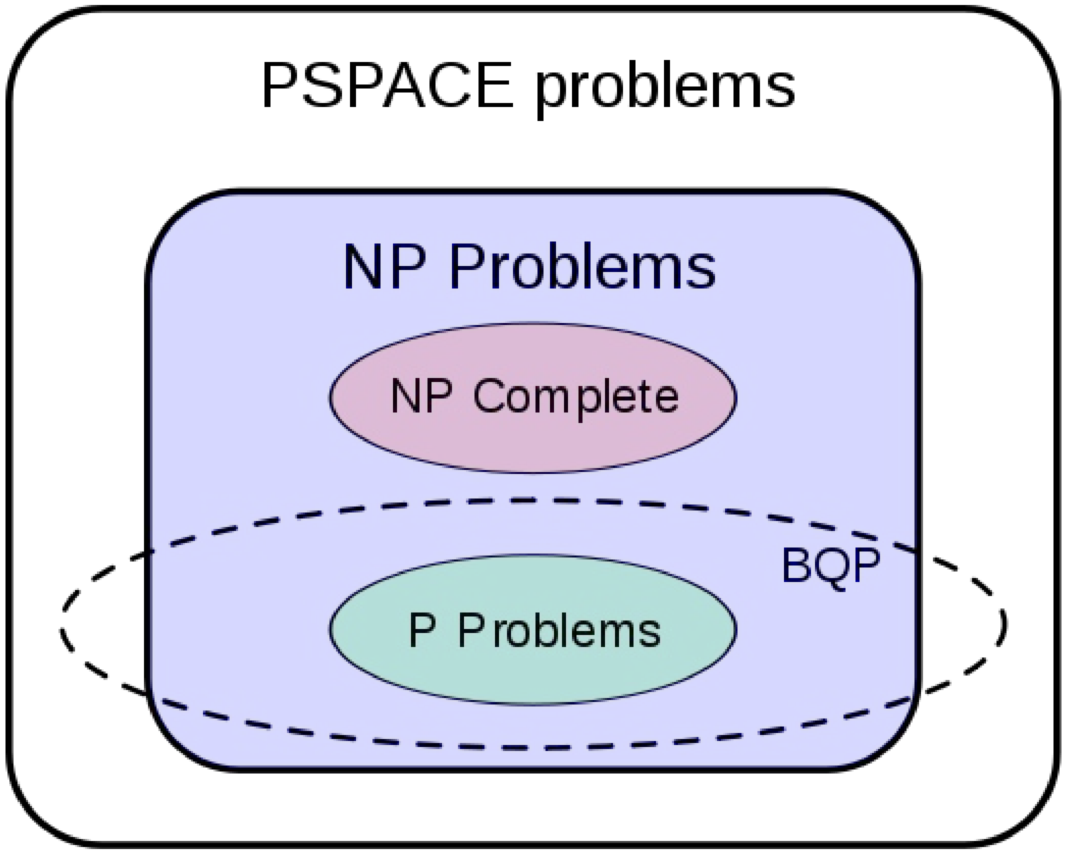
Complexity Classification of Commuting 2-qubit Hamiltonians

Laura Mančinska
University of Bristol

Joint work with Adam Bouland and Lucy Zhang

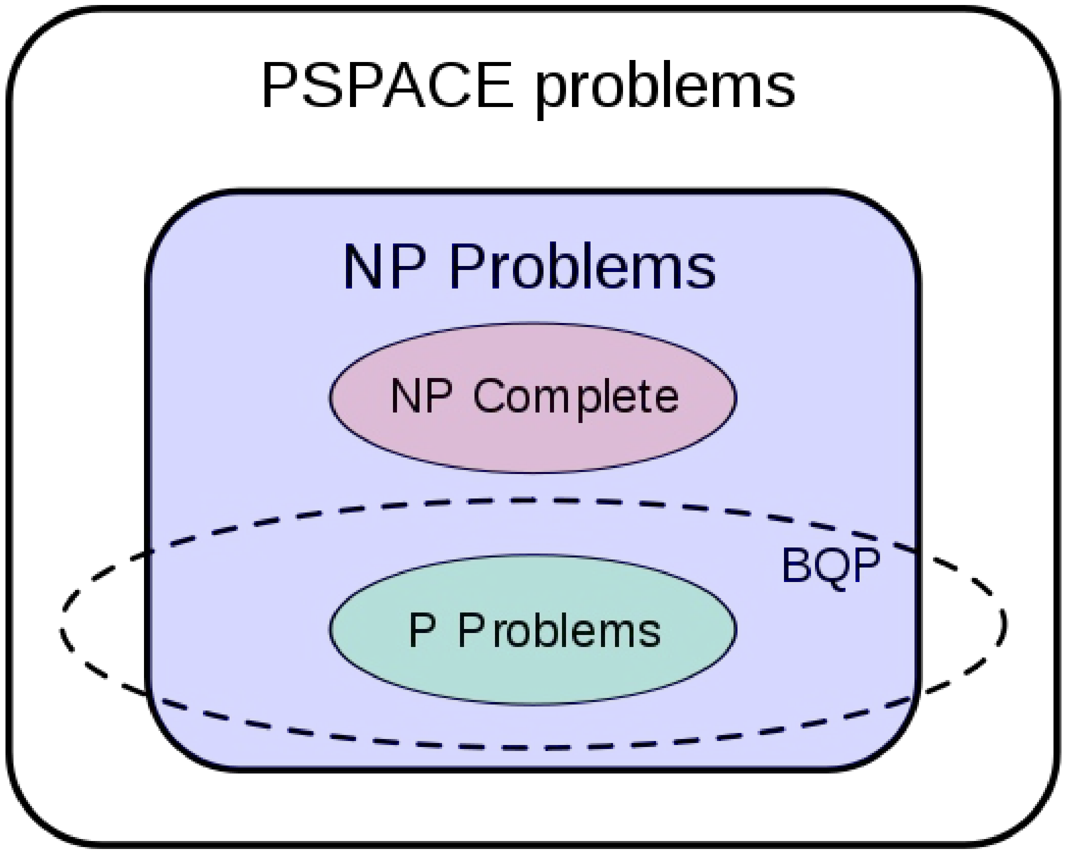
Establishing quantum advantage



Establishing quantum advantage

Considerations

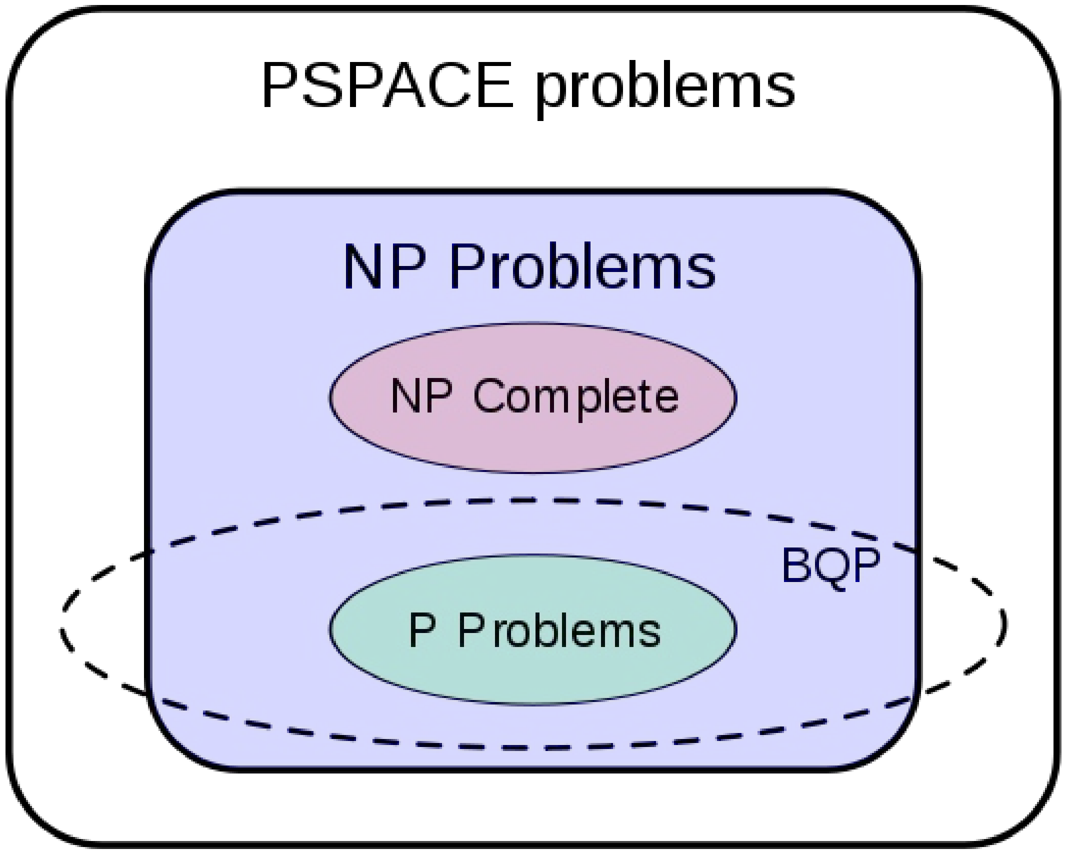
- Proving quantum advantage for a decision problem shows that P ≠ PSPACE



Establishing quantum advantage

Considerations

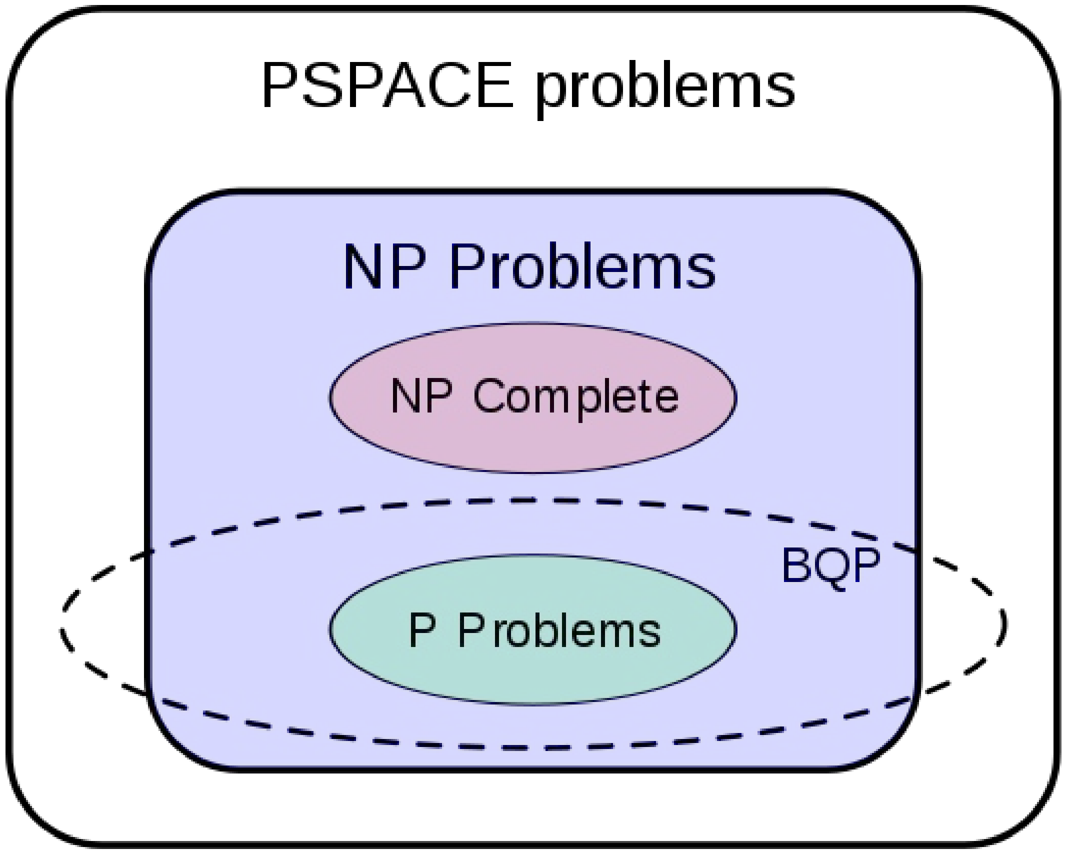
- Proving quantum advantage for a decision problem shows that P ≠ PSPACE
- Implementing a fault-tolerant large-scale quantum computers is an engineering challenge



