Algorithmic Quantum Simulation

Barry C. Sanders

9 Sep 2014

Table of Contents

What is Q Simulation?

Employing a computational machine to mimic certain physical Q systems thereby answering relevant C–intractable questions accurately and efficiently.

- Accuracy: bounded error ϵ.
- Efficiency: cost (e.g., time and space) of simulation scales 'reasonably' (polynomially) with the problem size.

An aim of Q sim: simulating Schrödinger's Equation

- Schrödinger's equation:

$$\mathrm{i}\frac{\mathrm{d}}{\mathrm{d}t}|\psi(t)\rangle = \hat{H}(t)|\psi(t)\rangle.$$

- Unitary dynamics ($\hbar \equiv 1$):
$\hat{H} = \hat{H}^\dagger \implies |\psi(t)\rangle = \mathcal{T}\exp\left\{-\mathrm{i}\int_0^t \mathrm{d}u \hat{H}(u)\right\}|\psi(0)\rangle.$
- Time-independent: $|\psi(t)\rangle = \exp\left\{-\mathrm{i}\hat{H}t\right\}|\psi(0)\rangle$
- Different solutions with different complexity:
  - solve $|\psi(t)\rangle$ over some time domain;
  - determine the spectrum of $\hat{H}$;
  - find eigenvectors of $\hat{H}$, e.g. the ground state;
  - estimate the mean of an observable $\langle\psi(t)|\hat{O}|\psi(t)\rangle$.
- Some quantities could be tractable whereas others not so.

Some $\mathbb{C}$ methods for simulating Schrödinger's Equation

- Diagonalize $\hat{H}$; then algebraïc.
- Integrate:
  - Runge-Kutta;
  - Magnus expansions = Baker-Campbell-Hausdorff method;
  - Product formulæ:
    - Forest-Ruth = symplectic integration;
    - Trotter-Suzuki;
    - ....
- Quantum Monte Carlo simulations:
  - Stochastic Green functions;
  - Variational, diffusion or path-integral Monte-Carlo methods.
- Density matrix renormalization group.

Wiebe Berry Høyer BCS *J. Phys. A* **43** 065203 (2010).

Feynman: Simulating Physics with Computers

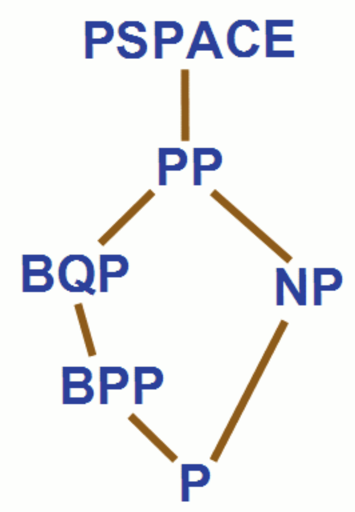
§5. Can Q systems be probabilistically simulated by a C computer?

Can a Q system be probabilistically simulated by a C (probabilistic, I'd assume) universal computer? In other words, a computer which will give the same probabilities as the Q system does. If you take the computer to be the C kind I've described so far, (not the Q kind described in the last section) and there're no changes in any laws, and there's no hocus-pocus, the answer is certainly, No! This is called the hidden-variable problem: it is impossible to represent the results of Q mechanics with a C universal device.

Feynman *Int. J. Th. Phys.* **21** (1982) pp. 467–488.

Decision problems & complexity (Aaronson's schematic)

- How hard to solve Yes/No problem.
- Employ algorithm (input, output, procedure using instruction set).
- Instance size: n bits for input.
- Complexity: resource scaling (T & S) vs n.
- PSPACE $\subset$ EXP.
- PP: Y $\Longrightarrow$ output Y w/pr$\geq 1/2$; N $\Longrightarrow$ output Y w/pr$\leq 1/2$.
- BPP: Y $\Longrightarrow$ output Y w/pr$\geq 2/3$; N $\Longrightarrow$ output Y w/pr$\leq 1/3$.



