Digital quantum simulation algorithms

Andrew Childs
University of Maryland

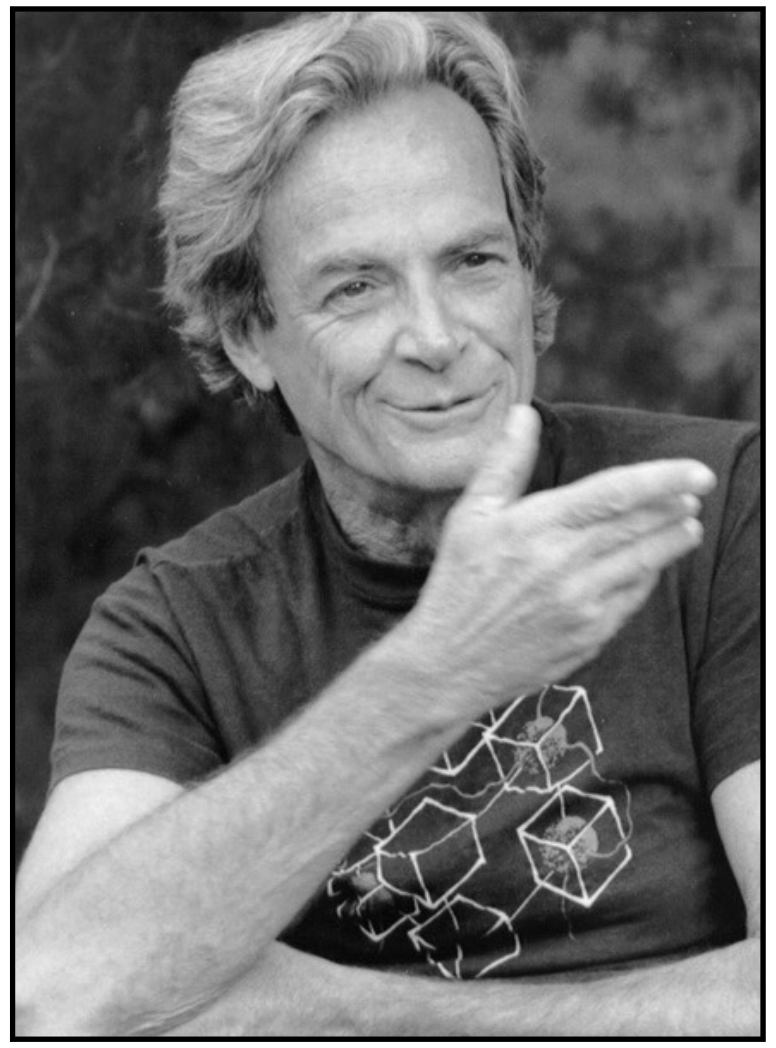
COMPUTER SCIENCE
UNIVERSITY OF MARYLAND

UMIACS
University of Maryland
Institute for Advanced
Computer Studies

JOINT CENTER FOR
QUANTUM INFORMATION
AND COMPUTER SCIENCE

NSF

Institute for
Robust Quantum
Simulation



"… 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)

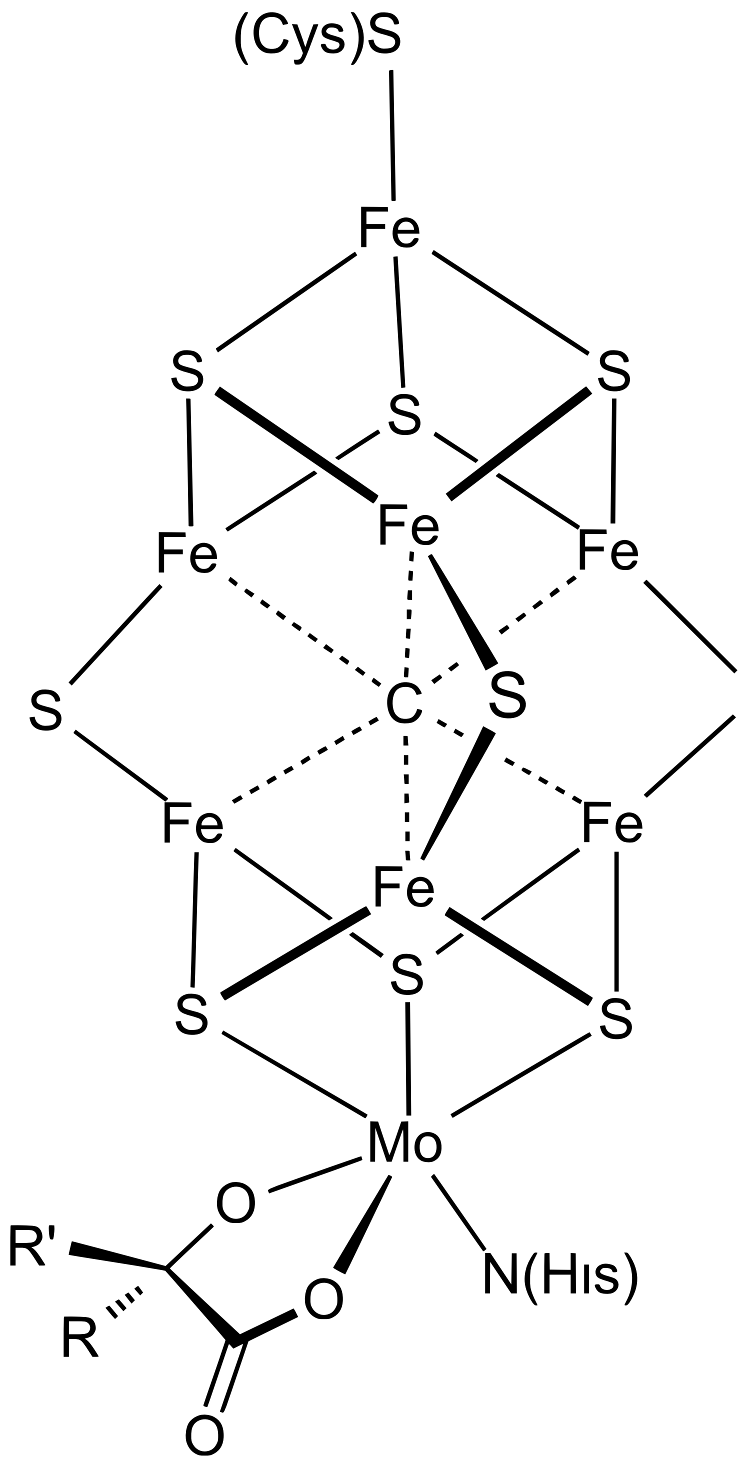
This talk: Algorithms for simulating quantum dynamics on digital, fault-tolerant quantum computers.

Outline

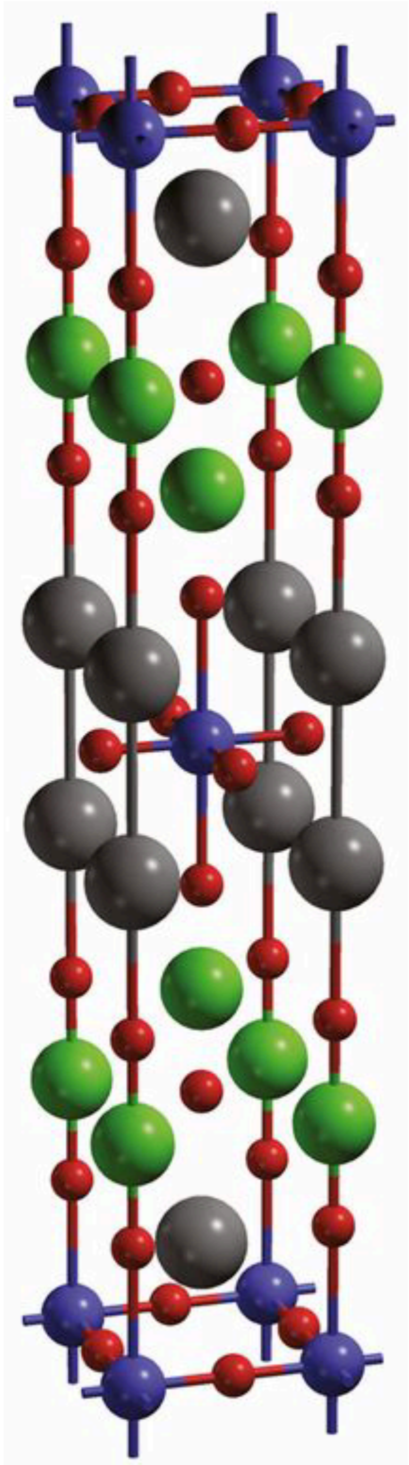
- Background
  - Applications of quantum simulation
  - Problem statement
  - Specifying Hamiltonians
- Complexity
  - BQP-hardness
  - No fast forwarding
- Product formulas
- Post-Trotter methods
- Special topics
  - Spatial locality
  - Interaction picture
  - Randomized simulation
  - Average-case simulation and entanglement
- Outlook

