Contracting Tensor Network on a Noisy Quantum Computer

Isaac H. Kim

Stanford Institute for Theoretical Physics

September 13th, 2018

arXiv:1703.02093, arXiv:1703.00032, arXiv:1711.07500(w. Brian Swingle(UMD)) + some unpublished tidbits

Before we start...

- I gave a different version of this talk at KITP.
  - You can google "kitp noise resilient" to find it online.
- The emphasis is different.
  - KITP talk: More on on noise-resilience
  - This talk: Broader overview

Bounty

The Quantum Advantage Prize Quantum Cloud Services has been designed from the bottom-up to accelerate the pursuit of quantum advantage. We don't know when the first demonstration of quantum advantage will be achieved, or what shape it will take, but one thing is certain: it will dramatically accelerate progress in unlocking the power of quantum computing for everyone. Recognizing the significance of this achievement, Rigetti Computing is offering a $1 million prize for the first conclusive demonstration of quantum advantage on QCS. More details of the prize will be announced on October 30th, 2018. Stay tuned!

Quantum Computing | Cloud Computing | Cloud Services

Noise

1. The theory of fault tolerance tells us that a quantum computer consisting of noisy components can simulate a noiseless quantum computer with a moderate overhead.
2. But this is possible only if the error rate is sufficiently low, i.e., lower than the **threshold value** p_{th}.
3. The leading approach has a threshold of $\sim 0.7\%$.
4. Noise rate below 0.7% can be realized in superconducting qubits/ion traps at small scales.
5. However, whether these systems can be scaled to $O(10^6)$ qubits while maintaining this noise rate is not clear at all.

Noise

1. The theory of fault tolerance tells us that a quantum computer consisting of noisy components can simulate a noiseless quantum computer with a moderate overhead.
2. But this is possible only if the error rate is sufficiently low, i.e., lower than the **threshold value** p_{th}.
3. The leading approach has a threshold of $\sim 0.7\%$.
4. Noise rate below 0.7% can be realized in superconducting qubits/ion traps at small scales.
5. However, whether these systems can be scaled to $O(10^6)$ qubits while maintaining this noise rate is not clear at all.

Lesson

Experimentalists are working hard! Theorists should help them out by finding useful applications of noisy quantum computer.

Goal

Solve a problem that we wanted to solve but couldn't solve before, with a noisy quantum computer.

Goal

Solve a problem that we wanted to solve but couldn't solve before, with a noisy quantum computer.

1. What problem?
2. Why couldn't we solve it before?
3. Why does quantum computer help?
4. Can we deal with noise?

Goal

Solve a problem that we wanted to solve but couldn't solve before, with a noisy quantum computer.

1. What problem? : Solve low-energy phase diagram of strongly interacting quantum many-body system.
2. Why couldn't we solve it before? : Not enough memory/speed on a classical computer.
3. Why does quantum computer help? : Because it removes the memory/speed bottleneck of an existing computational method.
4. Can we deal with noise? : Yes, without error correction.

Goal

Solve a problem that we wanted to solve but couldn't solve before, with a noisy quantum computer.

1. What problem? : Solve low-energy phase diagram of strongly interacting quantum many-body system.
2. Why couldn't we solve it before? : Not enough memory/speed on a classical computer.
3. Why does quantum computer help? : Because it removes the memory/speed bottleneck of an existing computational method.
4. Can we deal with noise? : Yes, without error correction.

Be careful!

It's easy to misinterpret the third point.

- I am not saying that quantum computer in itself will solve everything.

Checklist

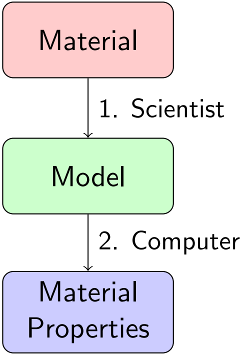
I do think this is a useful checklist for assessing a quantum algorithm, both as an algorithm inventor and as a customer.

1. What problem?
2. Why is it classically hard?
3. How can a quantum computer help us?
4. For near term: Can we deal with noise?