Feynman exegesis

- Heisenberg picture (matrices) $\implies$ q problems $\subset$ EXP.
- Feynman path integral $\implies$ q problems $\subset$ PP.[1]
- "give the same probabilities" $\implies$ q algorithm efficiently answers decision problems concerning expectation values $\langle\psi|\hat{\mathcal{O}}|\psi\rangle$ with bounded error.
- "classical kind . . . the answer is certainly, No!" $\implies$ some of these problems $\not\subset$ BPP.[2]
  - P=BPP generally believed.
  - Implication BPP$\subset$BQP would be significant if proved.
  - Feynman says "No!" because of "the hidden-variable problem: it is impossible to represent the results of quantum mechanics with a classical universal device". Correct?
- Aside: post-selected quantum computing is PostBQP=PP.

Approximate simulation of (known) $\hat{H}$-generated evolution

Simulating within tolerance ϵ.

- Treat case of time-independent $\hat{H}^{(n)}$;
- Resultant evolution over time t: $U = \exp\left\{-\mathrm{i}\hat{H}^{(n)}t\right\}$;
- Evolution: $|\psi(t)\rangle = \exp\left\{-\mathrm{i}\hat{H}t\right\}|\psi(0)\rangle$;
- Simulated state $|\tilde{\psi}(t)\rangle$ has error: $\|\,|\tilde{\psi}(t)\rangle - |\psi(t)\rangle\,\|$;
- Input: $\epsilon =$ upper bound to allowed worst-case error.

Raeisi Wiebe BCS *New J. Phys.* **14** 103017 (2012).

Decomposing an n-qubit k-local $\hat{H}^{(n)}$

Write the Hamiltonian as a sum of simpler Hamiltonians

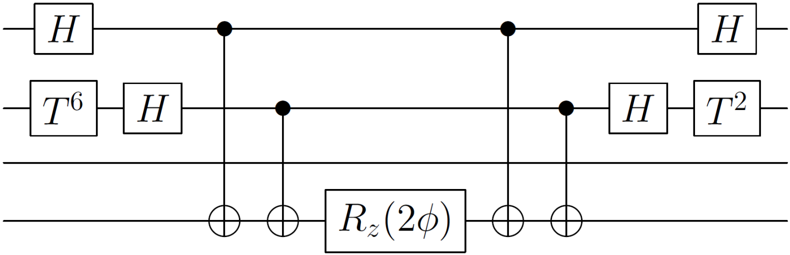
- Express evolution as sequence of evolutions generated by simpler Hamiltonians;
- Let $\hat{\mathfrak{h}}_j^{(n)} = \otimes_{\ell=1}^{n} \hat{\Xi}_{j\ell}^{(n)}$ act on $k \in \text{polylog}(n)$ qubits;
- Each $\hat{\Xi}_{j\ell}^{(n)}$ drawn from
$$\left\{\mathbb{1}, X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \mathrm{i}\begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}\right\}$$
& is non-$\mathbb{1}$ for $\leq k$ instances in tensor product;
- k-local $\hat{H}^{(n)}$: $\sum_{j=1}^{m\in\text{poly}(n)} \hat{\mathfrak{h}}_j^{(n)}$.

Raeisi Wiebe BCS *New J. Phys.* **14** 103017 (1982).

Quantum circuit component for Pauli evolution

Unitary evolution generated by $\mathfrak{h}_j$

$$U_j = \exp\left\{-\mathrm{i}\mathfrak{h}_j^{(n)} t\right\}$$



$$\exp\left\{-\mathrm{i}\phi X \otimes Y \otimes \mathbb{1} \otimes Z\right\}$$

Sequence of $\hat{H}$-generated evolutions

Generating and multiplying evolution operators.

- Partition time interval $\Delta t = t/r$, namely $(t_1, \ldots, t_M)$;
- $U(\Delta t) \approx \exp\left\{-\mathrm{i} a_{j_M} \mathfrak{h}_{j_M}^{(n)} t_M\right\} \cdots \exp\left\{-\mathrm{i} a_{j_1} \mathfrak{h}_{j_1}^{(n)} t_1\right\}$.

General case: time-ordered exponential

$$\mathcal{T} \exp\left\{-\mathrm{i} \int_t^{t+\Delta t} \mathrm{d}u \sum_{j=1}^m \hat{H}_j(u)\right\} \approx \prod_{q=1}^M \exp\left(-\mathrm{i}\hat{H}_{j_q}(t_q)\Delta t_q\right).$$

Trotter product formula

$\mathrm{e}^{\mathrm{i}t\left(\hat{\mathfrak{h}}+\hat{\mathfrak{h}}'\right)} \to \lim_{n\to\infty} \left(\mathrm{e}^{\mathrm{i}t\hat{\mathfrak{h}}/n} \mathrm{e}^{\mathrm{i}t\hat{\mathfrak{h}}'/n}\right)^n$. Error ϵ is important.

