Hamiltonian simulation with nearly optimal dependence on all parameters

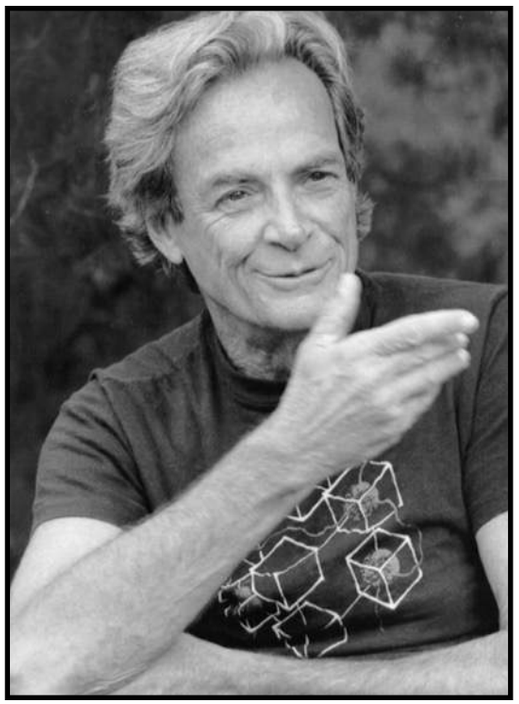
Andrew Childs

CS, UMIACS, & QuICS
University of Maryland

JOINT CENTER FOR
QUANTUM INFORMATION
AND COMPUTER SCIENCE

Joint work with Dominic Berry (Macquarie) and Robin Kothari (MIT),
building on previous work also with Richard Cleve (Waterloo) and Rolando Somma (Los Alamos)

arXiv:1312.1414 / STOC 2014 arXiv:1412.4687 / to appear in PRL arXiv:1501.01715



"... nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy."

Richard Feynman
Simulating physics with computers (1981)

Why simulate quantum mechanics?

Computational chemistry/physics

- chemical reactions
- properties of materials

Implementing quantum algorithms

- continuous-time quantum walk (e.g., for formula evaluation)
- adiabatic quantum computation (e.g., for optimization)
- linear/differential equations

Quantum dynamics

The dynamics of a quantum system are determined by its *Hamiltonian*.

$$i\frac{\mathrm{d}}{\mathrm{d}t}|\psi(t)\rangle = H|\psi(t)\rangle$$
$$\Downarrow$$
$$|\psi(t)\rangle = e^{-iHt}|\psi(0)\rangle$$

Quantum simulation problem: Given a description of the Hamiltonian H, an evolution time t, and an initial state $|\psi(0)\rangle$, produce the final state $|\psi(t)\rangle$ (to within some error tolerance ϵ)

A classical computer cannot even represent the state efficiently

A quantum computer cannot produce a complete description of the state, but by performing measurements on the state, it can answer questions that (apparently) a classical computer cannot

Local and sparse Hamiltonians

Local Hamiltonians [Lloyd 96]

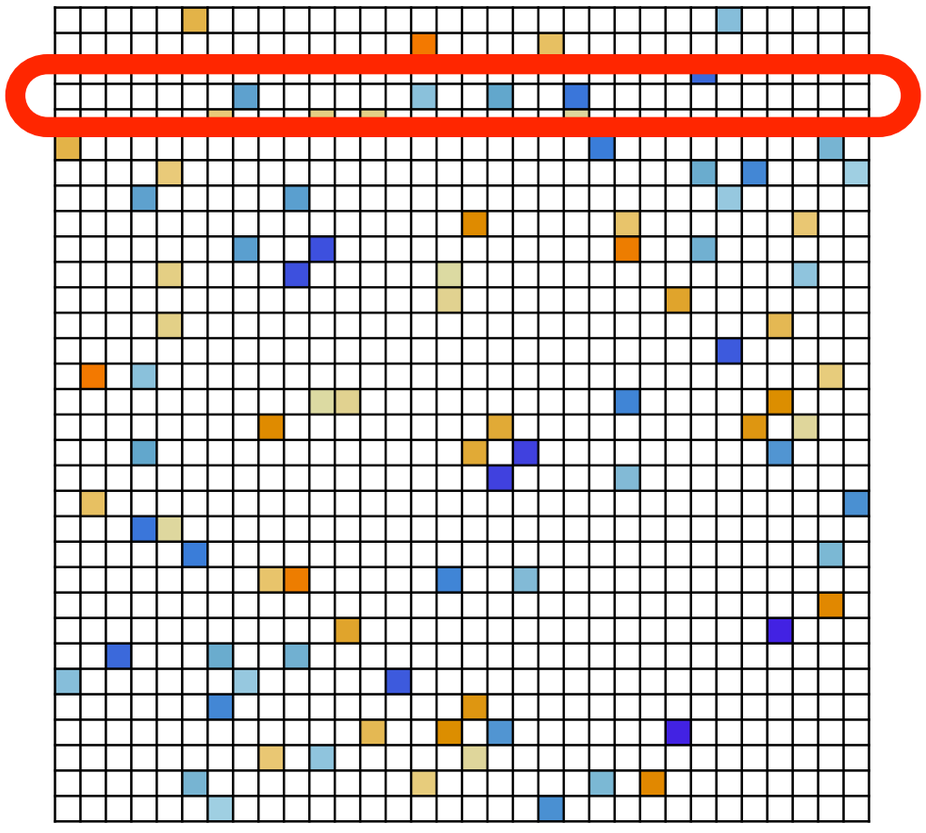
$H = \sum_{j=1}^{m} H_j$ where each H_j acts on $k = O(1)$ qubits

