Hybrid classical-quantum linear solver on NISQ machines

Shiue Yuan Shiau

National Center for Theoretical Sciences (NCTS)

Hsinchu

August 20, 2020

Noisy intermediate-scale quantum (NISQ) era

- Feynman (1982)

- Fully fault-tolerant quantum computer

  still a long way to go

  HHL linear solver (2009), Grover's search (1995), Shor's factorization (1994)

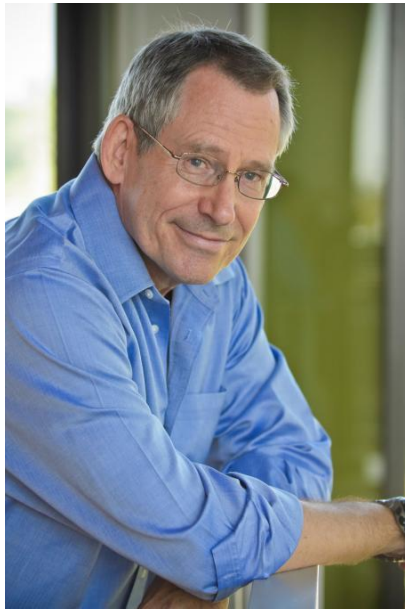
- NISQ era:

  1. Intermediate scale: 50 ~ 100 of qubits
  2. Quantum error correction not available
  3. Error rate 0.1% per gate ⟶ 1000 gates at most
  4. Task-oriented algorithms

     quantum approximate optimization algorithm

     variational quantum eigensolver

     quantum auto encoder



John Preskill

Preskill, Q2B conference 2017

Quantum 2, 79 (2018)

Quantum technology and business is booming

Software & Consultants

- |ħ⟩ Quantum Consultants
- 1QBit
- Entanglement Partners_ >
- Zapata Computing
- CQC Cambridge Quantum Computing Limited
- Atos
- Q^xBranch
- Entropica Labs – Quantum computing, at your service
- qb quantum benchmark
- Q&I
- ProteinQure
- Netramark
- Qu & Co
- Everettian
- Artiste-qb.net
- QC Ware
- Bra-Ket Science
- Quantika – quantum deep tech
- |EeroQ⟩
- Qilimanjaro
- Strange Works
- ⟨Qubit|Era⟩
- Horizon Quantum Computing
- River Lane Research

Quantum Computers

- IBM
- D-Wave – The Quantum Computing Company™
- Google
- Microsoft
- Rigetti
- Silicon Quantum Computing
- IonQ
- Alibaba Group
- NTT
- Xanadu
- QCI
- Baidu
- Nokia Bell Labs
- |EeroQ⟩
- Qilimanjaro
- Alpine Quantum Technologies
- Oxford Quantum Circuits

Enabling Technologies

- Zurich Instruments
- Qubitekk
- BlueFors
- Leiden Cryogenics – Leader in Low Temperature Techniques
- Oxford Cryosystems
- Anyon – Computing with Manufactured Quanta
- Intel
- Keysight Technologies
- Signadyne
- Metempsy
- Raytheon BBN Technologies
- Quandela
- Hypres – The Digital Superconductor Company
- Janis
- Spice Labs
- Delft Circuits – quantum computing hardware
- Q-CTRL