Raeisi Wiebe BCS *New J. Phys.* **14** 103017 (1982).

Minimizing time cost using Suzuki's iterative algorithm

Suzuki's generalization of the Trotter formula

$$S_2(\lambda) = \prod_{j=1}^{m} e^{\hat{H}_j \lambda/2} \prod_{j'=m}^{1} e^{\hat{H}_{j'} \lambda/2},$$

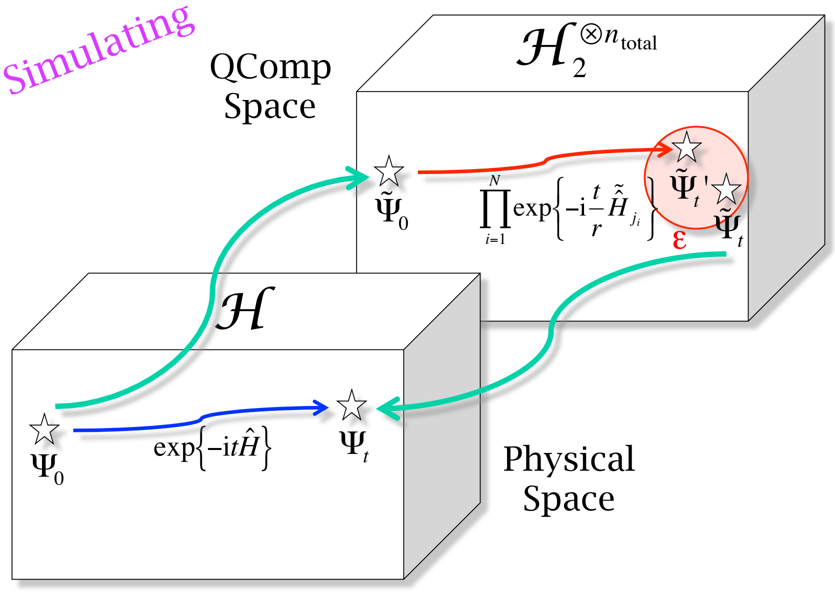
$$S_{2k}(\lambda) = [S_{2k-2}(p_k\lambda)]^2 \, S_{2k-2}\left((1-4p_k)\,\lambda\right) [S_{2k-2}(p_k\lambda)]^2 \, ,$$

for $p_k = \left(4 - 4^{1/(2k-1)}\right)^{-1}$. Each iteration k has $5\times$ as many terms as for iteration $k-1$.

Suzuki proves for small λ:

$$\left\| \exp\left\{\sum_{j=1}^{m} \hat{H}_j \lambda\right\} - S_{2k-1}(\lambda) \right\| \in O\left(|\lambda|^{2k+1}\right).$$

Suzuki *Phys. Lett. A* **146** 319 (1990),
Suzuki *J. Math. Phys.* **32** 400 (1991).


Simulating
QComp Space
$\mathcal{H}_2^{\otimes n_{\text{total}}}$
$\tilde{\Psi}_0$
$\prod_{i=1}^{N} \exp\left\{-\mathrm{i}\frac{t}{r}\tilde{\hat{H}}_{j_i}\right\}$
$\tilde{\Psi}_t{}'$
$\tilde{\Psi}_t$
ε
$\mathcal{H}$
Ψ_0
$\exp\{-\mathrm{i}t\hat{H}\}$
Ψ_t
Physical Space

Hamiltonian in a black-box

- Previously designed algorithm exploits knowledge of $\hat{H}^{(n)}$;
- Black-box setting: algorithm without knowledge of $\hat{H}^{(n)}$;
- $\hat{H}^{(n)}$ is queried during algorithm;
- $\hat{H}^{(n)}$ is exponentially large in n;
- Require simplifying promises for $\hat{H}^{(n)}$ to reduce cost;
- Objective is to construct an efficient algorithm for any Hamiltonian subject to reasonable promises.

