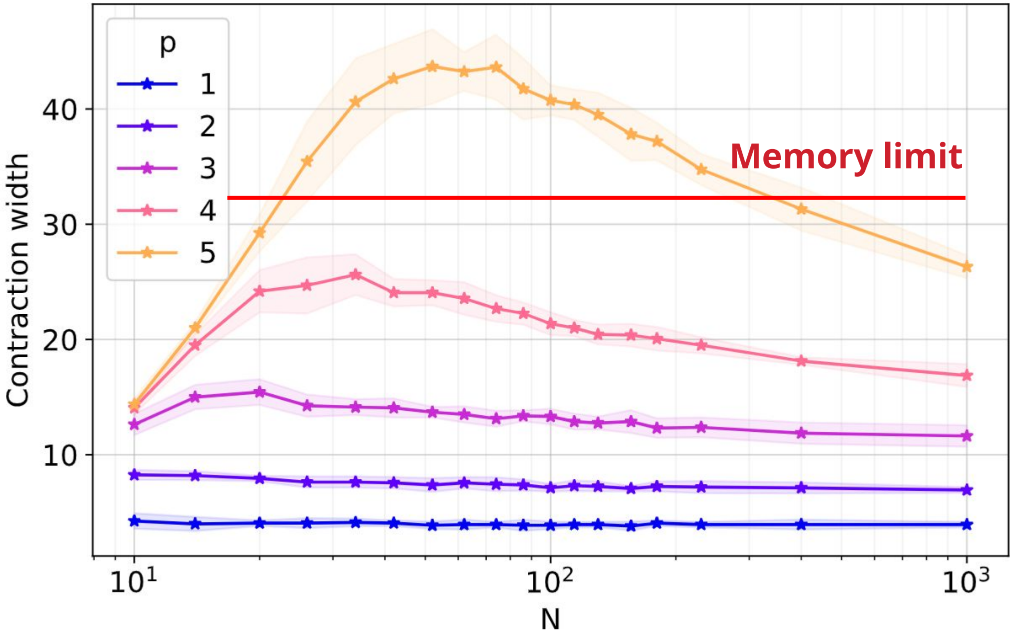
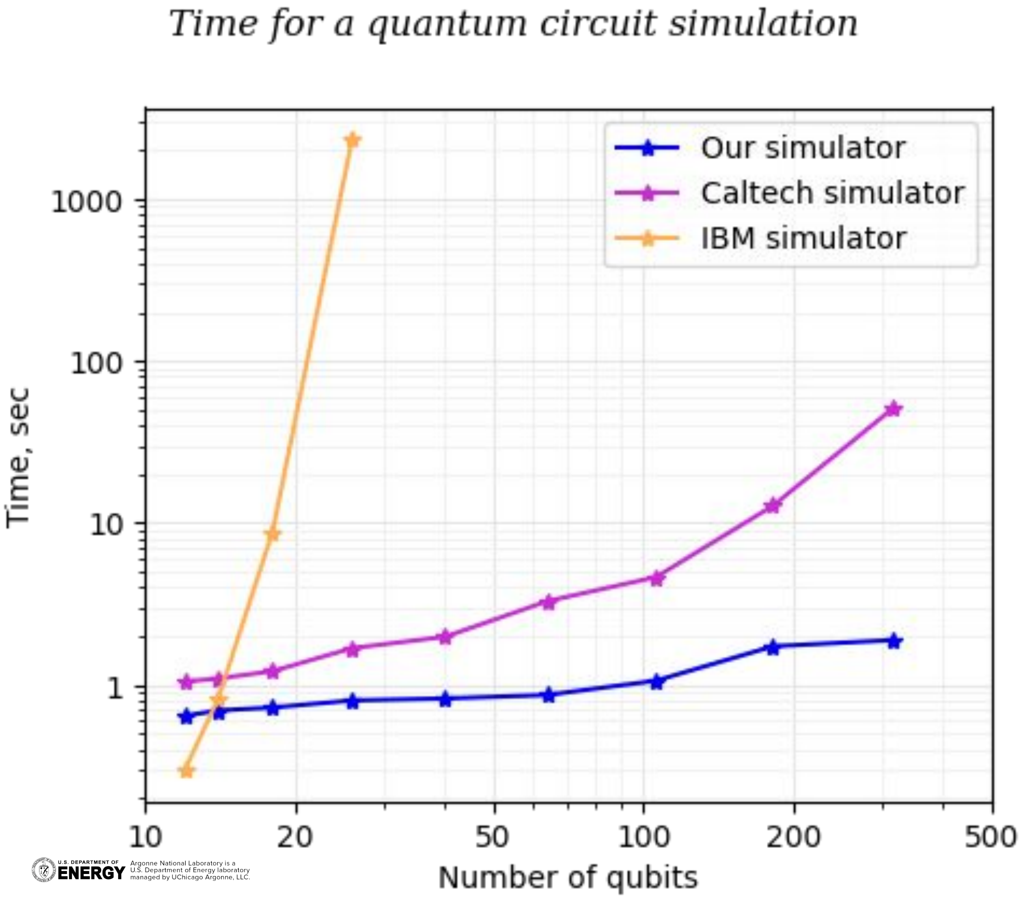