Establishing quantum advantage

Considerations

- Proving quantum advantage for a decision problem shows that P ≠ PSPACE
- Implementing a fault-tolerant large-scale quantum computers is an engineering challenge



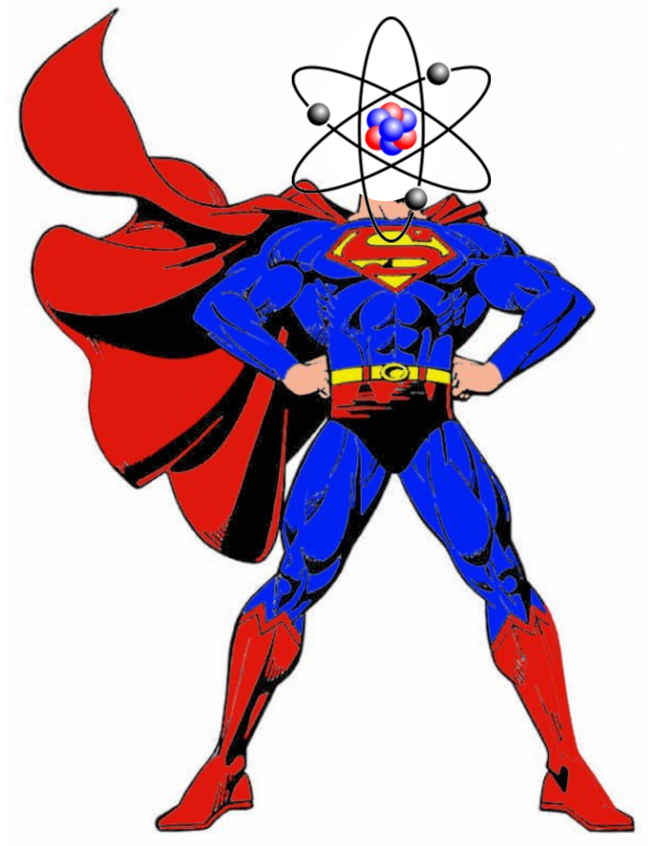
Approach

Establish quantum advantage under accepted complexity assumptions for a problem that promises easier experimental implementation.

Previous work

Sampling problems with quantum advantage

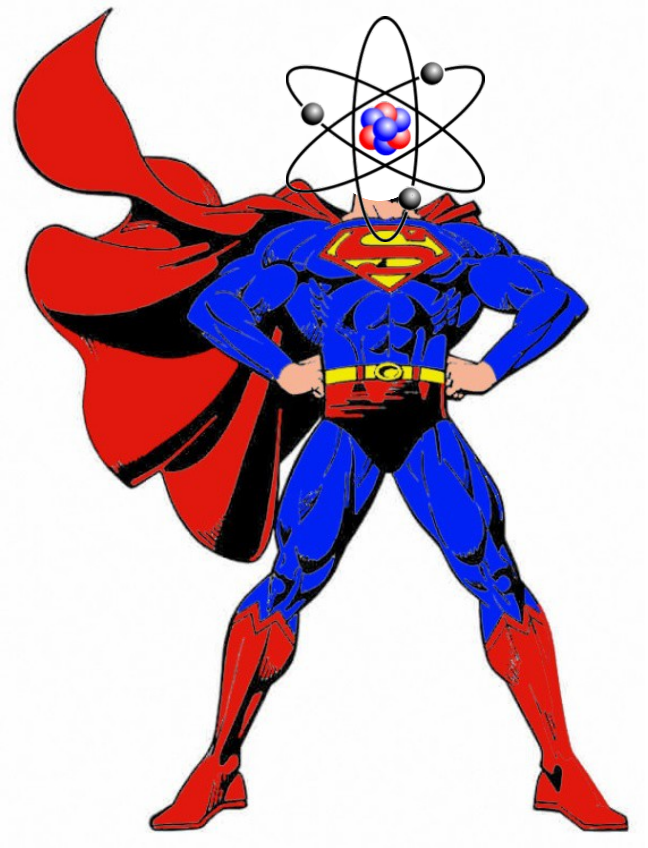
- One-clean qubit model
  [Knill-Laflamme, Morimae-Fuji-Fitzsimons]
- Instantaneous quantum computation (IQP)
  [Bremner-Josza-Montanaro-Shepherd]
- Boson Sampling [Aaronson-Arkhipov]
- Fourier sampling [Fefferman-Umans]



Previous work

Sampling problems with quantum advantage

- One-clean qubit model
  [Knill-Laflamme, Morimae-Fuji-Fitzsimons]
- Instantaneous quantum computation (IQP)
  [Bremner-Josza-Montanaro-Shepherd]
- Boson Sampling [Aaronson-Arkhipov]
- Fourier sampling [Fefferman-Umans]

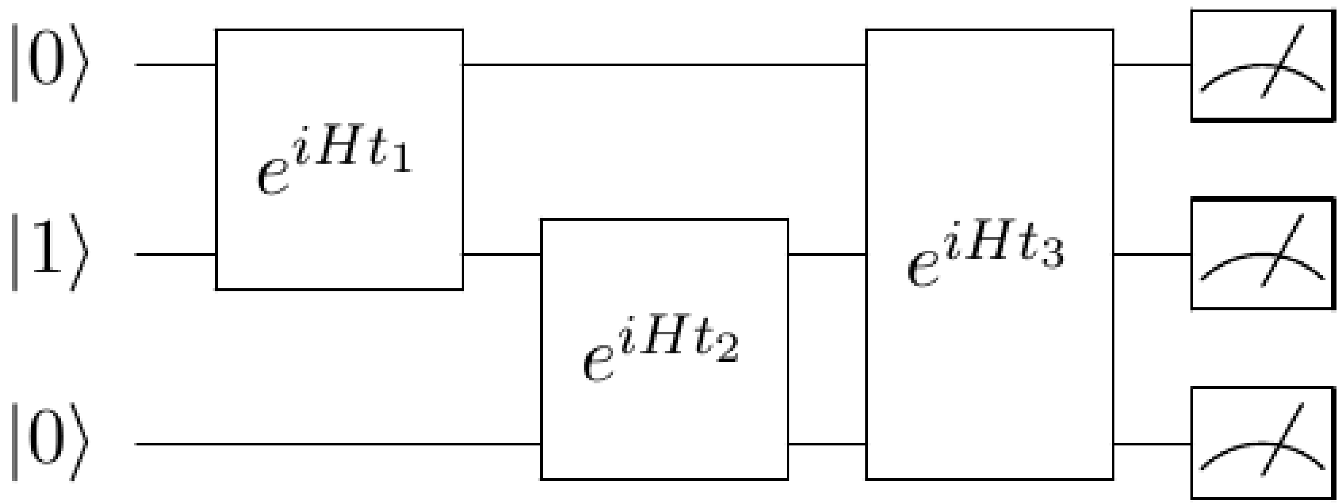


This work: Classify when this quantum advantage occurs

The model: computations with a fixed two-qubit Hamiltonian *H*

Here we can

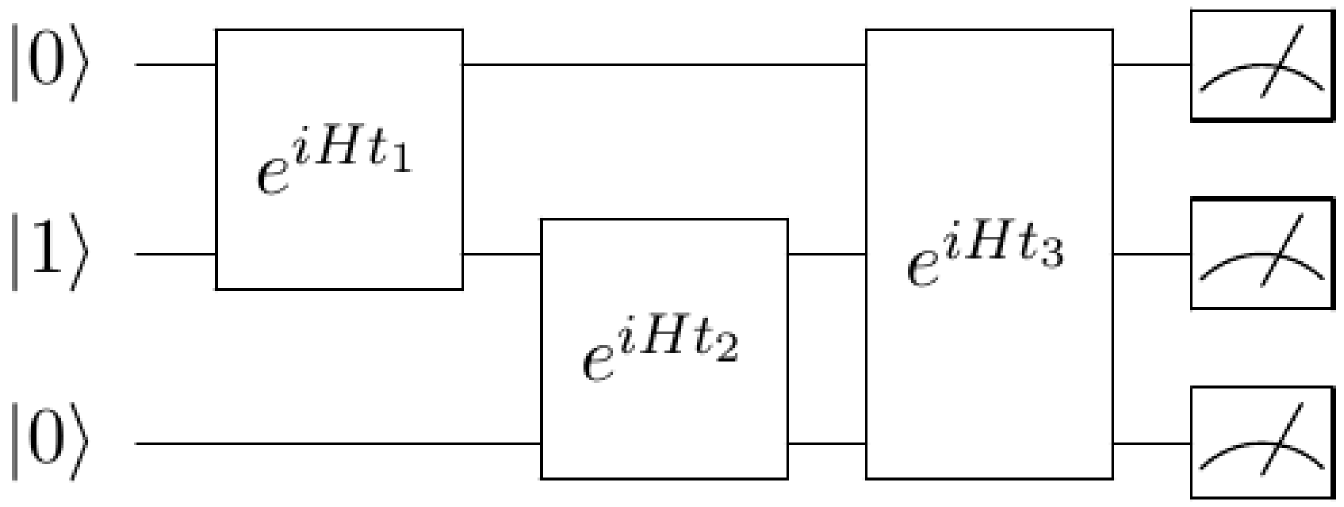
- Initialize qubits in any standard basis states
- Evolve any ordered pair of qubits according to *H* for any time *t*
- Measure qubits in standard basis



The model: computations with a fixed two-qubit Hamiltonian *H*

Here we can

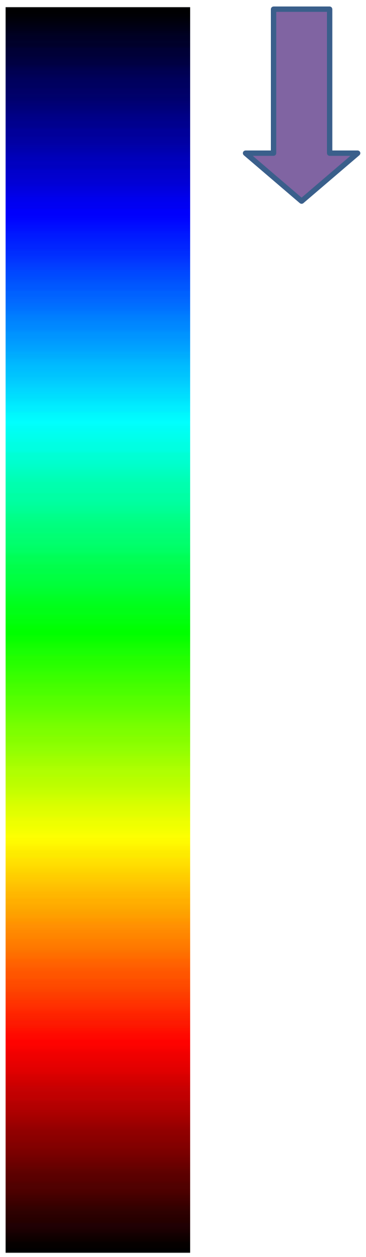
- Initialize qubits in any standard basis states
- Evolve any ordered pair of qubits according to *H* for any time *t*
- Measure qubits in standard basis



Question

For a given *H*, what is the computational power of this model?