Lloyd's 1996 formalization of efficient Q computing

Assumed tensor-product structure and used

$$\exp\left\{-\mathrm{i}t\sum_{j=1}^{m}\hat{H}_j\right\} = \left(\prod_{i=1}^{N}\exp\left\{-\mathrm{i}\frac{t}{r}\hat{H}_{j_i}\right\}\right)^r + \sum_{j>j'}\left[\hat{H}_j, \hat{H}_{j'}\right]\frac{t^2}{2r} + \epsilon$$

to prove poly$n$ time T and space S costs.

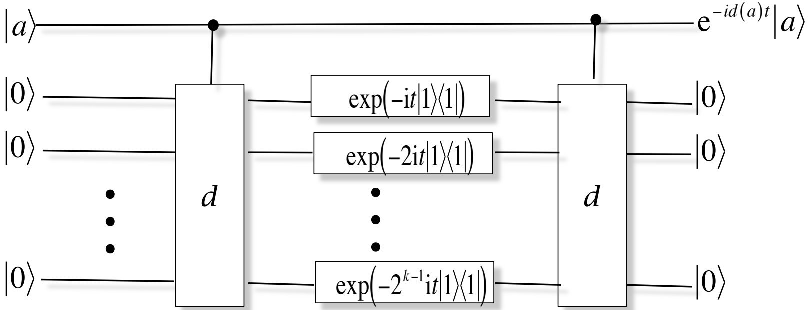
Simulating evolution for one-sparse $\hat{H}^{(n)}$

Simulating evolution for diagonal $\hat{H}$ with $d(a) = \langle a|\hat{H}|a\rangle \in \{0,1\}^k$.

$|a,0\rangle \mapsto |a,d(a)\rangle \mapsto \exp\{-\mathrm{i}td(a)\}|a,d(a)\rangle \mapsto \exp\{-\mathrm{i}td(a)\}|a,0\rangle$.

Circuit for one-sparse Hamiltonian evolution is a minor modification of diagonal-$\hat{H}^{(n)}$ circuit.

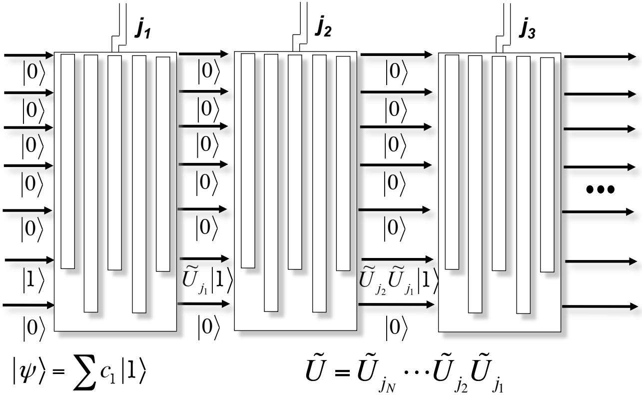
Childs Cleve Deotto Farhi Gutmann Spielman *STOC'03* **146** 59–68.



Simulating evolution for one-sparse $\hat{H}$

General evolution as sequence of one-sparse Hamiltonian evolutions

Approximately & efficiently decompose the overall evolution $U \approx \prod_{\nu=1}^{N} U_{j_\nu}$ each generated by one-sparse $\hat{H}_{j_i}$.



Q state generation [Aharonov & Ta-Shma (AT) 2003]

- Motivated by claims of adiabatic Q computing solving NP-Hard problems (still relevant today[3]).
- Consider which Q states can be efficiently generated.
- Oracle setting: efficiently queries elements of $\hat{H}$.
- No assumption of tensor-product structure (c.f. Lloyd).
- Demonstrate equivalence between QSG and statistical zero knowledge (SZK) problems.
    - ZK proof: prove knowledge of secret without revealing secret.
    - SZK problems: discrete log, quadratic residuosity, ...
    - Specifically show SZK problems reducible to QSG problems.

Considerations for efficient quantum simulation

- Problem size: Number n of qubits in the system.
- Accuracy: The answer is no worse than ϵ (appropriate metric).
- Efficient: Solve with resource consumption $\in O\left(\text{poly}\frac{n}{\epsilon}\right)$.
- Generality: Solves problems for a broad class of systems.

