The relationship between continuous- and discrete-time quantum walk

Andrew Childs

Department of Combinatorics & Optimization
and Institute for Quantum Computing
University of Waterloo

arXiv:0810.0312 PacNQuInT 2009

Quantum walk algorithms

Exponential speedups

- Black box graph traversal [CCDFGS 03]
- Hidden sphere problem [CSV 07]

Polynomial speedups

- Search on graphs [Shenvi, Kempe, Whaley 02], [CG 03, 04], [Ambainis, Kempe, Rivosh 04]
- Element distinctness [Ambainis 03]
- Triangle finding [Magniez, Santha, Szegedy 03]
- Checking matrix multiplication [Buhrman, Špalek 04]
- Testing group commutativity [Magniez, Nayak 05]
- Formula evaluation [Farhi, Goldstone, Gutmann 07], [ACRŠZ 07], [Cleve, Gavinsky, Yeung 08], [Reichardt, Špalek 08]
- Unstructured search [Grover 96] (+ many applications)

Two models, both alike in dignity

	Continuous	**Discrete**
state space		
simplicity		
ease of implementation		

Two models, both alike in dignity

	Continuous	Discrete
state space	vertices	directed edges (“quantum coin”)
simplicity		
ease of implementation		

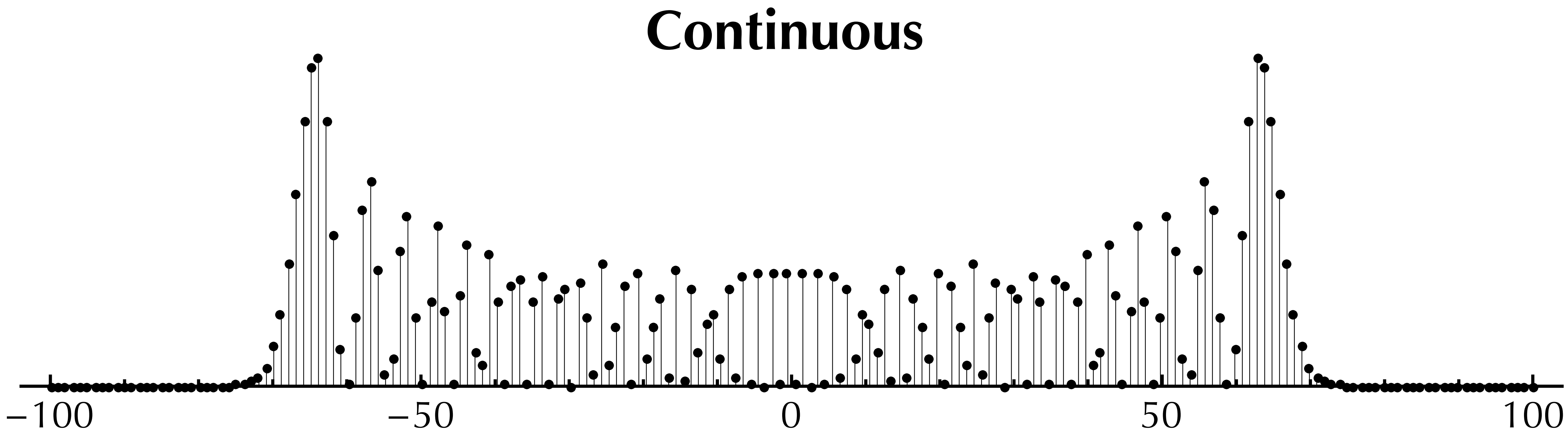
Two models, both alike in dignity

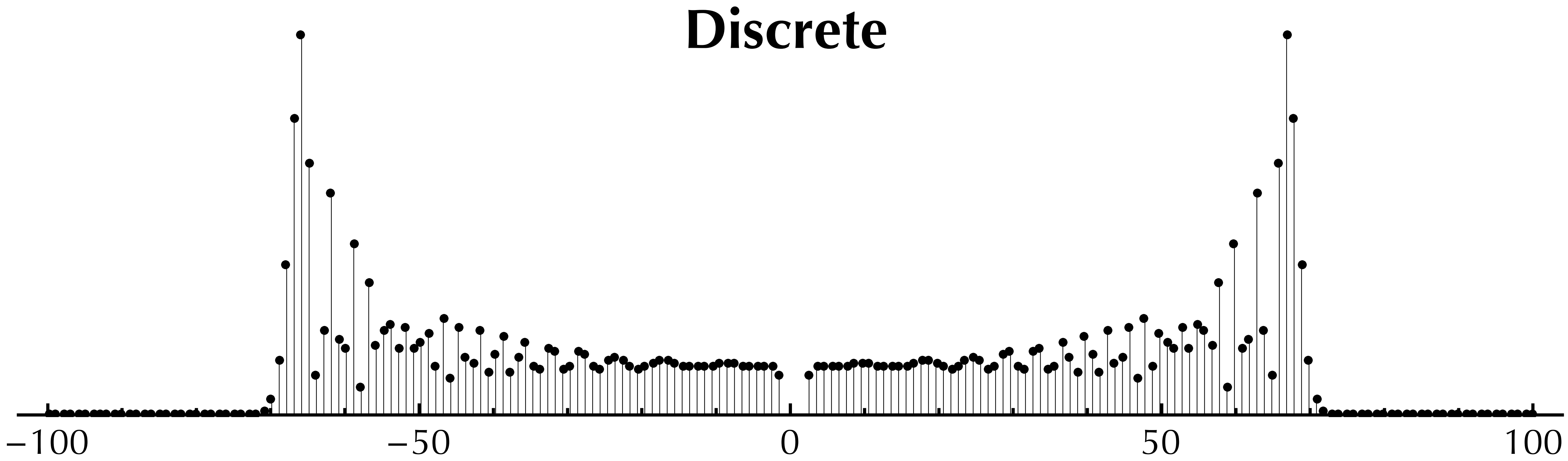
	Continuous	Discrete
state space	vertices	directed edges (“quantum coin”)
simplicity	✓	
ease of implementation		

Two models, both alike in dignity

	Continuous	**Discrete**
state space	vertices	directed edges (“quantum coin”)
simplicity	✓	
ease of implementation		✓

Walks on lines





Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)		
glued trees		
element distinctness		
balanced binary AND-OR trees		

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d > 4$ [CG 03]	
glued trees		
element distinctness		
balanced binary AND-OR trees		

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d > 4$ [CG 03]	$O(\sqrt{N})$ for $d > 2$ [AKR 04]
glued trees		
element distinctness		
balanced binary AND-OR trees		