The complexity of *H*-computations



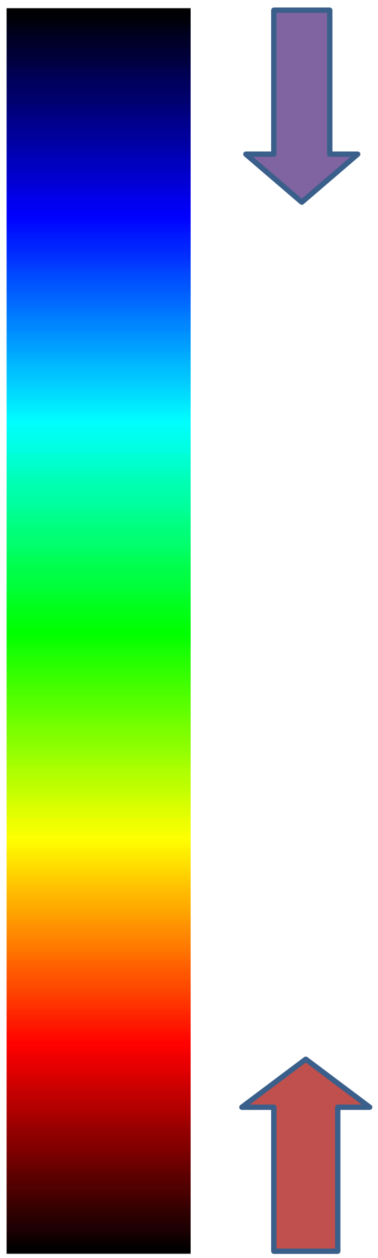
Universal [Childs-Leung-Mančinska-Ozols]

(Densely generate the *n*-qubit unitary group)

Surprisingly still unknown:

Which two-qubit Hamiltonians are universal

The complexity of *H*-computations



Universal [Childs-Leung-Mančinska-Ozols]

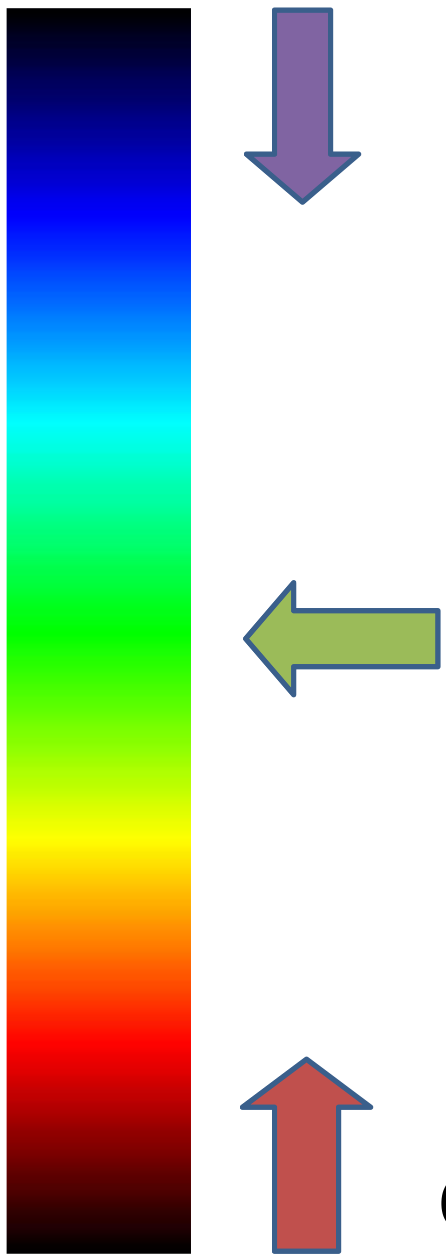
(Densely generate the *n*-qubit unitary group)

Surprisingly still unknown:

Which two-qubit Hamiltonians are universal

Classically simulable

The complexity of *H*-computations



Universal [Childs-Leung-Mančinska-Ozols]
(Densely generate the *n*-qubit unitary group)

Surprisingly still unknown:

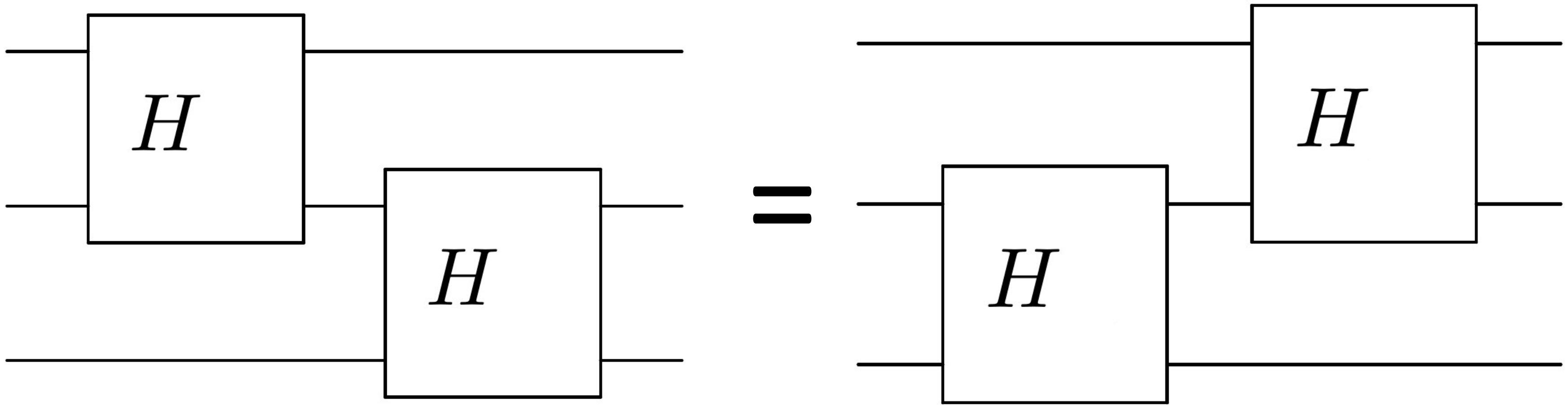
Which two-qubit Hamiltonians are universal

This work

Classically simulable

We study: *H*-computations for commuting 2-qubit *H*

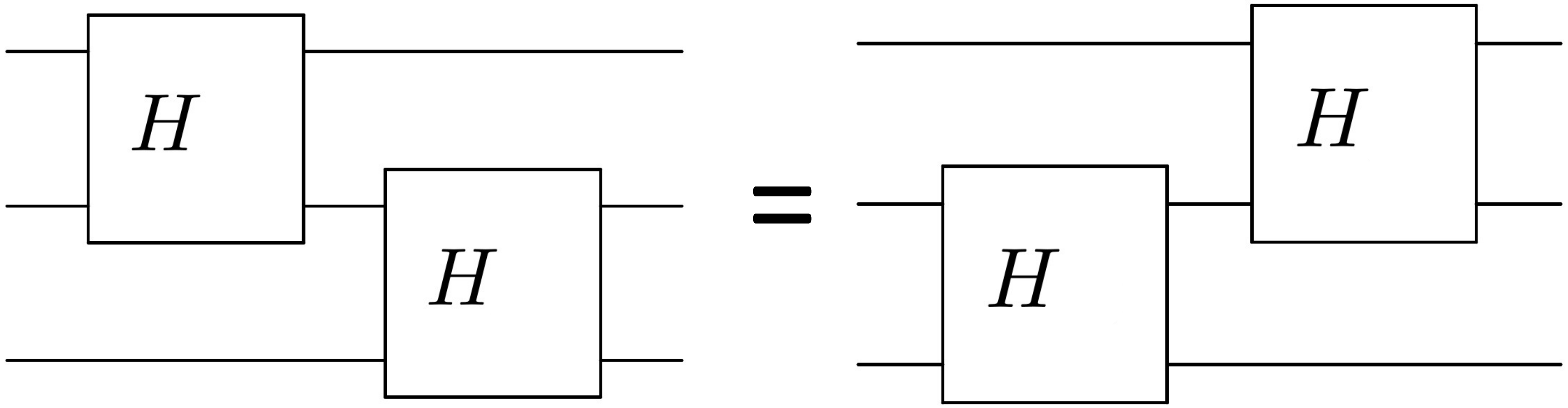
Definition. A 2-qubit Hamiltonian *H* is commuting if it gives rise only to mutually commuting gates.



In other words, $[H_{ij}, H_{kl}] = 0$ for all *i, j, k* and *l.*

We study: *H*-computations for commuting 2-qubit *H*

Definition. A 2-qubit Hamiltonian *H* is commuting if it gives rise only to mutually commuting gates.



In other words, $[H_{ij}, H_{kl}] = 0$ for all *i, j, k* and *l.*

An efficient *H*-computation is specified by:

- times t_{ij} (each specified by polynomially many bits and with polynomially bounded magnitude)
- initial computational basis state

Inspiration: Commuting circuits

Studied by Bremner, Josza, Montanaro and Shepherd

Inspiration: Commuting circuits

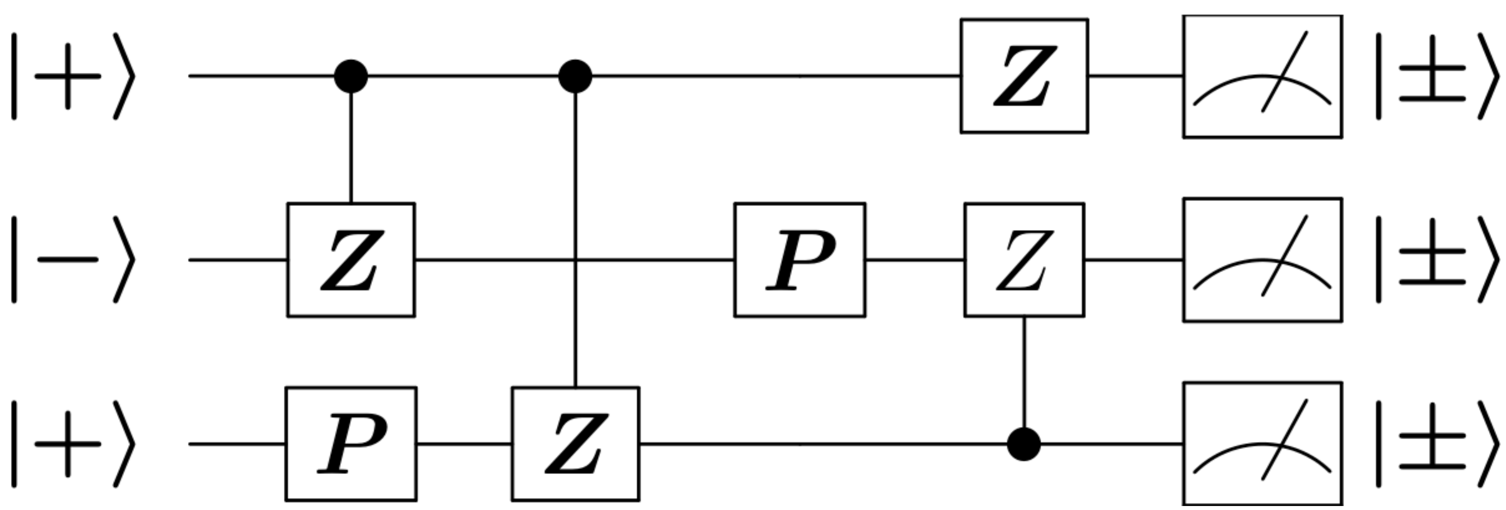
Studied by Bremner, Josza, Montanaro and Shepherd

Theorem. Consider circuits over gate set

G = { Z, controlled-Z, π/8-phase gate P}

where each qubit is initialized as plus or minus state and measured in the X-basis at the end.

Some of these circuits yield probability distributions which are hard to sample from classically unless polynomial hierarchy collapses.



Inspiration: Commuting circuits

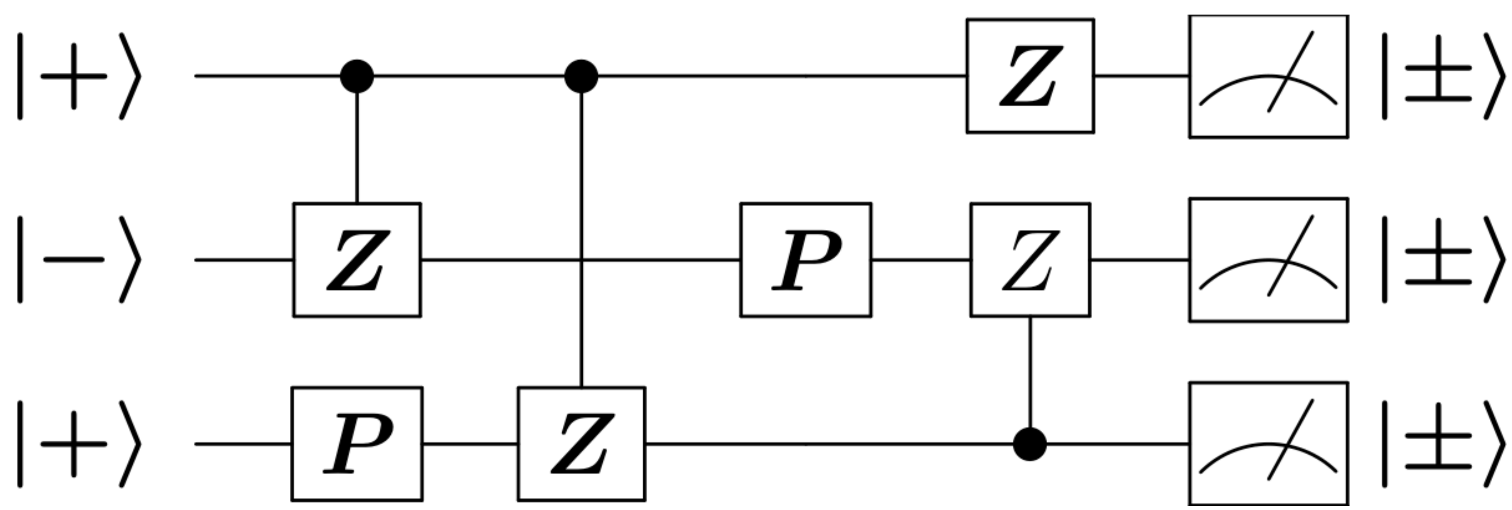
Studied by Bremner, Josza, Montanaro and Shepherd

Theorem. Consider circuits over gate set

G = { Z, controlled-Z, π/8-phase gate P}

where each qubit is initialized as plus or minus state and measured in the X-basis at the end.

