About universal quantum simulation

Barry C. Sanders, University of Calgary

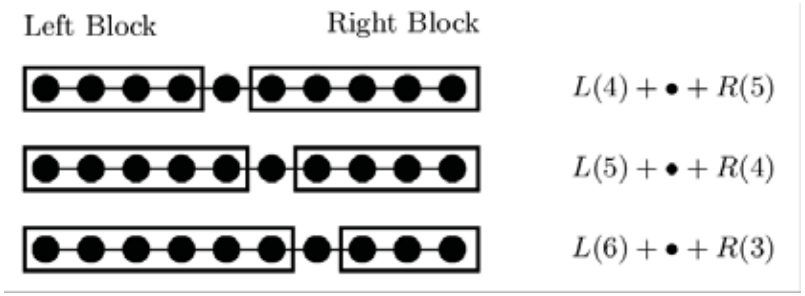
Goal: Solve Schrödinger's Equation

$$\mathrm{i}\frac{\mathrm{d}}{\mathrm{d}t}\left|\psi(t)\right\rangle = \hat{H}(t)\left|\psi(t)\right\rangle,\ \hat{H} = \hat{H}^{+} \Rightarrow \text{Unitary dynamics}$$

- Solve state $\left|\psi(t)\right\rangle$ as a function of t.
- Determine the spectrum of $\hat{H}$.
- Find eigenvectors of $\hat{H}$, e.g. ground state.
- Estimate mean of some operator $\left\langle\psi(t)\right|\hat{O}\left|\psi(t)\right\rangle$.
- Applications: Chemistry, Physics, coupled linear equations, differential equations, ….

Solve with (classical) computer

- Diagonalize the Hamiltonian
- Integrate Schrödinger's equation
  - Runge-Kutta
  - Magnus expansions
  - Product formulæ (e.g. Forest-Ruth and Trotter-Suzuki)
- Q Monte Carlo simulations
  - Stochastic Green Function
  - Variational, Diffusion or Path-Integral Monte Carlo, …
- Density matrix renormalization group



International Journal of Theoretical Physics, Vol. 21, Nos. 6/7, 1982

Simulating Physics with Computers

Richard P. Feynman

Department of Physics, California Institute of Technology, Pasadena, California 91107

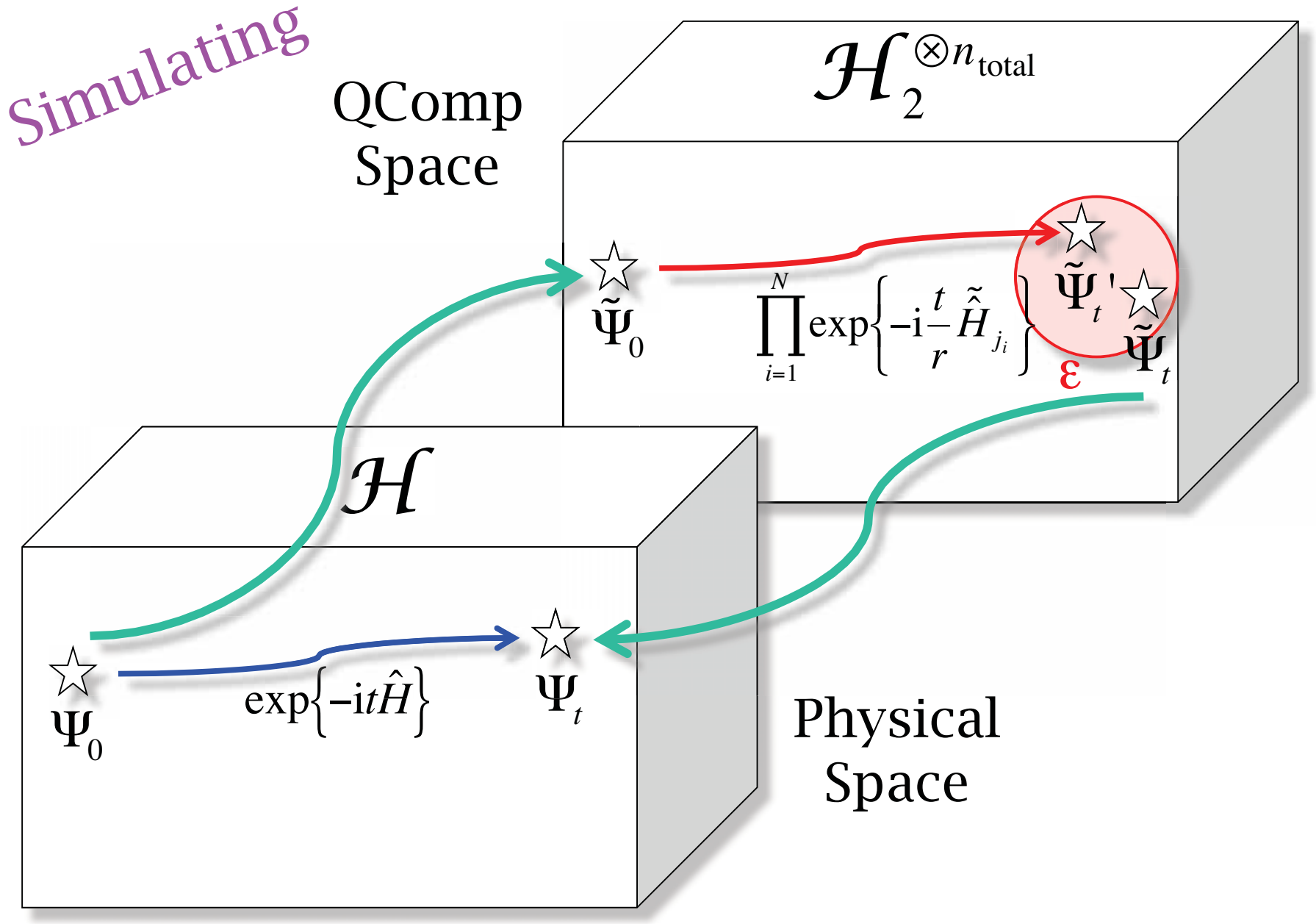
Received May 7, 1981

1. INTRODUCTION

On the program it says this is a keynote speech—and I don't know what a keynote speech is. I do not intend in any way to suggest what should be in this meeting as a keynote of the subjects or anything like that. I have

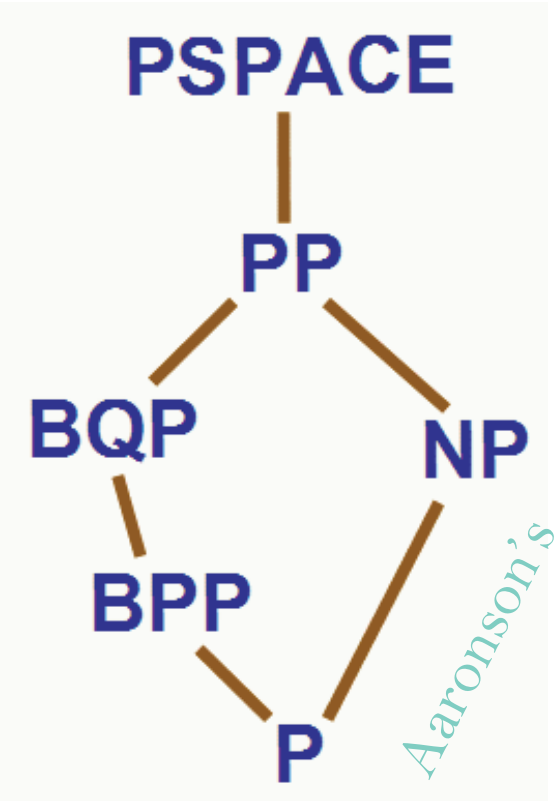
5. CAN QUANTUM SYSTEMS BE PROBABILISTICALLY SIMULATED BY A CLASSICAL COMPUTER?

Now the next question that I would like to bring up is, of course, the interesting one, i.e., Can a quantum system be probabilistically simulated by a classical (probabilistic, I'd assume) universal computer? In other words, a computer which will give the same probabilities as the quantum system does. If you take the computer to be the classical kind I've described so far, (not the quantum 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 quantum mechanics with a classical universal device. To learn a little bit


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

Computational Complexity Classes

- Decision problem: yes/no answer.
- Algorithm: procedure using tool set to solve decision problem.
- Complexity: characterized by how cost of limited resources (e.g. disk space, memory, time) scales with problem size (e.g. bits to specify input) in order to solve problem.
- PSPACE is set of all decision problems solvable on a Turing machine with polynomial-space.



Interpreting Feynman's conjecture

- Complexity of Heisenberg and Schrödinger q-mechanic techniques are in EXP.
- Feynman path integral method is in PSPACE.
- BQP is set of problems solved by a q computer (e.g. Deutsch's 1985) in polynomial-time with "yes" error no larger than 1/3.
- Feynman's says "certainly" $\mathrm{BPP} \subset \mathrm{PSPACE}$.
- However, we only know that $\mathrm{BPP} \subseteq \mathrm{PSPACE}$.
- Proving Feynman's claim would be highly significant in computer science.

Can we simulate any Hamiltonian?

- Find solution s_x to problem x by simulating

$$\hat{H} = \sum_x \left(|0, s_x\rangle\langle x, 0| + |0, x\rangle\langle s_x, 0| \right)$$

- Seems too easy to solve any problem so we impose Childs's rules: H is
  - a sum of H_i's each acting on O(1) qubits, or
  - is a ($i\times$) commutator of two simulatable H's, or
  - is related to a simulatable H by an efficiently implementable unitary conjugation, or
  - is sparse and efficiently computable.

Problem

Construct efficient q algorithm for accurately simulating evolution of general q systems.

- efficient → polynomial overhead
- algorithm → instruction set for q computer
- accurate → bounded error
- isolated & q → Hamiltonian evolution
- general → Hamiltonian held by oracle

From continuous to discrete time

Lie - Trotter product formula:

$$e^{it(A+B)} \xrightarrow[n\to\infty]{} \left(e^{itA/n} e^{itB/n}\right)^n .$$

$$\mathrm{T}\exp\left(-i\int_t^{t+\Delta t}\sum_{j=1}^{m} H_j(u)\,\mathrm{d}u\right) \approx \prod_{q=1}^{N_{\exp}} \exp\left(-\mathrm{i}H_{j_q}(t_q)\Delta t_q\right)$$

Q Computing & Feynman's Conjecture

- Lloyd (1996): proved Feynman's conjecture:
  - Cost for N iterations: poly(r,m,n).
  - Assumed tensor product structure.
  - Time-independent Hamiltonian.
  - Runtime is O(t^2) and Space cost (register) is O(n).

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

Q State Generation

- Aharonov & Ta-Shma (STOC 2003)
- Motivated by claims of efficient adiabatic q algorithms to solve NP-Hard problems.
- Raises questions about which q states can be efficiently generated.
- Equivalent to statistical zero-knowledge (SZK) complexity class of problems.

Sparse Hamiltonian Lemma

- Hamiltonian is d-sparse

  If $\hat{H}$ on n qubits is row - sparse

  $\left(\#\ \text{non - zero entries is poly}(n)\text{ - bounded}\right)$

  and row - computable

  $\left(\text{efficiently computable list of nonzero } H_{ij} \text{ in row } i\right)$

  with $\left\|\hat{H}\right\| \le \text{poly}(n)$, then $\hat{H}$ is simulatable.

- Runtime $\in O\left(n^9 d^4/\varepsilon\right), O\left(t^{3/2}\right)$.