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d > 4$ [CG 03] $O(\sqrt{N})$ for $d > 2$ [CG 04]	$O(\sqrt{N})$ for $d > 2$ [AKR 04]
glued trees		
element distinctness		
balanced binary AND-OR trees		

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d>4$ [CG 03] $O(\sqrt{N})$ for $d>2$ [CG 04]	$O(\sqrt{N})$ for $d>2$ [AKR 04]
glued trees	exponential speedup over classical [CCDFGS 03]	
element distinctness		
balanced binary AND-OR trees		

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d > 4$ [CG 03] $O(\sqrt{N})$ for $d > 2$ [CG 04]	$O(\sqrt{N})$ for $d > 2$ [AKR 04]
glued trees	exponential speedup over classical [CCDFGS 03]	?
element distinctness		
balanced binary AND-OR trees		

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d > 4$ [CG 03] $O(\sqrt{N})$ for $d > 2$ [CG 04]	$O(\sqrt{N})$ for $d > 2$ [AKR 04]
glued trees	exponential speedup over classical [CCDFGS 03]	?
element distinctness		$O(N^{2/3})$ [Ambainis 03]
balanced binary AND-OR trees		

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d > 4$ [CG 03] $O(\sqrt{N})$ for $d > 2$ [CG 04]	$O(\sqrt{N})$ for $d > 2$ [AKR 04]
glued trees	exponential speedup over classical [CCDFGS 03]	?
element distinctness	?	$O(N^{2/3})$ [Ambainis 03]
balanced binary AND-OR trees		

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d > 4$ [CG 03] $O(\sqrt{N})$ for $d > 2$ [CG 04]	$O(\sqrt{N})$ for $d > 2$ [AKR 04]
glued trees	exponential speedup over classical [CCDFGS 03]	?
element distinctness	?	$O(N^{2/3})$ [Ambainis 03]
balanced binary AND-OR trees	$O(\sqrt{N})$ [FGG 07]	

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d>4$ [CG 03] $O(\sqrt{N})$ for $d>2$ [CG 04]	$O(\sqrt{N})$ for $d>2$ [AKR 04]
glued trees	exponential speedup over classical [CCDFGS 03]	?
element distinctness	?	$O(N^{2/3})$ [Ambainis 03]
balanced binary AND-OR trees	$O(\sqrt{N})$ [FGG 07] $N^{\frac{1}{2}+o(1)}$ in circuit model [CCJY 07]	

Dueling algorithms

	Continuous	Discrete
searching a grid (d dimensions)	$O(\sqrt{N})$ for $d > 4$ [CG 03] $O(\sqrt{N})$ for $d > 2$ [CG 04]	$O(\sqrt{N})$ for $d > 2$ [AKR 04]
glued trees	exponential speedup over classical [CCDFGS 03]	?
element distinctness	?	$O(N^{2/3})$ [Ambainis 03]
balanced binary AND-OR trees	$O(\sqrt{N})$ [FGG 07] $N^{\frac{1}{2}+o(1)}$ in circuit model [CCJY 07]	$O(\sqrt{N})$ [ACRŠZ 07]

A formal equivalence?

Is there a single framework describing both kinds of walks?

A formal equivalence?

Is there a single framework describing both kinds of walks?

E.g., do the walks behave the same in some limit?

A formal equivalence?

Is there a single framework describing both kinds of walks?

E.g., do the walks behave the same in some limit?

Of course not! The state spaces aren't even the same!

Reconciliation

In fact, there is a close correspondence between the continuous- and discrete-time models (suitably defined).

In particular:

- There is a sequence of discrete-time quantum walks whose behavior (in an appropriate subspace) converges to the dynamics of the continuous-time quantum walk.
- By applying phase estimation instead of taking that limit, we can obtain the continuous-time quantum walk more efficiently. ($\Rightarrow$ improved simulations of Hamiltonian dynamics)

Outline

- Models
  - Classical and quantum, continuous- and discrete-time
  - Szegedy's theorem
  - Szegedizing Hamiltonians
- Continuous-time walk as a limit of discrete-time walks
- Hamiltonian simulation
- Applications
  - Algorithms
  - Hamiltonian oracles
- Open question: A sign problem for Hamiltonian simulation

Models

Random walk

A Markov process on a graph $G = (V, E)$.

Random walk

A Markov process on a graph $G = (V, E)$.

In discrete time:

Stochastic matrix $W \in \mathbb{R}^{|V| \times |V|}$ $\quad (W_{kj} \geq 0, \ \sum_k W_{kj} = 1)$

with $W_{kj} \neq 0$ iff $(j, k) \in E$

↑

probability of taking a step from j to k

Random walk

A Markov process on a graph $G = (V, E)$.

In discrete time:

Stochastic matrix $W \in \mathbb{R}^{|V| \times |V|}$ $\quad (W_{kj} \geq 0, \ \ \sum_k W_{kj} = 1)$

with $W_{kj} \neq 0$ iff $(j, k) \in E$

↑

probability of taking a step from j to k

Dynamics: $\quad p_t = W^t p_0 \qquad p_t \in \mathbb{R}^{|V|} \qquad t = 0, 1, 2, \ldots$

Random walk

A Markov process on a graph $G = (V, E)$.

In discrete time:

Stochastic matrix $W \in \mathbb{R}^{|V| \times |V|}$ $\quad (W_{kj} \geq 0, \;\; \sum_k W_{kj} = 1)$

with $W_{kj} \neq 0$ iff $(j, k) \in E$

↑

probability of taking a step from j to k

Dynamics: $\quad p_t = W^t p_0 \qquad p_t \in \mathbb{R}^{|V|} \qquad t = 0, 1, 2, \ldots$

Ex: Simple random walk. $\quad W_{kj} = \begin{cases} \frac{1}{\deg j} & (j, k) \in E \\ 0 & (j, k) \notin E \end{cases}$

Random walk

A Markov process on a graph $G = (V, E)$.

In continuous time:

Random walk

A Markov process on a graph $G = (V, E)$.

In continuous time:

Generator matrix $M \in \mathbb{R}^{|V| \times |V|}$ $(\sum_k M_{kj} = 0)$

with $M_{kj} \neq 0$ iff $(j, k) \in E$

↑

probability *per unit time* of taking a step from j to k

Random walk

A Markov process on a graph $G = (V, E)$.

In continuous time:

Generator matrix $M \in \mathbb{R}^{|V|\times|V|}$ $(\sum_k M_{kj} = 0)$

with $M_{kj} \neq 0$ iff $(j, k) \in E$

↑

probability *per unit time* of taking a step from j to k

Dynamics: $\frac{d}{dt}p(t) = Mp(t)$ $\quad p(t) \in \mathbb{R}^{|V|}$ $\quad t \in \mathbb{R}$

Random walk

A Markov process on a graph $G = (V, E)$.