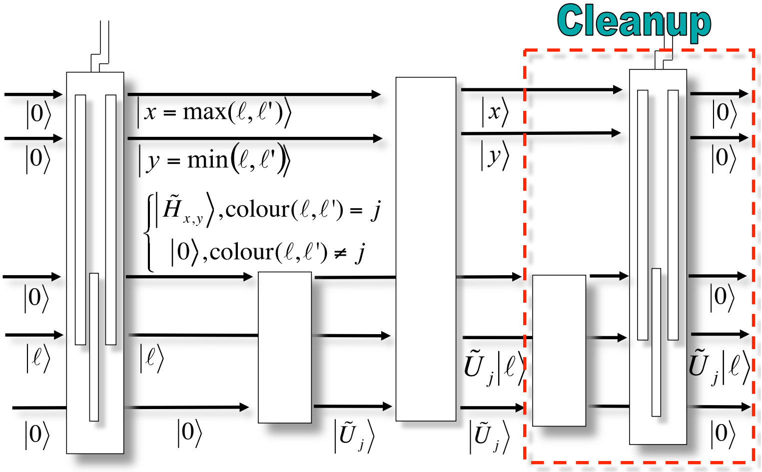
Sparse Hamiltonian Lemma (Aharonov & Ta-Shma STOC 2003)

If $\hat{H}$ acting on n qubits is d-sparse s.t. $d \in O(\text{poly}n)$ & the list of nonzero entries in each row is efficiently computable, then $\hat{H}$ is *simulatable* if $\|\hat{H}\| \leq \text{poly}n$.

Childs's rules for simulatability

- $\sum_i \hat{H}_i$ with each $\hat{H}_i$ acting on $O(1)$ qubits or
- is a $\sqrt{-1}\times$ commutator of two simulatable $\hat{H}_i$s or
- convertible to simulatable $\hat{H}$ by efficient unitary conjugation or
- is sparse and efficiently computable

Aharonov & Ta-Shma Circuit (Wiebe's picture)



Simulation cost is slightly superlinear in time $t^{1+o(1)}$

Theorem [Berry, Ahokas, Cleve, Sanders 2007 (BACS)]

$$M \leq \frac{m5^{2k}(mq_k\tau)^{1+1/2k}}{2\left[(2k+1)!\epsilon\right]^{1/2k}}$$

Optimize

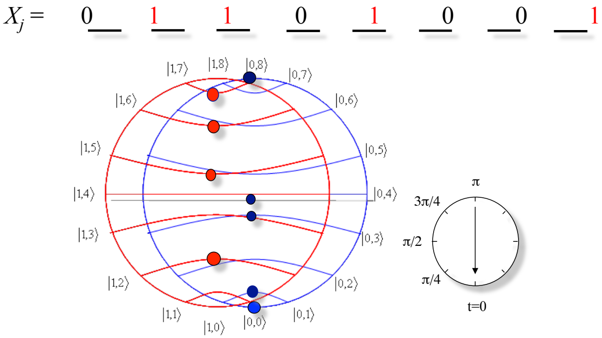
$$k \approx \frac{1}{2}\sqrt{\log_5\left(\frac{m\tau}{\epsilon}\right)}.$$

$$M \leq 2m^2\tau\exp\left\{2\sqrt{\log_5\left(\frac{m\tau}{\epsilon}\right)}\right\} \approx \frac{1}{2}\sqrt{\log_{5/\sqrt{3}}\left(\frac{m\tau}{\epsilon}\right)} \qquad (1)$$

Black-box Q simulation must be superlinear in time

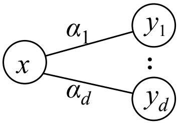
Theorem (No Quantum Speedup)

For all positive integers N $\exists$ a row-computable two-sparse $\hat{H}$ s.t. simulating $\hat{H}$-generated evolution for (scaled) time $\tau = \pi N/2$ within precision $1/4$ requires $\geq \tau/2\pi$ queries to $\hat{H}$.