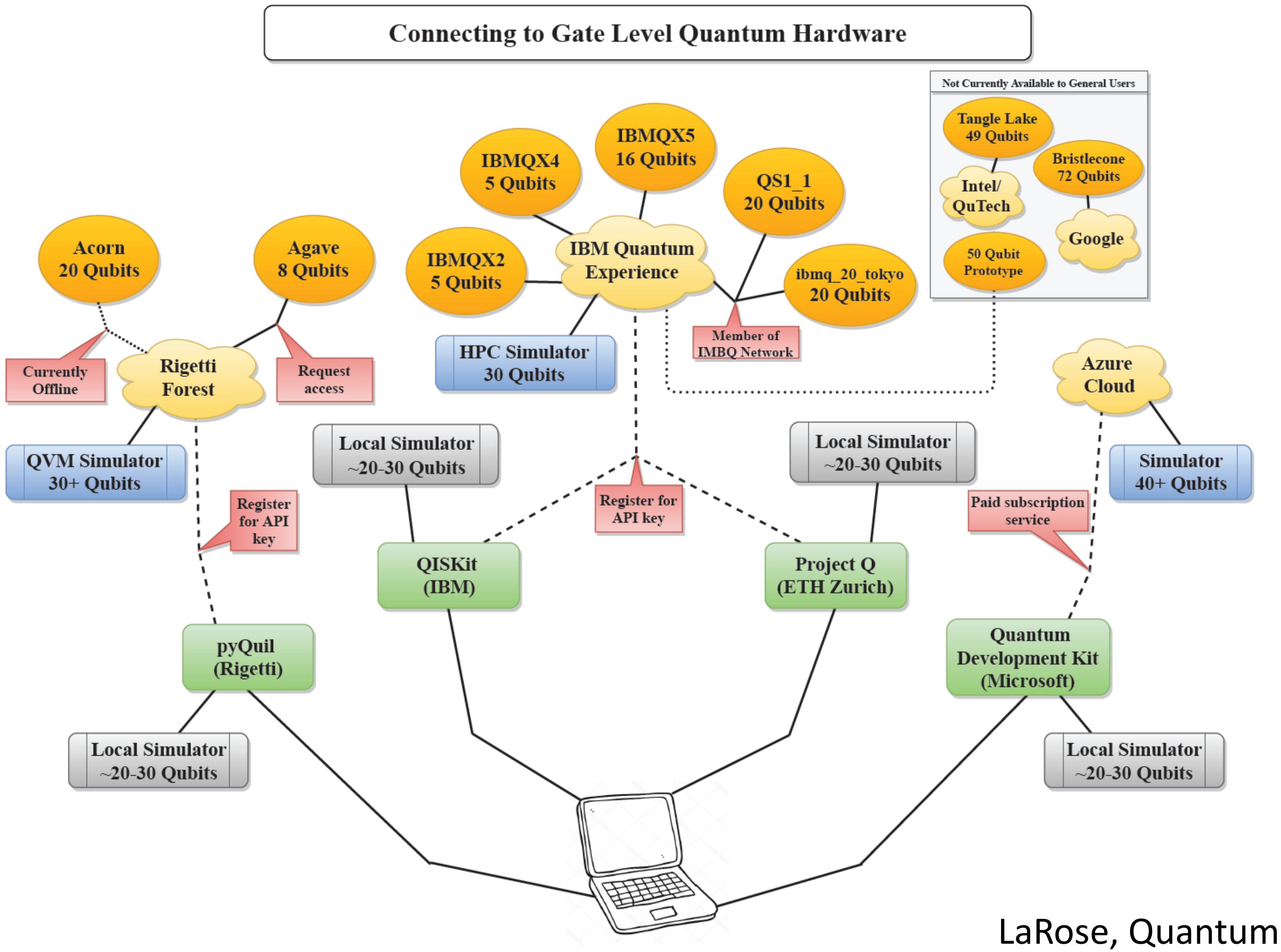
New Funding Strategies

- Qubit Protocol
- QuantumX
- Creative Destruction Lab
- The Quantum Revolution Fund
- Quantum Valley Investments

Representative list of players. A very active ecosystem!

Quantum World Association

Currently available quantum computing services



LaRose, Quantum 3, 130 (2019)

Why quantum computer ?

- Quantum algorithms for classically intractable problems

  Classically hard but quantum easy, such as Shor's integer factorization

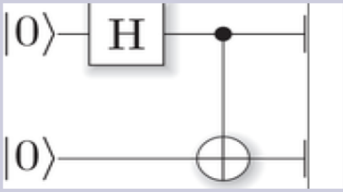
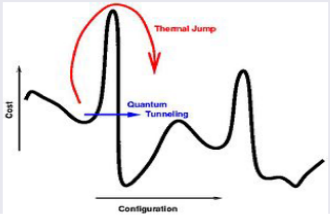
- Complexity arguments

  Quantum entanglement/correlation is hard to obtain by a classical means

- Classical algorithms cannot simulate quantum computer

Preskill, Quantum 2, 79 (2018)

Quantum computing models

Abstract Model	Theory	Commercial Product/ Experiments
Circuit/Gate-based 	Deutsch (1985)	IBM Q System One 
Quantum annealing/ Adiabatic 	Kadowaki & Nishimori (1998) Farhi, Goldstone, Gutmann, & Sipser (2000)	D-Wave 2000 Q 
One-way/ Measurement-based	Raussendorf & Briegel (2001)	Commercially N/A. Walther et al. Nature (2005) photon 4 qubits.

1. Each model has various physical realizations, superconductor, semiconductor, photon… etc.
2. Topological quantum computing is beyond the scope of this talk.

Curtesy CC Chen

Classical vs quantum circuits

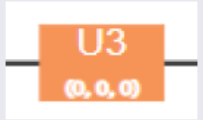
	Classical	Quantum
State	n classical bits	n qubits
Data	n Boolean data	2^n complex numbers $\lvert\psi\rangle = \sum_{\sigma_1,\ldots\sigma_n=0}^{1} c_{\sigma_1\ldots\sigma_n}\lvert\sigma_1,\ldots,\sigma_n\rangle$
Readout	Deterministic	Probabilistic
Gate Operation	Boolean function	Unitary transformation

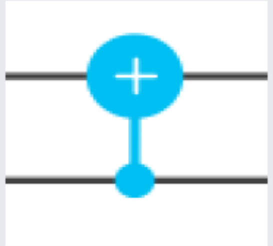
Curtesy CC Chen

Universal circuit construction

> Any classical/quantum circuit can be constructed using only a small set of gates.