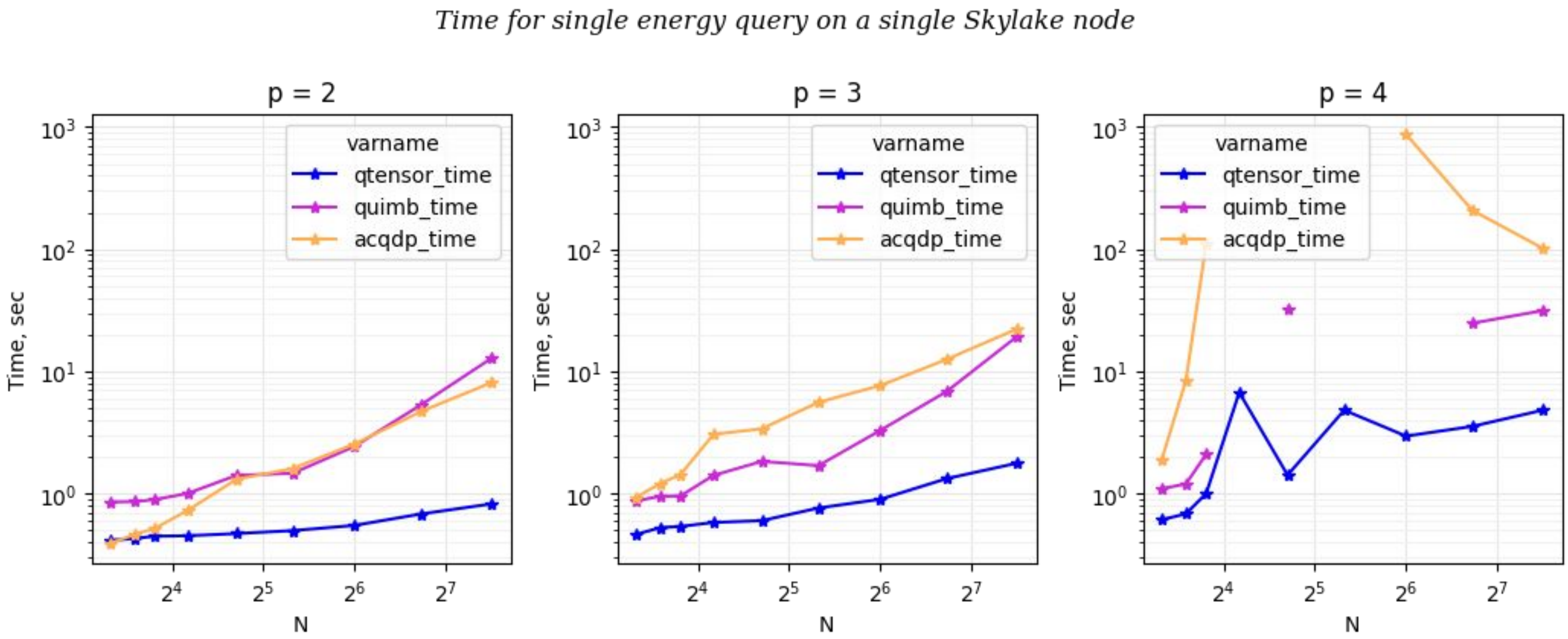
Figure 2: Contraction width for energy calculations for Max-Cut on graphs of different size.

QTensor: Energy Calculations



The problem to solve is MaxCut with QAOA for p=3 and d=3 on 56 Intel Xeon CPUs

QTensor: Energy Calculations



* optimization + contraction time

QTensor: Energy Calculations

	$d = 3$	$d = 4$	$d = 5$
$p = 1$	1.04	1.65	2.16
$p = 2$	1.46	2.3	4.36
$p = 3$	2.42	10.2	$45.1^{\dagger}$
$p = 4$	6.83		
$p = 5$	58.0^{*}		

Table 1: QAOA Energy simulation time in seconds for 1000 node regular graphs. All calculations were done using QTensor simulator using NumPy backend on a single Intel Xeon Platinum 8180M CPU @ 2.50GHz with 56 physical cores.

Parallel Simulations

We calculated the QAOA expectation value for a 1,000,000 qubit circuit with depth p=6 in 1 hour and 20 minutes. The simulations were performed on the Theta supercomputer with 512 nodes.

QTensor Features

- Possibility to use Qiskit circuit as input
- Efficient simulation of probability amplitudes
- Simulation of batches of amplitudes for the same cost
- Efficient simulation of expectation values
- Parallelization support
- Automatic differentiation with respect to gate parameters

Quantum Simulator Team

Yuri Alexeev

Project Supervisor

ANL Principal Project Specialist

Alexey Galda

ANL Visiting Scientist

UChicago Research Assistant Professor

Cameron Ibrahim

ANL Consultant

PhD Student at University of Delaware

Publications

1. Submitted to ACM Transactions for Quantum Computing
   https://arxiv.org/pdf/2012.02430.pdf

   Tensor Network Quantum Simulator With Step-Dependent Parallelization

   DANYLO LYKOV, Argonne National Laboratory, USA
   ROMAN SCHUTSKI, Rice University, USA
   ALEXEY GALDA, University of Chicago Argonne National Laboratory, USA
   VALERII VINOKUR, Argonne National Laboratory, USA
   YURI ALEXEEV, Argonne National Laboratory, USA

2. In preparation the paper “QTensor: the fastest QAOA energy simulator” for NPJ Quantum Information

3. In preparation the paper for the 2nd International Workshop on Quantum Computing: Circuits Systems Automation and Applications (QC-CSAA)

Acknowledgements

https://github.com/danlkv/QTensor

Contact information:

- Danyl Lykov dlykov@anl.gov
- Yuri Alexeev yuri@anl.gov
- Alexey Galda agalda@anl.gov

Funding:

DOE ECP: *This research was partially supported by the Exascale Computing Project (17-SC-20-SC), a joint project of the U.S. Department of Energy's Office of Science and National Nuclear Security Administration, responsible for delivering a capable exascale ecosystem, including software, applications, and hardware technology, to support the nation's exascale computing imperative*

DOD DARPA: *This research was partially supported by the the Defense Advanced Research Projects Agency (DARPA) project*

Optimization time vs simulation time?