Background

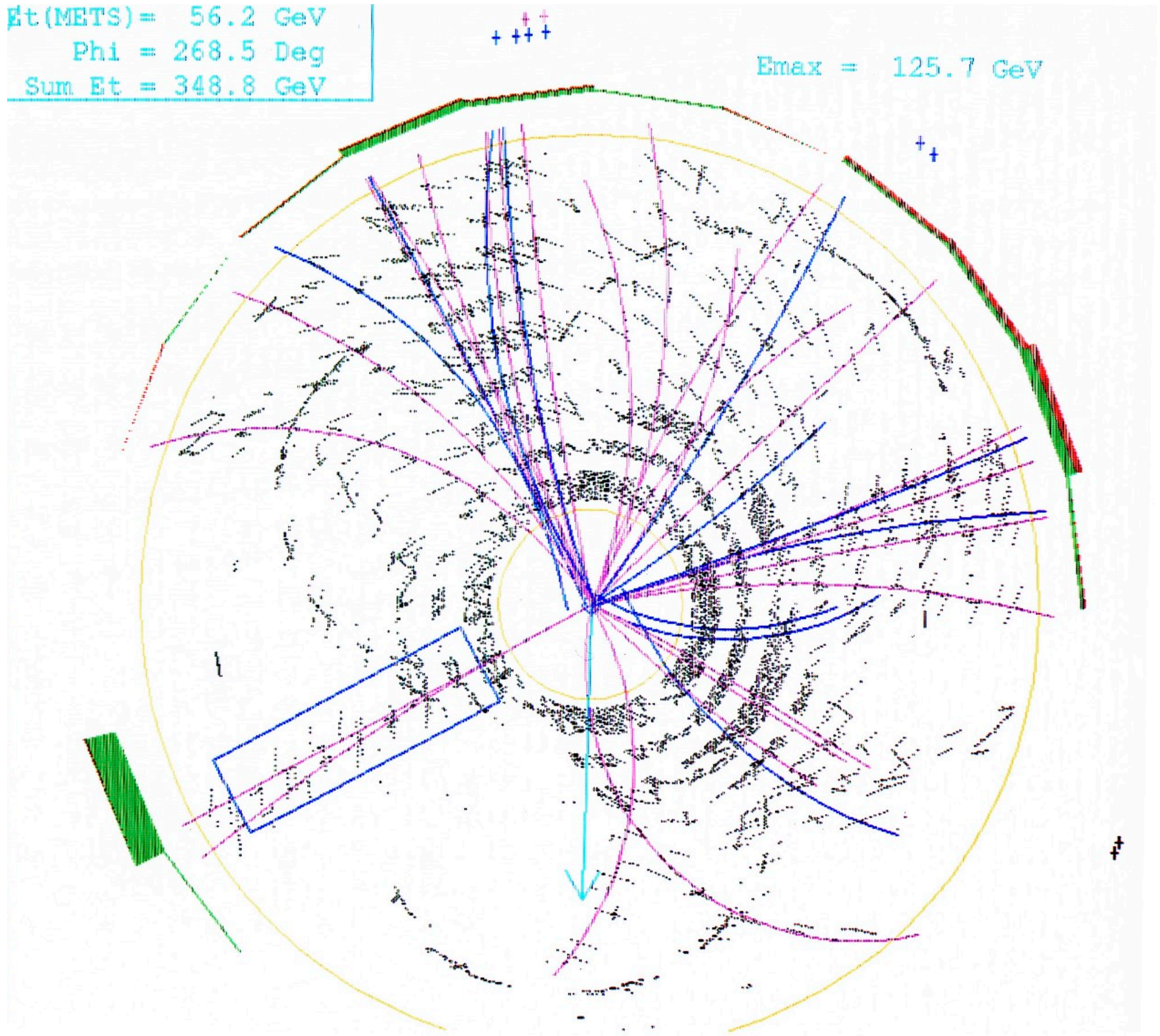
Computational quantum physics



quantum chemistry

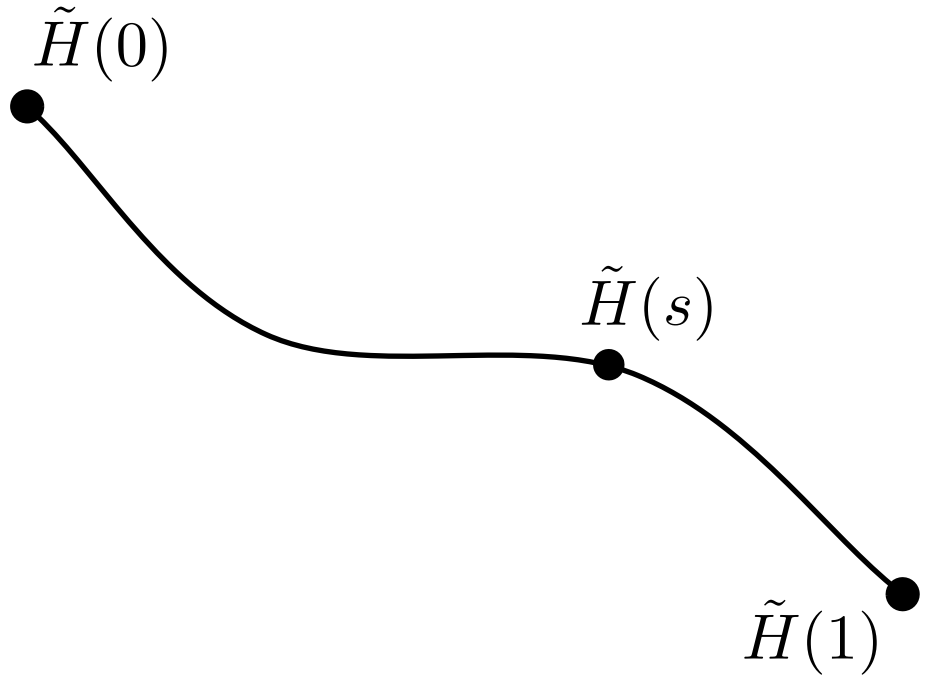


condensed matter physics/
materials science

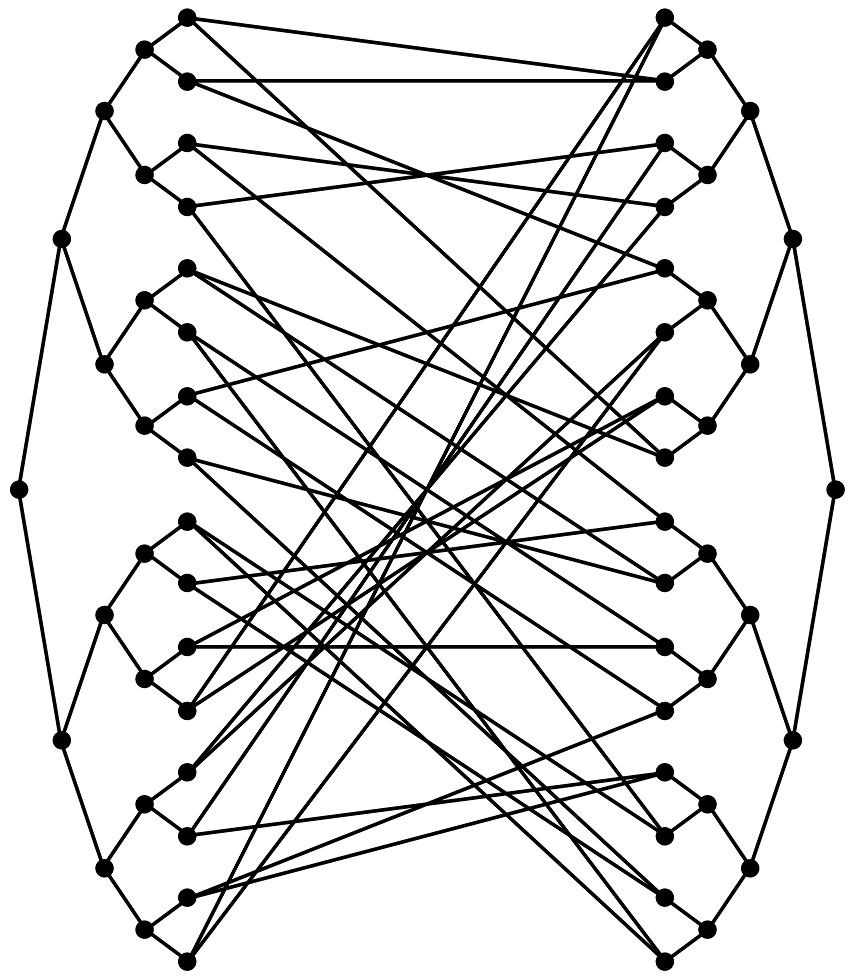


nuclear/particle
physics

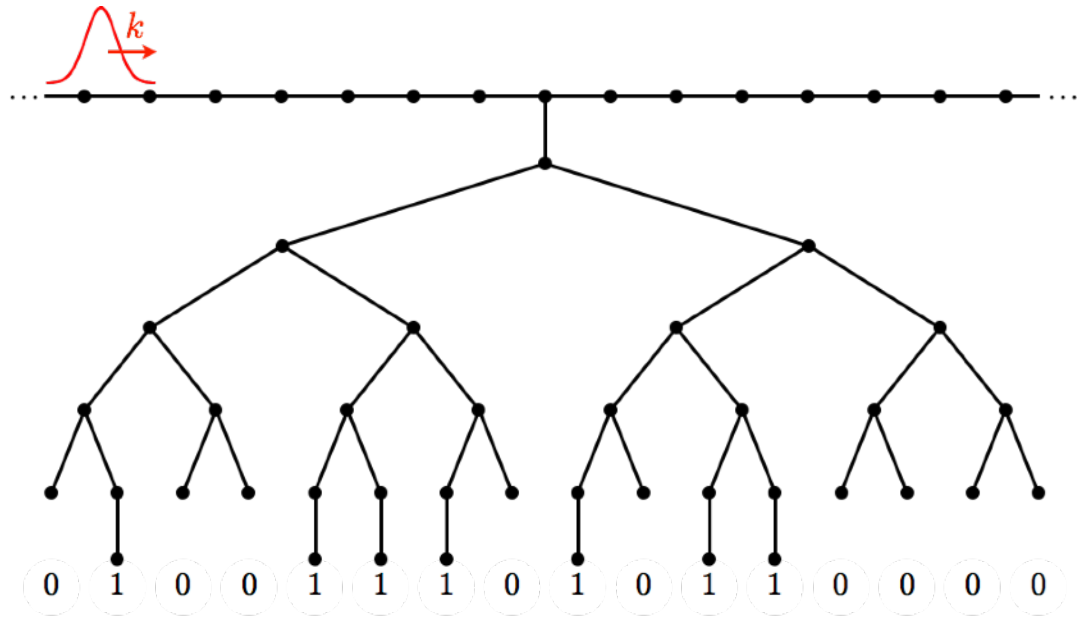
Implementing quantum algorithms



adiabatic optimization



exponential speedup by quantum walk



evaluating Boolean formulas

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

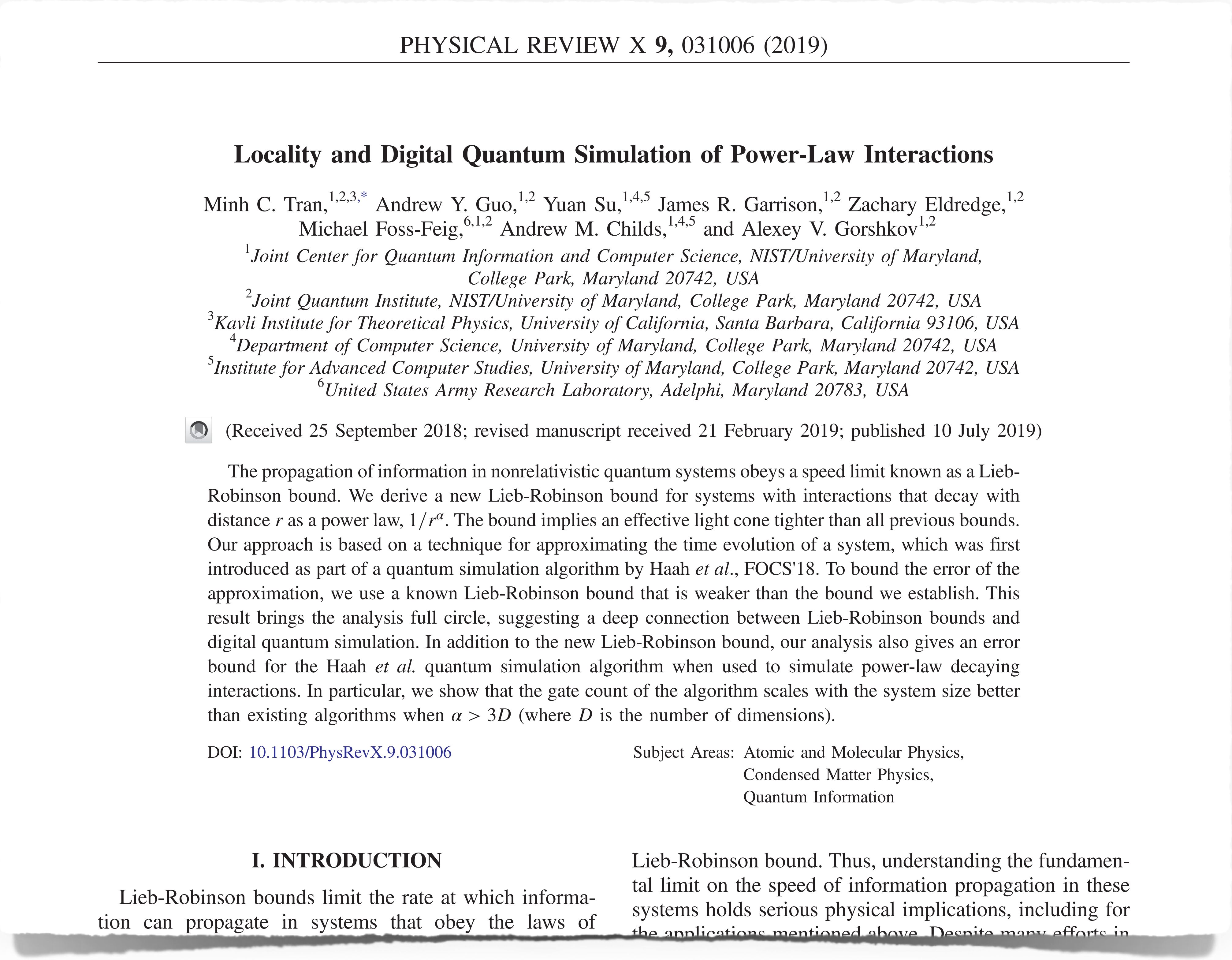
linear/differential equations, convex optimization

Simulation as a theoretical tool

Quantum simulation algorithms provide convenient descriptions of the time evolution operator.

Can we use them as theoretical tools to reason about physics?

Example: Lieb-Robinson bounds from quantum simulation algorithms



PHYSICAL REVIEW X **9**, 031006 (2019)

Locality and Digital Quantum Simulation of Power-Law Interactions

Minh C. Tran,[1,2,3,*] Andrew Y. Guo,[1,2] Yuan Su,[1,4,5] James R. Garrison,[1,2] Zachary Eldredge,[1,2]
Michael Foss-Feig,[6,1,2] Andrew M. Childs,[1,4,5] and Alexey V. Gorshkov[1,2]

[1] *Joint Center for Quantum Information and Computer Science, NIST/University of Maryland, College Park, Maryland 20742, USA*
[2] *Joint Quantum Institute, NIST/University of Maryland, College Park, Maryland 20742, USA*
[3] *Kavli Institute for Theoretical Physics, University of California, Santa Barbara, California 93106, USA*
[4] *Department of Computer Science, University of Maryland, College Park, Maryland 20742, USA*
[5] *Institute for Advanced Computer Studies, University of Maryland, College Park, Maryland 20742, USA*
[6] *United States Army Research Laboratory, Adelphi, Maryland 20783, USA*

(Received 25 September 2018; revised manuscript received 21 February 2019; published 10 July 2019)