In continuous time:

Generator matrix $M \in \mathbb{R}^{|V| \times |V|}$ $\left(\sum_k M_{kj} = 0\right)$

with $M_{kj} \neq 0$ iff $(j, k) \in E$

↑

probability *per unit time* of taking a step from j to k

Dynamics: $\dfrac{d}{dt} p(t) = M p(t) \qquad p(t) \in \mathbb{R}^{|V|} \qquad t \in \mathbb{R}$

Ex: Laplacian walk. $M_{kj} = \begin{cases} -\deg j & j = k \\ 1 & (j, k) \in E \\ 0 & (j, k) \notin E \end{cases}$

Continuous-time quantum walk

Quantum analog of a random walk on a graph $G = (V, E)$.

Continuous-time quantum walk

Quantum analog of a random walk on a graph $G = (V, E)$.

Idea: Replace probabilities by quantum amplitudes.

$$|\psi(t)\rangle = \sum_{v \in V} q_v(t)|v\rangle$$

↖ amplitude for vertex v at time t

Continuous-time quantum walk

Quantum analog of a random walk on a graph $G = (V, E)$.

Idea: Replace probabilities by quantum amplitudes.

$$|\psi(t)\rangle = \sum_{v \in V} q_v(t) |v\rangle$$

↖ amplitude for vertex v at time t

Define time-homogeneous, local dynamics on G.

$$i \frac{d}{dt} |\psi(t)\rangle = H |\psi(t)\rangle$$

Continuous-time quantum walk

Quantum analog of a random walk on a graph $G = (V, E)$.

Idea: Replace probabilities by quantum amplitudes.

$$|\psi(t)\rangle = \sum_{v \in V} q_v(t) |v\rangle$$

↖ amplitude for vertex v at time t

Define time-homogeneous, local dynamics on G.

$$i \frac{d}{dt} |\psi(t)\rangle = H |\psi(t)\rangle$$

$$H = H^\dagger \text{ with } H_{kj} \neq 0 \text{ iff } (j, k) \in E$$

Continuous-time quantum walk

Quantum analog of a random walk on a graph $G = (V, E)$.

Idea: Replace probabilities by quantum amplitudes.

$$|\psi(t)\rangle = \sum_{v \in V} q_v(t)|v\rangle$$

↖ amplitude for vertex v at time t

Define time-homogeneous, local dynamics on G.

$$i\frac{d}{dt}|\psi(t)\rangle = H|\psi(t)\rangle$$

$$H = H^\dagger \text{ with } H_{kj} \neq 0 \text{ iff } (j,k) \in E$$

Ex: Adjacency matrix. $H_{kj} = \begin{cases} 1 & (j,k) \in E \\ 0 & (j,k) \notin E \end{cases}$

Discrete-time quantum walk

We can also define a quantum walk with discrete steps. [Watrous 99]

Discrete-time quantum walk

We can also define a quantum walk with discrete steps. [Watrous 99]

Unitary matrix $U \in \mathbb{C}^{|V| \times |V|}$ with $U_{kj} \neq 0$ iff $(j, k) \in E$

Discrete-time quantum walk

We can also define a quantum walk with discrete steps. [Watrous 99]

Unitary matrix $U \in \mathbb{C}^{|V| \times |V|}$ with $U_{kj} \neq 0$ iff $(j, k) \in E$

Ex: On an infinite line, $|x\rangle \mapsto \frac{1}{\sqrt{2}}(|x-1\rangle + |x+1\rangle)$

Discrete-time quantum walk

We can also define a quantum walk with discrete steps. [Watrous 99]

Unitary matrix $U \in \mathbb{C}^{|V| \times |V|}$ with $U_{kj} \neq 0$ iff $(j, k) \in E$

Ex: On an infinite line, $|x\rangle \mapsto \frac{1}{\sqrt{2}}(|x-1\rangle + |x+1\rangle)$

but then $|x+2\rangle \mapsto \frac{1}{\sqrt{2}}(|x+1\rangle + |x+3\rangle)$

which is not orthogonal!

Discrete-time quantum walk

We can also define a quantum walk with discrete steps. [Watrous 99]

~~Unitary matrix $U \in \mathbb{C}^{|V| \times |V|}$ with $U_{kj} \neq 0$ iff $(j,k) \in E$~~

[Meyer 96], [Severini 03]

Ex: On an infinite line, $|x\rangle \mapsto \frac{1}{\sqrt{2}}(|x-1\rangle + |x+1\rangle)$

but then $|x+2\rangle \mapsto \frac{1}{\sqrt{2}}(|x+1\rangle + |x+3\rangle)$

which is not orthogonal!

In general, we must enlarge the state space.

Szegedy's discrete-time quantum walk

Szegedy's discrete-time quantum walk

State space: $\mathrm{span}\{|j,k\rangle, |k,j\rangle : (j,k) \in E\}$

Szegedy's discrete-time quantum walk

State space: $\mathrm{span}\{|j,k\rangle, |k,j\rangle : (j,k) \in E\}$

Let W be a stochastic matrix (a discrete-time random walk).

Szegedy's discrete-time quantum walk

State space: $\mathrm{span}\{|j,k\rangle, |k,j\rangle : (j,k) \in E\}$

Let W be a stochastic matrix (a discrete-time random walk).

Define $\quad |\psi_j\rangle := \sum_{k \in V} \sqrt{W_{kj}} |j,k\rangle \qquad$ (note $\langle \psi_j | \psi_k \rangle = \delta_{j,k}$)

Szegedy's discrete-time quantum walk

State space: $\mathrm{span}\{|j,k\rangle, |k,j\rangle : (j,k) \in E\}$

Let W be a stochastic matrix (a discrete-time random walk).

Define $$|\psi_j\rangle := \sum_{k \in V} \sqrt{W_{kj}}\,|j,k\rangle \qquad (\text{note } \langle\psi_j|\psi_k\rangle = \delta_{j,k})$$

$$R := 2\sum_{j \in V} |\psi_j\rangle\langle\psi_j| - I$$

Szegedy's discrete-time quantum walk

State space: $\mathrm{span}\{|j,k\rangle, |k,j\rangle : (j,k) \in E\}$

Let W be a stochastic matrix (a discrete-time random walk).

Define $$|\psi_j\rangle := \sum_{k \in V} \sqrt{W_{kj}}|j,k\rangle \qquad \text{(note } \langle\psi_j|\psi_k\rangle = \delta_{j,k}\text{)}$$

$$R := 2\sum_{j \in V} |\psi_j\rangle\langle\psi_j| - I$$

$$S|j,k\rangle := |k,j\rangle$$

Szegedy's discrete-time quantum walk

State space: $\mathrm{span}\{|j,k\rangle, |k,j\rangle : (j,k) \in E\}$

Let W be a stochastic matrix (a discrete-time random walk).