<table>
<tr><th>Circuit</th><th>Universal Gates</th><th>Graph</th><th>Truth Table/Matrix</th></tr>
<tr><td>Classical</td><td>Boole/Shannon:
{AND,OR,NOT}
Sheffer (1913):
{NAND}</td><td>NAND:

</td><td><table>
<tr><th colspan="2">INPUT</th><th>OUTPUT</th></tr>
<tr><th>A</th><th>B</th><th>A NAND B</th></tr>
<tr><td>0</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
</table></td></tr>
<tr><td>Quantum</td><td>{U3,CX}</td><td>U3:


CX:
</td><td>$U_3(\lambda,\phi,\theta) = \begin{pmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i(\lambda+\phi)}\cos(\theta/2) \end{pmatrix}$

$\mathrm{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$</td></tr>
</table>

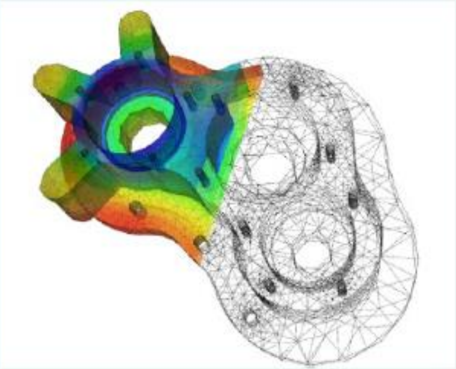
Nielsen and Chuang book

Curtesy CC Chen

Solving systems of linear equations

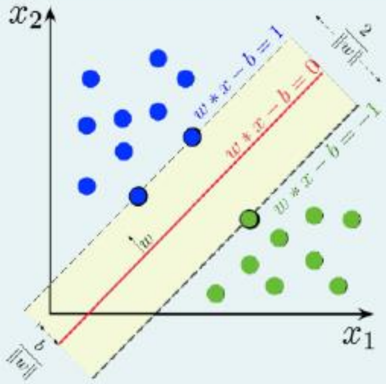
has vast implications

Engineering

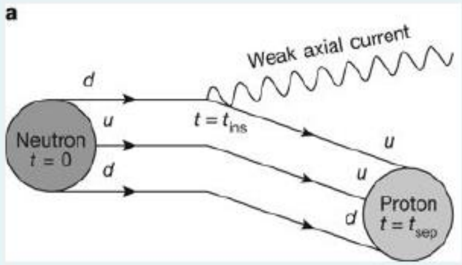


Finance

Machine Learning



Science



Supercomputer Benchmark

TOP 500
The List.

Solving systems of linear equations

Given a A matrix and a b vector, solve x $Ax = b$

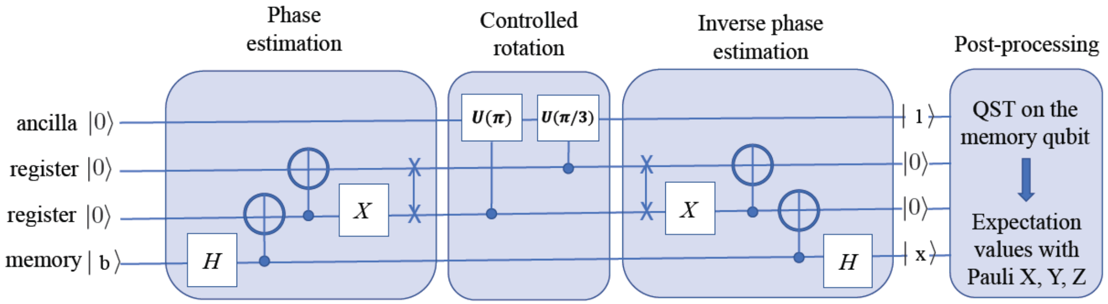
$$\begin{pmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{N1} & \cdots & A_{NN} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_N \end{pmatrix}$$

Classical algorithm runs in polynomial time $O(N^d)$ at best

Can quantum linear solver run faster ?

Harrow-Hassidim-Lloyd algorithm

Solve $A|x\rangle = |b\rangle$



Quantum circuit for solving 2 x 2 matrix

arXiv:1804.03719

- Complexity O(logN) for sparse matrices
- BQP-complete (can solve other problems)
- Input |b> and output |x> are quantum states
- Strong coherence demand

HHL, Phys. Rev. Lett. **103**, 150502 (2009)

SCIENTIFIC REPORTS
natureresearch

OPEN

Hybrid classical-quantum linear solver using Noisy Intermediate-Scale Quantum machines

Chih-Chieh Chen[1*], Shiue-Yuan Shiau[2], Ming-Feng Wu[1] & Yuh-Renn Wu[3*]