Sparse Hamiltonians [Aharonov, Ta-Shma 03]

At most d nonzero entries per row, $d = \mathrm{poly}(\log N)$ (where H is $N \times N$)

In any given row, the location of the jth nonzero entry and its value can be computed efficiently (or is given by a black box)

$H =$



Note: A k-local Hamiltonian with m terms is d-sparse with $d = 2^k m$

Previous simulation methods

Product formulas

- Decompose Hamiltonian into a sum of terms that are easy to simulate
- Recombine the terms by alternating between them

$$\left(e^{-iAt/r}e^{-iBt/r}\right)^r = e^{-i(A+B)t} + O(t^2/r)$$

$$\left(e^{-iAt/2r}e^{-iBt/r}e^{-iAt/2r}\right)^r = e^{-i(A+B)t} + O(t^3/r^2)$$

$$\vdots$$

Quantum walk

- Define an easy-to-implement unitary operation (a step of a quantum walk) whose spectrum is related to the Hamiltonian
- Use phase estimation to obtain information about the spectrum
- Introduce phases to give the desired evolution

Complexity of previous simulation methods

Parameters: dimension N sparsity d

evolution time t allowed error ϵ

[Lloyd 96]: $\mathrm{poly}(\log N)\,(\|H\|t)^2/\epsilon$ (for local Hamiltonians only)

[Aharonov, Ta-Shma 02]: $\mathrm{poly}(d, \log N)\,(\|H\|t)^{3/2}/\sqrt{\epsilon}$

[Childs 04]: $O\big((d^4 \log^4 N \|H\|t)^{1+\delta}/\epsilon^{\delta}\big)$ (for any $\delta > 0$)

[Berry, Ahokas, Cleve, Sanders 07]: $O\big((d^4 \log^* N \|H\|t)^{1+\delta}/\epsilon^{\delta}\big)$

[Childs, Kothari 11]: $O\big((d^3 \log^* N \|H\|t)^{1+\delta}/\epsilon^{\delta}\big)$

[Childs10; Berry, Childs 12]: $O(d\|H\|_{\max}t/\sqrt{\epsilon})$

New result: $O\big(\tau \frac{\log(\tau/\epsilon)}{\log\log(\tau/\epsilon)}\big)$ ~~$\tau := d^2\|H\|_{\max}t$~~ $\tau := d\|H\|_{\max}t$

Improved simulation algorithms

We have developed a novel approach that directly implements the Taylor series of the evolution operator

New tools:

- Implementing linear combinations of unitary operations
- Oblivious amplitude amplification

Dependence on simulation error is $\mathrm{poly}(\log(1/\epsilon))$, an exponential improvement over previous work

Algorithms are also simpler, with less overhead

Linear combinations of unitaries

LCU Lemma: Given the ability to perform unitaries V_j with unit complexity, one can perform the operation $U = \sum_j \beta_j V_j$ with complexity $O(\sum_j |\beta_j|)$. Furthermore, if U is (nearly) unitary then this implementation can be made (nearly) deterministic.