Define $$|\psi_j\rangle := \sum_{k \in V} \sqrt{W_{kj}}\,|j,k\rangle \qquad \text{(note } \langle\psi_j|\psi_k\rangle = \delta_{j,k}\text{)}$$

$$R := 2\sum_{j \in V} |\psi_j\rangle\langle\psi_j| - I$$

$$S|j,k\rangle := |k,j\rangle$$

Then a step of the walk is the unitary operator $U := iSR$.

Szegedy's spectral theorem

Theorem. Let $|\psi_j\rangle := \sum_k \sqrt{W_{kj}}|j,k\rangle$ where $\sum_k |W_{kj}| = 1$.

Suppose the matrix $\sum_{j,k} \sqrt{W_{jk}^* W_{kj}}|k\rangle\langle j|$ has an eigenvector $|\lambda\rangle$ with eigenvalue λ.

Let $T := \sum_j |\psi_j\rangle\langle j|$.

Then $\dfrac{I - e^{\pm i \arccos \lambda} S}{\sqrt{2(1-\lambda^2)}} T|\lambda\rangle$

are eigenvectors of $U := iS(2TT^\dagger - I)$ with eigenvalues $\pm e^{\pm i \arcsin \lambda}$.

Szegedizing a Hamiltonian

Idea: Let H be a Hermitian matrix. If we find a matrix W with $\sum_k |W_{kj}| = 1$ and

$$H_{jk} = h\sqrt{W_{jk}W^*_{kj}}$$

for some real number h, then W defines a discrete-time quantum walk closely related to H.

[ACRŠZ 07]

Szegedizing a Hamiltonian

Idea: Let H be a Hermitian matrix. If we find a matrix W with $\sum_k |W_{kj}| = 1$ and

$$H_{jk} = h\sqrt{W_{jk}W_{kj}^*}$$

for some real number h, then W defines a discrete-time quantum walk closely related to H.

Two strategies:

[ACRŠZ 07]

Szegedizing a Hamiltonian

Idea: Let H be a Hermitian matrix. If we find a matrix W with $\sum_k |W_{kj}| = 1$ and

$$H_{jk} = h\sqrt{W_{jk}W^*_{kj}}$$

for some real number h, then W defines a discrete-time quantum walk closely related to H.

Two strategies:

1. Let $\mathrm{abs}(H)$ denote the matrix with elements $\mathrm{abs}(H)_{jk} = |H_{jk}|$.

Let $|d\rangle = \sum_j d_j|j\rangle$ be the principal eigenvector of $\mathrm{abs}(H)$.

Then $W_{jk} = \frac{H_{jk}}{\|\mathrm{abs}(H)\|}\frac{d_k}{d_j}$ gives $h = \|\mathrm{abs}(H)\|$.

[ACRŠZ 07]

Szegedizing a Hamiltonian

Idea: Let H be a Hermitian matrix. If we find a matrix W with $\sum_k |W_{kj}| = 1$ and

$$H_{jk} = h\sqrt{W_{jk} W^*_{kj}}$$

for some real number h, then W defines a discrete-time quantum walk closely related to H.

Two strategies:

2. Let $\|H\|_1 := \max_j \sum_k |H_{jk}|$.

Introduce another state, denoted $|\varnothing\rangle$.

Then $W = \frac{H}{\|H\|_1} + \sum_k \left(1 - \sum_j \frac{|H_{jk}|}{\|H\|_1}\right) |\varnothing\rangle\langle k|$ gives $h = \|H\|_1$.

[ACRŠZ 07]

Continuous-time walk as a limit of discrete-time walks

Classical case

Discrete-time random walk: $p_{t+1} = W p_t$

Classical case

Discrete-time random walk: $p_{t+1} = W p_t$

"Lazy walk": Only move with probability ϵ, so $W \to \epsilon W + (1-\epsilon) I$

Classical case

Discrete-time random walk: $p_{t+1} = W p_t$

"Lazy walk": Only move with probability ϵ, so $W \to \epsilon W + (1-\epsilon)I$

$$p_{t+1} = [\epsilon W + (1-\epsilon)I]p_t$$

Classical case

Discrete-time random walk: $p_{t+1} = W p_t$

"Lazy walk": Only move with probability ϵ, so $W \to \epsilon W + (1-\epsilon) I$

$$
\begin{aligned}
p_{t+1} &= [\epsilon W + (1-\epsilon) I] p_t \\
\frac{p_{t+1} - p_t}{\epsilon} &= (W - I) p_t
\end{aligned}
$$

Classical case

Discrete-time random walk: $p_{t+1} = W p_t$

"Lazy walk": Only move with probability ϵ, so $W \to \epsilon W + (1-\epsilon)I$

$$p_{t+1} = [\epsilon W + (1-\epsilon)I]p_t$$

$$\frac{p_{t+1} - p_t}{\epsilon} = (W - I)p_t$$

As $\epsilon \to 0$ with $\tau = \epsilon t$, $\frac{d}{d\tau}p(\tau) = (W - I)p(\tau)$

Quantum case

With a suitable "lazy discrete-time quantum walk" U_ϵ, defined by an isometry $T_\epsilon := \sum_{j,k} \sqrt{W_{kj}(\epsilon)}|j,k\rangle\langle j|$,

Quantum case

With a suitable "lazy discrete-time quantum walk" U_ϵ, defined by an isometry $T_\epsilon := \sum_{j,k} \sqrt{W_{kj}(\epsilon)} |j,k\rangle\langle j|$,

1. Apply $\frac{I + iS}{\sqrt{2}} T_\epsilon$ to the input state $|\psi\rangle$.

Quantum case

With a suitable "lazy discrete-time quantum walk" U_ϵ, defined by an isometry $T_\epsilon := \sum_{j,k} \sqrt{W_{kj}(\epsilon)}|j,k\rangle\langle j|$,

1. Apply $\dfrac{I + iS}{\sqrt{2}} T_\epsilon$ to the input state $|\psi\rangle$.
2. Perform ht/ϵ steps of the walk, $U_\epsilon := iS(2T_\epsilon T_\epsilon^\dagger - I)$ (where $h = \|\mathrm{abs}(H)\|$ or $\|H\|_1$).