Time and space for time-independent H

- Aharonov & TaShma 2003
  - $T \in \mathrm{O}(n^9 d^4 t^{3/2}/\varepsilon)$ & $S \in \mathrm{O}(n)$.
- Childs 2003
  - symmetrized Lie-Trotter formula
  - Improved (Linial) graph coloring algorithm
  - $T \in \mathrm{O}(d^{4+o(1)} n^2 t^{3/2}/\sqrt{\varepsilon})$ & $S \in \mathrm{O}(n)$.
- Berry, Ahokas, Cleve & BCS 2007
  - Lie-Trotter-Suzuki
  - deterministic coin tossing approach to graph colouring
  - $T \in \mathrm{O}(d^{4+o(1)} \log^* n \; t^{1+1/2k}/\varepsilon^{1/2k})$ & $S \in \mathrm{O}(n \log^* n)$.

Approximate evolution as ordered product

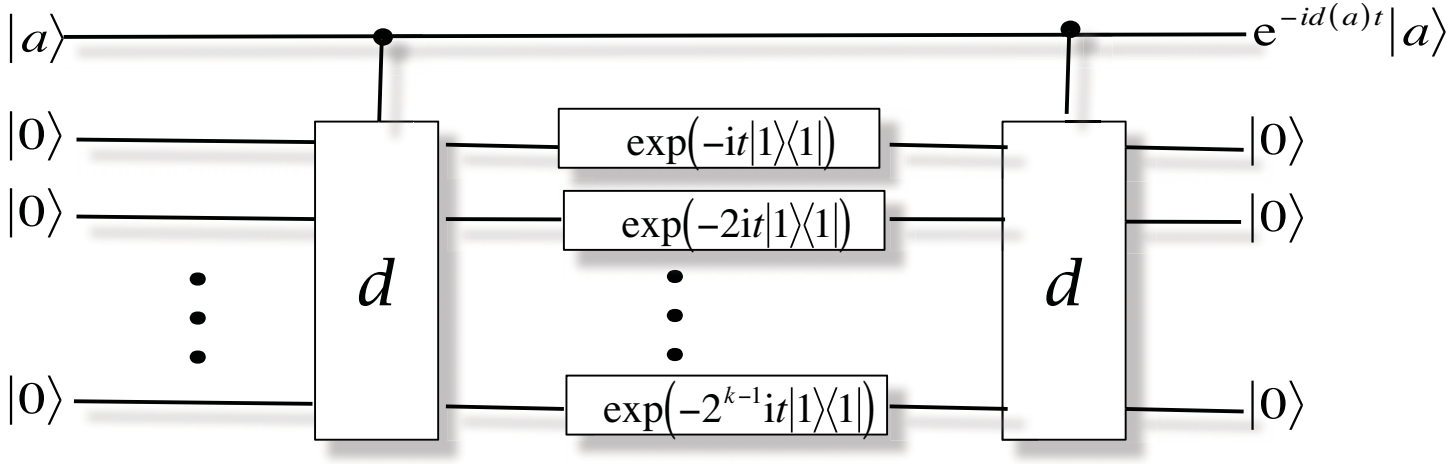
$$\hat{H} = \sum_{j=1}^{m} \hat{H}_j, U_j = \exp\left(-\mathrm{i}\hat{H}_j t/\hbar\right),$$

$$U = \exp\left(-\mathrm{i}\hat{H}t/\hbar\right) \approx \prod_{\nu=1}^{N} U_{j_\nu} .$$

There are *m* types (colours) of *U* but a sequence of *N* U_{j_ν}. We refer to *j* as the colour index.

Efficiently simulating diagonal *H* (Childs)

$d(a) = \langle a|\hat{H}|a\rangle \in \{0,1\}^k$ can be efficiently computed classically.



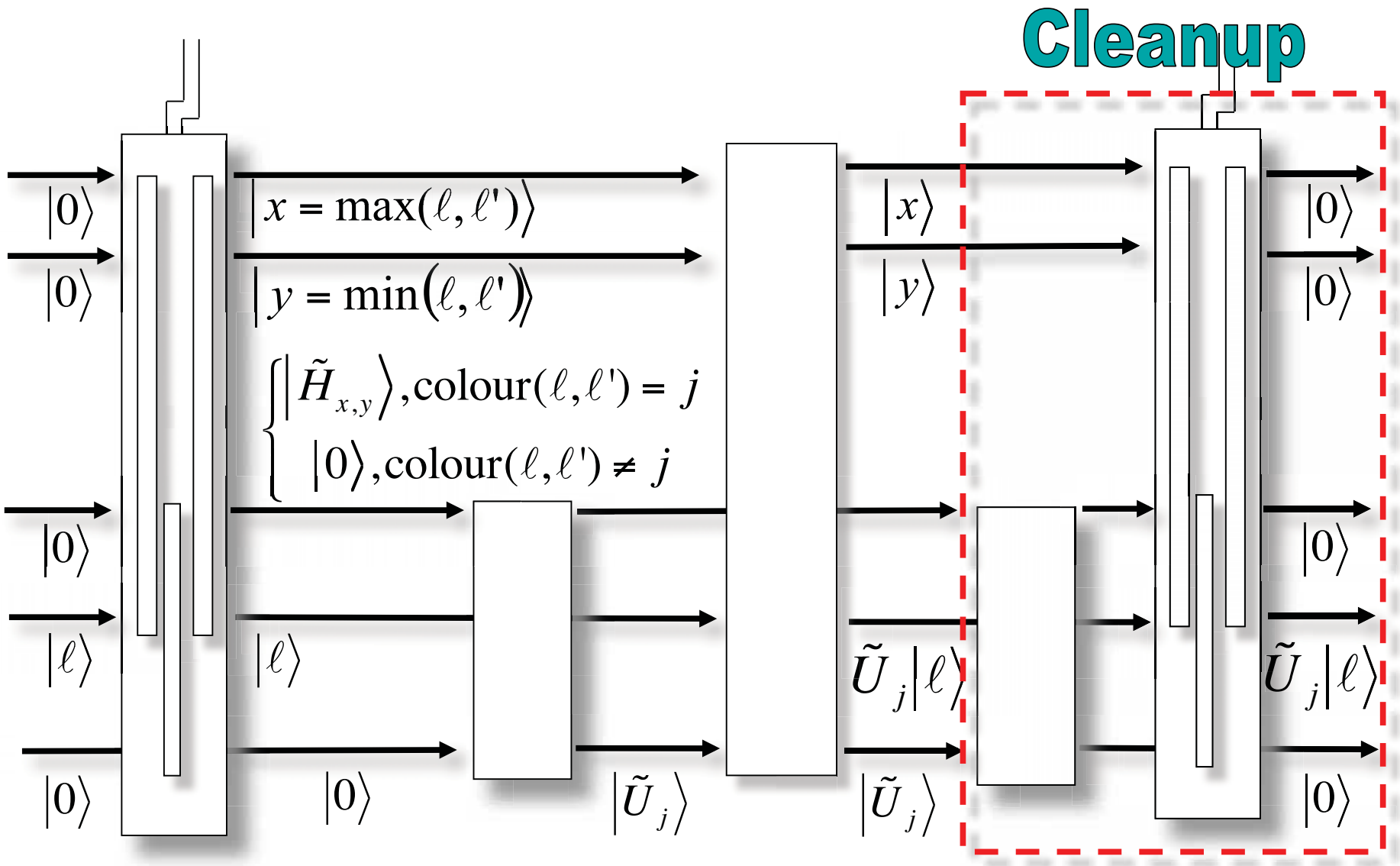
$$|a,0\rangle \mapsto |a,d(a)\rangle \mapsto e^{-itd(a)}|a,d(a)\rangle \mapsto e^{-itd(a)}|a,0\rangle = e^{-i\hat{H}t}|a,0\rangle$$

Simulating one-sparse *H* evolution

- No more than one nonzero element in each row or column of the Hamiltonian matrix.
- Use the Childs-Cleve-Deotto-Farhi-Gutman-Spielman (CCDFGS03) [STOC 2003] circuit to simulate one-sparse Hamiltonian evolution.
- If H is expressed as a sum of one-sparse Hamiltonians H_j, with j the colour index, then we can use the CCDFGS03 circuit to implement the H_j–generated evolution U_j.

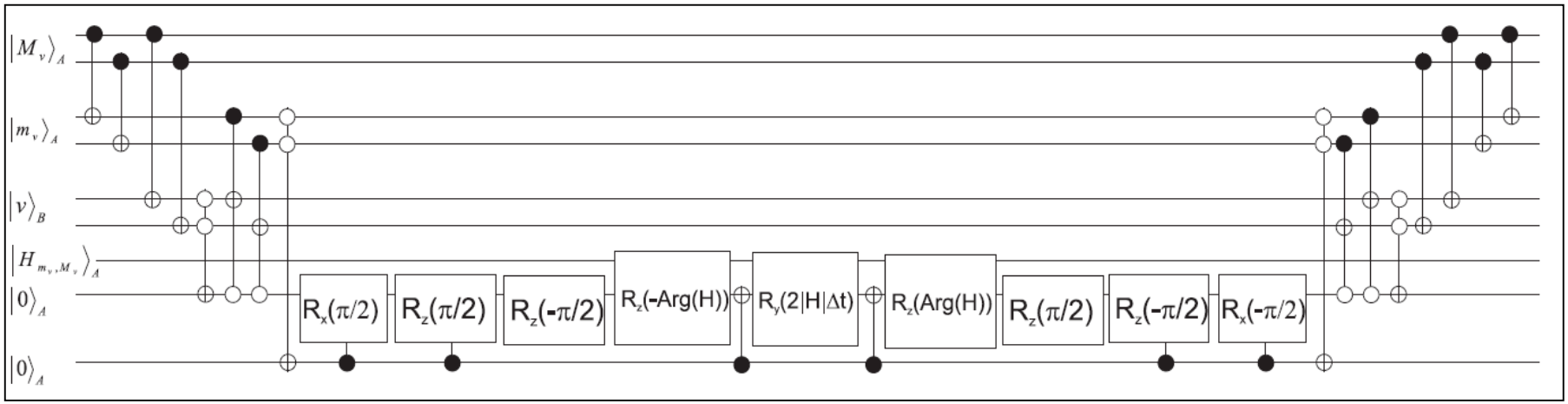
The Aharonov-TaShma circuit

- Input state $|\psi\rangle = \sum_{\ell=1}^{n} \psi_\ell |\ell\rangle$.
- For given colour j_ν, implement evolution U_ν.
- Clean up ancillæ for next step.
- Repeat process to implement the Lie-Trotter-Suzuki approximation of U.