The propagation of information in nonrelativistic quantum systems obeys a speed limit known as a Lieb-Robinson bound. We derive a new Lieb-Robinson bound for systems with interactions that decay with distance r as a power law, $1/r^\alpha$. The bound implies an effective light cone tighter than all previous bounds. Our approach is based on a technique for approximating the time evolution of a system, which was first introduced as part of a quantum simulation algorithm by Haah *et al.*, FOCS'18. To bound the error of the approximation, we use a known Lieb-Robinson bound that is weaker than the bound we establish. This result brings the analysis full circle, suggesting a deep connection between Lieb-Robinson bounds and digital quantum simulation. In addition to the new Lieb-Robinson bound, our analysis also gives an error bound for the Haah *et al.* quantum simulation algorithm when used to simulate power-law decaying interactions. In particular, we show that the gate count of the algorithm scales with the system size better than existing algorithms when $\alpha > 3D$ (where D is the number of dimensions).

DOI: 10.1103/PhysRevX.9.031006

Subject Areas: Atomic and Molecular Physics,
Condensed Matter Physics,
Quantum Information

I. INTRODUCTION

Lieb-Robinson bounds limit the rate at which information can propagate in systems that obey the laws of

Lieb-Robinson bound. Thus, understanding the fundamental limit on the speed of information propagation in these systems holds serious physical implications, including for the applications mentioned above. Despite many efforts in

Quantum dynamics

The dynamics of a quantum system are determined by its *Hamiltonian* H, a Hermitian operator whose spectrum describes the energy levels of the system.