Quantum case

With a suitable "lazy discrete-time quantum walk" U_ϵ, defined by an isometry $T_\epsilon := \sum_{j,k} \sqrt{W_{kj}(\epsilon)}|j,k\rangle\langle j|$,

1. Apply $\frac{I + iS}{\sqrt{2}} T_\epsilon$ to the input state $|\psi\rangle$.
2. Perform ht/ϵ steps of the walk, $U_\epsilon := iS(2T_\epsilon T_\epsilon^\dagger - I)$ (where $h = \|\mathrm{abs}(H)\|$ or $\|H\|_1$).
3. Measure in the basis $\left\{ \frac{I + iS}{\sqrt{2}} T_\epsilon |j\rangle \right\}$.

Quantum case

With a suitable "lazy discrete-time quantum walk" U_ϵ, defined by an isometry $T_\epsilon := \sum_{j,k} \sqrt{W_{kj}(\epsilon)}|j,k\rangle\langle j|$,

1. Apply $\dfrac{I + iS}{\sqrt{2}} T_\epsilon$ to the input state $|\psi\rangle$.
2. Perform ht/ϵ steps of the walk, $U_\epsilon := iS(2T_\epsilon T_\epsilon^\dagger - I)$ (where $h = \|\mathrm{abs}(H)\|$ or $\|H\|_1$).
3. Measure in the basis $\left\{ \dfrac{I + iS}{\sqrt{2}} T_\epsilon |j\rangle \right\}$.

Then as $\epsilon \to 0$, $\Pr(j) \to |\langle j|e^{-iHt}|\psi\rangle|^2$.

Quantum case

With a suitable "lazy discrete-time quantum walk" U_ϵ, defined by an isometry $T_\epsilon := \sum_{j,k} \sqrt{W_{kj}(\epsilon)}|j,k\rangle\langle j|$,

1. Apply $\frac{I + iS}{\sqrt{2}} T_\epsilon$ to the input state $|\psi\rangle$.
2. Perform ht/ϵ steps of the walk, $U_\epsilon := iS(2T_\epsilon T_\epsilon^\dagger - I)$ (where $h = \|\mathrm{abs}(H)\|$ or $\|H\|_1$).
3. Measure in the basis $\left\{ \frac{I + iS}{\sqrt{2}} T_\epsilon |j\rangle \right\}$.

Then as $\epsilon \to 0$, $\Pr(j) \to |\langle j|e^{-iHt}|\psi\rangle|^2$.

(We can get error at most δ in $O(ht, (\|H\|t)^{3/2}/\sqrt{\delta})$ steps.)

Hamiltonian simulation

Simulating sparse Hamiltonians

Problem: For a given Hamiltonian H, simulate the unitary time evolution e^{-iHt} for any desired t.

Simulating sparse Hamiltonians

Problem: For a given Hamiltonian H, simulate the unitary time evolution e^{-iHt} for any desired t.

Suppose H is **sparse**: for any x, we can efficiently compute all the nonzero matrix elements $\langle y|H|x\rangle$ (so in particular, there are only polynomially many such y).

Simulating sparse Hamiltonians

Problem: For a given Hamiltonian H, simulate the unitary time evolution e^{-iHt} for any desired t.

Suppose H is **sparse**: for any x, we can efficiently compute all the nonzero matrix elements $\langle y|H|x\rangle$ (so in particular, there are only polynomially many such y).

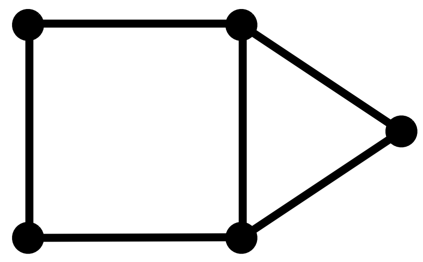
Approach: Color the graph of H. Then the simulation breaks into small pieces that are easy to handle. [Aharonov, Ta-Shma 03], [CCDFGS 03]

Simulating sparse Hamiltonians

Problem: For a given Hamiltonian H, simulate the unitary time evolution e^{-iHt} for any desired t.

Suppose H is **sparse**: for any x, we can efficiently compute all the nonzero matrix elements $\langle y|H|x\rangle$ (so in particular, there are only polynomially many such y).

Approach: Color the graph of H. Then the simulation breaks into small pieces that are easy to handle. [Aharonov, Ta-Shma 03], [CCDFGS 03]

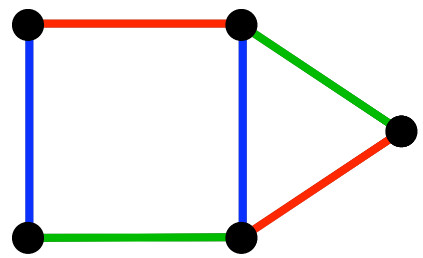


Simulating sparse Hamiltonians

Problem: For a given Hamiltonian H, simulate the unitary time evolution e^{-iHt} for any desired t.

Suppose H is **sparse**: for any x, we can efficiently compute all the nonzero matrix elements $\langle y|H|x\rangle$ (so in particular, there are only polynomially many such y).

Approach: Color the graph of H. Then the simulation breaks into small pieces that are easy to handle. [Aharonov, Ta-Shma 03], [CCDFGS 03]

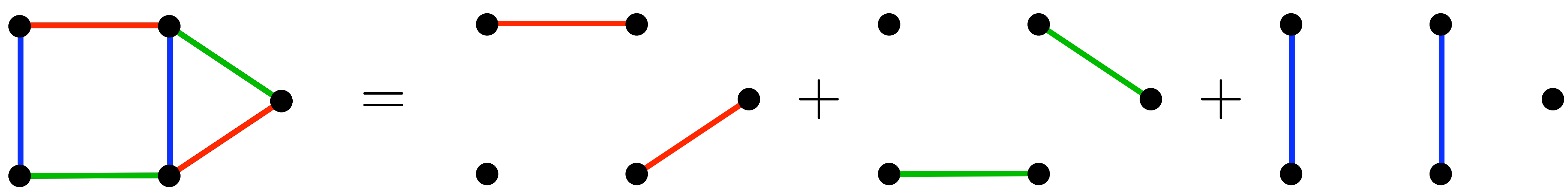


Simulating sparse Hamiltonians

Problem: For a given Hamiltonian H, simulate the unitary time evolution e^{-iHt} for any desired t.

Suppose H is **sparse**: for any x, we can efficiently compute all the nonzero matrix elements $\langle y|H|x\rangle$ (so in particular, there are only polynomially many such y).