Agenda

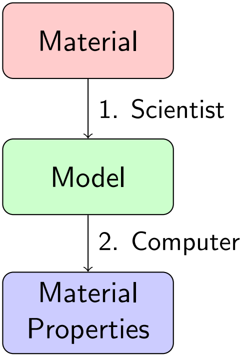
- **What problem?**
- **Why is it classically hard?**
- How can a quantum computer help us?
- Can we deal with noise?

What problem?

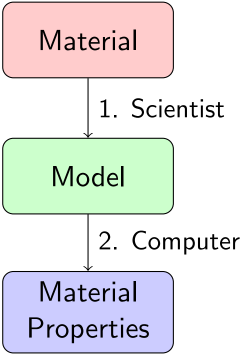


The second part can be computationally demanding.

Why is it classically hard?

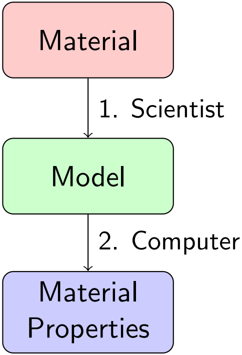


Why is it classically hard?



To even store the wavefunction of 100 electrons, one needs $\sim 10^{30}$ bytes.
Note: Petabyte $= 10^{15}$bytes.

Why is it classically hard?



We could solve special cases, free electron system, small correlation, etc. However, we do not have a general-purpose computational tool for **strongly correlated system.**

Why is it classically hard?

Different computational methods suffer from different problems.

1. Exact diagonalization: Memory problem
2. Quantum Monte Carlo: Sign problem
3. Variational methods aka Tensor network methods : Memory/time scales polynoially with the number of parameters, but the scaling is bad.
    - For example, $O(n^{16})$ time algoithm is not very practical!
    - Typical corridor conversation with my tensor network friends:
        - Me: “How’s it going?”
        - Friend: “Dude. My matrix doesn’t fit into my 128GB RAM.”
        - * Conversation stops. *
    - I should say that Steve White’s DMRG method works extremely well, but only in 1D and in some limited 2D settings.

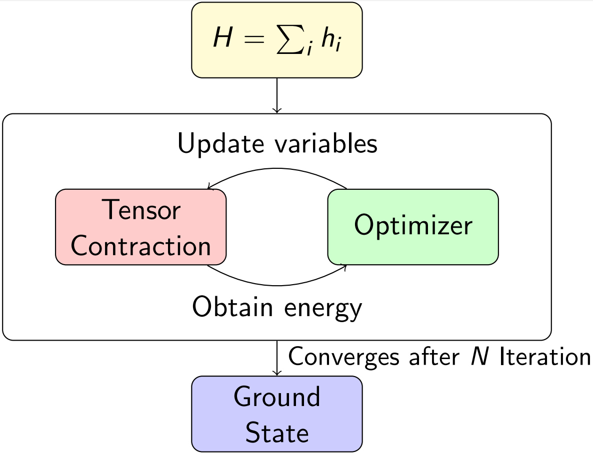
Why is it classically hard?

At the end of the day, what matters is whether we can solve a given problem. *How* we solve that problem is irrelevant, as long as the tool we use is legitimate. So we should ask, why use a quantum computer when we have awesome classical computers that are cheap and reliable?

- I already claimed that the memory/time scaling for tensor network method tends to be bad.
- If you dig deeper, you will quickly find out that the bottleneck of the computation consists of elementary linear algebra operations on large matrices.
- Improvement in the tensor network algorithms seem to require a speedup in elementary linear algebra operations. I don't expect any drastic improvements in the near future. Also, the memory problem will never go away.

A good figure of merit: Contraction time

Contraction time = Time to compute expectation value of local observable



Optimization Time = $N *$ Contraction Time

Who is the culprit?

What is the main bottleneck in the contraction algorithm?

Efficient vs. Practical

- n : # of lattice sites
- χ : # of variational parameters/site

Classical Methods

Method	n	χ
MPS	$O(n)$	$O(\chi^3)$
PEPS(2D)	$O(n^9)$? $O(e^n)$?	?
MERA(1D)	$O(\log n)$	$O(\chi^7)$
MERA(2D)	$O(\log n)$	$O(\chi^{16})$
fc-PEPS[a] (2D)	$O(n^9)$? $O(e^{\sqrt{n}})$?	?
DMERA($d-$dim)[b]	$O(\log n)$	$O(e^{\chi})$

[a] K (2017)
[b] K, B. Swingle (2017)

Agenda

- What problem?
- Why is it classically hard?
- **How can a quantum computer help us?**
- Can we deal with noise?