Ulam-von Neumann algorithm

Monte Carlo linear solver

- Solve $\boldsymbol{A}x = b$ through

$$x = \boldsymbol{A}^{-1}b = (1-\gamma\boldsymbol{P})^{-1}b = \sum_{s=0}^{\infty}\gamma^{s}\,\boldsymbol{P}^{\,s}\,b$$

- Random walk in finite steps to get approximate solution x

$$x_{i_0} = \sum_{s=0}^{c}\gamma^{s}\sum_{i_0,i_1,\ldots=0}^{N-1}P_{i_0,i_1}\ldots P_{i_{s-1},i_s}b_{i_s}$$

- Complexity O(N) for N X N stochastic matrix **P** coded in N bits

maybe useful in reinforcement learning

Barto, Duff (1993), Sutton, Barto (1998)

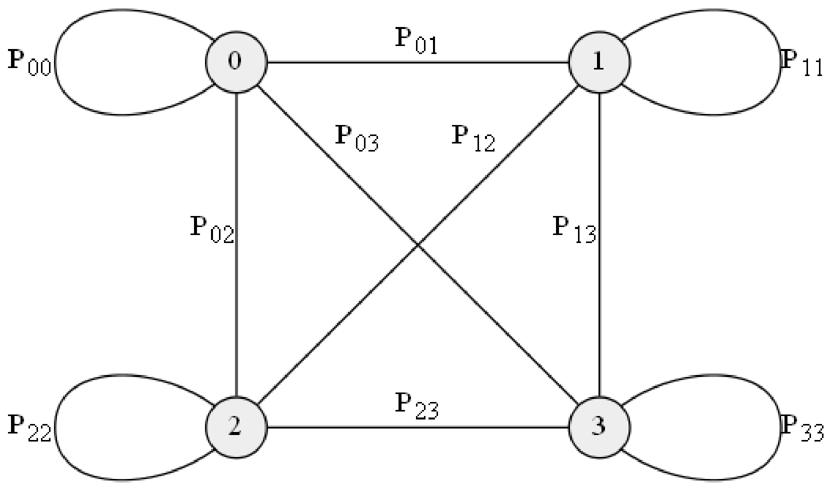
Classical random walk on a N-vertex graph

$P_{i,j} \geq 0$ transition probability of going from vertex i to vertex j or vice versa

$P_{i,j} = P_{j,i}$ symmetric (undirected)

vertices labeled by i= (0,1,2,3)

$$\boldsymbol{P} = \begin{pmatrix} P_{00} & P_{01} & P_{02} & P_{03} \\ P_{10} & P_{11} & P_{12} & P_{13} \\ P_{20} & P_{21} & P_{22} & P_{23} \\ P_{30} & P_{31} & P_{32} & P_{33} \end{pmatrix}$$



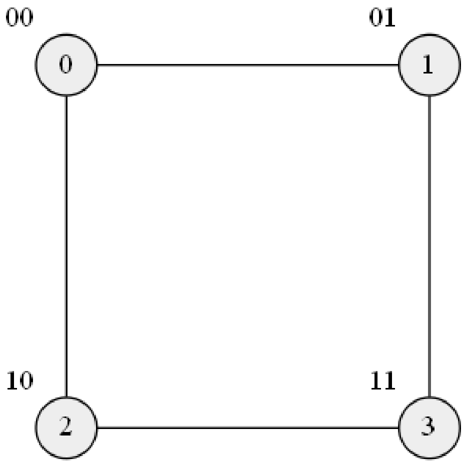
Encoding N X N matrix usually takes N bits

but encoding on a Hamming cube

requires only $\log_2 N$ bits

4-vertex graph requires 2 bits:

Node 0 represented by 2-bit state	|00>
Node 1	|01>
Node 2	|10>
Node 3	|11>

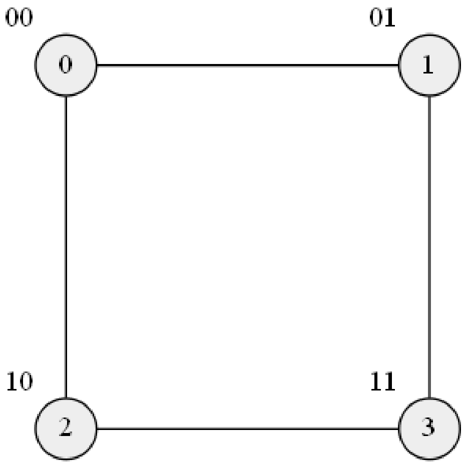


Hamming cube

hold true for quantum bits

Constructing 4 X 4 transition probability matrix **P**

Between 2-bit state $|j_1,j_0\rangle$ and $|j'_1,j'_0\rangle$
the m-th bit flips with probability $\sin^2(\theta_m/2)$
or not with probability $\cos^2(\theta_m/2)$