Some of these circuits yield probability distributions which are hard to sample from classically unless polynomial hierarchy collapses.



In this work

- rather than studying fixed gate sets, we fix a Hamiltonian
- rather than showing that there exist hard instances we characterize which instances are hard (turns out that almost all!)

Main result: classifcation + dichotomy

Theorem.
For a commuting 2-qubit Hamiltonian *H,* we have that EITHER

- *H* cannot generate entanglement from standard basis states, then *H*-computations, are **classically simulable**

OR

- *H*-computations produce probability distributions which are **hard to sample from classically** unless polynomial hierarchy collapses.

Main result: classifcation + dichotomy

Theorem.
For a commuting 2-qubit Hamiltonian *H,* we have that EITHER

- *H* cannot generate entanglement from standard basis states, then *H*-computations, are **classically simulable**

OR

- *H*-computations produce probability distributions which are **hard to sample from classically** unless polynomial hierarchy collapses.

So generic commuting *H*, can be used to demonstrate quantum advantage, assuming PH does not collapse.

Main result: classifcation + dichotomy

Theorem.
For a commuting 2-qubit Hamiltonian *H,* we have that EITHER

- *H* cannot generate entanglement from standard basis states, then *H*-computations, are **classically simulable**

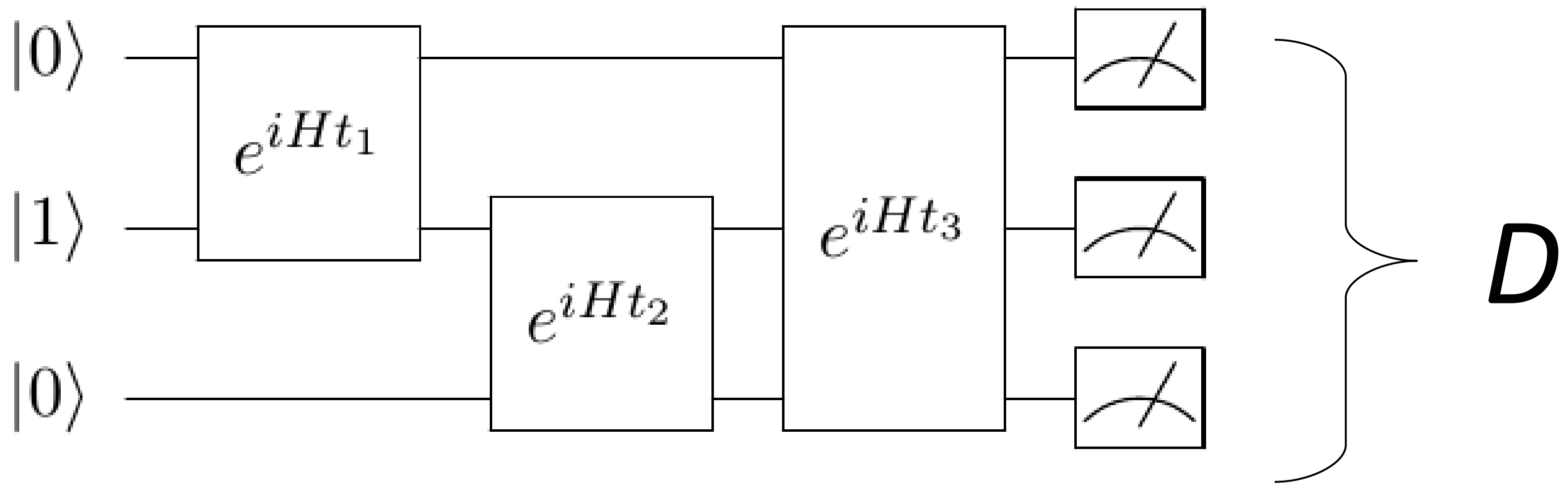
OR

- *H*-computations produce probability distributions which are **hard to sample from classically** unless polynomial hierarchy collapses.

So generic commuting *H*, can be used to demonstrate quantum advantage, assuming PH does not collapse.

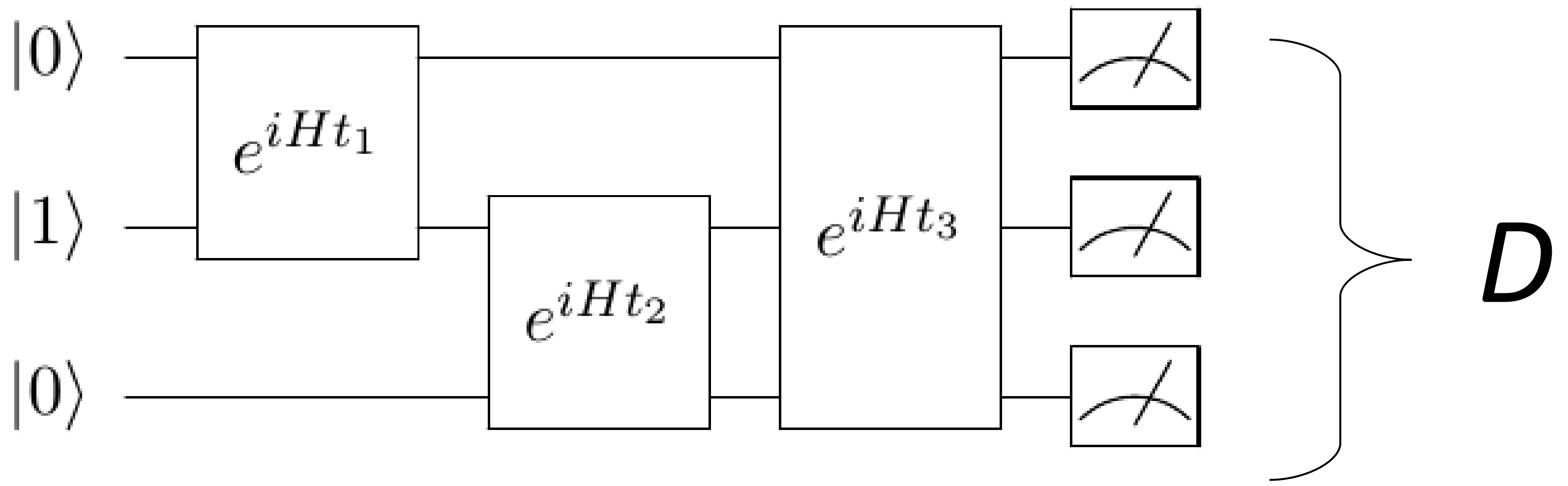
Classically-hard probability distributions

samp-*H*: probability distributions *D* produced by efficient *H*-computations



Classically-hard probability distributions

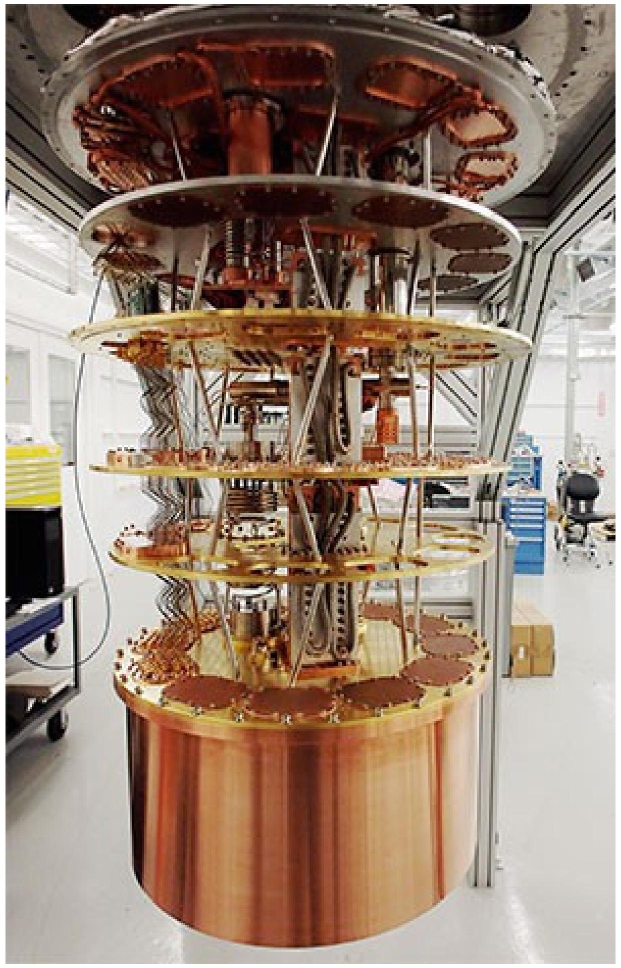
samp-*H*: probability distributions *D* produced by efficient *H*-computations



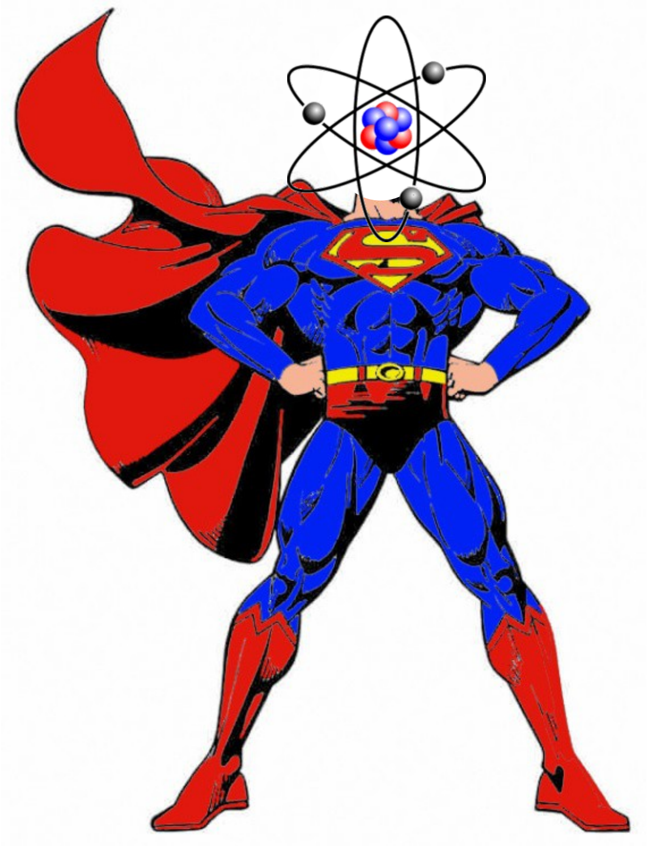
For some distributions *D* in samp-*H* no efficient randomized algorithm *M* satisfies

$$\frac{1}{\sqrt{2}} \Pr[M \text{ outputs } y] \leq \mathcal{D}(y) \leq \sqrt{2} \Pr[M \text{ outputs } y]$$

Experimental considerations



Superconducting chip
Martinis' group



Aliferis et al. '09: Commuting computations allow lower fault-tolerance thresholds in super-conducting implementations.

Comparison of quantum supremacy proposals

Under some assumptions, quantum computers can sample from distributions that are intractable for classical computers.

Comparison of quantum supremacy proposals

Under some assumptions, quantum computers can sample from distributions that are intractable for classical computers.