Approach: Color the graph of H. Then the simulation breaks into small pieces that are easy to handle. [Aharonov, Ta-Shma 03], [CCDFGS 03]

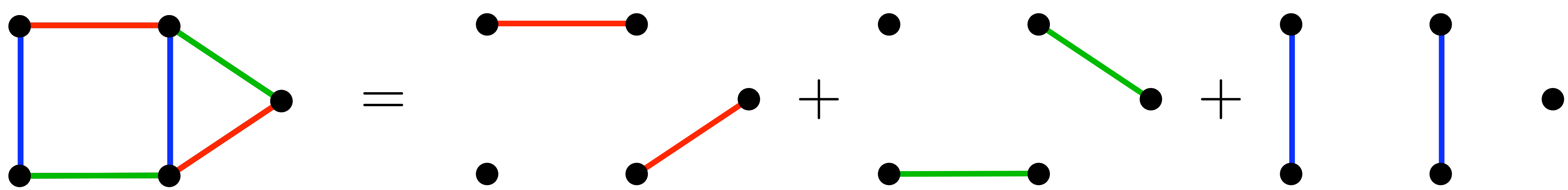


Simulating sparse Hamiltonians

Problem: For a given Hamiltonian H, simulate the unitary time evolution e^{-iHt} for any desired t.

Suppose H is **sparse**: for any x, we can efficiently compute all the nonzero matrix elements $\langle y|H|x\rangle$ (so in particular, there are only polynomially many such y).

Approach: Color the graph of H. Then the simulation breaks into small pieces that are easy to handle. [Aharonov, Ta-Shma 03], [CCDFGS 03]



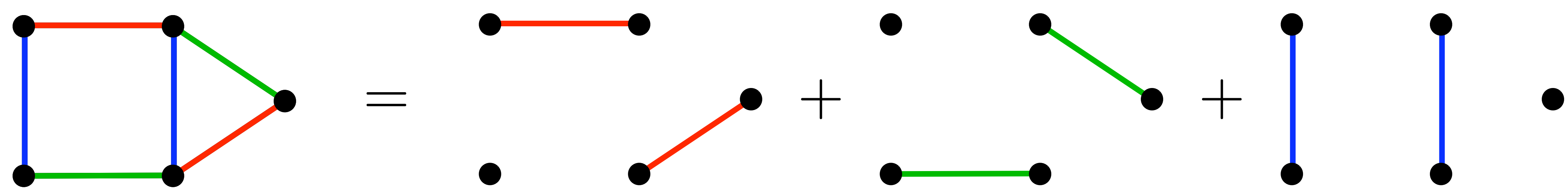
Any sufficiently sparse graph can be efficiently colored using only local information. [Linial 87]

Simulating sparse Hamiltonians

Problem: For a given Hamiltonian H, simulate the unitary time evolution e^{-iHt} for any desired t.

Suppose H is **sparse**: for any x, we can efficiently compute all the nonzero matrix elements $\langle y|H|x\rangle$ (so in particular, there are only polynomially many such y).

Approach: Color the graph of H. Then the simulation breaks into small pieces that are easy to handle. [Aharonov, Ta-Shma 03], [CCDFGS 03]



Any sufficiently sparse graph can be efficiently colored using only local information. [Linial 87]

So we can simulate H in poly(deg(H), log dim(H), t, $1/\epsilon$) steps.

Simulating a sum of Hamiltonians

Lie product formula: $\lim_{n\to\infty} \left(e^{-iA/N} e^{-iB/n}\right)^n = e^{-i(A+B)}$

[C 04], [Berry, Ahokas, Cleve, Sanders 05]

Simulating a sum of Hamiltonians

Lie product formula: $\lim_{n\to\infty} \left(e^{-iA/N} e^{-iB/n}\right)^n = e^{-i(A+B)}$

We can approximately simulate $A + B$ for time t using

$$\left(e^{-iAt/n} e^{-iBt/n}\right)^n \approx e^{-i(A+B)t}.$$

To get error ϵ, it suffices to use $O(t^2/\epsilon)$ steps.

[C 04], [Berry, Ahokas, Cleve, Sanders 05]

Simulating a sum of Hamiltonians

Lie product formula: $\lim_{n\to\infty} \left(e^{-iA/N} e^{-iB/n}\right)^n = e^{-i(A+B)}$

We can approximately simulate $A + B$ for time t using

$$\left(e^{-iAt/n} e^{-iBt/n}\right)^n \approx e^{-i(A+B)t}.$$

To get error ϵ, it suffices to use $O(t^2/\epsilon)$ steps.

We can do better using a higher-order formula:

$$\left(e^{-iAt/2n} e^{-iBt/n} e^{-iAt/2n}\right)^n \approx e^{-i(A+B)t}.$$

Then $O(t^{3/2}/\epsilon^{1/2})$ steps suffice.

[C 04], [Berry, Ahokas, Cleve, Sanders 05]

Simulating a sum of Hamiltonians

Lie product formula: $\lim_{n\to\infty} \left(e^{-iA/N} e^{-iB/n}\right)^n = e^{-i(A+B)}$

We can approximately simulate $A + B$ for time t using

$$\left(e^{-iAt/n} e^{-iBt/n}\right)^n \approx e^{-i(A+B)t}.$$

To get error ϵ, it suffices to use $O(t^2/\epsilon)$ steps.

We can do better using a higher-order formula:

$$\left(e^{-iAt/2n} e^{-iBt/n} e^{-iAt/2n}\right)^n \approx e^{-i(A+B)t}.$$

Then $O(t^{3/2}/\epsilon^{1/2})$ steps suffice.

Using even better approximations (systematically constructed by Suzuki), we can simulate $A + B$ for time t in $t^{1+o(1)}$ steps.

[C 04], [Berry, Ahokas, Cleve, Sanders 05]

The no fast-forwarding theorem

Can we simulate H for time t using a number of operations that is sublinear in t?

In special cases, yes! (e.g., whenever $e^{-iH\tau} = I$ for a small τ)

[Berry, Ahokas, Cleve, Sanders 05]

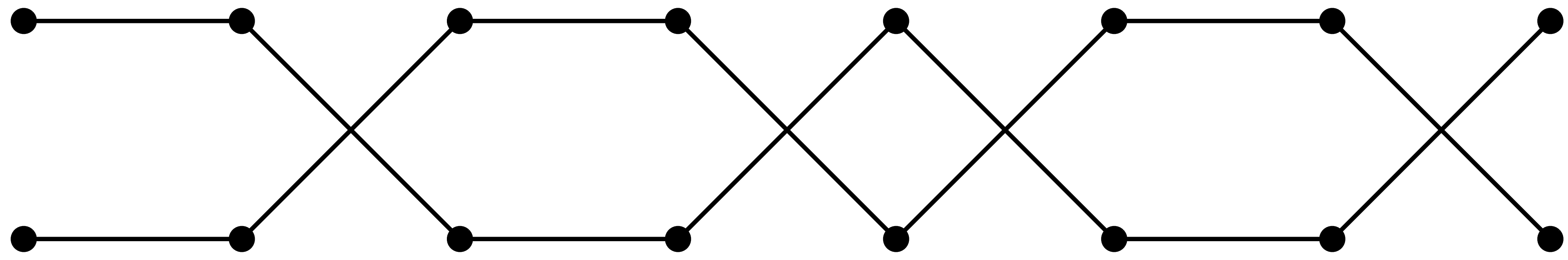
The no fast-forwarding theorem

Can we simulate H for time t using a number of operations that is sublinear in t?