Institute for Quantum Information Science at the University of Calgary

Detailed circuit for evolution



Volume 146, number 6 PHYSICS LETTERS A 4 June 1990

FRACTAL DECOMPOSITION OF EXPONENTIAL OPERATORS WITH APPLICATIONS TO MANY-BODY THEORIES AND MONTE CARLO SIMULATIONS

Masuo SUZUKI

Department of Physics, Faculty of Science, University of Tokyo, Bunkyo-ku, Tokyo 113, Japan

Received 6 February 1990; accepted for publication 28 March 1990
Communicated by A.R. Bishop

A new systematic scheme of decomposition of exponential operators is presented, namely $\exp[x(A+B)]=S_m(x)+O(x^{m+1})$ for any positive integer m, where $S_m(x)=e^{t_1A}e^{t_2B}e^{t_3A}e^{t_4B}\ldots e^{t_MA}$. A general scheme of construction of $\{t_j\}$ is given explicitly. The decomposition $\exp[x(A+B)]=[S_m(x/n)]^n+O(x^{m+1}/n^m)$ yields a new efficient approach to quantum Monte Carlo simulations.

$$\exp\left[x\left(A+B\right)\right]=S_m\left(x\right)+O\left(x^{m+1}\right),$$

$$S_m\left(x\right)=\mathrm{e}^{t_1A}\mathrm{e}^{t_2B}\mathrm{e}^{t_3A}\mathrm{e}^{t_4B}\cdots\mathrm{e}^{t_MA},$$

$$\exp\left[x\left(A+B\right)\right]=\left[S_m\left(x/n\right)\right]^n+O\left(x^{m+1}/n^m\right).$$

Hamiltonian *H* generates unitary: break up

- H as sum of local Hamiltonians $H = \sum_{i=1}^{m} H_i$
- Symmetrized $(m=2)$ Trotter:
  - $e^{iHt} \approx (e^{iH_1 t/2r} e^{iH_2 t/r} e^{iH_1 t/2r})^r$, $H \approx H_1 + H_2$.
- Number of exponentials for q computer $N \propto t^{3/2}$.

Hamiltonian *H* generates unitary: break up

- *H* as sum of local Hamiltonians $H = \sum_{i=1}^{m} H_i$
- Symmetrized *(m=2)* Trotter:
  - $e^{iHt} \approx (e^{iH_1 t/2r} e^{iH_2 t/r} e^{iH_1 t/2r})^r$, $H \approx H_1 + H_2$.
- Number of exponentials for q computer $N \propto t^{3/2}$.
- Suzuki's generalization of Trotter formula:

$$S_2(\lambda) = \prod_{j=1}^{m} e^{H_j \lambda/2} \prod_{j'=m}^{1} e^{H_{j'} \lambda/2} \quad , p_k = \left(4 - 4^{1/(2k-1)}\right)^{-1}$$

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

5 terms

Hamiltonian *H* generates unitary: break up

- *H* as sum of local Hamiltonians $H = \sum_{i=1}^{m} H_i$
- Symmetrized *(m=2)* Trotter:
  - $e^{iHt} \approx (e^{iH_1 t/2r} e^{iH_2 t/r} e^{iH_1 t/2r})^r$, $H \approx H_1 + H_2$.
- Number of exponentials for q computer $N \propto t^{3/2}$.
- Suzuki's generalization of Trotter formula:

$$S_2(\lambda) = \prod_{j=1}^{m} e^{H_j \lambda/2} \prod_{j'=m}^{1} e^{H_{j'} \lambda/2} \quad , p_k = \left(4 - 4^{1/(2k-1)}\right)^{-1}$$

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

- Suzuki proves for small λ:

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

Strict bound on Suzuki formula (BACS)

For $q_k = \prod_{k'=2}^{k} (1 - 4p_{k'})$ and

$$12m5^{k-1}q_k\tau / r \le 1,$$

$$\frac{3}{2}\frac{\left(2m5^{k-1}q_k\tau\right)^{2k+1}}{(2k+1)!r^{2k}} \le 1.$$

$$\tau = t \times \max\left\|H_j\right\|$$

$$\varepsilon \le 1 \le \frac{4m5^{k-1}q_k\tau}{(2k+1)!6^{2k}}$$

Lemma:

$$\left\|\exp\left(-it\sum_{i=1}^{m} H_i\right) - \left[S_{2k}\left(-i\frac{t}{r}\right)\right]^r\right\| \le 2\frac{\left(2m5^{k-1}q_k\tau\right)^{2k+1}}{(2k+1)!r^{2k}}$$

With $r = \left\lceil (2m5^{k-1}q_k\tau)^{1+1/2k}\left[\frac{2}{(2k+1)!\epsilon}\right]^{1/2k}\right\rceil$

for $\epsilon \le 1 \le (4m5^{k-1}q_k\tau)/[(2k+1)!6^{2k}]$ the Lemma's constraints are satisfied, and the error does not exceed ε. The number of exponentials in $S_{2k}(\lambda)$ does not exceed $2m5^{k-1}r$ so using this value for r implies the result.

Simulation cost is almost linear in time

Theorem: $$N \le \frac{m5^{2k}\left(mq_k\tau\right)^{1+1/2k}}{2\left[\left(2k+1\right)!\varepsilon\right]^{1/2k}}$$

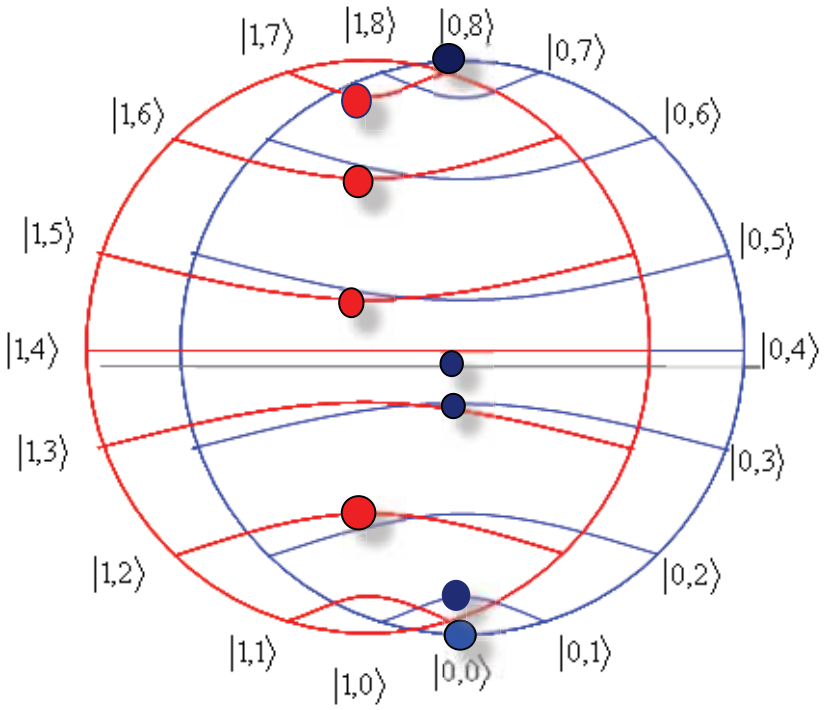
Optimize k: $$k \approx \frac{1}{2}\sqrt{\log_5\left(\frac{m\tau}{\varepsilon}\right)}$$

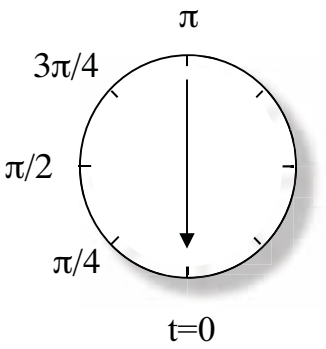
Then $$N \le 4m^2\tau\exp\left[2\sqrt{\log_5\left(m\tau/\varepsilon\right)}\right]$$

BACS: Cannot be sublinear in *t*

Theorem: For all positive integers N there exists a row-computable 2-sparse Hamiltonian H such that simulating the evolution of H for scaled time $\tau = \pi N/2$ within precision $1/4$ requires at least $\tau/2\pi$ queries to H.

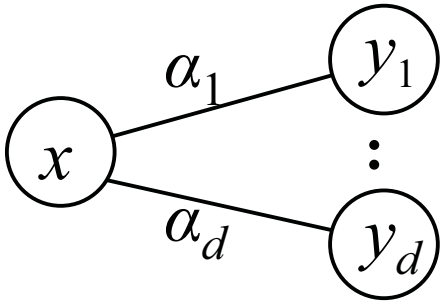
$X_j =$ 0 1 1 0 1 0 0 1

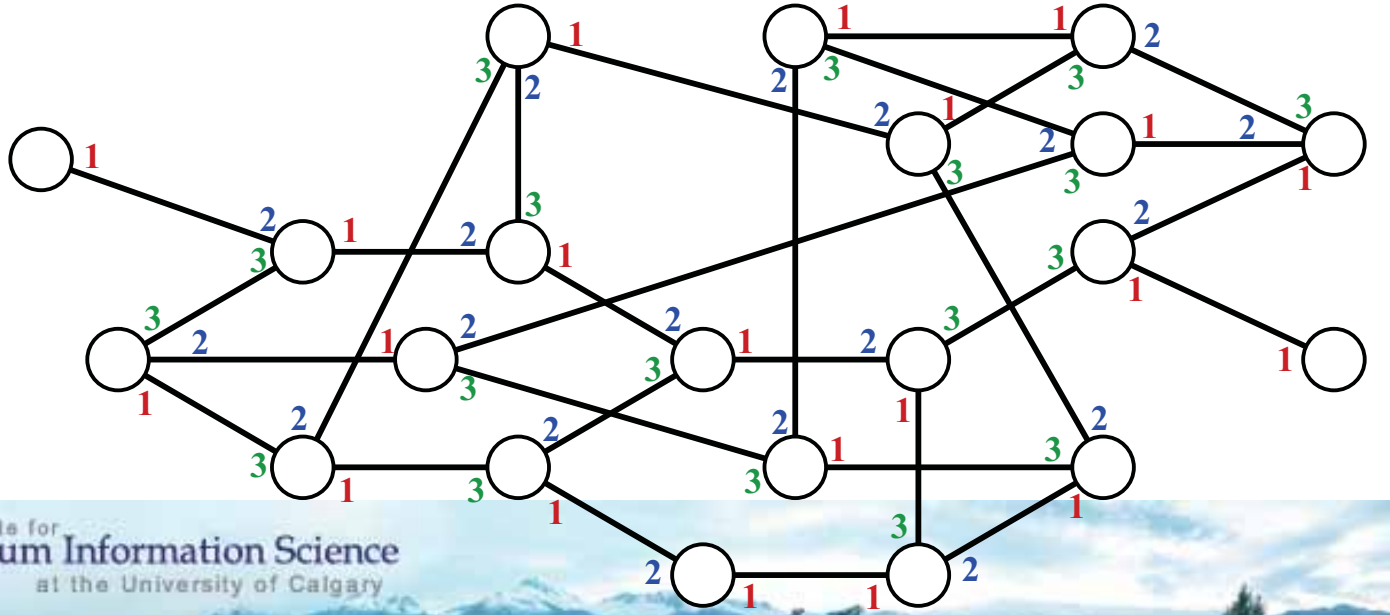




Graph for sparse H (Cleve's slides)

