Intermediate-Scale Full State Quantum Circuit Simulation by Using Lossy Data Compression

Xin-Chuan Wu¹, Sheng Di², Franck Cappello², Hal Finkel², Yuri Alexeev², and Frederic T. Chong¹

EPIQC@University of Chicago¹
Argonne National Laboratory²
Why Quantum Circuit Simulation?

Physically realizing a quantum information processor is difficult as quantum systems are extremely sensitive to environmental effects.

Simulation of quantum circuits is vital for the study of quantum computing.
- Validate the quantum circuit and quantify the circuit fidelity on real quantum machines.
- Assess correctness and performance of new quantum algorithms.

Simulation: calculate quantum state amplitudes.
- Using classical computing systems to simulate the behavior of quantum computers.

Advantages of full state vector update simulation
- General circuits
- Good at quantum software development and debugging
Challenges of Quantum Circuit Simulation

For n-qubit systems, the number of amplitudes is $2^n$.
- Each amplitude is a double-precision complex number: 16 Bytes
- For n-qubit system simulation, the state vector takes $2^{n+4}$ Bytes.
- 50-qubit systems: The memory requirement of the state vector is $2^{54}$ Bytes (16PB).
- People believe it is impossible to classically simulate a 50-qubit quantum computer.

<table>
<thead>
<tr>
<th>System</th>
<th>Memory (PB)</th>
<th>Max qubits</th>
</tr>
</thead>
<tbody>
<tr>
<td>TACC Stampede</td>
<td>0.192</td>
<td>43</td>
</tr>
<tr>
<td>Titan</td>
<td>0.71</td>
<td>45</td>
</tr>
<tr>
<td>Theta</td>
<td>0.8</td>
<td>45</td>
</tr>
<tr>
<td>Sunway TaihuLight</td>
<td>1.3</td>
<td>46</td>
</tr>
<tr>
<td>K computer</td>
<td>1.4</td>
<td>46</td>
</tr>
<tr>
<td>Summit</td>
<td>2.8</td>
<td>47</td>
</tr>
</tbody>
</table>
Data Compression on State Vector

- Reduce memory requirement
- +1 qubit in simulation
- +2 qubits in simulation
- +m qubits in simulation ( Depending on the compression ratio)
Error-Bounded Lossy Compressor: ANL SZ

Applying lossy compression to the state vector.

SZ is an error-bounded lossy data compressor allowing user-controlled loss of accuracy during the compression with significantly reduced data size.

- SZ allows user to set the error bound, denoted $\delta$.
- The decompressed data $D_i'$ must be in the range $[D_i - \delta, D_i + \delta]$, where $D_i'$ is referred as the decompressed value and $D_i$ is the original data value.
- SZ can compress 1-D dataset efficiently.

$\delta \uparrow$: Simulation accuracy $\downarrow$, compression ratio $\uparrow$
State Vector Update Simulation

Intel-QS

- Full state vector update simulator (with MPI)
- $|\psi_{t+1}\rangle = A_t |\psi_t\rangle$

$$A = I \otimes I \otimes ... \otimes U \otimes ... \otimes I \otimes I$$

$$U = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}$$

- We do not need to construct the entire $A$.

- For example, applying a single-qubit gate to the first qubit
  - Writing the amplitude indices in binary format.
  - Applying $U$ to every pair of amplitudes, whose indices have 0 and 1 in the first bit, while all other bits remain the same.

- In $n$-qubit systems, apply $U$ to the $k$-th qubit: $a^* \times 0$ for $k$-th $\times$, and $a^* \times 1$ for $k$-th $\times$.
  - $2^{n-1}$ pairs $\rightarrow 2^{n-1}$ matrix multiplications

---

Getting the Pair of Amplitudes \((a_{*0}^{k-th}, a_{*1}^{k-th})\)

**Message Passing Interface** (total R ranks)

- **RANK 0**: State Vector [0]
- **RANK 1**: State Vector [1]
- **RANK 2**: State Vector [2]
- **RANK r-1**: State Vector [R-1]

**Data Compression** (Block size = B)

- Compressed Block [0]
- Compressed Block [1]
- Compressed Block [2]
- Compressed Block [3]
- ... (for R-1 ranks)
- Compressed Block [n_b-1]

**Memory**

- Amplitude index
  - 000...000
  - 000...001
  - 000...010
  - 000...011
  - ...
  - 111...100
  - 111...101
  - 111...110
  - 111...111

**Amplitude index**

- n-1
- n - \(\log_2 R\)
- \(\log_2 B\)
- 0

**Explanation**

- Both amplitudes are in the same block.
- Both amplitudes are in the same rank, different blocks.
- The pair of amplitudes are in the different ranks.
Two-Qubit Gate

In a control-U gate, control qubit position: C-th qubit
If the C-th qubit is 1, apply U to k-th qubit; otherwise left unmodified.
Compression Ratio (QFT)

Set a compression ratio threshold $\theta$. Relax the error bound to meet the threshold.

QFT results:
Conclusion

We propose a lossy data compression strategy that could be used for quantum circuit simulation.

Our approach compresses the state vector to reduce the memory requirement, so we can simulate a larger quantum system with the same memory capacity.

- Trade computation time for memory capacity.
- Trade fidelity for compression ratio.

For 50-qubit systems, our preliminary results suggest that we are able to reduce the memory requirement from 16PB to 1PB.
Thank You!

Acknowledgment:

This research used resources of the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357. This research was supported by the Exascale Computing Project (ECP), Project Number: 17-SC-20-SC, a collaborative effort of two DOE organizations the Office of Science and the National Nuclear Security Administration, responsible for the planning and preparation of a capable exascale ecosystem, including software, applications, hardware, advanced system engineering and early testbed platforms, to support the nations exascale computing imperative. The material was supported by the U.S. Department of Energy, Office of Science, and supported by the National Science Foundation under Grant No. 1619253. This work is funded in part by EPIQC, an NSF Expedition in Computing, under grant CCF-1730449. This work is also funded in part by NSF PHY-1818914 and a research gift from Intel.

https://www.epiqc.cs.uchicago.edu/
Quantum State Vector Update Simulation

n-qubit system

\[ |\Psi\rangle = a_0 |000000\ldots000000\rangle + a_1 |000000\ldots000001\rangle + \ldots + a_{2^n-1} |111111\ldots111111\rangle \]

Simulation: \[ |\Psi_{t+1}\rangle = A_t |\Psi_t\rangle \], for \( t = 0, \ldots, d \) at each layer

- A is a unitary matrix
- d is the depth of the circuit