In special cases, yes! (e.g., whenever $e^{-iH\tau} = I$ for a small τ)

But this is not possible in general: for some Hamiltonians, $\Omega(t)$ operations are required.

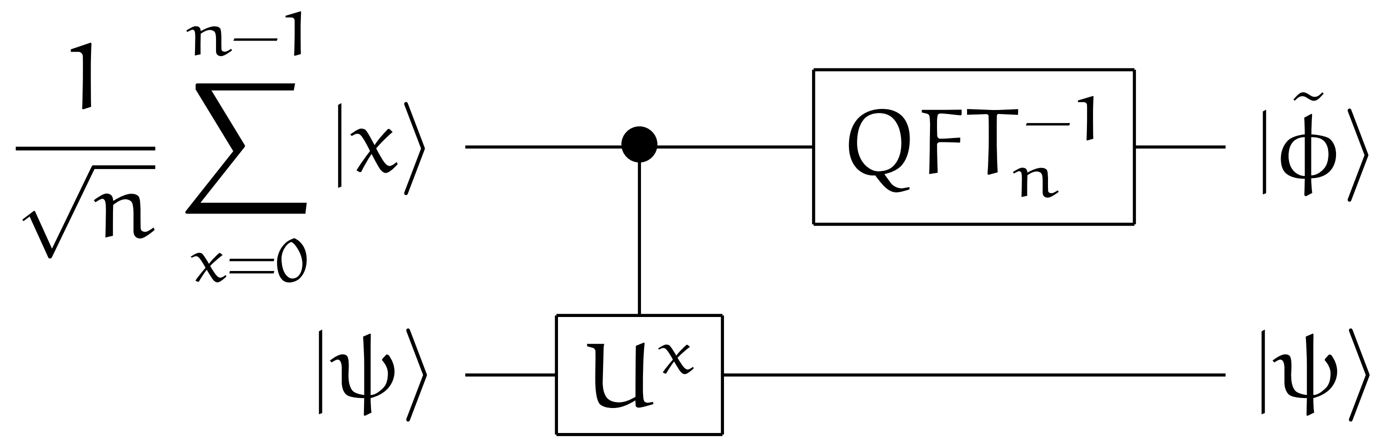
Proof is by reduction of parity to simulating a Hamiltonian.



[Berry, Ahokas, Cleve, Sanders 05]

Phase estimation

$$U|\psi\rangle = e^{i\phi}|\psi\rangle$$



Precision δ with error probability at most ϵ using $O(1/\delta\epsilon)$ applications of U.

Hamiltonian simulation by discrete-time quantum walk

To simulate H for time t:

Hamiltonian simulation by discrete-time quantum walk

To simulate H for time t:

1. Apply T to the input state $|\psi\rangle$.

Hamiltonian simulation by discrete-time quantum walk

To simulate H for time t:

1. Apply T to the input state $|\psi\rangle$.
2. Perform phase estimation with $U = iS(2TT^\dagger - I)$, estimating a phase $\pm e^{\pm i \arcsin \lambda}$ for the component of $T|\psi\rangle$ corresponding to an eigenvector of H with eigenvalue λ.

Hamiltonian simulation by discrete-time quantum walk

To simulate H for time t:

1. Apply T to the input state $|\psi\rangle$.
2. Perform phase estimation with $U = iS(2TT^\dagger - I)$, estimating a phase $\pm e^{\pm i \arcsin \lambda}$ for the component of $T|\psi\rangle$ corresponding to an eigenvector of H with eigenvalue λ.
3. Use the estimate of $\arcsin \lambda$ to estimate λ, and apply the phase $e^{-i\lambda t}$.

Hamiltonian simulation by discrete-time quantum walk

To simulate H for time t:

1. Apply T to the input state $|\psi\rangle$.
2. Perform phase estimation with $U = iS(2TT^\dagger - I)$, estimating a phase $\pm e^{\pm i \arcsin \lambda}$ for the component of $T|\psi\rangle$ corresponding to an eigenvector of H with eigenvalue λ.
3. Use the estimate of $\arcsin \lambda$ to estimate λ, and apply the phase $e^{-i\lambda t}$.
4. Uncompute the phase estimation and T, giving an approximation of $e^{-iHt}|\psi\rangle$.

Hamiltonian simulation by discrete-time quantum walk

To simulate H for time t:

1. Apply T to the input state $|\psi\rangle$.
2. Perform phase estimation with $U = iS(2TT^\dagger - I)$, estimating a phase $\pm e^{\pm i \arcsin \lambda}$ for the component of $T|\psi\rangle$ corresponding to an eigenvector of H with eigenvalue λ.
3. Use the estimate of $\arcsin \lambda$ to estimate λ, and apply the phase $e^{-i\lambda t}$.
4. Uncompute the phase estimation and T, giving an approximation of $e^{-iHt}|\psi\rangle$.