For time-independent H: $$i\frac{\mathrm{d}}{\mathrm{d}t}|\psi(t)\rangle = H|\psi(t)\rangle \quad \Rightarrow \quad |\psi(t)\rangle = e^{-iHt}|\psi(0)\rangle$$

For time-dependent $H(t)$: $$i\frac{\mathrm{d}}{\mathrm{d}t}|\psi(t)\rangle = H(t)|\psi(t)\rangle \quad \Rightarrow \quad |\psi(t)\rangle = \mathcal{T}e^{-i\int_0^t H(\tau)\,\mathrm{d}\tau}|\psi(0)\rangle$$

The Hamiltonian simulation problem

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

Standard measure of error: worst-case ℓ_2 distance

$$\max_{|\psi(0)\rangle} \| |\psi(t)\rangle - |\psi_{\mathrm{sim}}\rangle \| = \| e^{-iHt} - U_{\mathrm{sim}} \|$$

initial state

$e^{-iHt}|\psi(0)\rangle$

simulation output state

simulated unitary operation

Alternatives: state-dependent, subspace-dependent, or average-case bounds; diamond norm or other distance measures

Goal: Understand the cost of simulation as a function of t, ϵ, and parameters of H.

(cost could be circuit size, circuit depth, queries, …)

Also important for applications but not covered in this talk: state preparation, measurement, representing fermions/bosons using qubits

Local Hamiltonians

To make this problem well defined, we must indicate how the Hamiltonian is specified.

One natural model: k-local Hamiltonians

We say H is k-local if it has the form $H = \sum_{j=1}^{L} H_j$ where each H_j acts on k qubits

Advantages:

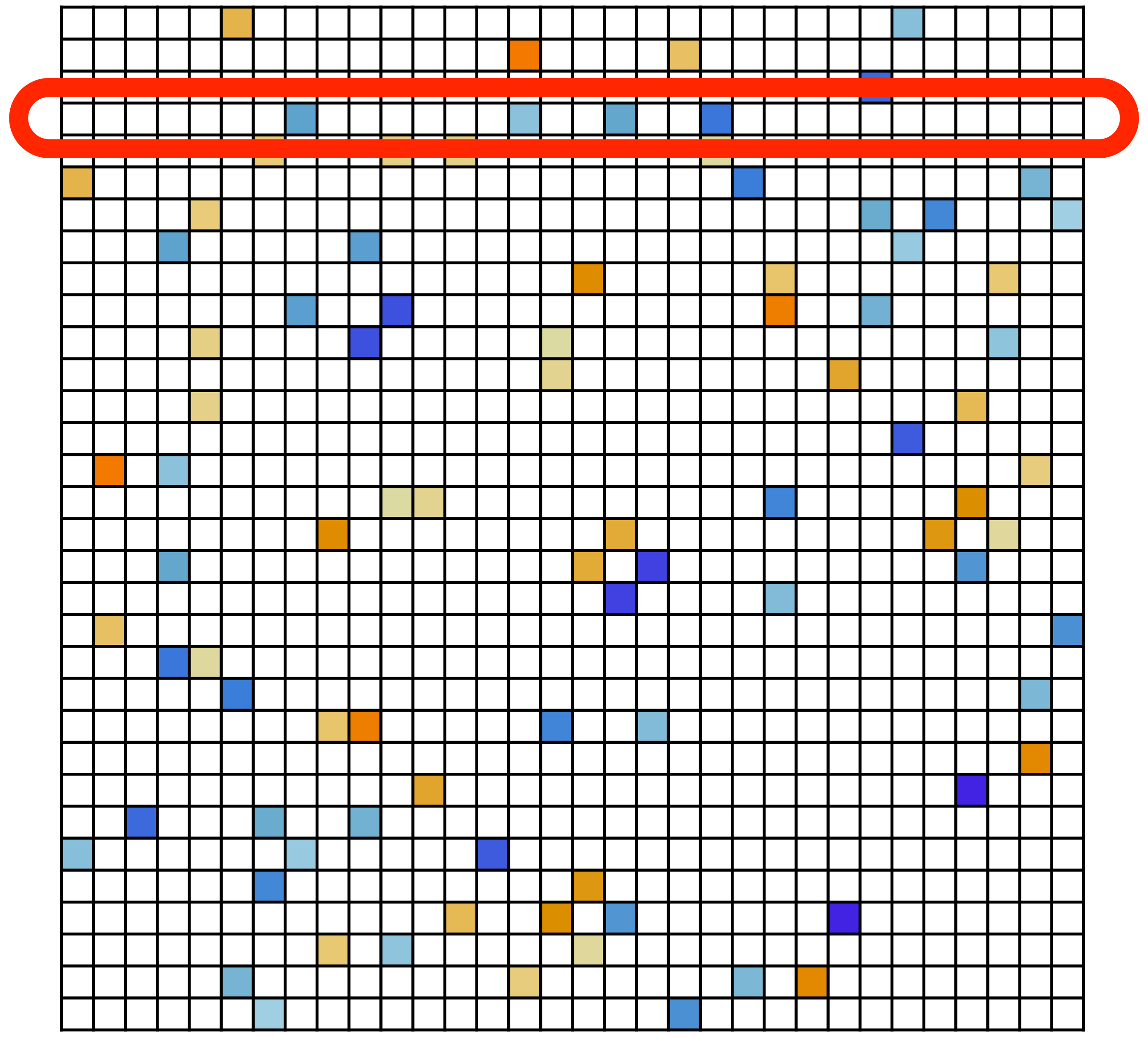
- Physics is local, so many natural Hamiltonians are k-local
- For $k = O(\log n)$, any k-local Hamiltonian has an efficient explicit description (at most $\binom{n}{k}$ terms, each acting on a space of dimension 2^k)
- Provides a decomposition that is useful in simulation algorithms

Sparse Hamiltonians

We say H is d-sparse if it has at most d nonzero entries per row

We can simulate sparse Hamiltonians that are *row-computable*: in any given row, the location of the jth nonzero entry and its value can be computed efficiently (locations and values provided by black boxes)

$H =$



Advantages:

- Generalizes local Hamiltonians (a k-local Hamiltonian with L terms is d-sparse with $d = 2^k L$)
- Can still be efficiently simulated
- Describes other Hamiltonians with sparse connectivity graphs (quantum walks, some chemistry simulations, $\ldots$)
- Black-box description facilitates query lower bounds

Complexity

How hard is quantum simulation?

A classical computer cannot even represent the state efficiently, so surely general simulation is classically hard.

But a quantum computer cannot produce a complete description of the state. How can we make a fair comparison?

Suppose we provide succinct descriptions of

- the initial state (say, as a small-depth quantum circuit acting on the $|0\rangle$ state),
- the Hamiltonian (say, as a k-local operator), and
- a final measurement (say, measurements of the individual qubits in some basis).

Then a quantum computer can efficiently sample from the output distribution.

How hard is this for a classical computer?

Quantum simulation is BQP-hard

Any quantum circuit can be efficiently implemented by the dynamics of a local Hamiltonian.

[Feynman 85]

Given a circuit $U = U_k \dots U_2 U_1$, append a "clock" register and define

$$H := \sum_{j=1}^{k} \alpha_j H_j \quad \text{where} \quad H_j := U_j \otimes |j\rangle\langle j-1| + U_j^\dagger \otimes |j-1\rangle\langle j|$$

Advancing the clock from $j-1$ to j corresponds to applying the gate U_j

With an appropriate choice of α_j, we have $e^{-iHk}|\psi\rangle \otimes |0\rangle = U|\psi\rangle \otimes |k\rangle$

If we represent the clock state $|j\rangle$ in unary, then we can implement $|j\rangle\langle j-1|$ with a 2-qubit operation, so if the circuit consists of 2-qubit gates, H is 4-local.

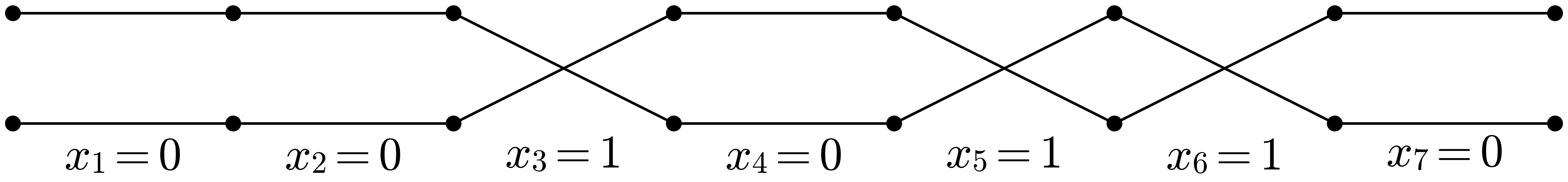
No fast forwarding

No fast-forwarding theorem: Simulating Hamiltonian dynamics for time t requires $\Omega(t)$ gates.