	Simulation error	Assumptions	Other
Boson sampling (Aaronson-Arkhipov'13)	additive	2 permanent conjectures	Implemented, error-correction
IQP I (Bremner et al. '11)	multiplicative	PH does not collapse	possibly lower FT-treshold
IQP II (Bremner et al. '15)	additive	Any one of two conjectures	possibly lower FT-treshold
Fourier sampling (Fefferman et al. '16)	additive	Variants of conjectures in A-A	probably requires universal QC

Comparison of quantum supremacy proposals

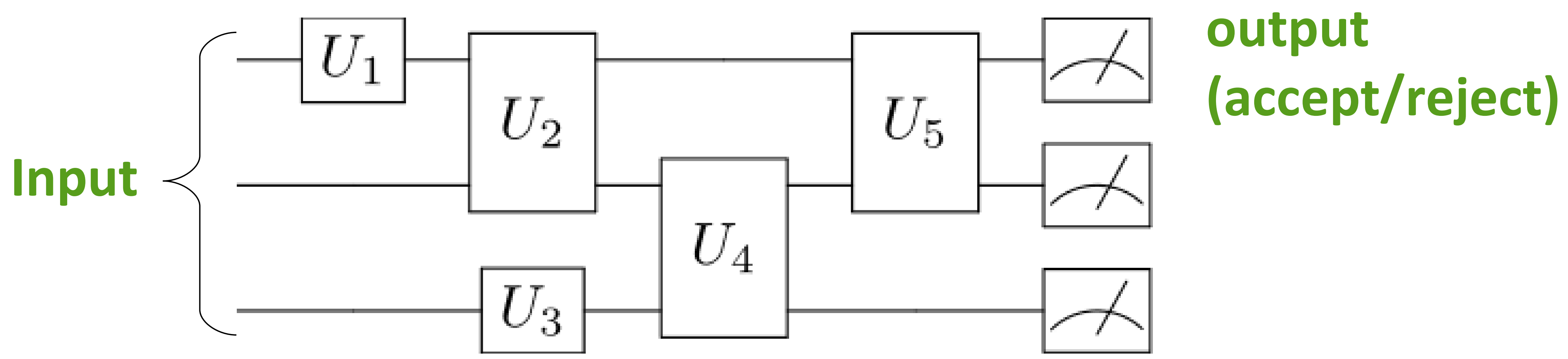
Under some assumptions, quantum computers can sample from distributions that are intractable for classical computers.

	Simulation error	Assumptions	Other
Boson sampling (Aaronson-Arkhipov'13)	**additive**	**2 permanent conjectures**	Implemented, error-correction
IQP I (Bremner et al. '11)	**multiplicative**	**PH does not collapse**	possibly lower FT-treshold
IQP II (Bremner et al. '15)	**additive**	**Any one of two conjectures**	possibly lower FT-treshold
Fourier sampling (Fefferman et al. '16)	**additive**	**Variants of conjectures in A-A**	probably requires universal QC
Our classification:	**multiplicative**	**PH does not collapse**	possibly lower FT-treshold

Part 2: Techniques

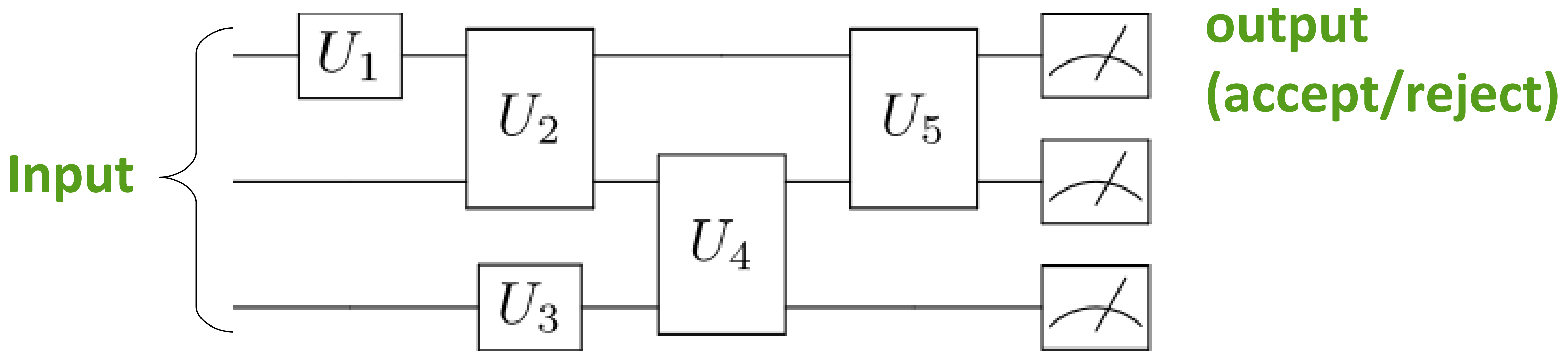
Computation with postselection

BQP – the class of languages decidable by uniform polysize quantum circuits



Computation with postselection

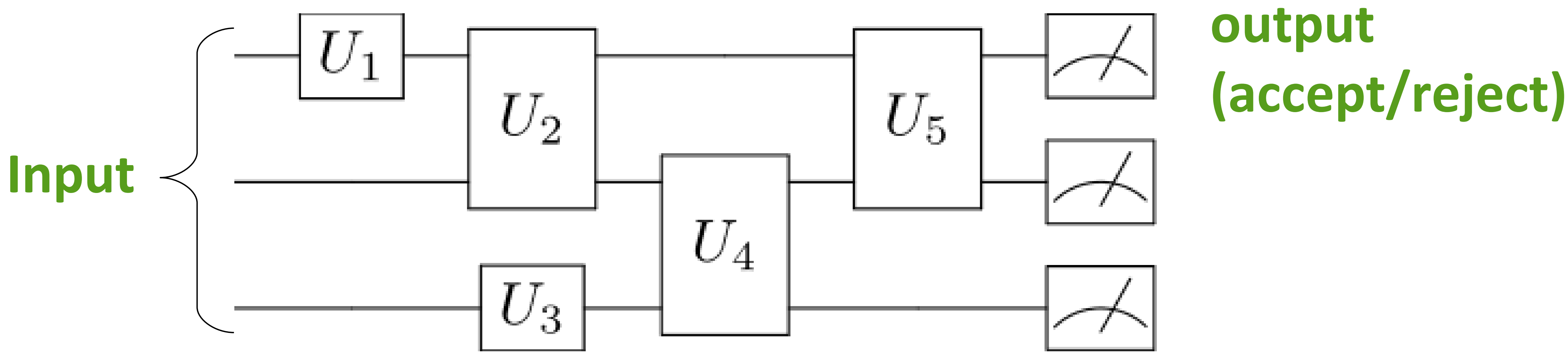
BQP – the class of languages decidable by uniform polysize quantum circuits



PostBQP – the class of languages decidable by uniform polysize quantum circuits **+ wishful thinking**

Computation with postselection

BQP – the class of languages decidable by uniform polysize quantum circuits

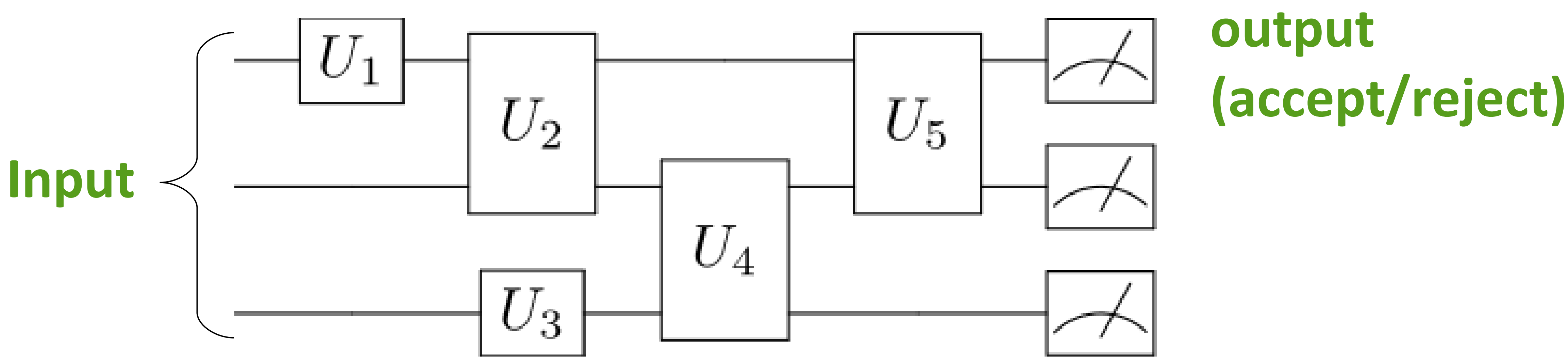


PostBQP – the class of languages decidable by uniform polysize quantum circuits **+ wishful thinking**

We can similarly define PostBPP and post-selected *H*-computations!

Computation with postselection

BQP – the class of languages decidable by uniform polysize quantum circuits



PostBQP – the class of languages decidable by uniform polysize quantum circuits **+ wishful thinking**

We can similarly define PostBPP and post-selected *H*-computations!

Aaronson '05: PostBQP = PP

Power of postselected oracles: PostBPP vs PostBQP

PostBQP is a very powerful oracle

$$P^{\mathbf{PostBQP}} = P^{\mathbf{PP}} \supseteq PH$$

(follows from PostBQP = PP and Toda's theorem)

Power of postselected oracles: PostBPP vs PostBQP

PostBQP is a very powerful oracle

$$P^{\mathbf{PostBQP}} = P^{\mathbf{PP}} \supseteq PH$$

(follows from PostBQP = PP and Toda's theorem)

PostBPP is less powerful

$$P^{\mathbf{PostBPP}} \subseteq P^{\mathbf{\Delta_3}} = \Delta_3$$

(follows from PostBPP $\subseteq \Delta_3$ where $\Delta_3 = P^{\mathbf{NP^{NP}}}$)

Power of postselected oracles: PostBPP vs PostBQP

PostBPP + quantum = PostBQP

Overall proof idea

Given: Entangling commuting 2-qubit Hamiltonian *H*
Want: If *H*-computations can be simulated classically then PH collapses to its third level

Overall proof idea

Given: Entangling commuting 2-qubit Hamiltonian *H*
Want: If *H*-computations can be simulated classically then PH collapses to its third level

1. Show that postselection boosts the power of *H*-computations to BQP (and hence also PostBQP).

Overall proof idea

Given: Entangling commuting 2-qubit Hamiltonian *H*
Want: If *H*-computations can be simulated classically then PH collapses to its third level

1. Show that postselection boosts the power of *H*-computations to BQP (and hence also PostBQP).

So PH $\subseteq$ P$^{\text{PostH}}$

Overall proof idea

Given: Entangling commuting 2-qubit Hamiltonian *H*
Want: If *H*-computations can be simulated classically then PH collapses to its third level

1. Show that postselection boosts the power of *H*-computations to BQP (and hence also PostBQP).

So PH ⊆ P^{PostH}

2. (Bremner et al.) If *H*-computations can be simulated in BPP, then PostH ⊆ PostBPP

Overall proof idea

Given: Entangling commuting 2-qubit Hamiltonian H
Want: If H-computations can be simulated classically then PH collapses to its third level

1. Show that postselection boosts the power of H-computations to BQP (and hence also PostBQP).

$$\textbf{So PH} \subseteq \textbf{P}^{\textbf{PostH}}$$

2. (Bremner et al.) If H-computations can be simulated in BPP, then PostH ⊆ PostBPP and

$$\textbf{P}^{\textbf{PostH}} \subseteq \Delta_3$$

Overall proof idea

Given: Entangling commuting 2-qubit Hamiltonian H
Want: If H-computations can be simulated classically then PH collapses to its third level

1. Show that postselection boosts the power of H-computations to BQP (and hence also PostBQP).

$$\textbf{So PH} \subseteq \textbf{P}^{\textbf{PostH}}$$

2. (Bremner et al.) If H-computations can be simulated in BPP, then PostH ⊆ PostBPP and

$$\textbf{P}^{\textbf{PostH}} \subseteq \Delta_3$$

3. **Summing up:** If BPP machines can simulate H-computations, then PH collapses to its third level.