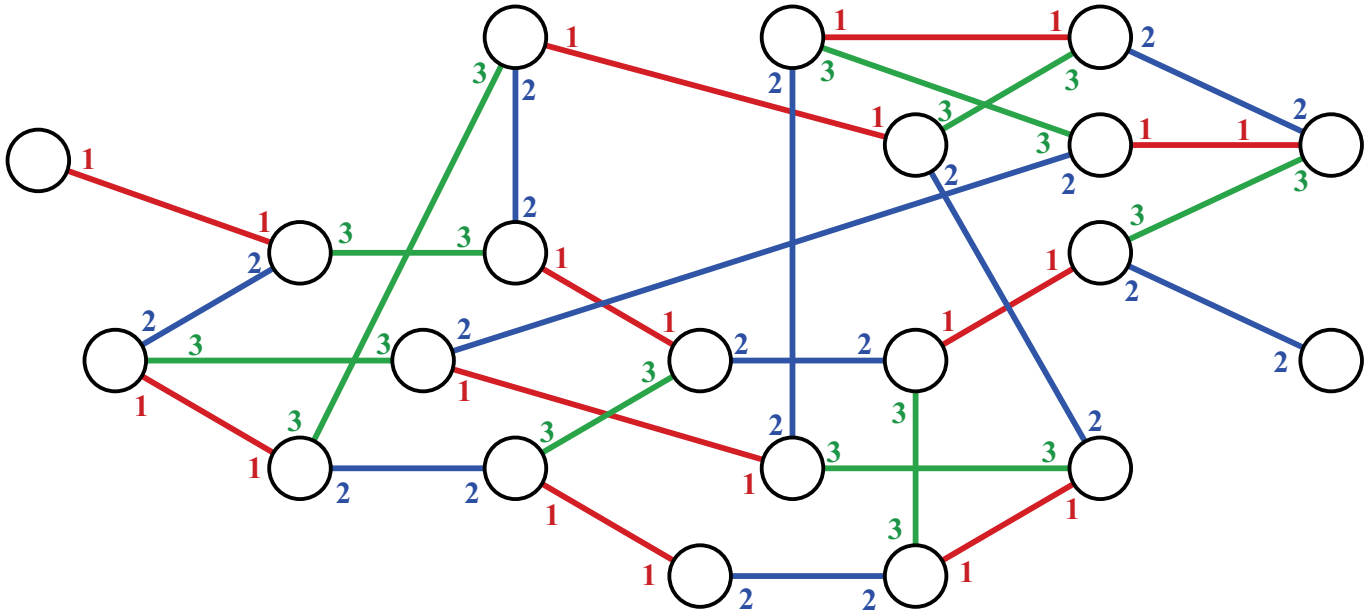
Connect x to $y_k(x)$ with an edge of weight $\alpha_k(x)$





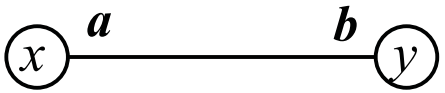
Institute for Quantum Information Science at the University of Calgary

Symmetrically labeled graphs

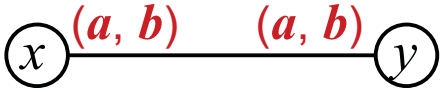


Non-symmetric case

Modify labeling to be symmetric (with an overhead cost)

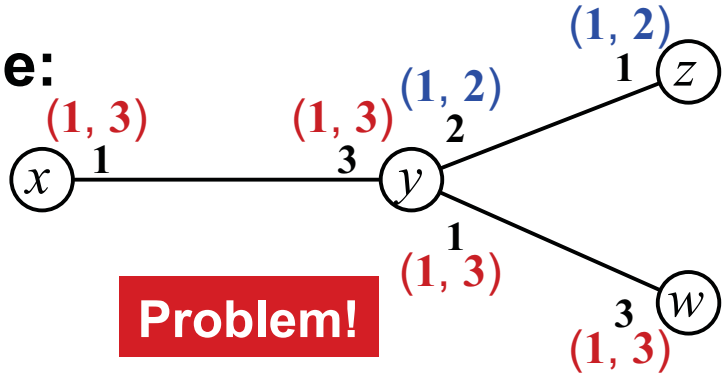


with $x < y$



We now have d^2 labels instead of d labels, but a ***symmetric*** labeling

Example:

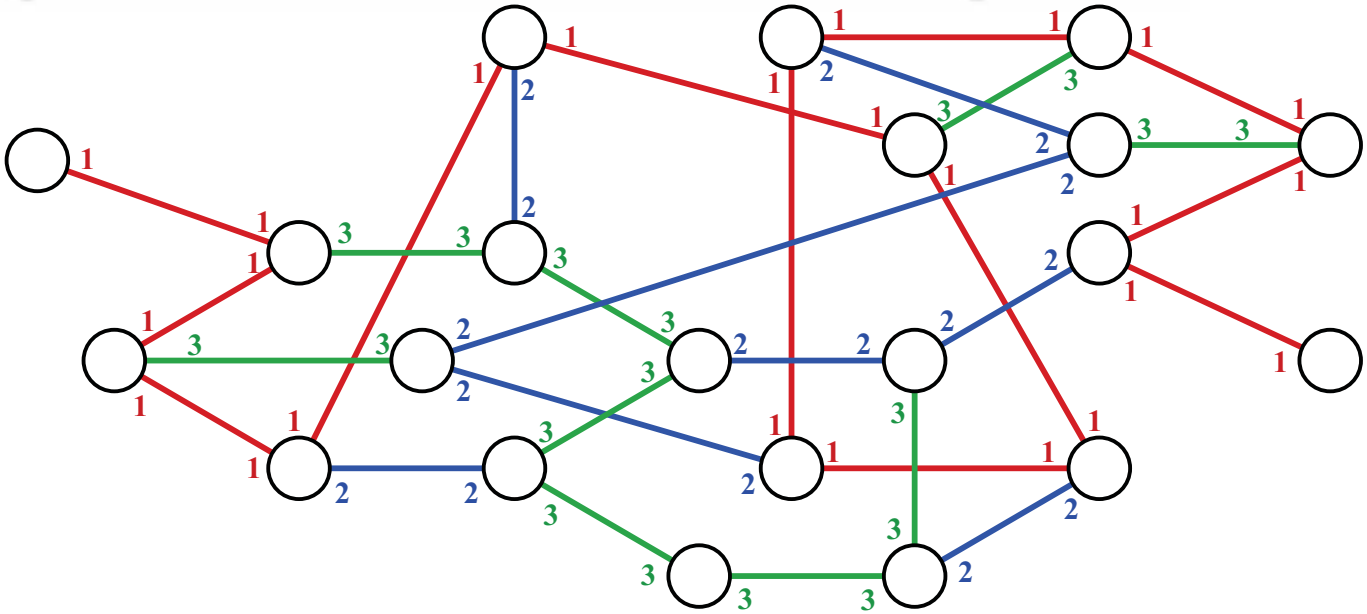


with $z < y$

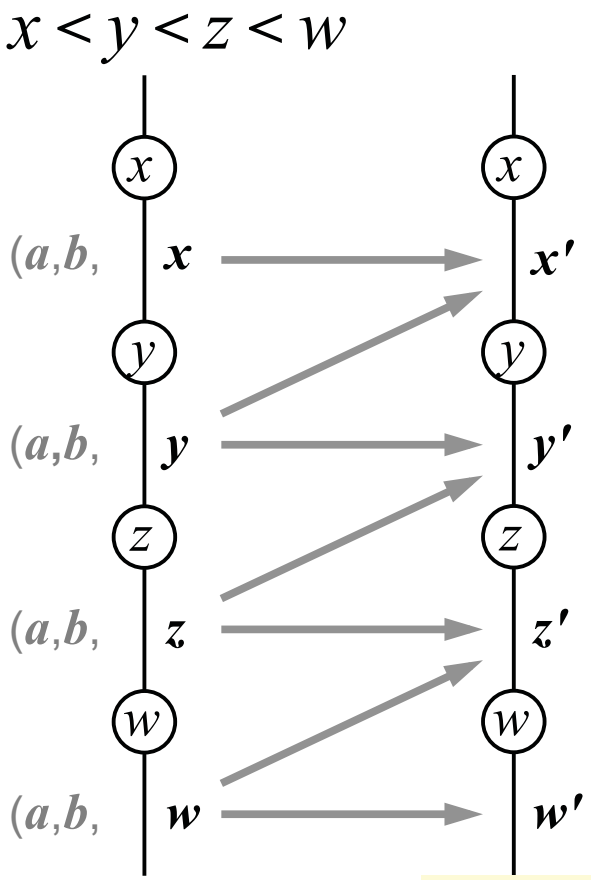
with $y < w$

Problem!

Graph with monochromatic paths



To break up the paths, we increase the number of colours



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

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

Example: $y = 01100101$ (010)

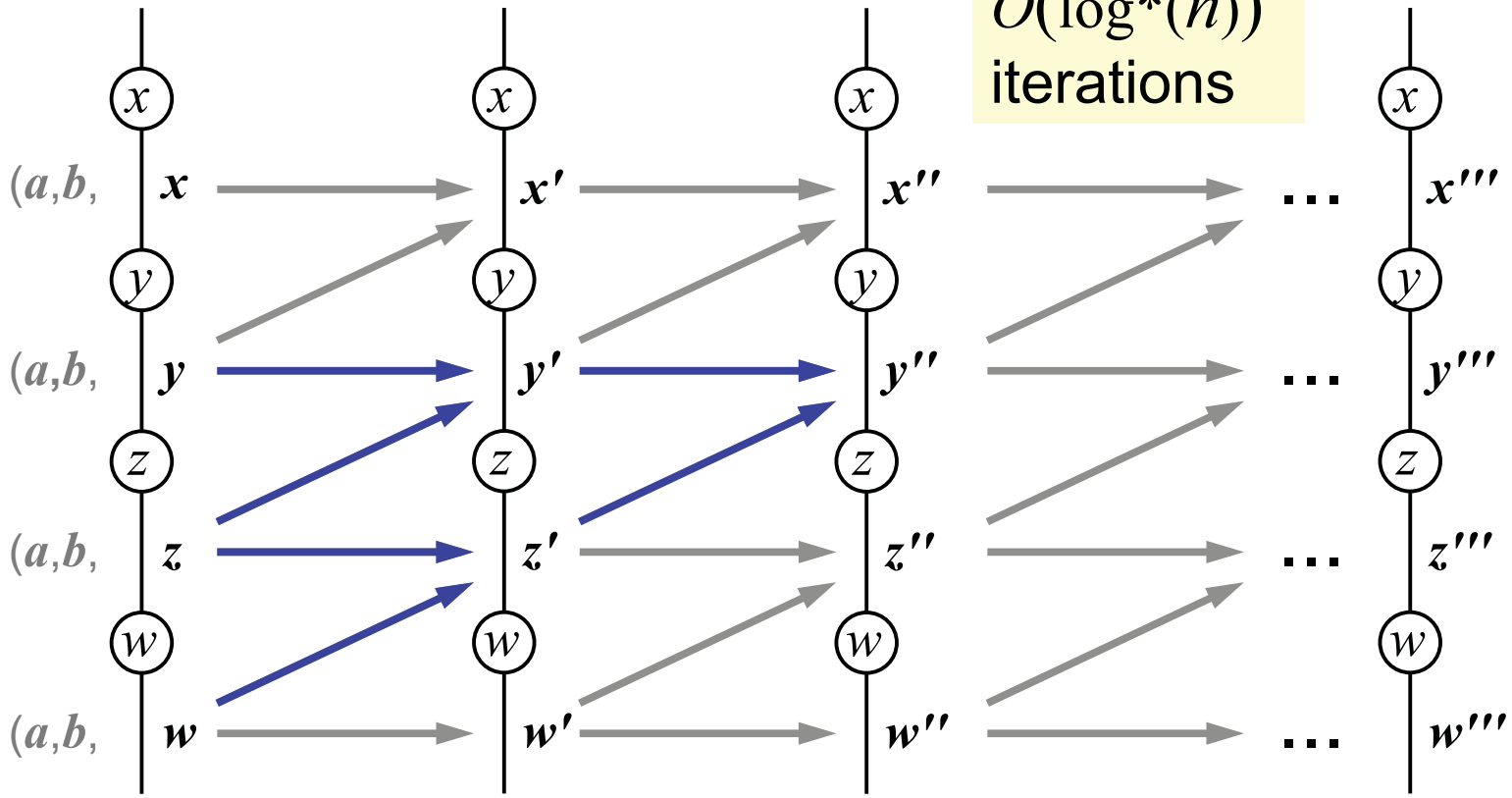
$z = 01001101$

Then $y' = (010,1)$

Note: still a valid coloring!