Proof is based on the hardness of computing parity: given a black box for the bits of a string $x \in \{0,1\}^n$, computing $x_1 \oplus \cdots \oplus x_n$ requires $\Omega(n)$ quantum queries.

[Beals, Buhrman, Cleve, Mosca, de Wolf 98; Farhi, Goldstone, Gutmann, Sipser 98]

Construct a sparse Hamiltonian that encodes the parity:



Simulation for time $t = O(n)$ can be used to determine the parity (again, choosing weights to ensure perfect transfer), so a general-purpose simulation method must use $\Omega(t)$ gates.

[Berry, Ahokas, Cleve, Sanders 05]

Product formulas

Product formula simulation

Suppose we want to simulate $H = \sum_{j=1}^{L} H_j$

Combine individual simulations with the Lie product formula. E.g., with two terms:

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

$$e^{-i(A+B)t} = 1 - i(A+B)t - \tfrac{1}{2}(A^2 + AB + BA + B^2)t^2 + O(t^3)$$

$$\begin{aligned} e^{-iAt}e^{-iBt} &= (1 - iAt - \tfrac{1}{2}A^2t^2 + O(t^3))(1 - iBt - \tfrac{1}{2}B^2t^2 + O(t^3)) \\ &= 1 - i(A+B)t - \tfrac{1}{2}(A^2 + 2AB + B^2)t^2 + O(t^3) \end{aligned}$$

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

To ensure error at most ϵ, take $r = O(t^2/\epsilon)$

[Lloyd 96]

Higher-order product formulas

To get a better approximation, use higher-order formulas.

E.g., second order: $S_2(t) := e^{-iAt/2} e^{-iBt} e^{-iAt/2} = e^{-i(A+B)t} + O(t^3)$

Can construct expansions to arbitrarily high order [Suzuki 92]:

$$\text{let } S_{2k}(t) := S_{2k-2}(p_k t)^2 S_{2k-2}(q_k t) S_{2k-2}(p_k t)^2$$

Then you can find numbers p_k and q_k (as functions of k) so that

$$S_{2k}(t) = e^{-i(A+B)t} + O(t^{2k+1})$$

Using the order-$2k$ expansion for an L-term Hamiltonian, the number of exponentials required for an approximation with error at most ϵ is at most [Berry, Ahokas, Cleve, Sanders 07]

$$5^{2k} L^2 ht \left(\tfrac{Lht}{\epsilon}\right)^{1/2k} \qquad h := \max_j \|H_j\|$$

[Childs 04; Berry, Ahokas, Cleve, Sanders 07]

Improved error analysis

Numerics suggest that product formula error bounds based on Taylor series expansion can be very loose in practice.

Alternative: *local error analysis* provides convenient integral representations of the error

[Descombes, Thalhammer 10]

Example (first order):

$$e^{-iBt}e^{-iAt} - e^{-i(A+B)t} = \int_0^t \mathrm{d}\tau_1 \int_0^{\tau_1} \mathrm{d}\tau_2 \, e^{-i(A+B)(t-\tau_1)} e^{i(\tau_2-\tau_1)B} [A,B] e^{-i\tau_2 B} e^{-i\tau_1 A}$$

Therefore $\|e^{-iBt}e^{-iAt} - e^{-i(A+B)t}\| \le \|[A,B]\| t^2$

Advantages:

- no explicit sum over higher-order terms
- commutator scaling
- can offer much better performance, both asymptotically and in practice (more on this later)

[Childs, Su 19]

A theory of Trotter error

Local error analysis can be generalized to give tight bounds on the error of product formula approximations depending on commutators of any number of terms.

Theorem. A pth-order product formula approximates the evolution of $H = \sum_{j=1}^{L} H_j$ with additive error $O(\alpha t^{p+1})$ where

$$\alpha := \sum_{j_1,\ldots,j_{p+1}} \big\|[H_{j_{p+1}}, [\cdots, [H_{j_2}, H_{j_1}]\cdots]]\big\|.$$

Therefore $O(L\alpha^{1/p}t^{1+1/p})$ gates suffice to simulate H for time t with constant accuracy.

This gives much tighter rigorous analysis of product formulation simulations, among other applications.

[Childs, Su, Tran, Wiebe, Zhu 19]

Post-Trotter methods

Linear-time simulation

Complexity of $2k$th-order product formula simulation is $O(5^{2k}t^{1+1/2k})$ (superlinear in t).

Can we give an algorithm with complexity precisely $O(t)$?

Pro: Systems simulate their own dynamics in real time!

Con: Mismatch between continuous-time dynamics and the discrete-time circuit model.

Hamiltonian simulation by discrete-time quantum walk

Quantum walk corresponding to H

Alternately reflect about $\mathrm{span}\{|\psi_j\rangle\}_{j=1}^{N}$, where

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

and swap the two registers.

If H is sparse, this walk is easy to implement.

Spectral theorem: 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).

Simulation by phase estimation

$$\begin{aligned} |\lambda\rangle &\mapsto |\lambda\rangle|\widetilde{\arcsin\lambda}\rangle && \text{(phase estimation)} \\ &\mapsto e^{-i\lambda t}|\lambda\rangle|\widetilde{\arcsin\lambda}\rangle && \\ &\mapsto e^{-i\lambda t}|\lambda\rangle && \text{(inverse phase est)} \end{aligned}$$

Theorem: $O(t/\sqrt{\epsilon})$ steps of this walk suffice to simulate H for time t with error at most ϵ.

[Childs10; Berry, Childs 12]

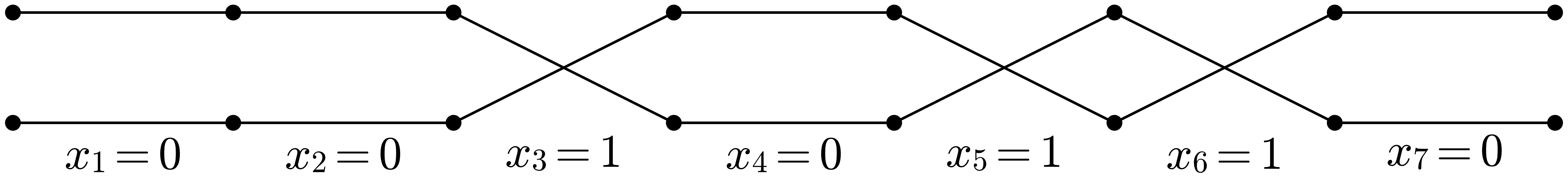
High-precision simulation?

Can we improve the dependence on ϵ?

Many approximate computations can be done with complexity $\mathrm{poly}(\log(1/\epsilon))$:

- computing numerical constants (e.g., π)
- boosting a bounded-error subroutine
- Solovay-Kitaev circuit synthesis
- and more…

Lower bound (based on the *unbounded-error* query complexity of parity): $\Omega(\frac{\log(1/\epsilon)}{\log\log(1/\epsilon)})$



Quantum walk simulation: $O(1/\sqrt{\epsilon})$

Product formulas ($2k$th order): $O(5^{2k}\epsilon^{-2k})$

Can we do better?

Hamiltonian simulation by linear combinations of unitaries

Main idea: Directly implement the series

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

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

Write $H = \sum_\ell \alpha_\ell H_\ell$ with H_ℓ unitary.

Then

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

is a linear combination of unitaries.

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:

- 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*

Query complexity: $O\left(t \frac{\log(t/\epsilon)}{\log\log(t/\epsilon)}\right)$

[Berry, Childs, Cleve, Kothari, Somma 14 & 15]

Partially implementing U

Given $U = \sum_j \beta_j V_j$

Let B be an operation that maps $|0\rangle$ to $\frac{1}{\sqrt{s}} \sum_j \sqrt{\beta_j}|j\rangle$, where $s := \sum_j |\beta_j|$

Let $W := B^\dagger \operatorname{select}(V) B$ where $\operatorname{select}(V) := \sum_j |j\rangle\langle j| \otimes V_j$

Then $$(\langle 0| \otimes I)W(|0\rangle \otimes |\psi\rangle) = \frac{1}{\sqrt{s}}(\langle 0| \otimes I)B^\dagger \operatorname{select}(V) \sum_j \sqrt{\beta_j}|j\rangle|\psi\rangle$$

$$= \frac{1}{\sqrt{s}}(\langle 0| \otimes I)B^\dagger \sum_j \sqrt{\beta_j}|j\rangle V_j|\psi\rangle$$

$$= \frac{1}{s}\sum_j \beta_j V_j|\psi\rangle = \frac{1}{s}U|\psi\rangle$$

so $W|0\rangle|\psi\rangle = \frac{1}{s}|0\rangle \otimes U|\psi\rangle + \sqrt{1 - \frac{1}{s^2}}|\Phi\rangle$ for some state $|\Phi\rangle$

[Berry, Childs, Cleve, Kothari, Somma 14 & 15]

Obvious amplitude amplification

We would like to boost the amplitude of the $U|\psi\rangle$ piece.
This is like amplitude amplification, but with an unknown initial state.