Current goal: For entangling H, PostH = BQP

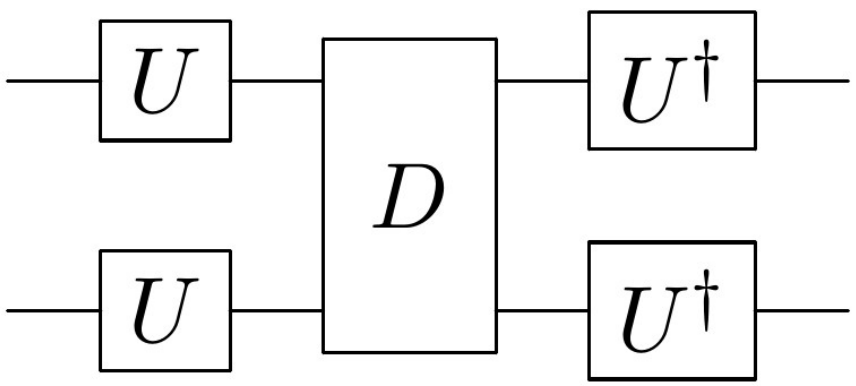
Lemma

Any commuting 2-qubit *H*, can be diagonalized locally:

$$U = \begin{pmatrix} \alpha & \beta \\ -\beta^* & \alpha^* \end{pmatrix}$$

$$D = \mathrm{diag}(a, b, c, d)$$

H =



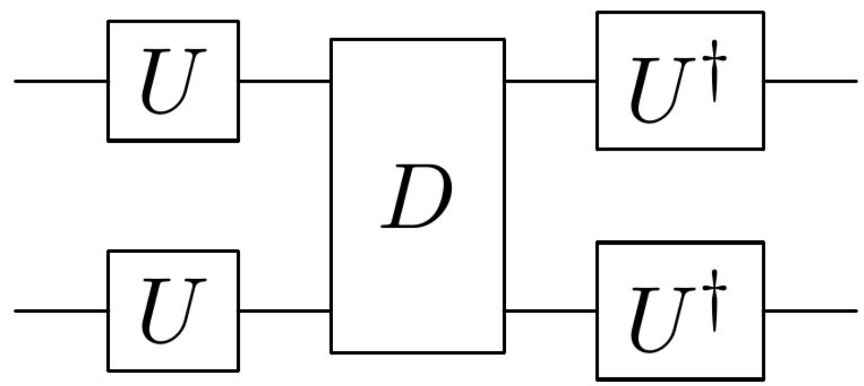
Current goal: For entangling H, PostH = BQP

Lemma

Any commuting 2-qubit *H*, can be diagonalized locally:

$$U = \begin{pmatrix} \alpha & \beta \\ -\beta^* & \alpha^* \end{pmatrix}$$

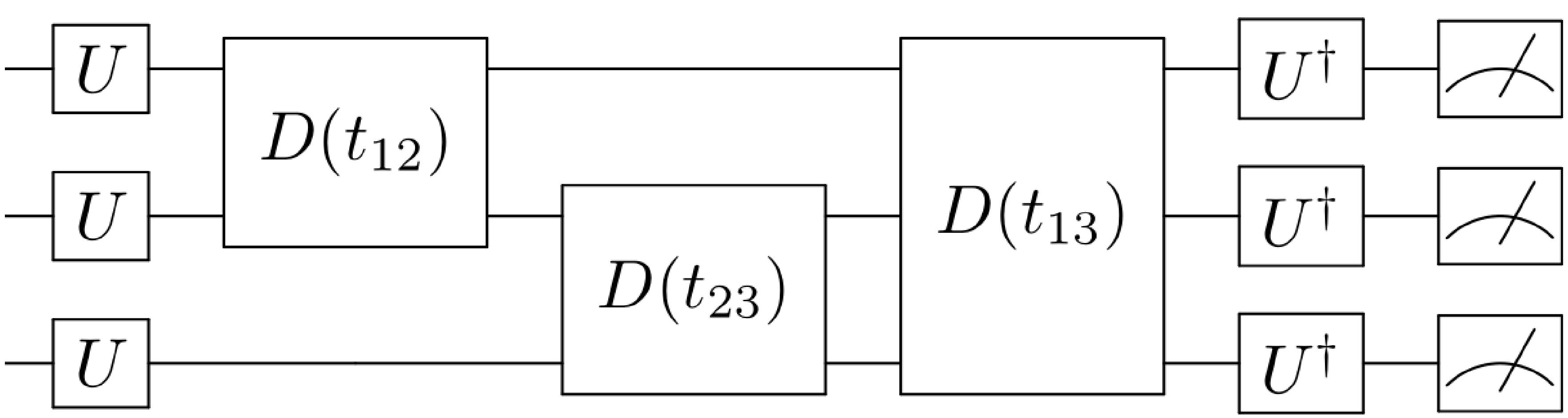
H =



$$D = \mathrm{diag}(a, b, c, d)$$

Observation

We can think of *H*-computations as circuits starting with a column of *U*'s and ending with a column of $U^\dagger$'s and diagonal *D* gates in between.



$$D(t) = e^{iDt}$$

Current goal: For entangling H, PostH = BQP

Theorem (Dodd et al. '02)

One-qubit gates + any entangling gate form a universal gate set.

Current goal: For entangling H, PostH = BQP

Theorem (Dodd et al. '02)
One-qubit gates + any entangling gate form a universal gate set.

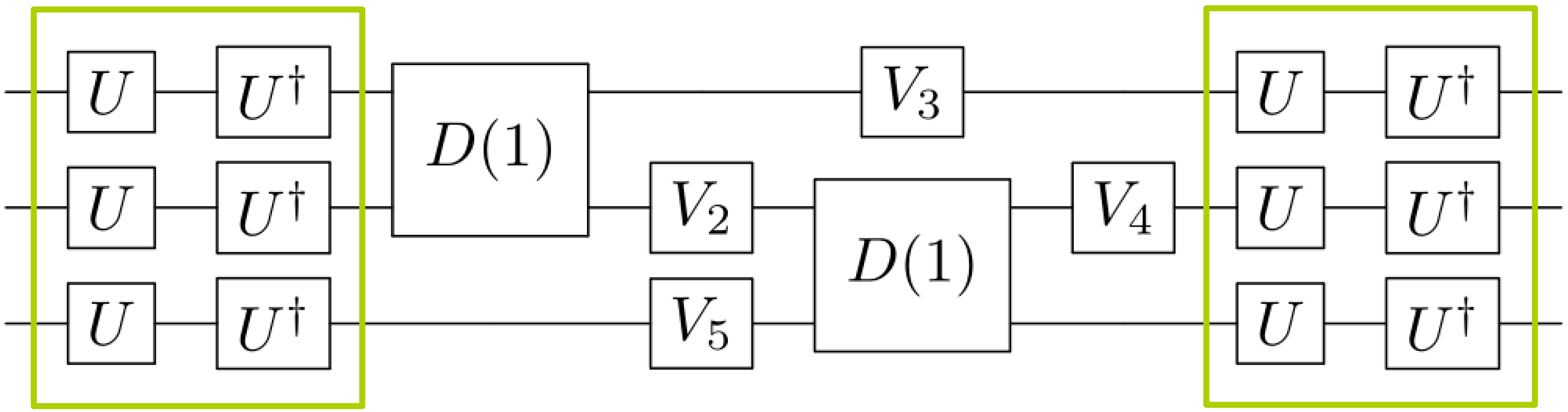
Note: $\exp(iD) = D(1)$ is an entangling gate.

Current goal: For entangling H, PostH = BQP

Theorem (Dodd et al. '02)
One-qubit gates + any entangling gate form a universal gate set.

Note: exp(iD) = D(1) is an entangling gate.

Given a BQP circuit C
1. Express C in terms of D(1) and 1-qubit gates.
2. Add $I = UU^\dagger$ in the beginning and at the end of every line.

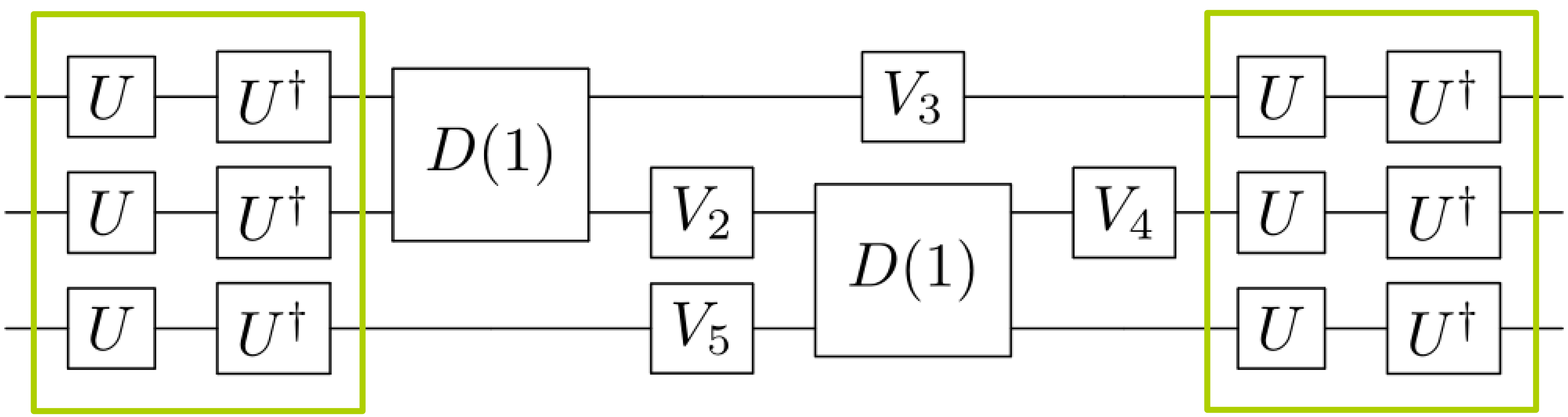


Current goal: For entangling H, PostH = BQP

Theorem (Dodd et al. '02)
One-qubit gates + any entangling gate form a universal gate set.

Note: $\exp(iD) = D(1)$ is an entangling gate.

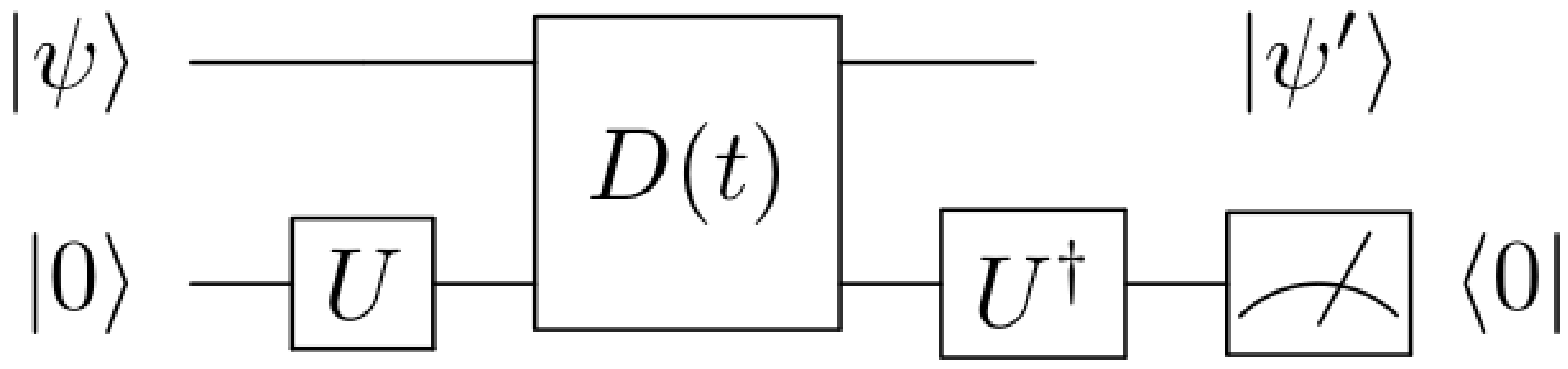
Given a BQP circuit *C*
1. Express C in terms of *D*(1) and 1-qubit gates.
2. Add $I = UU^\dagger$ in the beginning and at the end of every line.



Except for the one-qubit gates this looks just like an *H*-computations circuit from the previous slide.