$$
\mathbf{P}^{classical} = \begin{array}{cc}
\begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} & \\
\begin{bmatrix}
\cos^2\left(\frac{\theta_0}{2}\right)\cos^2\left(\frac{\theta_1}{2}\right) & \boxed{\sin^2\left(\frac{\theta_0}{2}\right)\cos^2\left(\frac{\theta_1}{2}\right)} & \cos^2\left(\frac{\theta_0}{2}\right)\sin^2\left(\frac{\theta_1}{2}\right) & \sin^2\left(\frac{\theta_0}{2}\right)\sin^2\left(\frac{\theta_1}{2}\right) \\
 & \cos^2\left(\frac{\theta_0}{2}\right)\cos^2\left(\frac{\theta_1}{2}\right) & \sin^2\left(\frac{\theta_0}{2}\right)\sin^2\left(\frac{\theta_1}{2}\right) & \cos^2\left(\frac{\theta_0}{2}\right)\sin^2\left(\frac{\theta_1}{2}\right) \\
 & & \cos^2\left(\frac{\theta_0}{2}\right)\cos^2\left(\frac{\theta_1}{2}\right) & \sin^2\left(\frac{\theta_0}{2}\right)\cos^2\left(\frac{\theta_1}{2}\right) \\
 & & & \cos^2\left(\frac{\theta_0}{2}\right)\cos^2\left(\frac{\theta_1}{2}\right)
\end{bmatrix} & \begin{array}{c} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{array}
\end{array}
$$

$$
= \begin{bmatrix} \cos^2\left(\frac{\theta_1}{2}\right) & \sin^2\left(\frac{\theta_1}{2}\right) \\ \sin^2\left(\frac{\theta_1}{2}\right) & \cos^2\left(\frac{\theta_1}{2}\right) \end{bmatrix} \otimes \begin{bmatrix} \cos^2\left(\frac{\theta_0}{2}\right) & \sin^2\left(\frac{\theta_0}{2}\right) \\ \sin^2\left(\frac{\theta_0}{2}\right) & \cos^2\left(\frac{\theta_0}{2}\right) \end{bmatrix},
$$

factorizable matrix

Classical random walk on N-vertex Hamming cube

- Doing coin-flipping over all bits takes O(log N) time

- Limited to factorizable matrices

  because classical bits uncorrelated

- N X N matrix **P**

$$P^{classical}_{J',J} = \prod_{\ell=0}^{n-1} \left|\cos^2\left(\frac{\theta_\ell}{2}\right)\right|^{1-i_\ell} \left|\sin^2\left(\frac{\theta_\ell}{2}\right)\right|^{i_\ell}$$

$$|I\rangle_c = |i_{n-1}, \ldots, i_1, i_0\rangle_c$$

$$|J'\rangle_c = |I\rangle_c \oplus |J\rangle_c$$

bitwise exclusive

$$j_0 \neq j'_0 \longrightarrow i_0 = 1$$
$$j_0 = j'_0 \longrightarrow i_0 = 0$$

Quantum walk on N-vertex Hamming cube

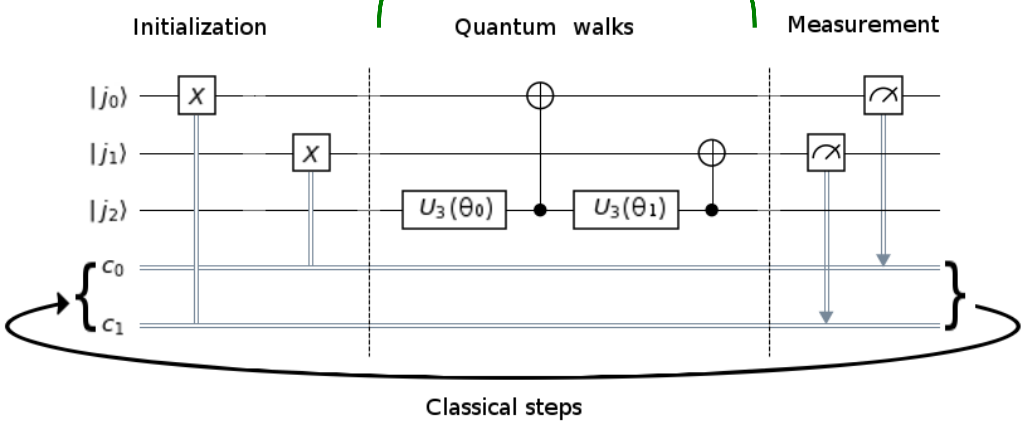
- Coin-flipping involves a rotation on the coin state and a CNOT gate to flip a qubit

- Doing coin-flipping over all qubits takes O(log N)

$$\mathcal{U} = \prod_{k=0}^{n-1}{}' \underbrace{(|0\rangle_\lozenge\langle 0|_\lozenge \otimes 1_q + |1\rangle_\lozenge\langle 1|_\lozenge \otimes X_k)}_{\text{CNOT gate}} \cdot \underbrace{(U_3(\mathbf{u}_k)}_{\text{coin rotation}} \otimes 1_q)$$

$$U_3(\mathbf{u}) = U_3(\theta, \phi, \lambda) = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -e^{i\lambda}\sin\left(\frac{\theta}{2}\right) \\ e^{i\phi}\sin\left(\frac{\theta}{2}\right) & e^{i(\lambda+\phi)}\cos\left(\frac{\theta}{2}\right) \end{bmatrix}$$