Efficient vs. Practical

- n : # of lattice sites
- χ : # of variational parameters/site

Quantum Methods

Method	n	χ
MPS	$O(n)$	$O(\chi^3)$
PEPS(2D)	$O(n^9)$? $O(e^n)$?	?
MERA(1D)	$O(\log n)$	$O(\chi^7)$
MERA(2D)	$O(\log n)$	$O(\chi^{16})$
fc-PEPS[a] (2D)	$O(n^9)$? $O(e^{\sqrt{n}})$? $\rightarrow O(\sqrt{n})$	? $\rightarrow O(\chi)$
DMERA(d−dim)[b]	$O(\log n)$	$O(e^\chi) \rightarrow O(\chi^{1/d}))$

[a] K (2017)

[b] K, B. Swingle (2017)

Requirement

- We consider a $n \times n$ 2D system.
- τ_2 : Maximum time to execute two-qubit gate.
- ϵ : Desired accuracy for energy/site.

Method	# of Qubits	Contraction Time	Gate Type
fc-PEPS	χn	$\frac{\tau_2 n \chi}{\epsilon^2}$	NN Local Gate
DMERA	χ	$\frac{\tau_2 \chi \log n}{\epsilon^2}$	Nonlocal Gate

Plausible?

I think so.

Method	# of Qubits	Contraction Time	Gate Type
fc-PEPS	χn	$\frac{\tau_2 n \chi}{\epsilon^2}$	NN Local Gate
DMERA	χ	$\frac{\tau_2 \chi \log n}{\epsilon^2}$	Nonlocal Gate

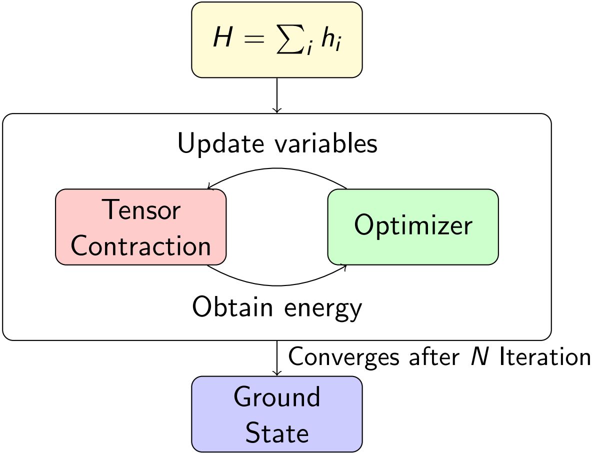
- For supeconducting qubits, NN local gate is possible. Gate time $50 \sim 300 ns$.
- For ion traps, Nonlocal gate is possible. Gate them $1 \sim 100 \mu s$.

For computation involving 50 qubits with gate depth of ~ 50,

	Memory	Time
Classical[a]	Terabytes	Hours
Quantum	50 Qubits	< 10s

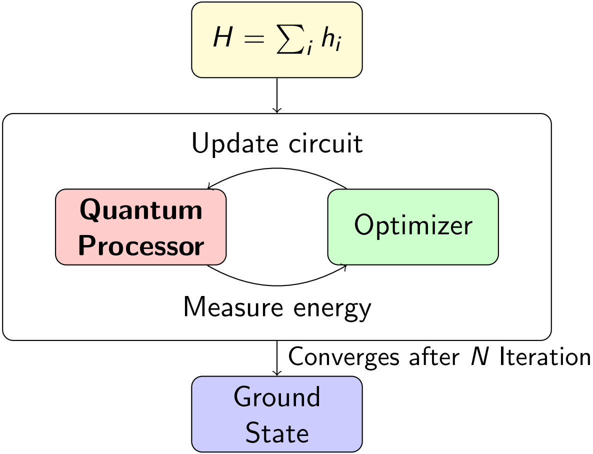
[a] Google, Intel, IBM, Alibaba, ETH, etc...