$x' \neq y'$ & $y' \neq z'$ & $z' \neq w'$

Breaking up the paths II



$d^2\,2^n$ colors

n bits

$\log(n)+1$ bits

$\log(\log(n)+1)+1$ bits

6 elements

Just 5 iterations for $n \leq 10^{10^{37}}$

Institute for Quantum Information Science at the University of Calgary

Theorem: The number of black-box calls for given k is

$$N_{\mathrm{bb}} \in O\left((\log^* n) d^2 5^{2k} (d^2 q_k \tau)^{1+1/2k} / [(2k+1)!\epsilon]^{1/2k}\right)$$

with $\log^* n \equiv \min\{r | \log_2^{(r)} n < 2\}$ (the $^{(r)}$ indicating the iterated logarithm).

Sketch of Proof:

of H_j's is $m=6d^2$. Need to call the black-box O($\log^* n$) times for each H_j.

Substituting into theorem for upper bound on N_{exp} gives result.

Further reducing time cost

- Childs & Kothari 2010: deterministic coin-tossing colouring of vertices rather than edges, decompose H into a sum of d galaxies (disjoint union of star graphs)
  - $T \in O([d^3+d^2\log^* n]t^{1+1/2k}/\varepsilon^{1/2k})$
  - $S \in O(nd+n\log^* n)$.
- Berry & Childs 2010: Q walk & H-eigen estimate replaces Lie-Trotter-Suzuki formula. Transform to H-eigenbasis, evolve then revert to computational basis.
  - $T \in O(|H|_{max} dt\varepsilon^{-1/2})$.

INTEGRATING SCHRÖDINGER EVOLUTION

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

Simulation of Many-Body Fermi Systems on a Universal Quantum Computer

Daniel S. Abrams
Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Seth Lloyd
Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139
(Received 7 November 1996)

We provide fast algorithms for simulating many-body Fermi systems on a universal quantum computer. Both first and second quantized descriptions are considered, and the relative computational complexities are determined in each case. In order to accommodate fermions using a first quantized Hamiltonian, an efficient quantum algorithm for antisymmetrization is given. Finally, a simulation of the Hubbard model is discussed in detail. [S0031-9007(97)04120-3]

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

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

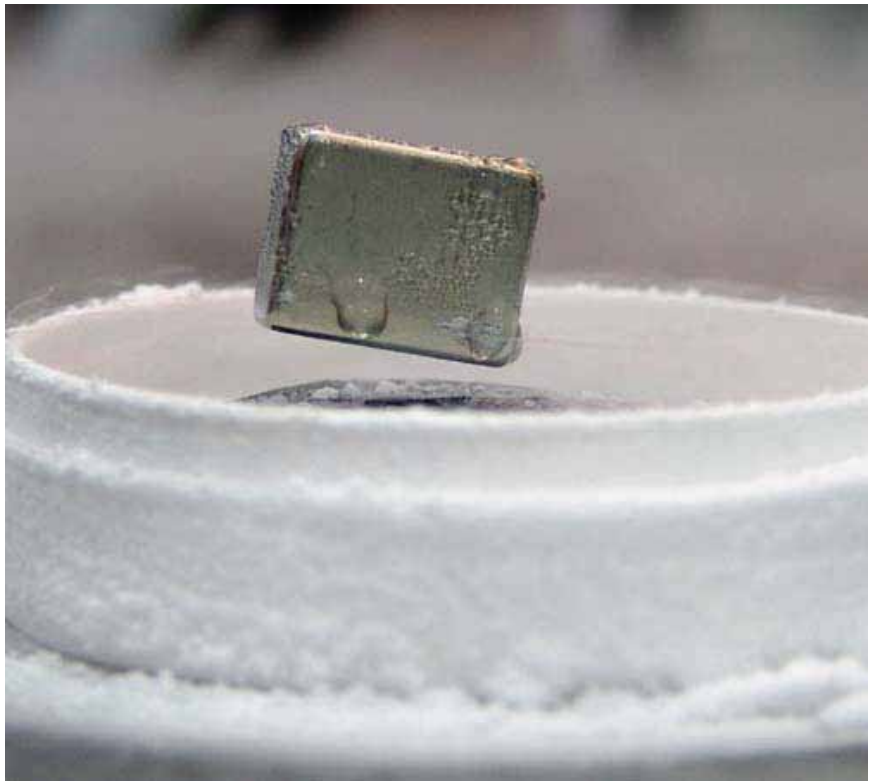
Simulation of Many-Body Fermi Systems on a Universal Quantum Computer

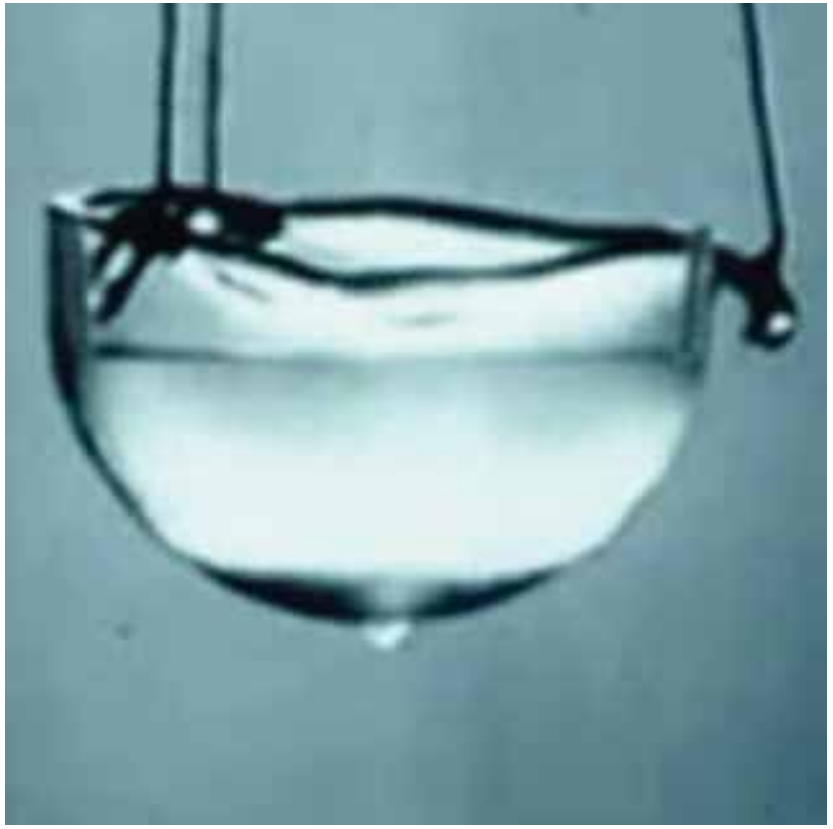
Daniel S. Abrams
Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Seth Lloyd
Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139
(Received 7 November 1996)

computer remain daunting [12–15]. But the problem of simulation — that is, the problem of modeling the full time evolution of an arbitrary quantum system — is less technologically demanding. While thousands of qubits and billions of quantum logic operations are needed to solve classically difficult factoring problems [16], it would be possible to use a quantum computer with only a few tens of qubits and a few thousand operations to perform simulations that would be classically intractable [17]. A quantum computer of this scale appears to be a realistic possibility.

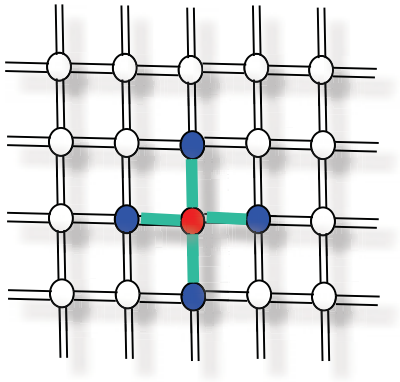
Quantum materials





Examples of spin systems for simulation

$$\hat{H}_{\text{Ising}} = J\sum_{\langle i,j\rangle} Z_i \otimes Z_j + B\sum_i X_i$$



$$\hat{H}_{\text{XY}} = J_x\sum_{\langle i,j\rangle} X_i \otimes X_j + J_y\sum_{\langle i,j\rangle} Y_i \otimes Y_j$$

$$\hat{H}_{\text{Heisenberg}} = J_x\sum_{\langle i,j\rangle} X_i \otimes X_j + J_y\sum_{\langle i,j\rangle} Y_i \otimes Y_j + J_z\sum_{\langle i,j\rangle} Z_i \otimes Z_j + B\sum_i X_i$$

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

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

Heisenberg Model Hamiltonian Matrices

$$H = \sum_{i=1}^{n-1} I^{\otimes i-1} \sigma_i \cdot \sigma_{i+1} I^{\otimes n-1-i}$$

The number of non-zero elements in a nearest neighbor 1-D Hamiltonian is proportional to the number of qubits in the chain:

n=2,d=2

1	0	0	0
0	-1	2	0
0	2	-1	0
0	0	0	1

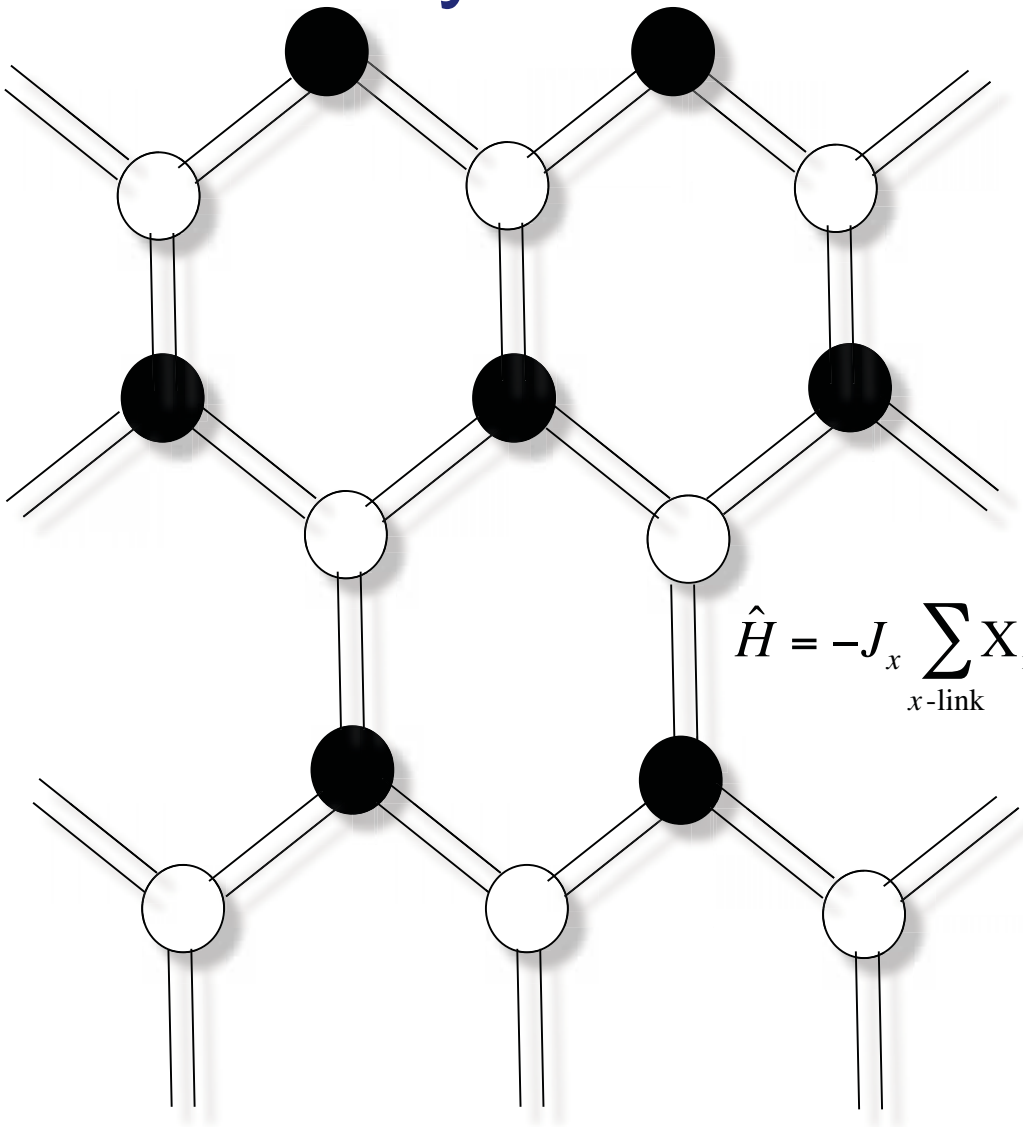
n=3,d=3

2	0	0	0	0	0	0	0
0	0	2	0	0	0	0	0
0	2	-2	0	2	0	0	0
0	0	0	0	0	2	0	0
0	0	2	0	0	0	0	0
0	0	0	2	0	-2	2	0
0	0	0	0	0	2	0	0
0	0	0	0	0	0	0	2