$\mathcal{U}$ operator



$x_{i_0} = \sum_{s=0}^{c} \gamma^s \sum_{i_0,i_1,\ldots=0}^{N-1} P_{i_0,i_1} \ldots P_{i_{s-1},i_s} b_{i_s}$

Quantum walk on N-vertex Hamming cube

- Lead to nonfactorizable **P** matrices
  because quantum bits correlated

- N X N matrix **P**

$$
\begin{aligned}
P_{J',J}^{quantum} &= \prod_{\ell=0}^{n-1} \left| U_3(\mathbf{u}_\ell)_{i_\ell, i_{\ell-1}} \right|^2 \\
&= \prod_{\ell=0}^{n-1} \left| \cos^2\left(\frac{\theta_\ell}{2}\right) \right|^{1-(i_\ell \oplus i_{\ell-1})} \left(\sin^2\left(\frac{\theta_\ell}{2}\right) \right)^{i_\ell \oplus i_{\ell-1}}
\end{aligned}
$$

neighboring qubits temporally correlated
non-Markovian memory effect

$$|J'\rangle_c = |I\rangle_c \oplus |J\rangle_c$$

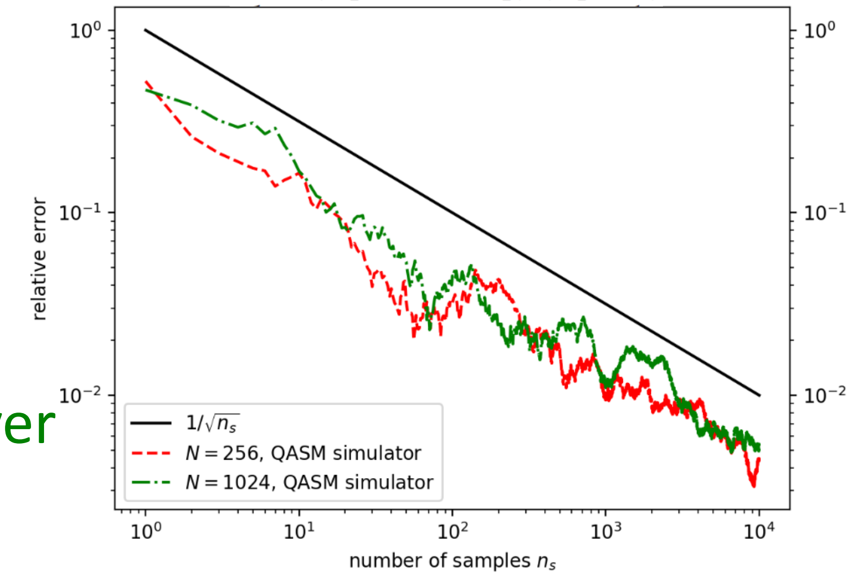
$$|I\rangle_c = |i_{n-1}, \ldots, i_1, i_0\rangle_c$$

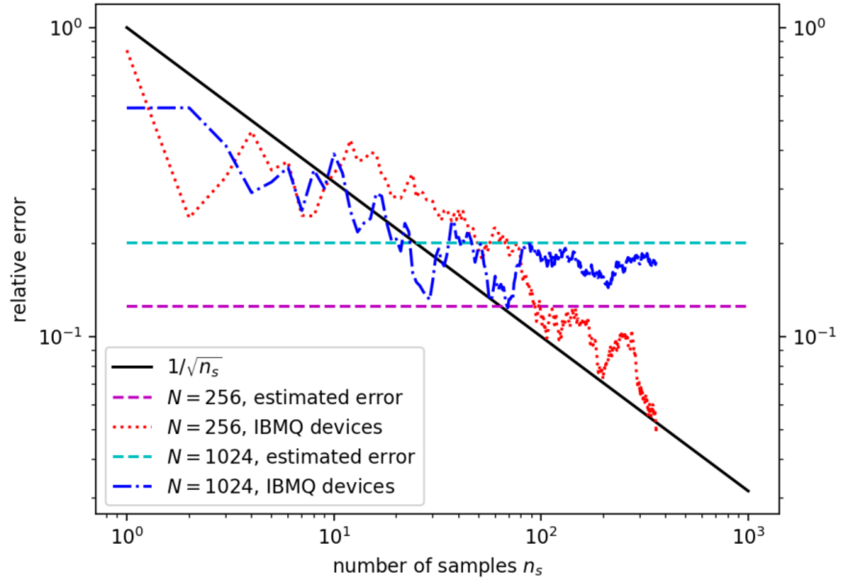
Results on N=(256, 1024) graphs

- Works on QASM simulator (Qiskit)
- Error reduction $1/\sqrt{n_s}$ like Monte Carlo solver
- Works on noisy IBM Q machine