Main ideas:

- Using controlled-V_j operations, implement U with some amplitude:

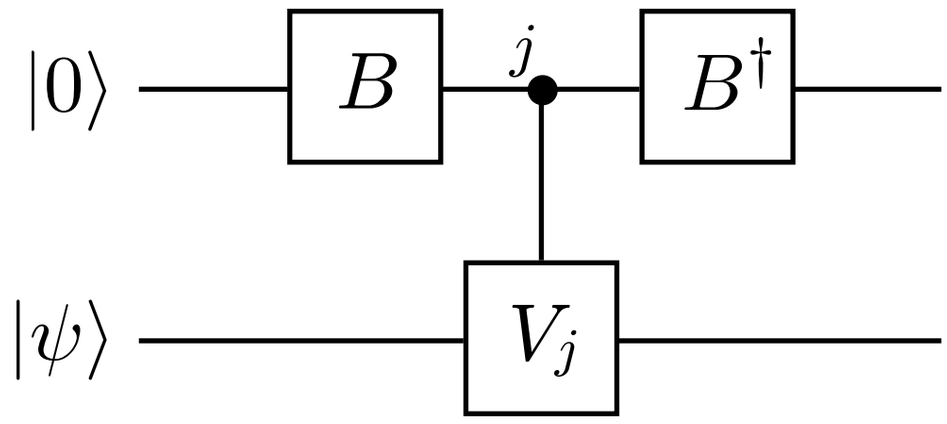
$$|0\rangle|\psi\rangle \mapsto \sin\theta|0\rangle U|\psi\rangle + \cos\theta|\Phi\rangle$$

- Boost the amplitude for success by oblivious amplitude amplification

Implementing U with some amplitude

$$U = \sum_j \beta_j V_j \qquad \text{(WLOG } \beta_j > 0\text{)}$$

Ancilla state: $B|0\rangle = \frac{1}{\sqrt{s}} \sum_j \sqrt{\beta_j}|j\rangle$ $\qquad s := \sum_j \beta_j$



$$\frac{1}{s}|0\rangle U|\psi\rangle + \sqrt{1 - \frac{1}{s^2}}\,|\Phi\rangle$$

with $\langle 0|\Phi\rangle = 0$

Oblivious amplitude amplification

Suppose W implements U with amplitude $\sin\theta$:

$$W|0\rangle|\psi\rangle = \sin\theta|0\rangle U|\psi\rangle + \cos\theta|\Phi\rangle$$

To perform U with amplitude close to 1: use amplitude amplification?

But the input state is unknown!

Using ideas from [Marriott, Watrous 05], we can show that a $|\psi\rangle$-independent reflection suffices to do effective amplitude amplification.

With this *oblivious amplitude amplification*, we can perform the ideal evolution with only about $1/\sin\theta$ steps.

We also give a robust version that works even when U is not exactly unitary.

Simulating the Taylor series

Taylor series of the dynamics generated by H:

$$e^{-iHt} = \sum_{k=0}^{\infty} \frac{(-iHt)^k}{k!}$$

$$\approx \sum_{k=0}^{K} \frac{(-iHt)^k}{k!}$$

Write $H = \sum_\ell \alpha_\ell H_\ell$ where each H_ℓ is unitary

Then $e^{-iHt} \approx \sum_{k=0}^{K} \sum_{\ell_1,\ldots,l_k} \frac{(-it)^k}{k!} \alpha_{\ell_1} \cdots \alpha_{\ell_k} H_{\ell_1} \cdots H_{\ell_k}$

is a linear combination of unitaries

Decomposing sparse Hamiltonians

To express H as a linear combination of unitaries:

- Edge coloring: $H = \sum_{j=1}^{d^2} H_j$ where each H_j is 1-sparse

  new trick: H is bipartite wlog since it suffices to simulate $H \otimes \sigma_x$

  d^2-coloring: $\mathrm{color}(\ell, r) = (\mathrm{idx}(\ell, r), \mathrm{idx}(r, \ell))$

- Approximately decompose into terms with all nonzero entries equal

  Ex: $\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 & 3 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$

- Remove zero blocks so that all terms are rescaled unitaries

  Ex: $\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} + \frac{1}{2} \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

Why $\mathrm{poly}(\log(1/\epsilon))$?

Lowest-order product formula:

$$(e^{-iA/r}e^{-iB/r})^r = e^{-i(A+B)} + O(1/r)$$

so we must take $r = O(1/\epsilon)$ to achieve error at most ϵ

Higher-order formulas exist, but they only improve the power of ϵ

The approximation $e^{-iHt} \approx \sum_{k=0}^{K} \frac{(-iHt)^k}{k!}$ has error ϵ provided

$$K = O\left(\frac{\log(1/\epsilon)}{\log\log(1/\epsilon)}\right)$$

A discrete-time quantum walk for any H

Another way to simulate an $N \times N$ Hamiltonian H is to implement a related discrete-time (Szegedy) quantum walk.

Expand space from $\mathbb{C}^N$ to $\mathbb{C}^{N+1} \otimes \mathbb{C}^{N+1}$.