n=4,d=4

3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0
0	2	-1	0	2	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	2	0	0	0	0	0	0	0	0	0	0
0	0	2	0	-1	0	0	0	2	0	0	0	0	0	0	0
0	0	0	2	0	-3	2	0	0	2	0	0	0	0	0	0
0	0	0	0	0	2	-1	0	0	0	2	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	2	0	0	0	0
0	0	0	0	2	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	2	0	0	0	-1	2	0	0	0	0	0
0	0	0	0	0	0	2	0	0	2	-3	0	2	0	0	0
0	0	0	0	0	0	0	2	0	0	0	-1	0	2	0	0
0	0	0	0	0	0	0	0	0	0	2	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	2	0	-1	2	0
0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3

Kitaev's honeycomb lattice



$$\hat{H} = -J_x \sum_{x\text{-link}} \mathrm{X}_\mathrm{i} X_j - J_y \sum_{y\text{-link}} \mathrm{Y}_\mathrm{i} Y_j - J_z \sum_{x\text{-link}} \mathrm{Z}_\mathrm{i} Z_j$$

Analogue vs Digital Q Simulator

- Analogue Q Simulator
  - designed to evolve similarly to system being simulated
  - e.g. q magnetism or superfluidity.
- Digital Q Simulator
  - universal or purpose-built programmable q computer
  - Simulate dynamics via a sequence of Hamiltonian-generated evolutions.

Algorithm for many-body q simulation I

- Input: Time t, tolerance ε and Hamiltonian

$$\hat{H}^{(n)} = \sum_{j=1}^{m} a_j^{(n)} \hat{h}_j^{(n)}, \hat{h}_j^{(n)} = \otimes_{\ell=1}^{n} \hat{\Xi}_{j\ell}^{(n)},$$

$$\hat{\Xi}_{j\ell}^{(n)} \in \left\{ I, X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = i\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, Z = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \right\}$$

(Total number of non - identity operators in each summand is at most n - independent constant k.)

- Output: Q Circuit

Algorithm for many-body q simulation II

- Expressed as string of bits

$$\left[\hat{H}^{(n)}\right] := \left\{a_j, \mathbf{l}_j, \mathbf{S}_j; j = 1,\ldots,j\right\},$$

$\mathbf{l}_j = \left(l_{X_j}, l_{Y_j}, l_{Z_j}\right)$ gives number of each type of Pauli operator,

$\mathbf{S}_j = \left(\mathbf{S}_{X_j}, \mathbf{S}_{Y_j}, \mathbf{S}_{Z_j}\right)$ comprises vectors of strings corresponding to positions of each of the Pauli operators.

This string is poly(n) in size.

Honeycomb model as an example

- One-site interaction Hamiltonian

$$\hat{H}^{(n)} = X^{\otimes 2} \otimes I^{\otimes 2} + 2\left(Y \otimes I\right)^{\otimes 2} + 4Z \otimes I^{\otimes 2} \otimes Z$$

- Bit-string representation

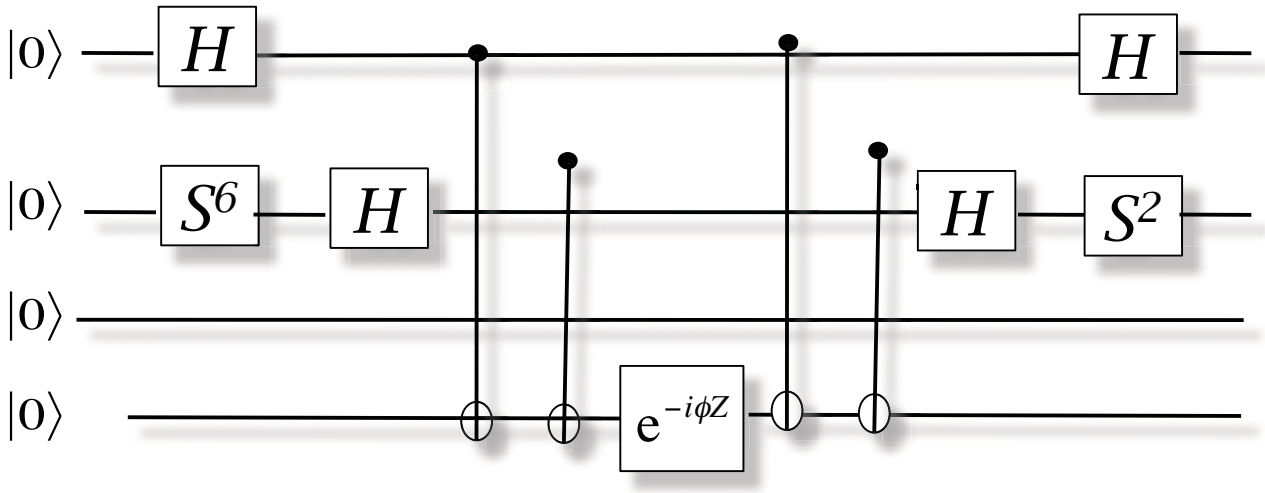
$$\mathbf{a} = (1,2,4),$$

$$\mathbf{l}_1 = (2,0,0), \mathbf{S}_{X_1} = (1,2),$$

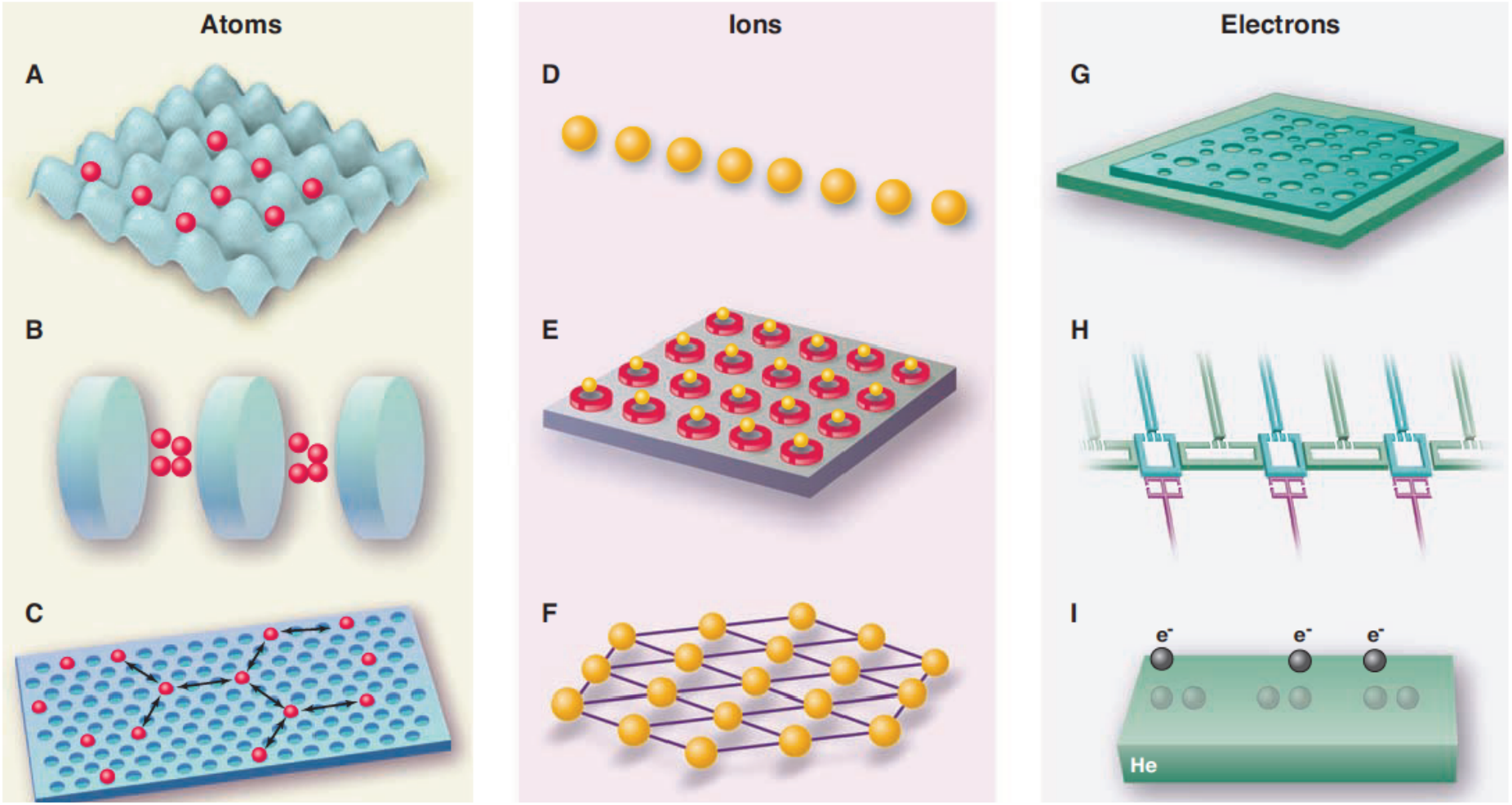
$$\mathbf{l}_2 = (0,2,0), \mathbf{S}_{Y_2} = (1,3),$$

$$\mathbf{l}_3 = (0,0,2), \mathbf{S}_{Z_3} = (1,4).$$

Q Circuit component for Pauli evolution



$$\exp\left(-i\phi X \otimes Y \otimes I \otimes Z\right)$$



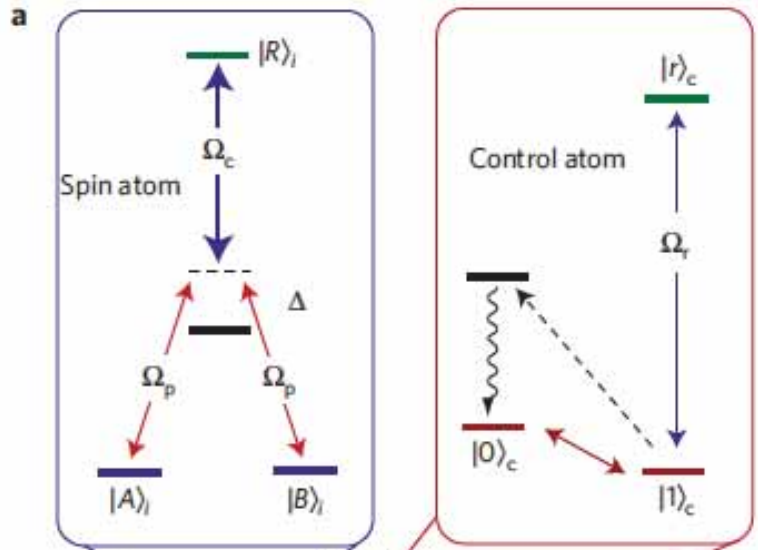
Buluta and Nori, “Quantum Simulators”, Science **326**, 108 (2009).