  bounded by machine error
  readout error
  condition number
  communication overhead

$$\varepsilon = |x_I^{exact} - x_I|/|x_I^{exact}|$$

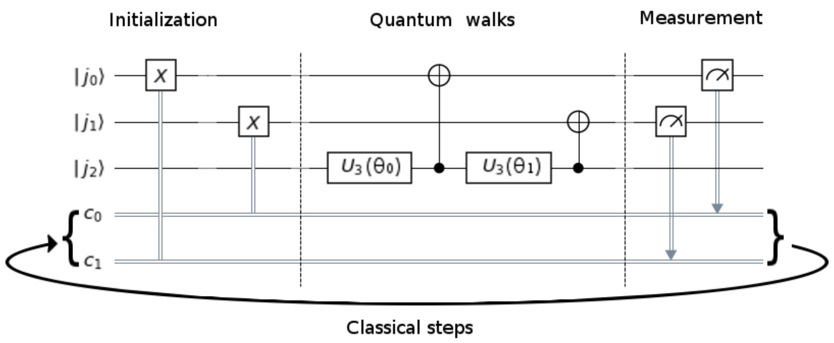




CC Chen, SY Shiau, MF Wu, YR Wu , Sci. Rep. **9**, 16251 (2019)

2 runs of quantum walk

$(\mathcal{U})^d$



$$P^{quantum}_{J',J} = \sum_{k=0}^{1} \left| \sum_{I} f(I,\, J' \oplus J \oplus I)\,\delta_{i_{n-1},k} \right|^2$$

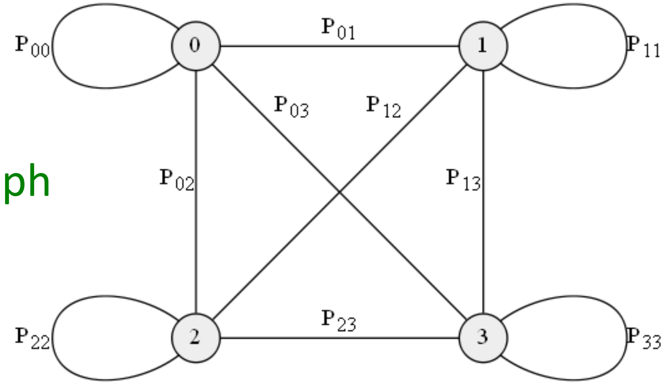
Square of the sum:

quantum interference comes into play

$$\begin{aligned} f(I,\,K) &= [U_3(\mathbf{u}_{n-1})]_{i_{n-1},i_{n-2}} \cdots [U_3(\mathbf{u}_0)]_{i_0,k_{n-1}} \\ &\quad \times [U_3(\mathbf{u}_{n-1})]_{k_{n-1},k_{n-2}} \cdots [U_3(\mathbf{u}_0)]_{k_0,0} \end{aligned}$$

$$U_3(\mathbf{u}) = U_3(\theta,\,\phi,\,\lambda) = \begin{bmatrix} \cos\left(\dfrac{\theta}{2}\right) & -e^{i\lambda}\sin\left(\dfrac{\theta}{2}\right) \\ e^{i\phi}\sin\left(\dfrac{\theta}{2}\right) & e^{i(\lambda+\phi)}\cos\left(\dfrac{\theta}{2}\right) \end{bmatrix}$$