Current goal: Simulate 1-qubit gates

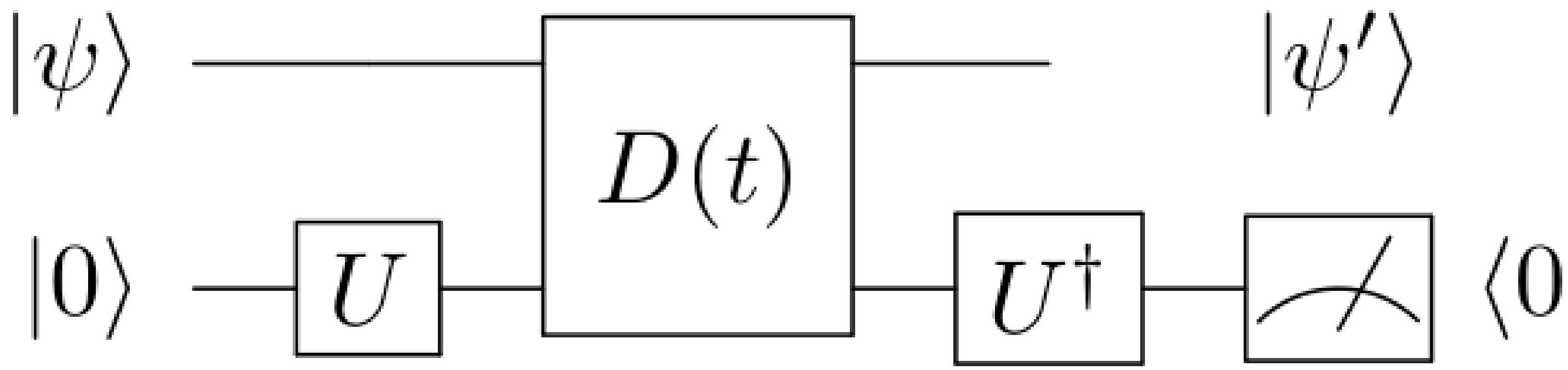
Postselection gadget



$$L(t) = \frac{1}{|\alpha||\beta|\sqrt{-2i\sin(2t)}} \begin{pmatrix} |\alpha|^2 e^{ia't} & \alpha\beta^* e^{it} \\ \alpha^*\beta e^{it} & |\beta|^2 e^{id't)} \end{pmatrix}$$

Current goal: Simulate 1-qubit gates

Postselection gadget



$$L(t) = \frac{1}{|\alpha||\beta|\sqrt{-2i\sin(2t)}} \begin{pmatrix} |\alpha|^2 e^{ia't} & \alpha\beta^* e^{it} \\ \alpha^*\beta e^{it} & |\beta|^2 e^{id't)} \end{pmatrix}$$

Caution: *L*(*t*) is not unitary!

But we can assume that *L*(*t*) is in SL(2,C).

Current goal: Simulate 1-qubit gates

We would be done, if we could approximate any 2x2 unitary using products of *L*(*t*)'s.

$$S = \overline{< L(t)'s >}$$

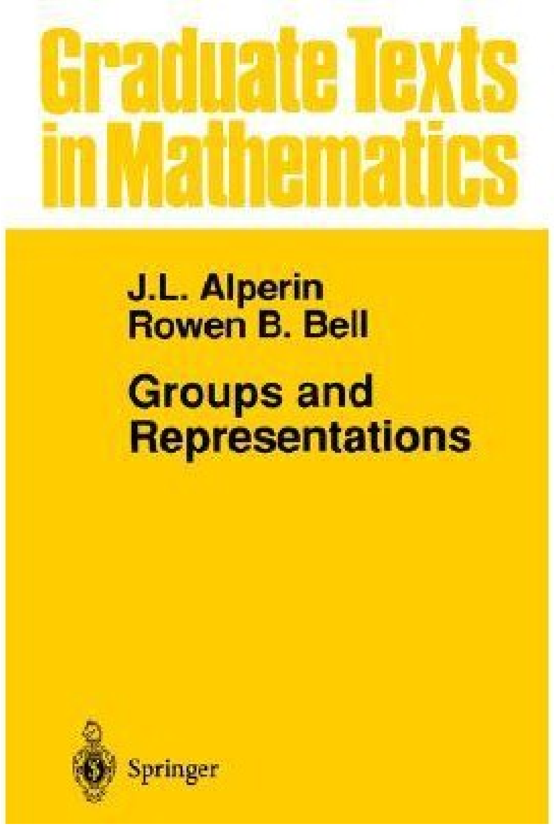
Current goal: Simulate 1-qubit gates

We would be done, if we could approximate any 2x2 unitary using products of $L(t)$'s.

$$S = \overline{< L(t)'s >}$$

If S was a group we could turn to group theory for help.

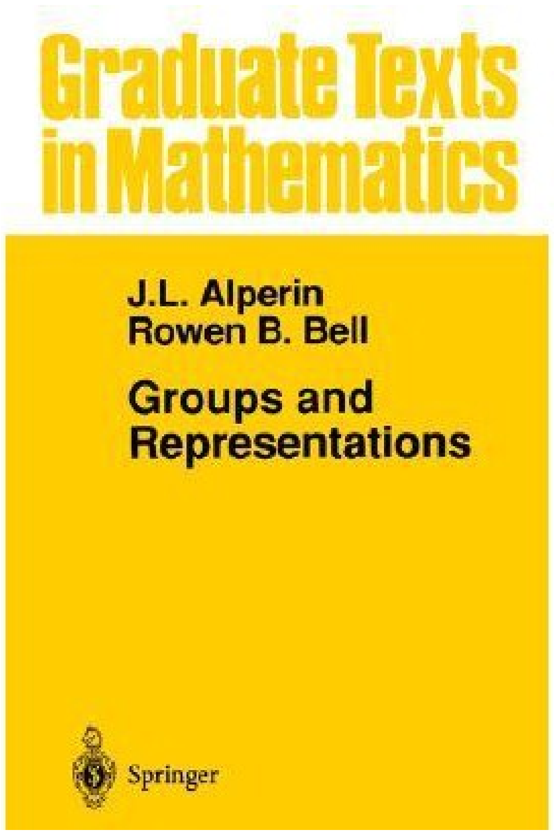
Unfortunately, S might not contain inverses.



Current goal: Simulate 1-qubit gates

We would be done, if we could approximate any 2x2 unitary using products of *L*(*t*)'s.

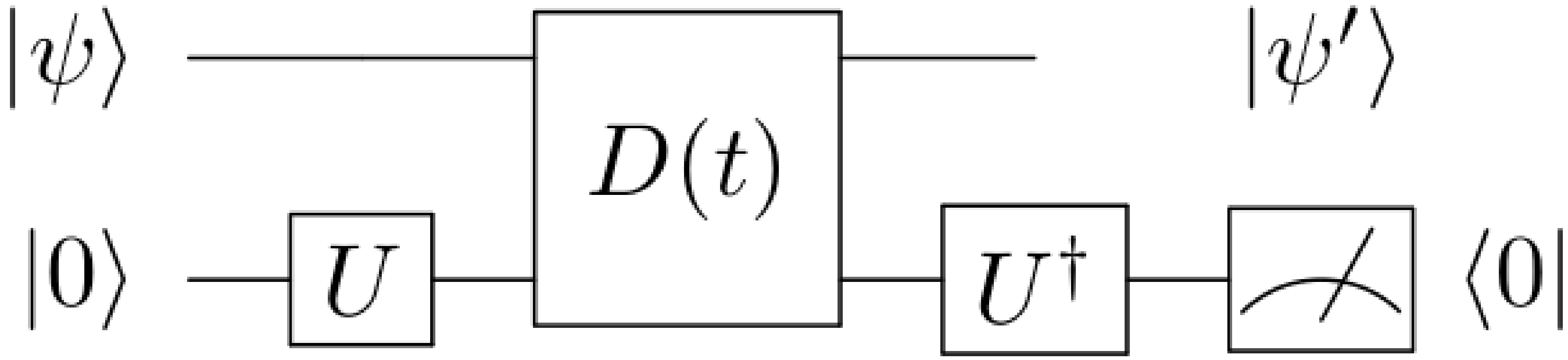
$$S = \overline{< L(t)'s >}$$



If *S* was a group we could turn to group theory for help.

Unfortunately, *S* might not contain inverses.

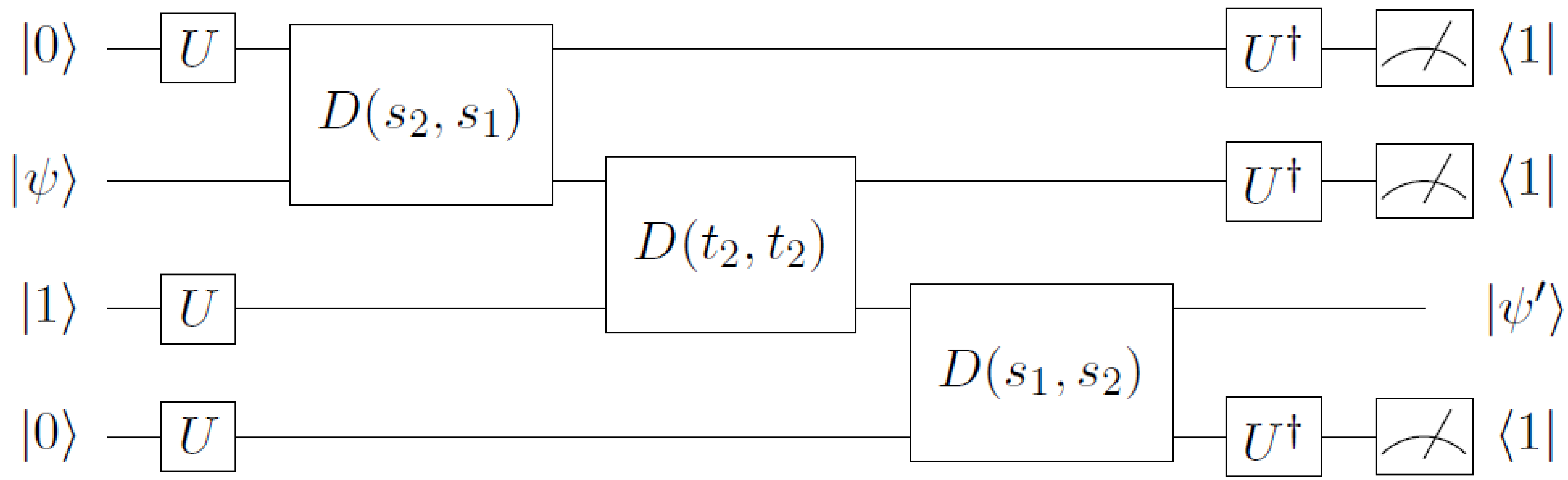
How can we invert our postselection gadget?



Current goal: Simulate 1-qubit gates

Inverting our postselection gadget:

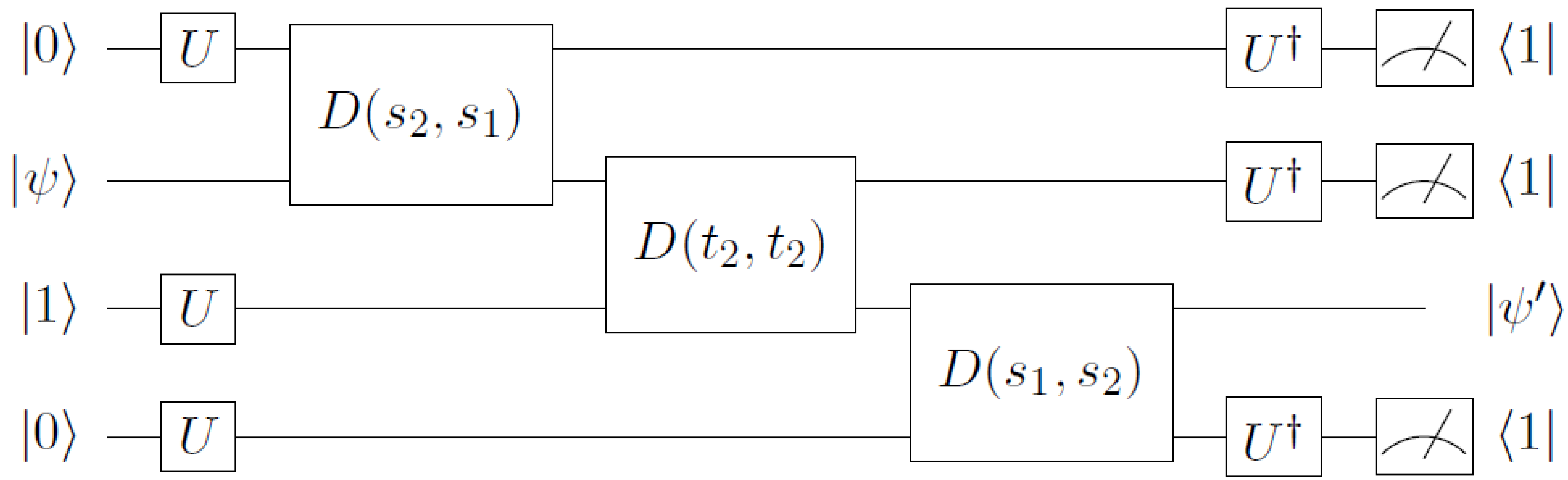
$$L(t)^{-1} =$$



Current goal: Simulate 1-qubit gates

Inverting our postselection gadget:

$$L(t)^{-1} =$$



Our construction becomes more complicated if *H* has degenerate eigenvalues.