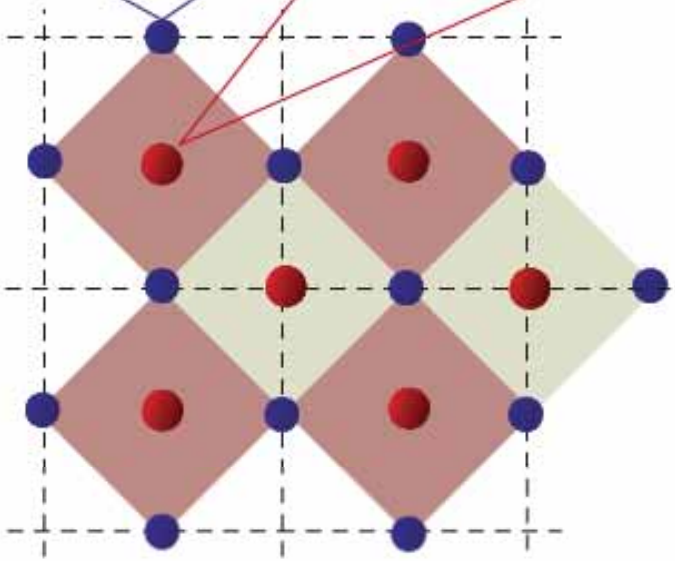
ARTICLES
PUBLISHED ONLINE: 14 MARCH 2010 | DOI: 10.1038/NPHYS1614
nature physics

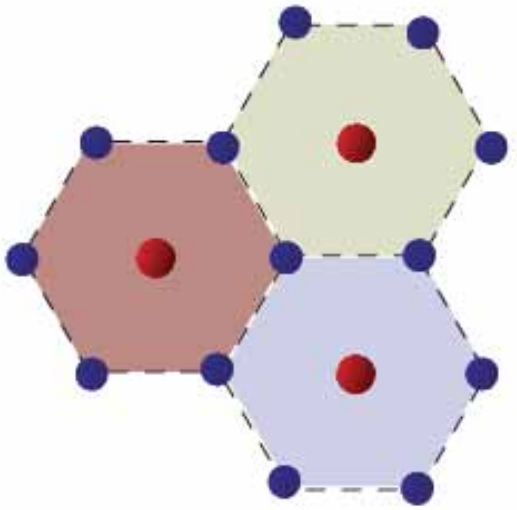
A Rydberg quantum simulator

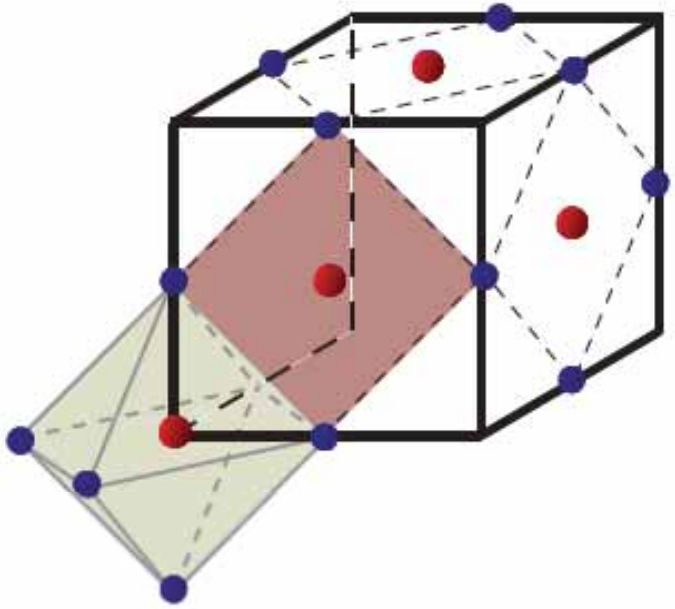
Hendrik Weimer[1]*, Markus Müller[2], Igor Lesanovsky[2,3], Peter Zoller[2] and Hans Peter Büchler[1]

A universal quantum simulator is a controlled quantum device that reproduces the dynamics of any other many-particle quantum system with short-range interactions. This dynamics can refer to both coherent Hamiltonian and dissipative open-system evolution. Here we propose that laser-excited Rydberg atoms in large-spacing optical or magnetic lattices provide an efficient implementation of a universal quantum simulator for spin models involving *n*-body interactions, including such of higher order. This would allow the simulation of Hamiltonians of exotic spin models involving *n*-particle constraints, such as the Kitaev toric code, colour code and lattice gauge theories with spin-liquid phases. In addition, our approach provides the ingredients for dissipative preparation of entangled states based on engineering *n*-particle reservoir couplings. The basic building blocks of our architecture are efficient and high-fidelity *n*-qubit entangling gates using auxiliary Rydberg atoms, including a possible dissipative time step through optical pumping. This enables mimicking the time evolution of the system by a sequence of fast, parallel and high-fidelity *n*-particle coherent and dissipative Rydberg gates.


a
$|R\rangle_i$
Ω_c
Spin atom
Δ
Ω_p
Ω_p
$|A\rangle_i$
$|B\rangle_i$
$|r\rangle_c$
Control atom
Ω_r
$|0\rangle_c$
$|1\rangle_c$


b


c


d

Universal Digital Quantum Simulation with Trapped Ions

B. P. Lanyon,[1,2*] C. Hempel,[1,2] D. Nigg,[2] M. Müller,[1,3] R. Gerritsma,[1,2] F. Zähringer,[1,2] P. Schindler,[2] J. T. Barreiro,[2] M. Rambach,[1,2] G. Kirchmair,[1,2] M. Hennrich,[2] P. Zoller,[1,3] R. Blatt,[1,2] C. F. Roos[1,2]

A digital quantum simulator is an envisioned quantum device that can be programmed to efficiently simulate any other local system. We demonstrate and investigate the digital approach to quantum simulation in a system of trapped ions. With sequences of up to 100 gates and 6 qubits, the full time dynamics of a range of spin systems are digitally simulated. Interactions beyond those naturally present in our simulator are accurately reproduced, and quantitative bounds are provided for the overall simulation quality. Our results demonstrate the key principles of digital quantum simulation and provide evidence that the level of control required for a full-scale device is within reach.

Although many natural phenomena are accurately described by the laws of quantum mechanics, solving the associated equations to calculate properties of physical systems, i.e., simulating quantum physics, is in general thought to be very difficult (*1*). Both the number of parameters and differential equations that describe a quantum state and its dynamics grow exponentially with the number of particles involved. One proposed solution is to build a

highly cor
ficiently p
quantum
several dif
ing the an
gous mode
the state a
and those
is dedicate
problems.

A digi
precisely
tem on wh
(gates) can
(*17*). The

[1]Institut für
reichische Ak
A-6020 Innsb
versity of In
Austria. [3]In
Innsbruck, Te

*To whom c
ben.lanyon@

Institute for Quantum Information Science at the University of Calgary

Time-dependent Hamiltonian evolution

For $H = \sum_{j=1}^{m} H_j$, $H_j : R \to C^{N\times N}$ P - differentiable, construct

$U(t, t+\Delta t) \equiv \mathrm{T}\exp\left(-i\int_{t}^{t+\Delta t} H(u)\mathrm{d}u\right)$ as a product of N exponentials

$\exp\left(-iH_{j_p}(t_p)\Delta t_p\right)$ within tolerance ε of $U(t, t+\Delta t)$, and find an upper bound for N.

Theorem

Each $H_j : R \to C^N \times C^N$ is $2k$ - differentiable on $[t, t+\Delta t]$.

$$\Lambda_{2k} = \sup_{p=0,1,\ldots,2k} \left(\sup_{u \in [t, t+\Delta t]} \left(\sum_{j=1}^{m} \left\| H_j^{(p)}(u) \right\| \right)^{\frac{1}{p+1}} \right)$$

For $\varepsilon \le (5/3)^{k-1} \Lambda_{2k} \Delta t, s_p = \dfrac{1}{4 - 4^{1/2p+1}}$,

$$N \le (2m-1)5^{k-1} r \le (2m-1)5^{k-1} \left\lceil 2\left(2k \cdot (5/3)^{k-1} \Lambda_{2k} \Delta t\right)^{1+1/2k} \varepsilon^{-1/2k} \right\rceil$$

Suzuki iteration

$$U_{p+1}\left(\mu+\Delta\lambda,\mu+\left[1-s_p\right]\Delta\lambda\right)\equiv U_p\left(\mu+\left[1-s_p\right]\Delta\lambda,\mu+\left[1-2s_p\right]\Delta\lambda\right)$$

$$\times U_p\left(\mu+\left[1-2s_p\right]\Delta\lambda,\mu+2s_p\Delta\lambda\right)U_p\left(\mu+2s_p\Delta\lambda,\mu+s_p\Delta\lambda\right)U_p\left(\mu+s_p\Delta\lambda,\mu\right),$$

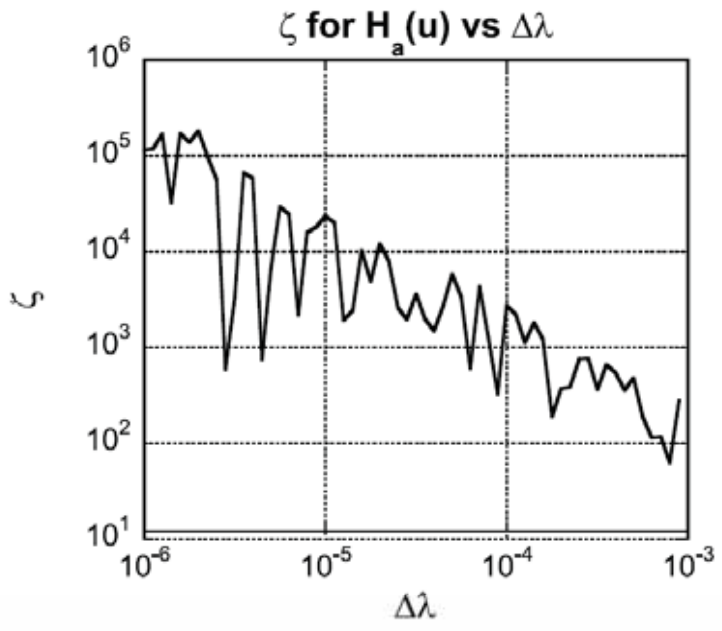
$$U_1\left(\mu+\Delta\lambda,\mu\right)\equiv\left(\prod_{j=1}^{m}\exp\left[A_j\left(\mu+\Delta\lambda/2\right)\Delta\lambda/2\right]\right)\left(\prod_{j=m}^{1}\exp\left[A_j\left(\mu+\Delta\lambda/2\right)\Delta\lambda/2\right]\right).$$

Problems arise for functions that are not smooth in time.