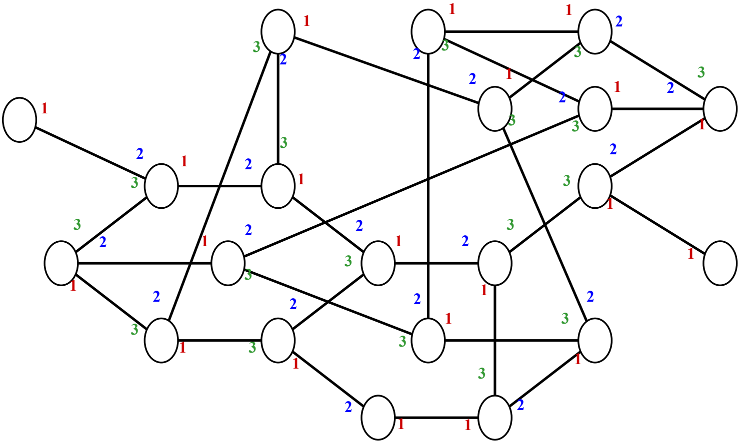


Hamiltonians as weighted graphs (Cleve's picture)

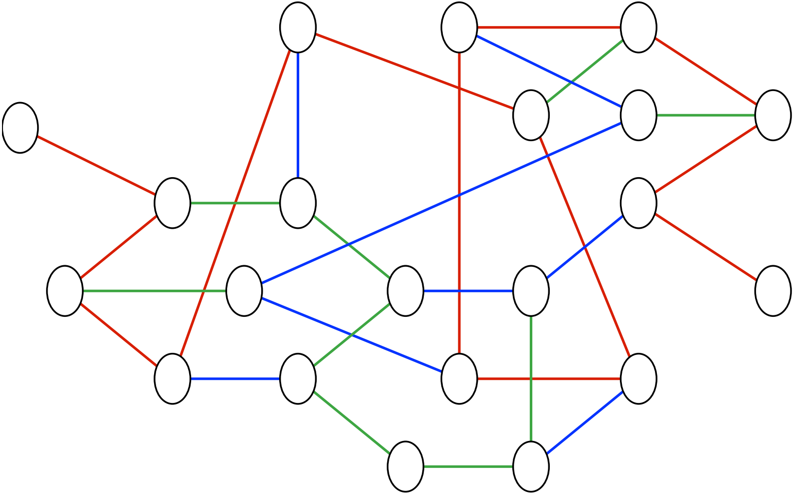
- For column x, only rows $y_{1,\ldots,d}$ hold nonzero matrix elements.
- The graph weight α_i is $\langle x|\hat{H}|y_i\rangle$.
- As $\hat{H}=\hat{H}^\dagger$, α_i^* is the weight for column y_i and row x.
- Hermitian $\hat{H}$ can be represented by a degree d graph.
- Goal: decompose $\hat{H}$ graph into disjoint union of $d=1$ graphs.



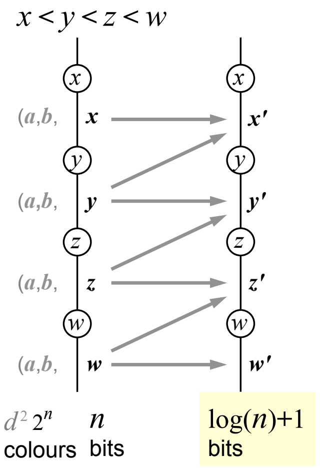
Colouring the graph for $\hat{H}$ with d^2 labels (Cleve's picture)



Problem: long monochromatic paths (Cleve's picture)



Colouring by Cole-Vishkin coin tossing [Cleve picture]



"Deterministic coin-tossing" [Cole & Vishkin '86]

$$y' \leftarrow (i, y_i), \text{ where } i = \min\{j : y_j \neq z_j\}$$

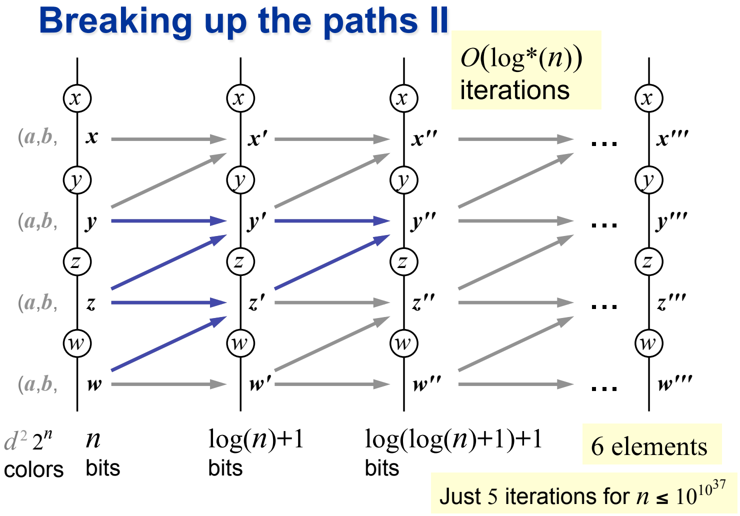
Example: $y = 01\textcolor{red}{1}00101$ (010 ↓)