Walk operator is the product of two reflections:

- Swap: $S|j,k\rangle = |k,j\rangle$
- Reflect about $\mathrm{span}\{|\psi_1\rangle, \ldots, |\psi_N\rangle\}$, where

$$|\psi_j\rangle := |j\rangle \otimes \left(\frac{1}{\sqrt{\|H\|_1}} \sum_{k=1}^{N} \sqrt{H_{jk}^*}\, |k\rangle + \nu_j |N+1\rangle \right)$$

$$\|H\|_1 := \max_j \textstyle\sum_{k=1}^{N} |H_{jk}|$$

i.e., $2TT^\dagger - \mathbb{1}$ where $T|j\rangle = |\psi_j\rangle$

[Childs 10]

Quantum walk simulation

Each eigenvalue λ of H corresponds to two eigenvalues $\pm e^{\pm i \arcsin \lambda}$ of the walk operator (with eigenvectors closely related to those of H)

Strategy: Use phase estimation to determine and correct the phase

Complexity: $O(\tau/\sqrt{\epsilon})$ $\quad \tau := d\|H\|_{\max} t$

[Childs 10], [Berry, Childs 12]

Linear combination of quantum walk steps

Another approach: find coefficients so that

$$e^{-iH} \approx T^\dagger \sum_{k=-K}^{K} \beta_k U^k T$$

and implement this using the LCU Lemma

By a generating series for Bessel functions,

$$e^{-i\lambda t} = \sum_{k=-\infty}^{\infty} J_k(-t)\, e^{ik \arcsin \lambda}$$

Coefficients drop off rapidly for large k, so we can truncate the series

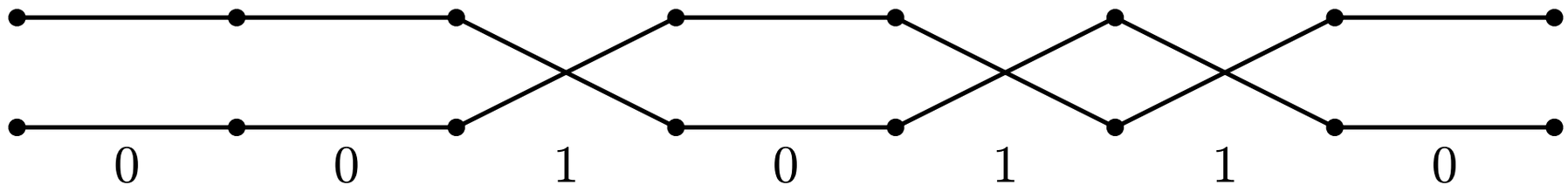
Query complexity of this approach: $O\left(\tau \frac{\log(\tau/\epsilon)}{\log\log(\tau/\epsilon)}\right)$

$$\tau := d\|H\|_{\max} t$$

Lower bounds

No-fast-forwarding theorem [BACS 07]: $\Omega(t)$

- Query complexity of computing the parity of n bits is $\Omega(n)$.
- There is a Hamiltonian that can compute parity by running for time $O(n)$.



New lower bound: $\Omega(\frac{\log(1/\epsilon)}{\log\log(1/\epsilon)})$

- Query complexity of parity is $\Omega(n)$ even for *unbounded error*.
- The same Hamiltonian as above computes parity with unbounded error by running for any positive time. Running for constant time gives the parity with probability $\Theta(1/n!)$.

New lower bound: $\Omega(dt)$

- Replacing each edge with $K_{d,d}$ effectively boosts Hamiltonian by d.

Query complexity of sparse Hamiltonian simulation

Quantum walk + phase estimation [BC 10]: $O\left(\frac{\tau}{\sqrt{\epsilon}}\right)$ $\tau := d\|H\|_{\max}t$

Quantum walk + LCU [BCK 15]: $O\left(\tau\frac{\log(\tau/\epsilon)}{\log\log(\tau/\epsilon)}\right)$

or for $\alpha \in (0,1]$: $O\left(\tau^{1+\alpha/2} + \tau^{1-\alpha/2}\log(1/\epsilon)\right)$

Lower bound: $\Omega\left(\tau + \frac{\log(1/\epsilon)}{\log\log(1/\epsilon)}\right)$

Notes:

- Gate complexity is only slightly larger than query complexity
- These techniques assume time-independent Hamiltonians (otherwise, use fractional queries/LCU on Dyson series [BCCKS 14])

Outlook

Improved simulation algorithms

- Optimal tradeoff for sparse Hamiltonian simulation
- Faster algorithms for structured problems
- Simulating open quantum systems

Applications to simulating physics

- What is the cost in practice for simulating molecular systems?
- How do recent algorithms compare to naive methods?

New quantum algorithms

- Improved algorithms for linear systems
- New applications of linear systems
- Other quantum algorithms from quantum simulation