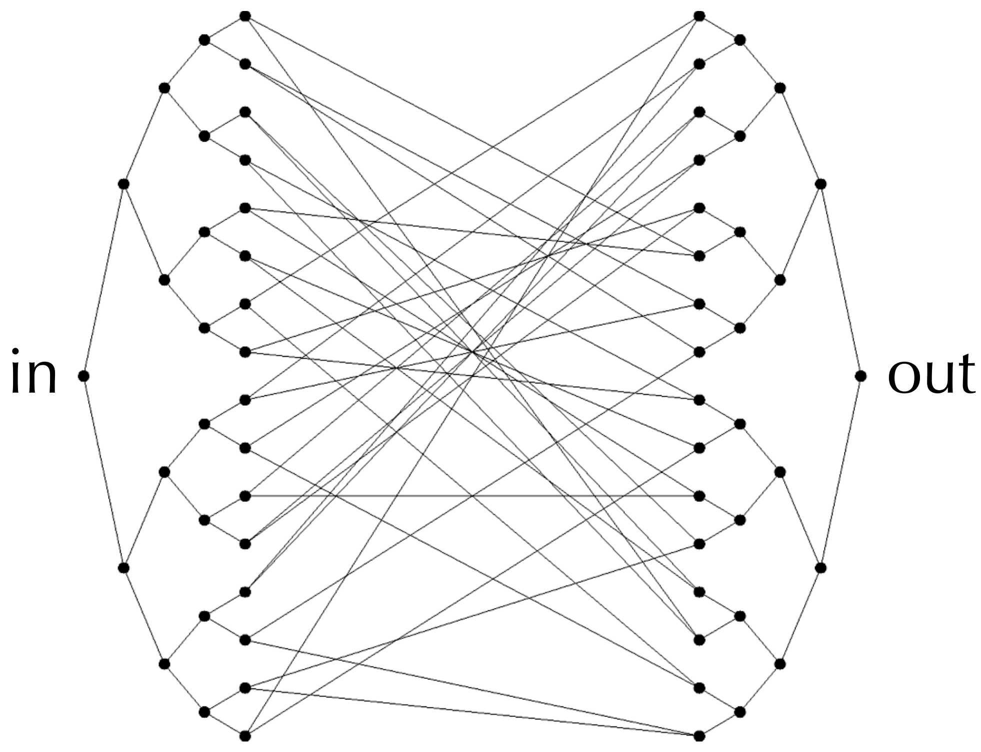
Theorem. To achieve fidelity $1 - \epsilon$, it suffices to use $O(\|\mathrm{abs}(H)\| t/\epsilon^{3/2})$ steps of the discrete-time quantum walk.

This is linear in t, and works even in cases where H is not sparse!

Applications

Algorithms

Glued trees



There is a discrete-time quantum walk that travels from “in” to “out” in polynomial time.

Element distinctness

Given a black box for $f : \{0, 1, \ldots, n\} \to S$, are there are distinct indices x,y such that $f(x) = f(y)$?

There is a continuous-time quantum walk algorithm that can be implemented with $O(N^{2/3})$ queries.

Walk takes place on a Johnson graph (not sparse).

Hamiltonian query model

Hamiltonian query model

Conventional quantum query model:

- Query operator Q_x, where $Q_x|i, b\rangle = |i, b \oplus x_i\rangle$.
- Unitary operators $U_0, U_1, ..., U_n$.
- Algorithm is $U_n Q_x ... Q_x U_1 Q_x U_0$.

Hamiltonian query model

Conventional quantum query model:

- Query operator Q_x, where $Q_x|i, b\rangle = |i, b \oplus x_i\rangle$.
- Unitary operators $U_0, U_1, \ldots, U_n$.
- Algorithm is $U_n Q_x \ldots Q_x U_1 Q_x U_0$.

Hamiltonian query model:

- Hamiltonian H_x that generates Q_x.
- Driving Hamiltonian $H_D(t)$.
- Algorithm is $H_D(t) + H_x$, from $t = 0$ to T.

Hamiltonian query model

Conventional quantum query model:

- Query operator Q_x, where $Q_x|i, b\rangle = |i, b \oplus x_i\rangle$.
- Unitary operators $U_0, U_1, \ldots, U_n$.
- Algorithm is $U_n Q_x \ldots Q_x U_1 Q_x U_0$.

Hamiltonian query model:

- Hamiltonian H_x that generates Q_x.
- Driving Hamiltonian $H_D(t)$.
- Algorithm is $H_D(t) + H_x$, from $t = 0$ to T.

The Hamiltonian model is potentially more powerful!

Hamiltonian query model

Conventional quantum query model:

- Query operator Q_x, where $Q_x|i, b\rangle = |i, b \oplus x_i\rangle$.
- Unitary operators $U_0, U_1, \ldots, U_n$.
- Algorithm is $U_n Q_x \ldots Q_x U_1 Q_x U_0$.

Hamiltonian query model:

- Hamiltonian H_x that generates Q_x.
- Driving Hamiltonian $H_D(t)$.
- Algorithm is $H_D(t) + H_x$, from $t = 0$ to T.

The Hamiltonian model is potentially more powerful!

Theorem. [Cleve, Gottesman, Mosca, Somma, Yonge-Mallo 08]
If a function can be evaluated with Hamiltonian queries for time T, then it can be evaluated with $O(T \log T)$ discrete queries.

Hamiltonian query model

Conventional quantum query model:

- Query operator Q_x, where $Q_x|i, b\rangle = |i, b \oplus x_i\rangle$.
- Unitary operators $U_0, U_1, \ldots, U_n$.
- Algorithm is $U_n Q_x \ldots Q_x U_1 Q_x U_0$.

Hamiltonian query model:

- Hamiltonian H_x that generates Q_x.
- Driving Hamiltonian $H_D(t)$.
- Algorithm is $H_D(t) + H_x$, from $t = 0$ to T.

The Hamiltonian model is potentially more powerful!

Theorem. [Cleve, Gottesman, Mosca, Somma, Yonge-Mallo 08]
If a function can be evaluated with Hamiltonian queries for time T, then it can be evaluated with $O(T \log T)$ discrete queries.

Theorem. If H_D is time-independent, $O(\|\mathrm{abs}(H_D)\| T)$ discrete queries suffice.

Open question:

A sign problem for Hamiltonain simulation

How hard is it to simulate H?

How hard is it to simulate H?

By the no fast forwarding theorem, $\Omega(\|H\|t)$ operations are necessary.

How hard is it to simulate H?

By the no fast forwarding theorem, $\Omega(\|H\|t)$ operations are necessary.

We have seen that $O(\|\mathrm{abs}(H)\|t)$ steps (of the corresponding discrete-time quantum walk) are sufficient. *Is this a fundamental barrier?*

How hard is it to simulate H?

By the no fast forwarding theorem, $\Omega(\|H\|t)$ operations are necessary.

We have seen that $O(\|\mathrm{abs}(H)\|t)$ steps (of the corresponding discrete-time quantum walk) are sufficient. *Is this a fundamental barrier?*

For $H \in \mathbb{C}^{N \times N}$, we have

$$\|H\| \le \|\mathrm{abs}(H)\| \le \sqrt{N}\|H\|$$

(and these bounds are the best possible).

How hard is it to simulate H?

By the no fast forwarding theorem, $\Omega(\|H\|t)$ operations are necessary.

We have seen that $O(\|\mathrm{abs}(H)\|t)$ steps (of the corresponding discrete-time quantum walk) are sufficient. *Is this a fundamental barrier?*

For $H \in \mathbb{C}^{N\times N}$, we have

$$\|H\| \le \|\mathrm{abs}(H)\| \le \sqrt{N}\|H\|$$

(and these bounds are the best possible).

Simulations using $O(\|H\|t)$ steps would have applications such as

- approximately computing exponential sums
- breaking pseudorandom generators derived from strongly regular graphs.