$z = 01\textcolor{red}{0}01101$

Then $y' = (010,1)$

Note: still a valid coloring!
$x' \neq y'$ & $y' \neq z'$ & $z' \neq w'$

Colouring by Cole-Vishkin coin tossing [Cleve picture]



Time and space costs for simulating $\hat{H}$-generated evolution

Who	Year	T	S
Lloyd	1996	$O(t^2)$	$O(n)$
AT[4]	2003	$O\left(n^9 d^4 \frac{t^2}{\epsilon}\right)$	$O(n)$
Childs[5]	2003	$O\left(n^2 d^{4+o(1)} \frac{t^{3/2}}{\sqrt{\epsilon}}\right)$	$O\left(n\right)$
BACS[6]	2007	$O\left(\log^* n d^{4+o(1)} \frac{t^{1+1/2k}}{\epsilon^{1/2k}}\right)$	$O(n \log^* n)$
CK[7]	2010	$O\left(\left[d^3 + d^2 \log^* n\right] \frac{t^{1+1/2k}}{\epsilon^{1/2k}}\right)$	$O\left(nd + n \log^* n\right)$
BC[8]	2010	$O\left(\Vert\hat{H}\Vert_{\max} d \frac{t}{\sqrt{\epsilon}}\right)$	•

Simulating many-body systems

Abrams & Lloyd PRL 1996

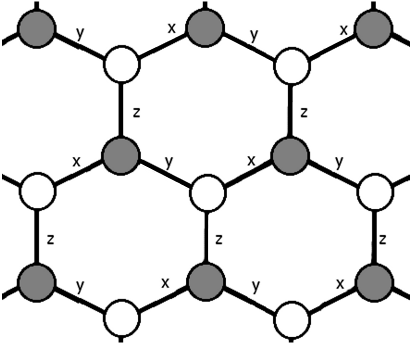
But the problem of simulation — that is, the problem of modeling the full time evolution of an arbitrary Q system — is less technologically demanding. While thousands of qubits and billions of Q logic operations are needed to solve C difficult factoring problems [16], it would be possible to use a Q computer with only a few tens of qubits and a few thousand operations to perform simulations that would be C intractable [17].

$$\hat{H}_{\text{Hubbard}} = -t \sum_{\langle i,j\rangle,\sigma} \left(\hat{c}^{\dagger}_{i\sigma}\hat{c}_{j\sigma} + \hat{c}^{\dagger}_{j\sigma}\hat{c}_{i\sigma} \right) + U \sum_{i=1}^{N} \hat{n}_{i\uparrow}\hat{n}_{i\downarrow},$$

$$\hat{H}_{\text{Bose-Hubbard}} = -t \sum_{\langle i,j\rangle} \left(\hat{c}^{\dagger}_{i\sigma}\hat{c}_{j} + \hat{c}^{\dagger}_{j}\hat{c}_{i} \right) + \frac{U}{2} \sum_{i=1}^{N} \hat{n}_{i}\left(\hat{n}_{i} - \mathbb{1}\right) - \mu \sum_{i=1}^{N} \hat{n}_{i}.$$

Examples: models for simulation

- $\hat{H}_{\text{Ising}} = J\sum_{\langle i,j\rangle} Z_i \otimes Z_j + B\sum_i X_i$.
- $\hat{H}_{\text{XY}} = J_x\sum_{\langle i,j\rangle} X_i \otimes X_j + J_y\sum_{\langle i,j\rangle} Y_i \otimes Y_j$.
- $\hat{H}_{\text{Heisenberg}} = J_x\sum_{\langle i,j\rangle} X_i \otimes X_j + J_y\sum_{\langle i,j\rangle} Y_i \otimes Y_j$
- $\hat{H}_{\text{honeycomb}} =$
  $-J_x\sum_{x\text{-link}} X_i \otimes X_j - J_y\sum_{y\text{-link}} Y_i \otimes Y_j - J_z\sum_{x\text{-link}} Z_i \otimes Z_j$