Define $P := |0\rangle\langle 0|$ and $R := (I - 2P) \otimes I$

Then $(\langle 0| \otimes I) W R W^\dagger R W (|0\rangle \otimes I)$

$$
\begin{aligned}
&= (\langle 0| \otimes I)\big(WW^\dagger W - 2WW^\dagger PW - 2WPW^\dagger W + 4WPW^\dagger PW\big)(|0\rangle \otimes I) \\
&= (\langle 0| \otimes I)\big(-3W + 4WPW^\dagger PW\big)(|0\rangle \otimes I) \\
&= -\frac{3}{s}U + \frac{4}{s^3}UU^\dagger U \\
&\approx -\left(\frac{3}{s} - \frac{4}{s^3}\right)U \quad \text{since } U \text{ is approximately unitary}
\end{aligned}
$$

For $s = 2$, the amplitude is boosted from $1/2$ to 1, as in 1-out-of-4 search.

[Berry, Childs, Cleve, Kothari, Somma 14 & 15]

Tradeoff between t and ϵ

Combining known lower bounds on the complexity of simulation as a function of t and ϵ gives

$$\Omega\Bigl(t + \frac{\log \frac{1}{\epsilon}}{\log\log \frac{1}{\epsilon}}\Bigr) \quad \text{vs. upper bound of} \quad O\Bigl(t \frac{\log \frac{t}{\epsilon}}{\log\log \frac{t}{\epsilon}}\Bigr)$$

An alternative method for implementing a linear combination of unitary operations, *quantum signal processing*, gives an optimal tradeoff. [Low, Chuang 16, 17]

Main idea: Encode the eigenvalues of H in a two-dimensional subspace; use a carefully-chosen sequence of single-qubit rotations to manipulate those eigenvalues.

Computing the rotation angles is challenging, but can be done efficiently (classically) [Haah 18]. Recent approaches are faster [Dong, Meng, Whaley, Lin 20; Chao, Ding, Gilyén, Huang, Szegedy 20; Ying 22].

Quantum signal processing (and more general *quantum singular value transformation*) gives a powerful framework for designing other quantum algorithms [Gilyén, Su, Low, Wiebe 19].

Block encoding

We say U is a block encoding of a matrix A if

$$U = \begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix} = |\mathbf{0}\rangle\langle\mathbf{0}| \otimes A + \cdots \qquad \text{(requires } \|A\| \leq 1\text{)}$$

where $|\mathbf{0}\rangle$ denotes the first computational basis state of an ancilla register.

Example: In the LCU approach, $W := B^\dagger \operatorname{select}(V) B$ is a block encoding of U/s.

Can directly give an efficient block encoding of a sparse matrix A. If

$$R\colon |0\rangle|0\rangle|i\rangle \mapsto |0\rangle \frac{1}{\sqrt{d}} \sum_{k=1}^{N} \sqrt{A_{ik}^*}|i\rangle|k\rangle + |1\rangle|i\rangle|\mu_i\rangle$$

$$C\colon |0\rangle|0\rangle|j\rangle \mapsto |0\rangle \frac{1}{\sqrt{d}} \sum_{\ell=1}^{N} \sqrt{A_{\ell j}}|\ell\rangle|j\rangle + |2\rangle|j\rangle|\nu_j\rangle$$

then $R^\dagger C$ is a block encoding of A/d.

Quantum signal processing

Problem: Given a block encoding of A, produce a block encoding of $f(A)$.

Block encoding of a scalar, $A = x$: $W(x) := \begin{pmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{pmatrix} = e^{i\arccos(x)\sigma_x}$.

Consider the gate sequence $W_\Phi(x) := e^{i\phi_0\sigma_z} W(x) e^{i\phi_1\sigma_z} W(x) \cdots W(x) e^{i\phi_k\sigma_z}$

What functions of W can we realize by some choice of $\Phi = (\phi_0, \ldots, \phi_k)$?

Lemma. *There exists $\Phi \in \mathbb{R}^{k+1}$ such that*

$$W_\Phi(x) = \begin{pmatrix} f(x) & ig(x)\sqrt{1-x^2} \\ ig^*(x)\sqrt{1-x^2} & f^*(x) \end{pmatrix}$$

if and only if $f, g \in \mathbb{C}[x]$ satisfy

(i) $\deg(f) \le k$ *and* $\deg(g) \le k-1$,
(ii) f *has parity* $k \bmod 2$ *and* g *has parity* $k-1 \bmod 2$, *and*
(iii) $\forall x \in [-1,1]$, $|f(x)|^2 + (1-x^2)|g(x)|^2 = 1$.

[Low, Yoder, Chuang 16; Gilyén, Su, Low, Wiebe 19]

Qubitization

To perform QSP on a high-dimensional block encoding, we can map it to a qubit by *qubitization*.

Let U be a block encoding of A, so $\Pi U \Pi = |\mathbf{0}\rangle\langle\mathbf{0}| \otimes A$ where $\Pi := |\mathbf{0}\rangle\langle\mathbf{0}| \otimes I$

Let $|\lambda\rangle := |\mathbf{0}\rangle \otimes |\psi_\lambda\rangle$ where $|\psi_\lambda\rangle$ is an eigenvector of A

Then U maps $\mathrm{span}\{|\lambda\rangle, (I-\Pi)U^\dagger|\lambda\rangle\}$ to $\mathrm{span}\{|\lambda\rangle, (I-\Pi)U|\lambda\rangle\}$, with

$$U \cong \bigoplus_\lambda \begin{pmatrix} \lambda & \sqrt{1-\lambda^2} \\ \sqrt{1-\lambda^2} & -\lambda \end{pmatrix}$$

Thus we can perform QSP on A by a sequence

$$e^{i\theta_1(2\Pi-1)}U^\dagger e^{i\theta_2(2\Pi-1)}U e^{i\theta_3(2\Pi-1)}U^\dagger e^{i\theta_4(2\Pi-1)}U \cdots e^{i\theta_{k-1}(2\Pi-1)}U^\dagger e^{i\theta_k(2\Pi-1)}U$$

to realize any function of A satisfying the conditions of the previous lemma.

[Low, Chuang 19]

Hamiltonian simulation by quantum signal processing

Task: Given a block encoding of H, produce a block encoding of e^{-iHt}

Jacobi-Anger formula: $$e^{itx} = J_0(t) + 2\sum_{k=1}^{\infty} i^k J_k(t) T_k(x)$$
$$\approx J_0(t) + 2\sum_{k=1}^{K} i^k J_k(t) T_k(x)$$

Approximation is within ϵ if we take $K = O\left(t + \frac{\ln(1/\epsilon)}{\ln(e + \ln(1/\epsilon)/t)}\right)$

For technical reasons, QSP can only give a block encoding of $e^{-iHt}/2$, but we can use oblivious amplitude amplification to boost this close to e^{-iHt}

This query complexity achieves the optimal tradeoff between t and ϵ!

[Low, Chuang 17]