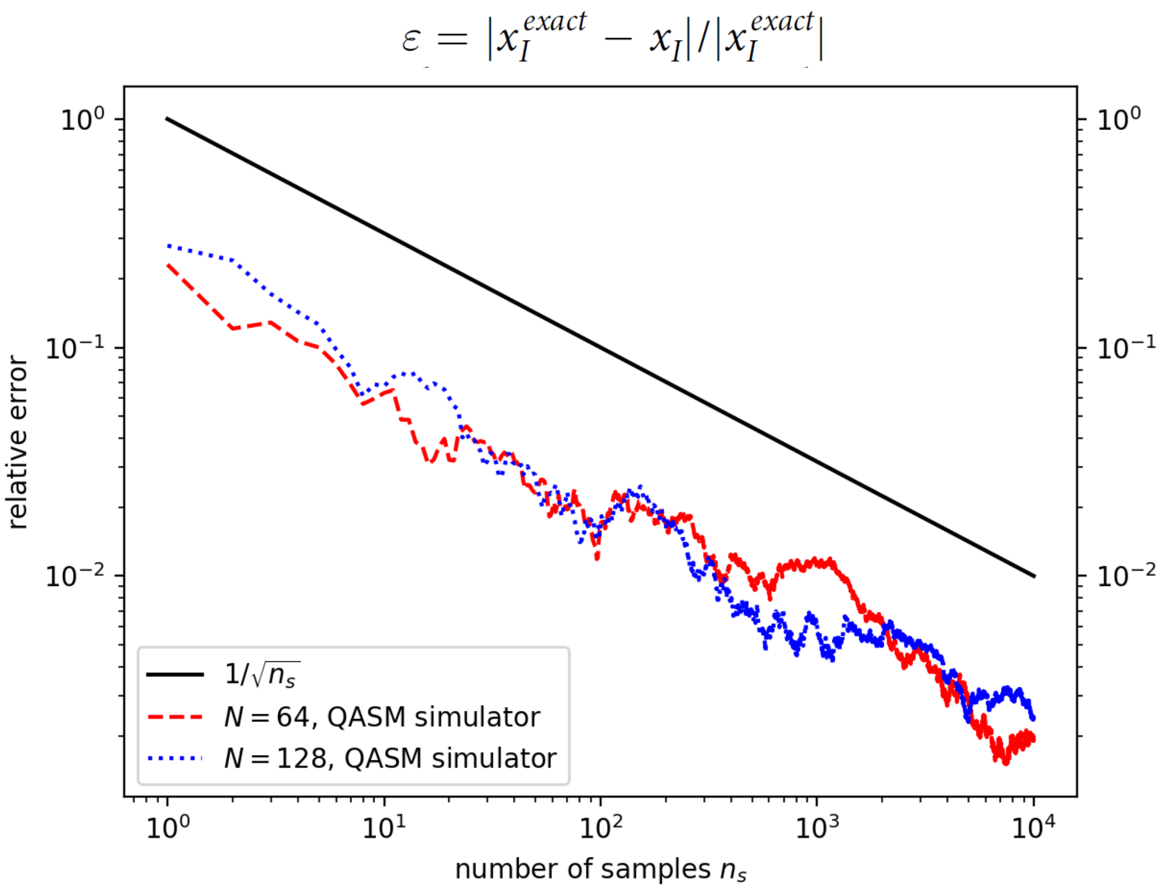
4-node graph



$$\begin{aligned} P_{00} &= P_{11} = P_{22} = P_{33} \\ &= \frac{1}{4}\sin^2\theta_0 + \frac{1}{8}(1 + \cos^2\theta_1 + \cos^2\theta_0 + 4\cos\theta_1\cos\theta_0 \\ &\quad + \cos^2\theta_1\cos^2\theta_0 - \sin^2\theta_1\sin^2\theta_0), \\ P_{01} &= P_{23} = \frac{1}{4}\sin^2\theta_1 \\ P_{02} &= P_{13} = \frac{1}{4}\sin^2\theta_1 \\ P_{03} &= P_{12} = \frac{1}{4}(1 - 2\cos\theta_1\cos\theta_0 + \cos^2\theta_1) \end{aligned}$$

Non-trivial phase dependence

Results on 2 runs of quantum walk



CC Chen, SY Shiau, MF Wu, YR Wu , Sci. Rep. 9, 16251 (2019)

Comparison of various classical/quantum algorithms

Algorithm	Time	Space for A	Input/Output
Classical Direct[2,3]	$\mathcal{O}(N^3)$	$\mathcal{O}(N^2)$	efficient for any $\mathbf{A}$, $\vec{x}$, $\vec{b}$
Classical Iterative[2,3]	$\mathcal{O}(N^2)$	$\mathcal{O}(N^2)$	efficient for any $\mathbf{A}$, $\vec{x}$, $\vec{b}$
Quantum HHL[4]	$\mathcal{O}(\log(N))$	$\mathcal{O}(\log(N))$ qubits	norm $\|\vec{x}\|$ not available difficult for $\mathbf{A}$, $\vec{x}$, $\vec{b}$
Classical MC[45,53,55] (for one component x_I)	$\mathcal{O}(N)$	$\mathcal{O}(N)$	efficient for any $\vec{x}$, $\vec{b}$ limited $\mathbf{A}$ (stochastic $\mathbf{P}$)
Classical RW on HC (for one component x_I)	$\mathcal{O}(\log(N))$	$\mathcal{O}(\log(N))$	efficient for any $\vec{x}$, $\vec{b}$ limited $\mathbf{A}$ (factorisable $\mathbf{P}$)
Hybrid QW on HC (for one component x_I)	$\mathcal{O}(\log(N))$	$\mathcal{O}(\log(N))$ qubits	efficient for any $\vec{x}$, $\vec{b}$ limited $\mathbf{A}$ (correlated $\mathbf{P}$)

Summary

- Discrete time coined quantum random walk linear solver
  on NISQ machines

- Complexity O(N) for both classical and quantum algorithms

- Classical algorithm can only deal with uncorrelated matrices

- Level of qubit correlation increases with the depth of quantum circuit
  difficult to evaluate using classical Monte Carlo sampling

- Likely to serve as quantum subroutine in a classical framework