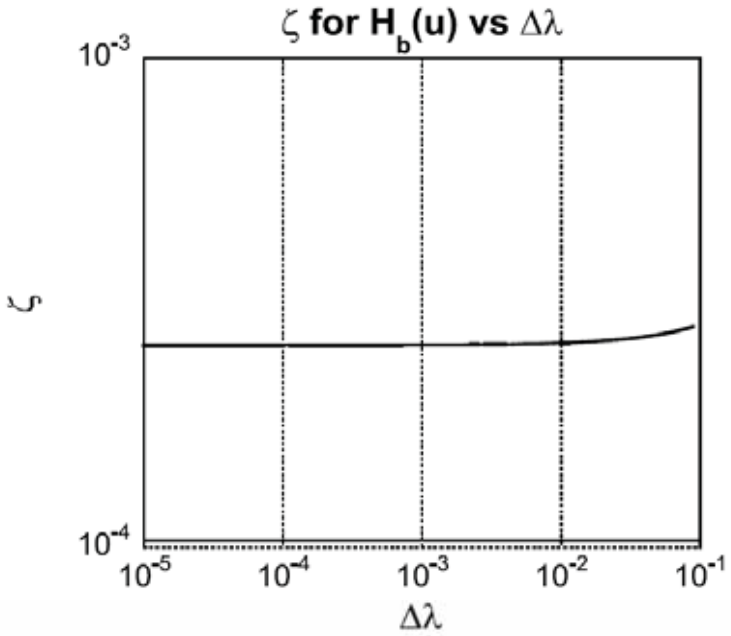
Problems with Lie-Trotter-Suzuki

- Suzuki's bound fails for non-analytic H
- Uses time-derivative super-operator

$$\zeta = \left\| U(\Delta\lambda,0) - U_2(\Delta\lambda,0) \right\|_2 / \Delta\lambda^5, H_a(\lambda) = \lambda^3 \sin(1/\lambda) I, H_b(\lambda) = \cos(\lambda) I$$



(a)



(b)

Lemma (no time-derivative superoperator)

$$T_0(t) \equiv 1,\ T_{p+1}(t) \equiv T_p(t)H(t) + i\dot{T}_p(t).$$

$$\left\| U(t,t+\Delta t) - \sum_{p=0}^{P} \frac{(-i\Delta t)^p T_p}{p!} \right\| \leq \frac{\max_{u\in[t,t+\Delta t]} \left\| T_{P+1}(u) \right\| \Delta t^{P+1}}{(P+1)!}$$

Optimal choice of *k*

$$k_{\text{optimal}} = \left\lceil \frac{1}{2}\left[\sqrt{\log_5\left(\frac{\Lambda\Delta t}{\varepsilon}\right) + 1} - 1 \right] \right\rceil$$

$$N_{\text{exp}} \leq 6m\Lambda\Delta t \exp\left[2\sqrt{\log_5\left(\Lambda\Delta t/\varepsilon\right)}\right]$$

Method

- Taylor expansion for U to $O(\Delta t^{2k+1})$
- Induction: U_k equals U to order Δt^{2k+1}
- Avoid Suzuki's time-derivative superoperator

$$\left\| U - \sum_{\xi=0}^{p} \frac{(\Delta t)^{\xi} T_{\xi}}{\xi!} \right\|, T_{\xi+1}(t) \equiv T_{\xi}(t) H(t) + \partial_t T_{\xi}(t)$$

- As Taylor expansions for U and U_k are identical for $(\Delta t)^{2k}$, the norm of the difference is bounded by the sum of terms $O((\Delta t)^{2k+1})$.
- Triangle inequality bounds norm of difference.

Theorem 1 *Let* $H(t) = \sum_{j=1}^{m} H_j(t)$ *where each* $H_j(t)$ *is differentiable* $2k$ *times on* $[\mu, \mu + \Delta\lambda]$. *Furthermore let the timescale* Λ *satisfy,*

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

$\epsilon \leq (9/10)(5/3)^k \Lambda \Delta\lambda$ *and* $\max_{x>y} \|U(x,y)\| \leq 1$, *then a decomposition* $\tilde{U}(\mu + \Delta\lambda, \mu)$ *can be constructed such that* $\|\tilde{U} - U\| \leq \epsilon$ *and the number of operator exponentials present in* $\tilde{U}$, N, *satisfies*

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

Method

- Compute the Taylor Series of U and U_k to order $\Delta\lambda^{2k+1}$
- Show that Suzuki's choice of s_k causes the error term to be 0, if Taylor series exists.
- τ is not needed in our analysis.

Steps to prove error bounds

- Compute Taylor expansion of $U(t,t+\Delta t)$ as powers of Δt with $U(t,t+\Delta t)$ computed iteratively for k the index of the iterant.
- # terms for Taylor expansion of U_k is exponential in k for truncation at $(\Delta t)^l$ for some l so we prove instead that

$$\left\| U - U_k \right\| \in O(\Delta t^{2k+1}).$$

Steps to prove error bounds

- As Taylor expansions for U and U_k are identical for $(\Delta t)^{2k}$, the norm of the difference is bounded by the sum of terms $O((\Delta t)^{2k+1})$.
- Use the triangle inequality to bound the norm of the difference.
- Obtain error as a function r intervals.

Quantum Algorithm for Linear Systems of Equations

Aram W. Harrow,[1] Avinatan Hassidim,[2] and Seth Lloyd[3]

[1]*Department of Mathematics, University of Bristol, Bristol, BS8 1TW, United Kingdom*
[2]*Research Laboratory for Electronics, MIT, Cambridge, Massachusetts 02139, USA*
[3]*Research Laboratory for Electronics and Department of Mechanical Engineering, MIT, Cambridge, Massachusetts 02139, USA*

(Received 5 July 2009; published 7 October 2009)

Solving linear systems of equations is a common problem that arises both on its own and as a subroutine in more complex problems: given a matrix A and a vector $\vec{b}$, find a vector $\vec{x}$ such that $A\vec{x} = \vec{b}$. We consider the case where one does not need to know the solution $\vec{x}$ itself, but rather an approximation of the expectation value of some operator associated with $\vec{x}$, e.g., $\vec{x}^\dagger M\vec{x}$ for some matrix M. In this case, when A is sparse, $N \times N$ and has condition number κ, the fastest known classical algorithms can find $\vec{x}$ and estimate $\vec{x}^\dagger M\vec{x}$ in time scaling roughly as $N\sqrt{\kappa}$. Here, we exhibit a quantum algorithm for estimating $\vec{x}^\dagger M\vec{x}$ whose runtime is a polynomial of $\log(N)$ and κ. Indeed, for small values of κ [i.e., poly $\log(N)$], we prove (using some common complexity-theoretic assumptions) that any classical algorithm for this problem generically requires exponentially more time than our quantum algorithm.

$$A\vec{x} = \vec{b}, \vec{b} \mapsto |b\rangle = \sum_{i=1}^{N} b_i |i\rangle, e^{iAt}|b\rangle, \ldots$$

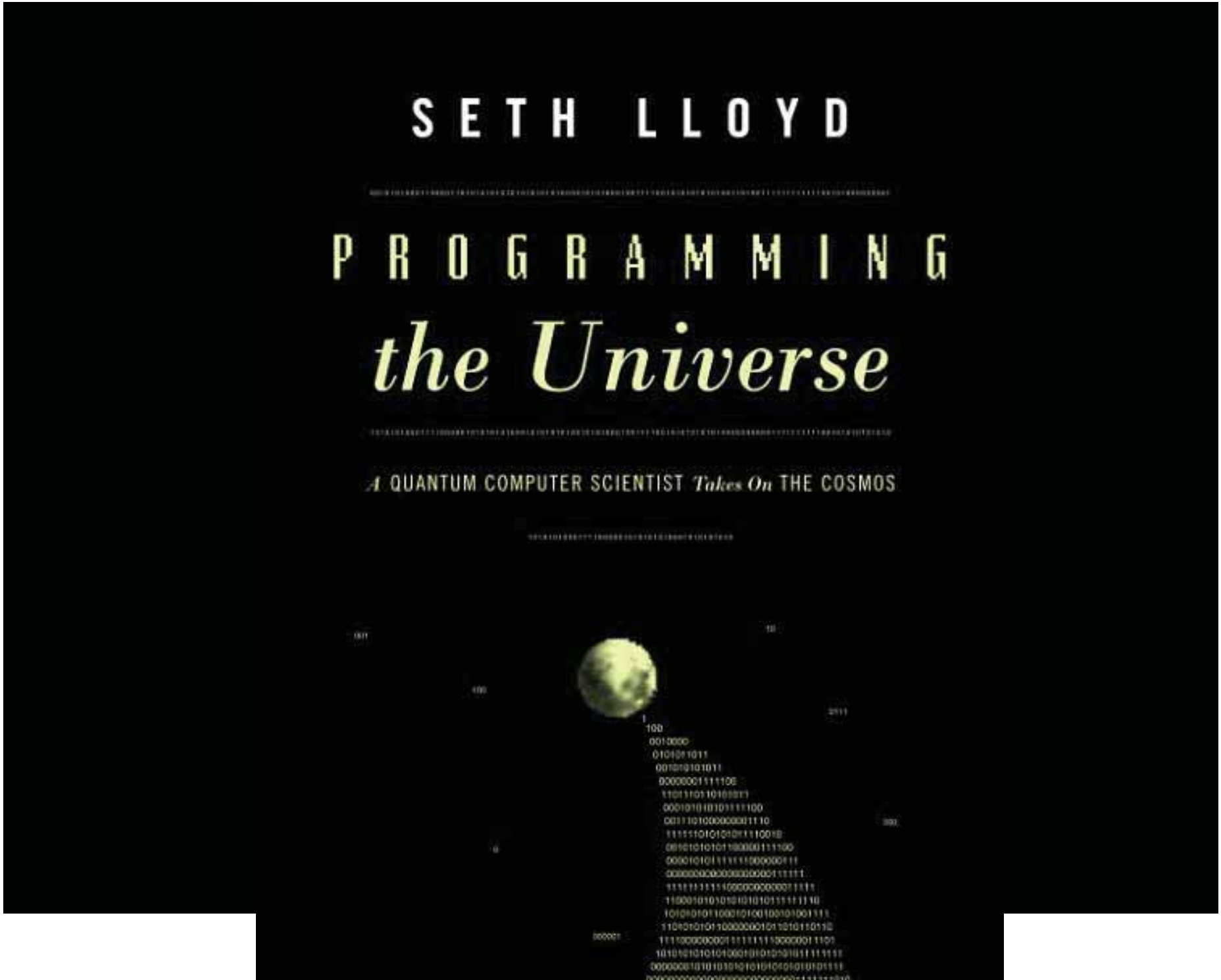
HHL Strategy

- Use (modified) Kitaev's eigenvalue estimation algorithm for Hermitian A.
- Prepare and inject $|b\rangle$ as input.
- Then approximate $|b\rangle$ in A-eigenbasis $\{|u_j\rangle\}$ with corresponding eigenvalues $\{\lambda_j\}$.
- Requires $O(n\log n)$ steps.
- Kitaev q algorithm output: eigenvalues and corresponding eigenvectors of A.
- Then controlled rotations on $|b\rangle$ and then undo to get $|x\rangle$.

SETH LLOYD

PROGRAMMING *the Universe*

A QUANTUM COMPUTER SCIENTIST *Takes On* THE COSMOS



“I [hypothesize] that ultimately physics will not require a mathematical statement, that in the end the machinery will be revealed, and the laws will turn out to be simple, like the checker board with all its apparent complexities.” - Feynman