Special topics

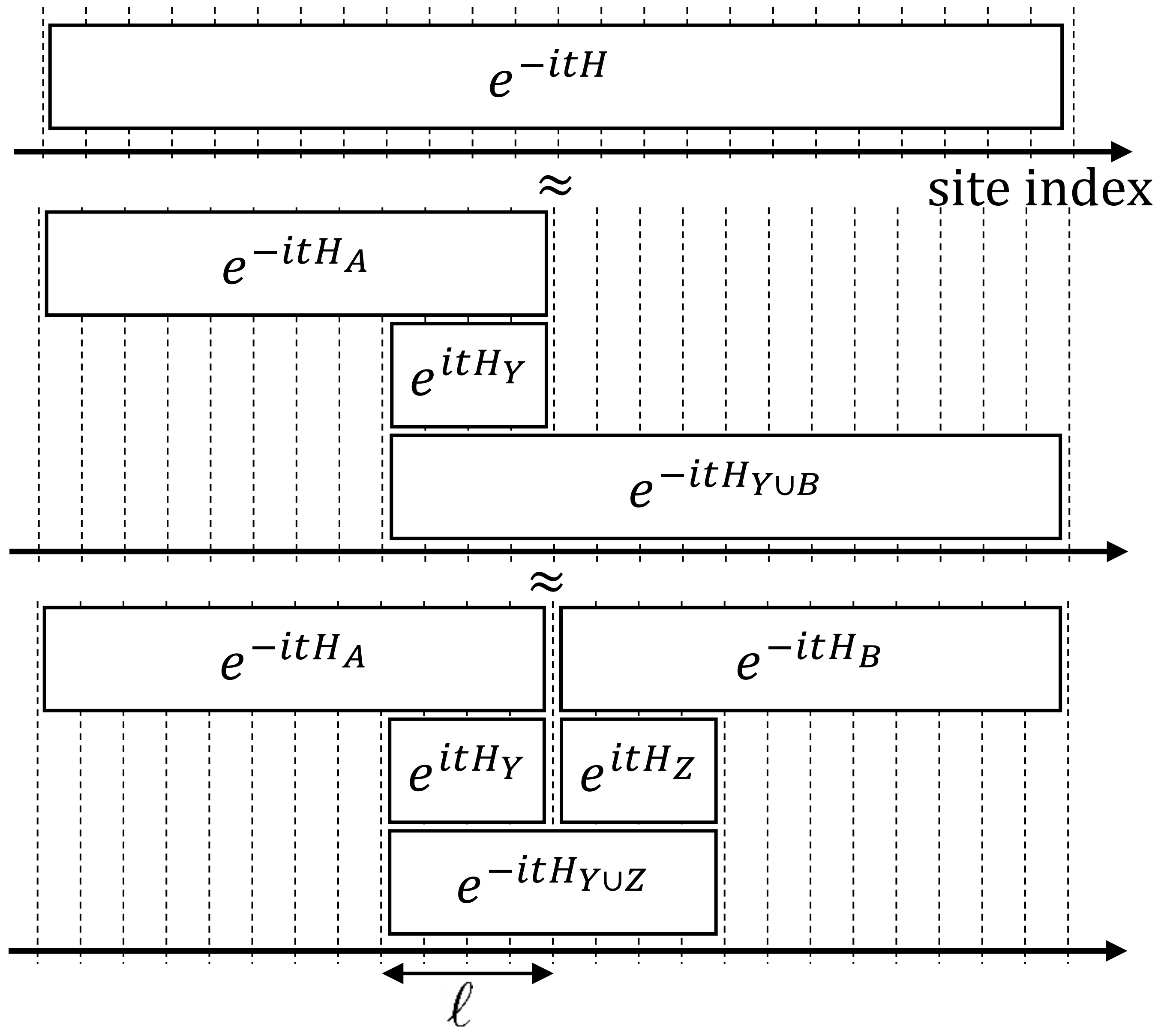
Lattice Hamiltonians

We've focused on the complexity as a function of t (evolution time) and ϵ (precision).
What about the dependence on system size?

Consider n spins with nearest-neighbor interactions on a grid of fixed dimension. To simulate for constant time, previous methods (LCU, QSP, high-order PF with standard analysis) give:

- total number of gates: $O(n^2)$
- circuit depth (execution time with parallel gates): $O(n)$

Execution time should not have to be extensive!



Can give a simulation with $\tilde{O}(n)$ gates, $\tilde{O}(1)$ depth (optimal!) [Haah, Hastings, Kothari, Low 18]

- Lieb-Robinson bound limits the speed of propagation
- Simulate small regions with negative-time evolutions to correct the boundaries

Product formula simulation of lattice Hamiltonians

Using standard error bounds for product formulas, simulation cost is $O\Big(5^{2k}L^2ht\big(\frac{Lht}{\epsilon}\big)^{1/2k}\Big)$

For an n-site lattice Hamiltonian of fixed dimension, $L=O(n)$, giving cost at least quadratic in n

Using improved analysis with commutator scaling, cost for an order-p formula is

$$O(L\alpha^{1/p}t^{1+1/p}), \qquad \alpha := \sum_{j_1,\ldots,j_{p+1}} \big\|[H_{j_{p+1}},[\cdots,[H_{j_2},H_{j_1}]\cdots]]\big\|.$$

For constant p, we have $\alpha = O(n)$ since for any fixed j_1, there are $O(1)$ nonzero nested commutators

This gives cost $O((nt)^{1+1/p})$, nearly matching the HHKL complexity for large p

Interaction picture simulation

Suppose we want to simulate $H = A + B$ where A is much larger than B, but easy to simulate

Product formula simulation will pay a cost that scales with $\|A\|$

Transform to *interaction picture*: $|\psi_I(t)\rangle := e^{iAt}|\psi(t)\rangle$

$$\begin{aligned} i\frac{\mathrm{d}}{\mathrm{d}t}|\psi_I(t)\rangle &= -A|\psi_I(t)\rangle + e^{iAt}H|\psi(t)\rangle \\ &= e^{iAt}Be^{-iAt}|\psi_I(t)\rangle \end{aligned}$$

so $|\psi_I(t)\rangle$ evolves according to the interaction picture Hamiltonian $H_I(t) := e^{iAt}Be^{-iAt}$

Simulating time-dependent Hamiltonians is more complicated that the time-independent case…

… but there are methods for doing this, and it can sometimes give a favorable tradeoff

[Low, Wiebe 18]

Randomized simulation

Another approach to speeding up simulation: introduce classical randomness

Example:

$$\begin{aligned} e^{-i(A+B)t} &= I - i(A+B)t - \tfrac{1}{2}(A^2 + AB + BA + B^2)t^2 + O(t^3) \\ e^{-iAt}e^{-iBt} &= I - i(A+B)t - \tfrac{1}{2}(A^2 + 2AB + B^2)t^2 + O(t^3) \\ e^{-iBt}e^{-iAt} &= I - i(A+B)t - \tfrac{1}{2}(A^2 + 2BA + B^2)t^2 + O(t^3) \end{aligned}$$

$$\Downarrow$$

$$\tfrac{1}{2}(e^{-iAt}e^{-iBt} + e^{-iBt}e^{-iAt}) = e^{-i(A+B)t} + O(t^3)$$

[Zhang 12]

Mixing lemma [Campbell 17, Hastings 17]: Error of the average operation is linear in the average error, quadratic in the error of individual operations.

Randomly permuting terms in a higher-order product formula also improves the approximation (though not the order of the formula), though in practice the improvement is typically small.

[Childs, Ostrander, Su 18]

qDRIFT

Alternative randomized simulation strategy: choose which term to simulate with a probability that depends on its weight

Suppose $H = \sum_j h_j H_j$ with $\|H_j\| = 1$; let $\lambda := \sum_j h_j$

Repeat $\nu = O(\lambda^2 t^2/\epsilon)$ times: sample H_j with probability h_j/λ and apply $e^{-i\lambda H_j t/\nu}$

Theorem: This gives a simulation that is accurate to within diamond norm distance ϵ.

Cost $O(\lambda^2 t^2/\epsilon)$, vs. $O\big(L^3(\max_j h_j)^2 t^2/\epsilon\big)$ for first-order Trotter

Con: Limited to first-order approximation; higher-order methods are better asymptotically

Pro: Simulation cost has no explicit dependence on the number of terms and may outperform deterministic product formulas non-asymptotically, especially when the terms have widely varying size

[Campbell 18]

Average-case simulation

Standard analysis of product formula error gives a worst-case guarantee (i.e., for any input state). For a pth-order formula, the error is $O(\alpha t^{p+1})$ where

$$\alpha := \sum_{j_1,\ldots,j_{p+1}} \left\| [H_{j_{p+1}}, [\cdots, [H_{j_2}, H_{j_1}] \cdots]] \right\|.$$

We can give a tighter bound if the input state is chosen at random from some distribution. For a 1-design ensemble on a d-dimensional system, the average error is $O(\alpha_F t^{p+1})$ where

$$\alpha_F := \frac{1}{\sqrt{d}} \sum_{j_1,\ldots,j_{p+1}} \left\| [H_{j_{p+1}}, [\cdots, [H_{j_2}, H_{j_1}] \cdots]] \right\|_F.$$

This is no worse, but can sometimes give a much better bound (for the same algorithm!).

For a nearest-neighbor n-qubit Hamiltonian, this reduces the error by a factor of $\sqrt{n}$.

[Zhao, Zhou, Shaw, Li, Childs 22]