Time-dependent Hamiltonian evolution

Problem:

For $\hat{H} = \sum_{j=1}^{m} \hat{H}_j$ with each Hamiltonian $\hat{H}_j : \mathbb{R} \to \mathbb{C}^{N\times N}$ P-differentiable, construct

$$U(t, t+\Delta T) := \mathcal{T} \exp\left\{-\mathrm{i}\int_t^{t+\Delta t} \mathrm{d}u \hat{H}(u)\right\}$$

as a product of N exponentials $\exp\left\{-\mathrm{i}\hat{H}_{j_P}(t_P)\Delta t_P\right\}$ within tolerance ϵ of $U(t, t+\Delta t)$, and find an upper bound for N.

Conditions for $\hat{H}(t)$ to be efficiently Q–simulatable

Theorem: Wiebe, Berry, Høyer, Sanders 2010

Let $\hat{H}(t) = \sum_{j=1}^{m} \hat{H}_j(t)$ with each $\hat{H}_j(t)$ $2k$-differentiable on $[\mu, \mu + \Delta\lambda]$. Furthermore let timescale Λ satisfy

$$\Lambda = \sup_{\lambda \in [\mu, \mu+\Delta\lambda]} \max_{q=0,\ldots,2k,\, j=1,\ldots,m} \left\| \partial_\lambda^q \hat{H}_j(t) \right\|^{1/(q+1)}$$

with

$$\epsilon \leq \frac{9}{10} \left(\frac{5}{3} \right)^k \Lambda \Delta\lambda$$

and $\max_{x>y} \|U(x,y)\| \leq 1$, then a decomposition $\tilde{U}(\mu + \Delta\lambda, \mu)$ can be constructed s.t. $\|\tilde{U} - U\| \leq \epsilon$ and s.t. the number of operator exponentials in $\tilde{U}$ satisfies

$$M \leq \left\lceil 3m\Lambda\Delta\lambda k \left(\frac{25}{3} \right)^k \left(\frac{\Lambda\Delta\lambda}{\epsilon} \right)^{1/2k} \right\rceil$$

Q linear equation solver [Harrow Hassidim, Lloyd 2009]

Typical problem statement

Given matrix A and vector b, find x such that $Ax = b$; or given matrix A, vector b, and matrix M, find a good approximation of $x^{\mathrm{T}}Mx$ such that x such that $Ax = b$.

- Replace b by $|b\rangle = \sum_{i=1}^{N} b_i|i\rangle$ in computational basis.
- Then $|x\rangle = \hat{\mathfrak{h}}^{-1}|b\rangle$, but inverting $\hat{\mathfrak{h}}$ is hard.
- $\hat{\mathfrak{h}}$ has eigenvalues λ_j and eigenvectors $|u_j\rangle$ for $j = 1, \ldots, N$.
- Express $|b\rangle = \sum_{j=1}^{N} \beta_j|u_j\rangle$.
- Idea: $|x\rangle = \hat{\mathfrak{h}}^{-1}|b\rangle \approx \sum_{j=1}^{N} \frac{\beta_j}{\lambda_j}|u_j\rangle$.
    - Kitaev phase-estimation approach: $\sum_{j=1}^{N} \beta_j|u_j\rangle|\lambda_j\rangle$.
    - Construct (non-unitary) linear map $|\lambda_j\rangle \mapsto \lambda_j^{-1}|\lambda_j\rangle$.
    - Uncompute $|\lambda_j\rangle$ to obtain approximate $|x\rangle$.

Summary

- Devised and costed efficient, accurate algorithms for Q simulation for $\hat{H}$ held by an oracle.
- For oracle setting, an efficient query technique is developed to construct the Q simulation as a concatenation of Q circuits for one-sparse $\hat{H}$ simulation.
- Run-time for Q algorithm is reduced by exploiting higher-order Suzuki method.
- Applications to many-body Q simulation.
- Q algorithms have been developed for time-dependent $\hat{H}$, which is relevant to adiabatic Q computing, controlled systems and Q phase transitions.
- Q could be used as a linear equation solver.