Classical Approach



$$\text{Optimization Time} = N * \text{Contraction Time}$$

Quantum Approach



Optimization Time $= N *$ Energy Measurement Time

Agenda

- What problem?
- Why is it classically hard?
- How can a quantum computer help us?
- **Can we deal with noise?**

Yes, without error correction.

Can we deal with noise?

For a particular tensor network proposed by us, e.g., fc-PEPS and DMERA,

1. For a noise strength of ϵ, expectation values of local observables deviate from the noiseless value by at most $O(\epsilon)$, even in the thermodynamic limit!
2. This is unexpected because the depth of the circuit scales with the system size.
3. For general large-depth quantum circuit, noise will accumulate too much and will have $O(1)$ effect.

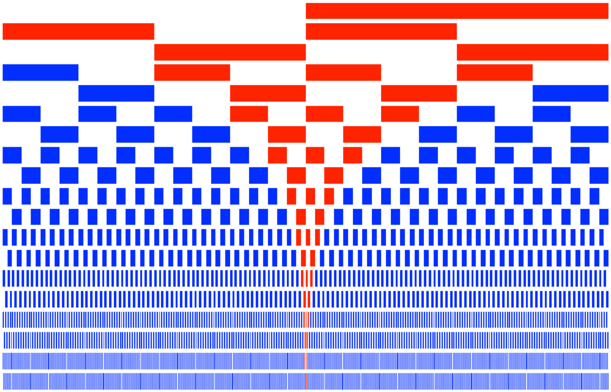
Noise still affects us, but more gracefully.

We call these circuits as *noise-resilient quantum circuits.*[a]

[a] K(2017), K and Swingle(2017)

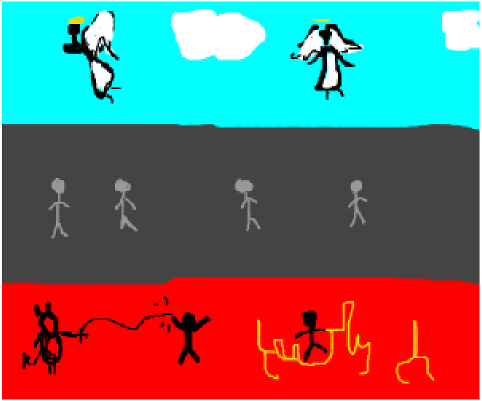
Intuition

Error occurred in the past decays exponentially.



$$\epsilon + \epsilon^2 + \cdots \epsilon^n = O(\epsilon)$$

Life after decoherence time



Fault-tolerant circuits

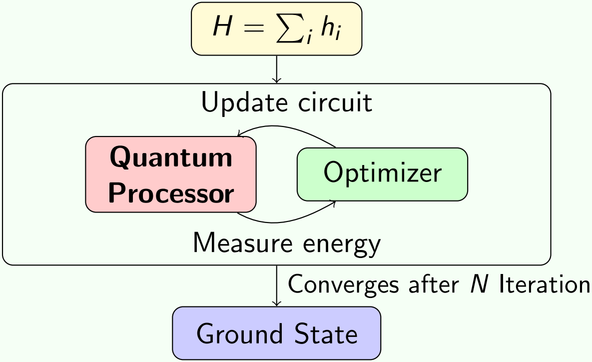
Noise-resilient circuits

Most circuits

Comments

1. Gate depth/count is a misleading figure of merit for noise-resilient quantum circuit.
2. I really think that we will be able to unlock the full potential of tensor network methods with near-term quantum computers.
3. One could have used this ansatz for machine-learning purposes. Noise-resilience is still there. By the same token, one can expect an advantage in using these ansatz for some machine learning purposes.
    - Whether such ansatz will be useful for industrial purposes is not clear at all.

Quantum Tensor Network Contraction



Opt Time $= N *$ Measurement Time

	Memory	Time
C	Pbytes	Hours
Q	50 Qubits	< 10s

- ~~Years of classical calculation~~
- Hours of quantum calculation.
- Hidden dissipative dynamics protects us from noise.

N empircally seems to be more than tens of thousands...