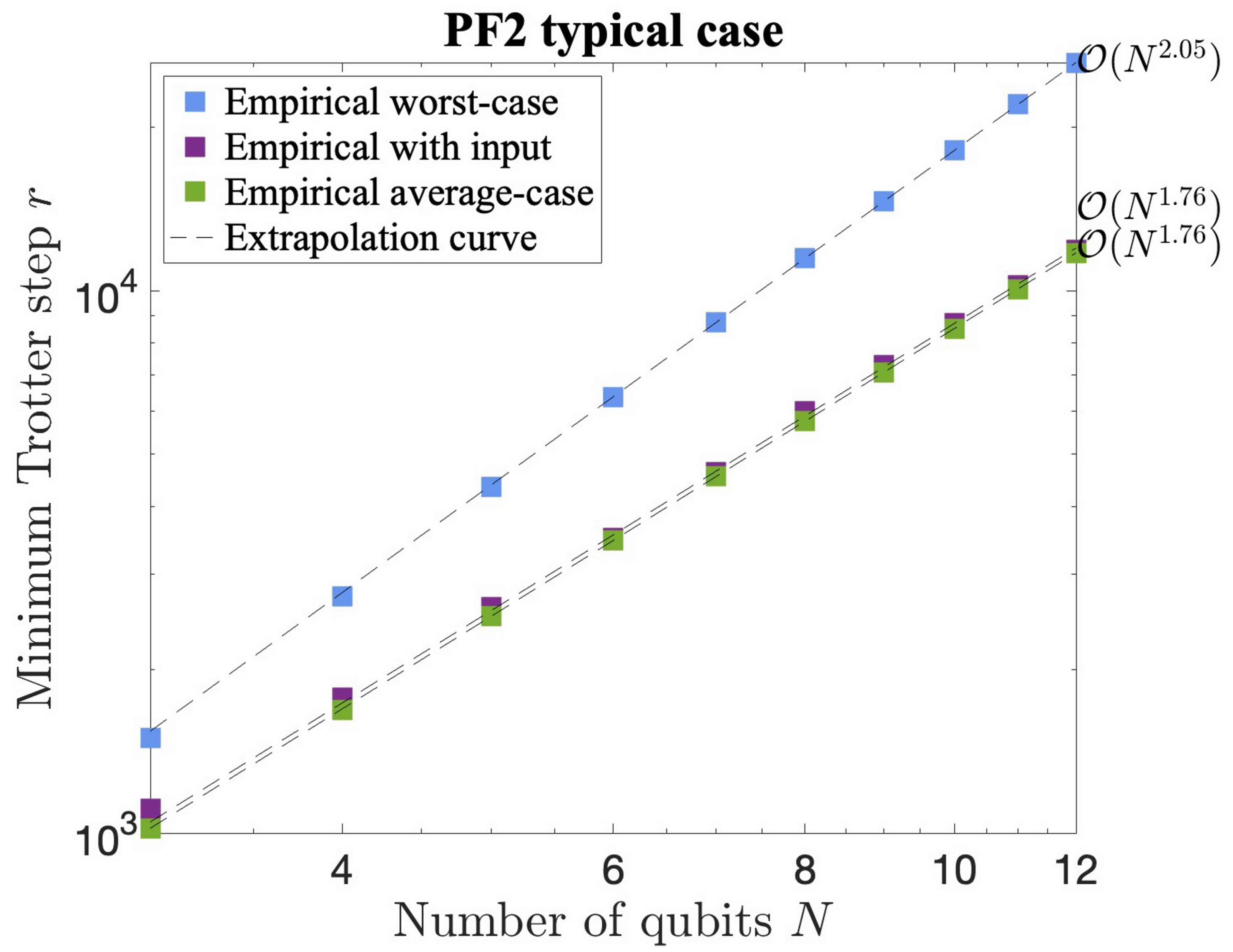
Entanglement accelerates quantum simulation

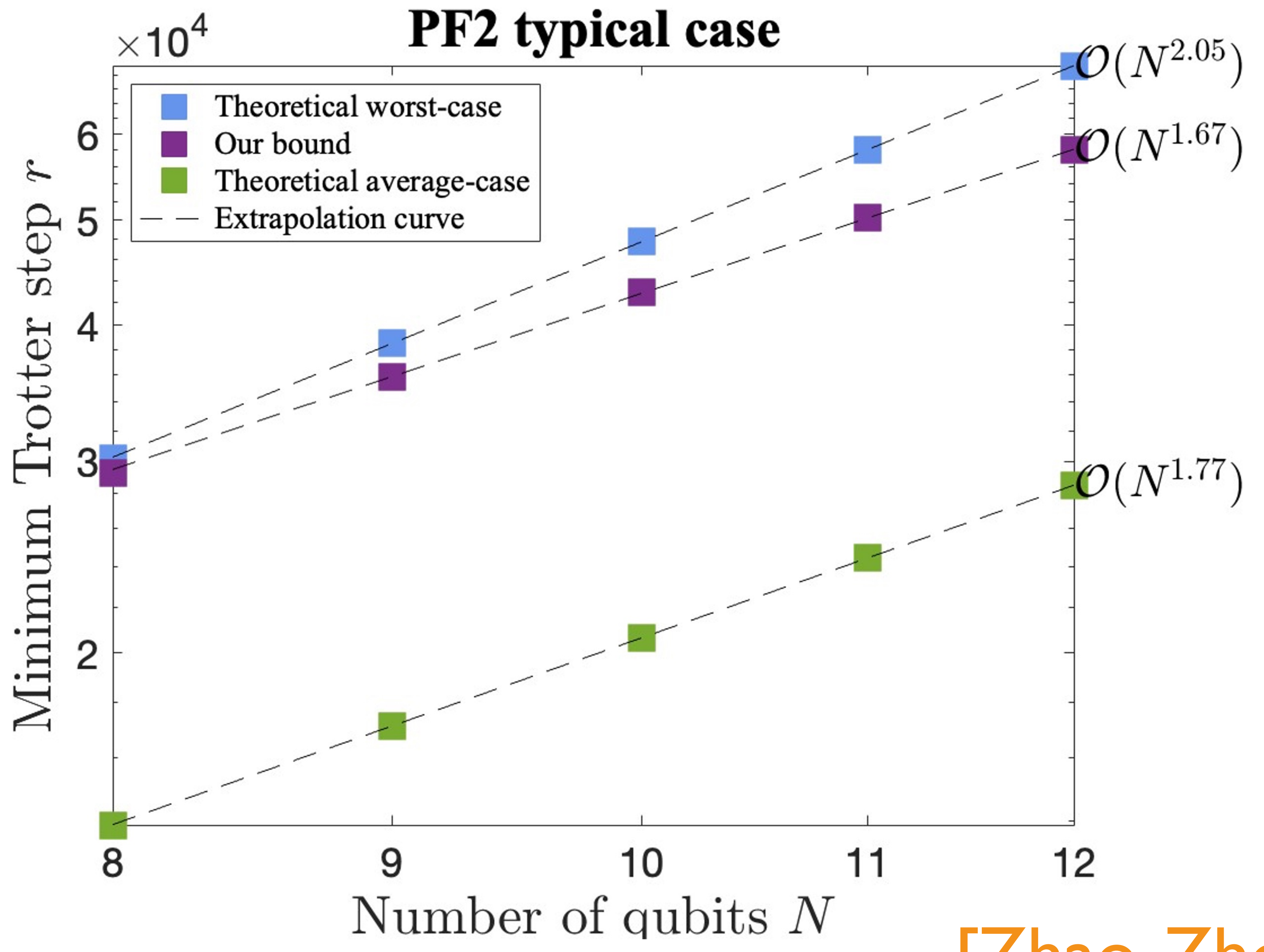
Intuitively, we should be be able to get similar error bounds for specific "typical" input states.

Similar error scaling to the average case holds provided the local marginals of the state (on the qubits of leading-order error terms of the product formula) are close to maximally mixed.

This is generic. Specifically, it holds if the state is highly entangled.

If the state were *not* highly entangled, it would be easy to simulate classically!





[Zhao, Zhou, Childs 24]

Outlook

Outlook: Theory

Improve quantum algorithms for Hamiltonian simulation

- Tighter error bounds for product formulas (e.g., going beyond the triangle inequality)
- Better product formulas/limitations on improvements to product formulas
- Understand the (t, ϵ) tradeoff for time-dependent Hamiltonians
- Faster simulation methods for structured Hamiltonians (including cases where fast-forwarding is possible)
- Optimized implementations of simulations; more efficient synthesis of the QSP circuit
- Alternative metrics: state-dependent simulation, simulation within a subspace, average-case simulation, …

Quantum simulation as an algorithmic tool

- Linear algebra in Hilbert space: linear systems, differential equations, convex optimization, …
- Find new quantum algorithms based on Hamiltonian simulation
- Use product formula error bounds to analyze classical algorithms (e.g., quantum Monte Carlo)

Quantum simulation as a theoretical tool

- Lieb-Robinson bounds
- Entanglement area laws
- Scrambling
- …

Outlook: Applications

Develop applications of quantum simulation to physics/chemistry

- Quantum chemistry
- Condensed matter
- Nuclear physics
- Particle physics

Explore prospects for near-term implementations

- Resource estimates under realistic hardware constraints
- Can we perform classically hard simulations without invoking fault tolerance?
- Noise-tolerant algorithms

Major goal: Develop (and then implement!) concrete end-to-end applications of quantum simulation with compelling evidence for speedup

Further reading

- Lecture notes on quantum algorithms (part V): https://www.cs.umd.edu/~amchilds/qa/
- Simulation gate counts: arXiv:1711.10980
- Product formula error: arXiv:1912.08854
- Martyn-Rossi-Tan-Chuang survey on QSP: arXiv:2105.02859
- Lin Lin lecture notes: arXiv:2201.08309

Lectures and seminars

- John Preskill simulation lecture: https://www.youtube.com/watch?v=t-qUR5kweCE
- Nathan Wiebe survey talk: https://www.youtube.com/watch?v=X4gegxIuh1o
- Yuan Su on Trotter error: https://www.youtube.com/watch?v=3AVnKJ3uCrU
- Robin Kothari on LCU/QSP: https://www.youtube.com/watch?v=mWg56DxtDy0
- András Gilyén on QSVT: https://www.youtube.com/watch?v=SMdLc36ysJE
- Longer version of this presentation: https://www.cs.umd.edu/~amchilds/talks/sim.pdf