Current goal: Simulate 1-qubit gates

The new gadgets let us take products of both *L*(t)'s and the inverses to approximate any desired 1-qubit unitary.

$$S = \overline{\left\langle L(t), L^{-1}(t) \right\rangle}$$

Current goal: Simulate 1-qubit gates

The new gadgets let us take products of both *L*(t)'s and the inverses to approximate any desired 1-qubit unitary.

$$S = \overline{\left\langle L(t), L^{-1}(t) \right\rangle}$$

Now *S* is a group. Moreover, it is a **closed** subgroup of the Lie group SL(2,C).

Current goal: Simulate 1-qubit gates

The new gadgets let us take products of both *L*(t)'s and the inverses to approximate any desired 1-qubit unitary.

$$S = \overline{\left\langle L(t), L^{-1}(t) \right\rangle}$$

Now *S* is a group. Moreover, it is a **closed** subgroup of the Lie group SL(2,C).

Theorem (Cartan)
Any closed subgroup of a Lie group is a Lie group.

Current goal: Simulate 1-qubit gates

The new gadgets let us take products of both *L*(t)'s and the inverses to approximate any desired 1-qubit unitary.

$$S = \overline{\left\langle L(t), L^{-1}(t) \right\rangle}$$

Now *S* is a group. Moreover, it is a **closed** subgroup of the Lie group SL(2,C).

Theorem (Cartan)
Any closed subgroup of a Lie group is a Lie group.

We would be done, if we could show that *S* = SL(2,C).

Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*

Consider Lie algebra Lie(*S*) of *S*.

Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*

Consider Lie algebra Lie(*S*) of *S*. Why??

Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*

Consider Lie algebra Lie(*S*) of *S*. Why??

Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*

Consider Lie algebra Lie(*S*) of *S*. Why??

	Lie group G	Lie algebra, Lie(G)
Closed under:	products	**real** linear combinations, commutators

Example: For Lie group U(2), the Lie algebra consists of all iH, where H is Hermitian.

Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*

Consider Lie algebra Lie(*S*) of *S*. Why??

	Lie group G	Lie algebra, Lie(G)
Closed under:	products	**real** linear combinations, commutators

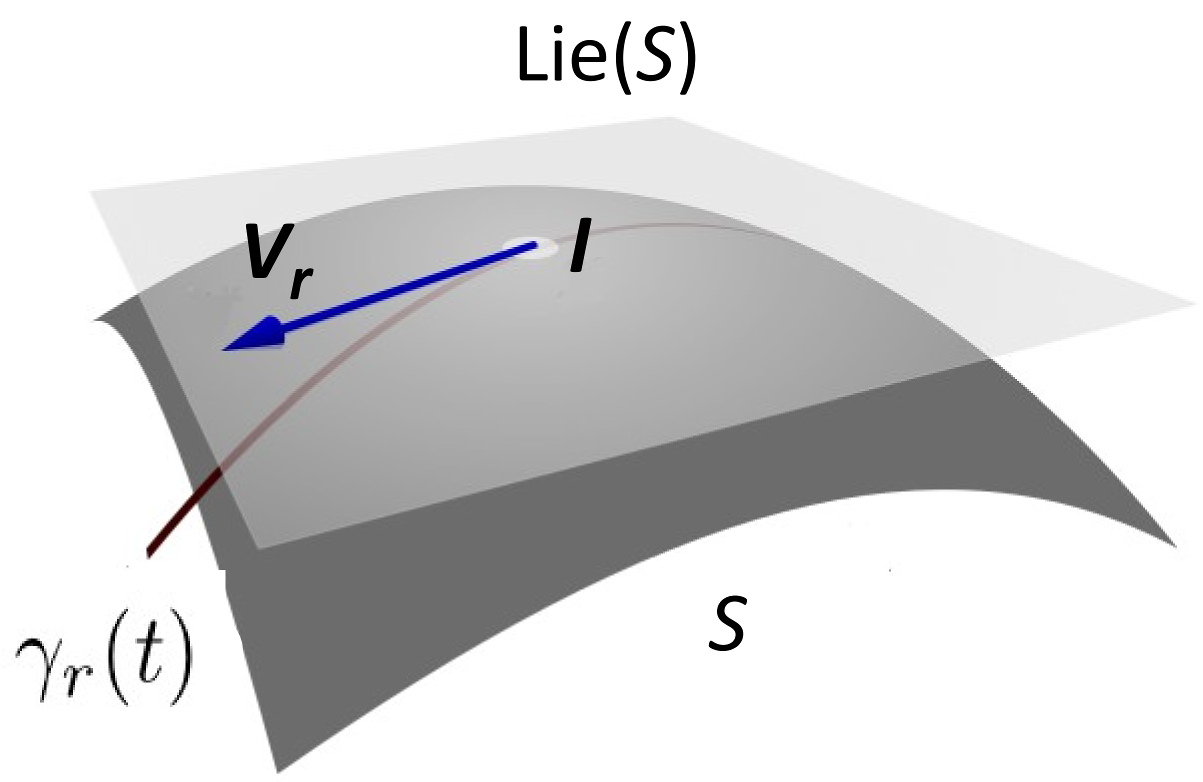
Example: For Lie group U(2), the Lie algebra consists of all *iH*, where *H* is Hermitian.

Exponential map

exp: Lie(G) → G

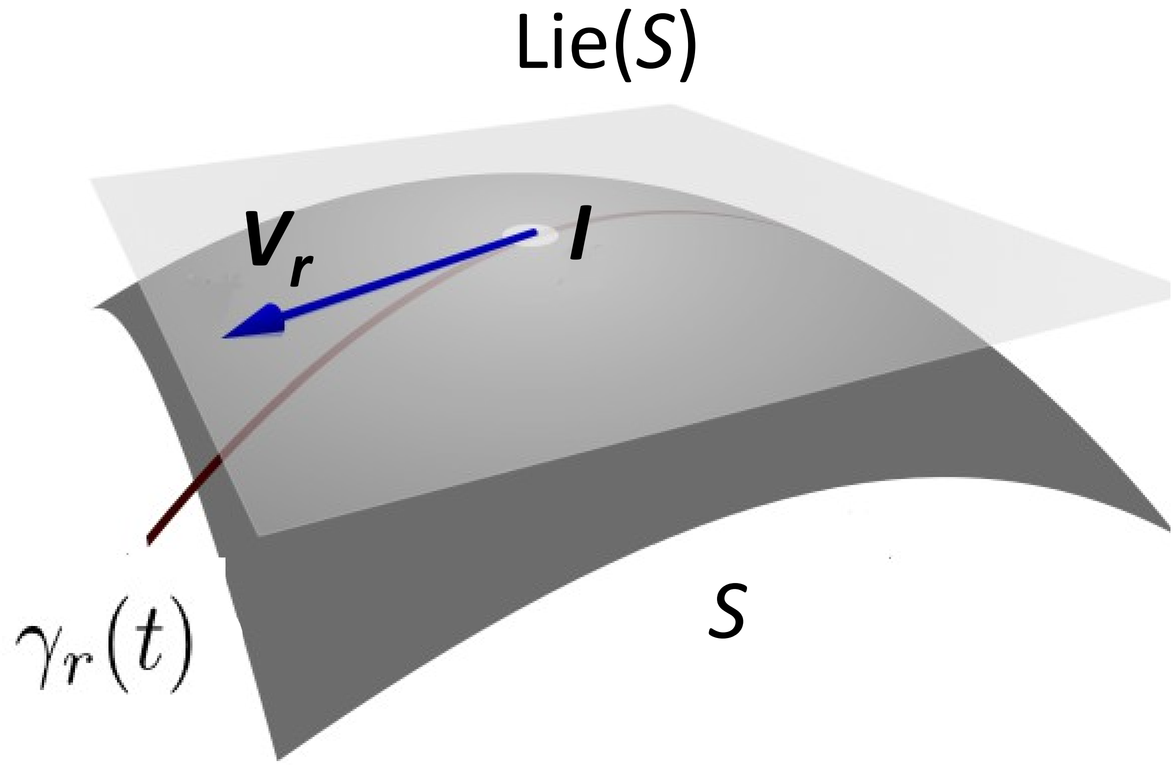
Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*



Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*

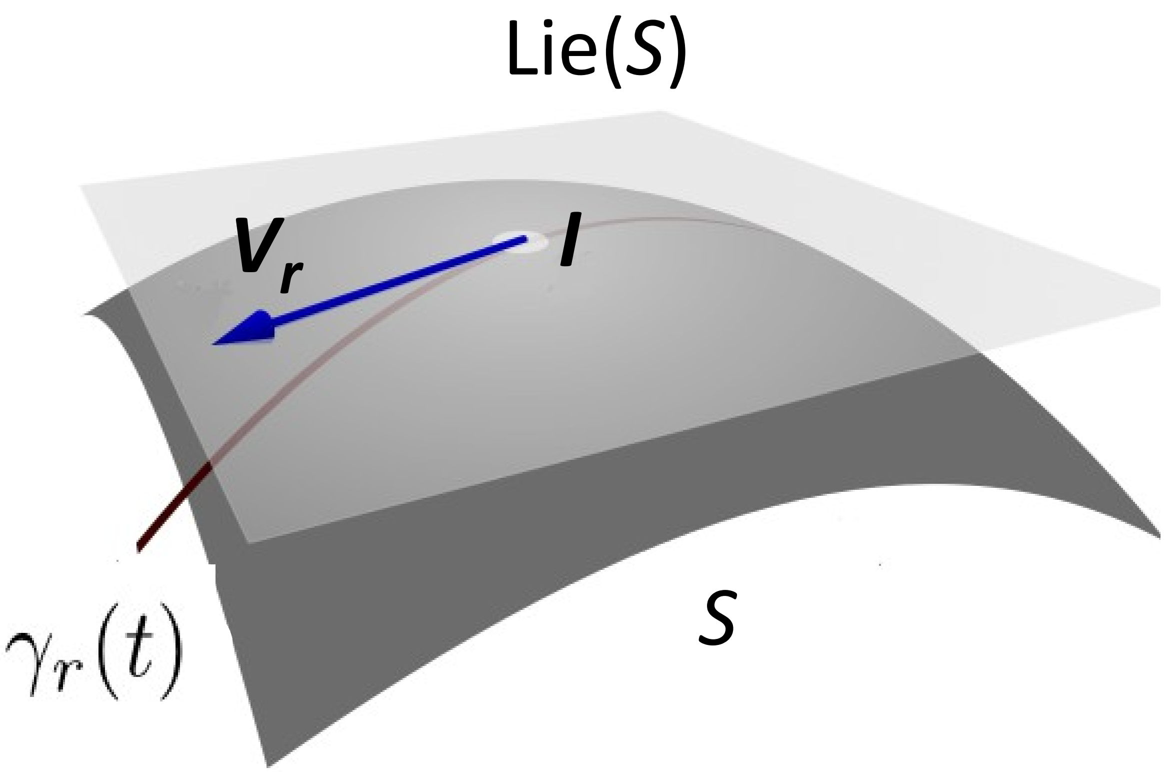


- Let $\gamma_r(t) := L(t)L^{-1}(r)$.
- Then $\gamma_r(r) = I$ and so for any $t \in \mathbb{R}$

$$\left.\frac{\partial \gamma_r}{\partial t}\right|_{t=r} =: V_r \in \mathrm{Lie}(S)$$

Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*



- Let $\gamma_r(t) := L(t)L^{-1}(r)$.
- Then $\gamma_r(r) = I$ and so for any $t \in \mathbb{R}$

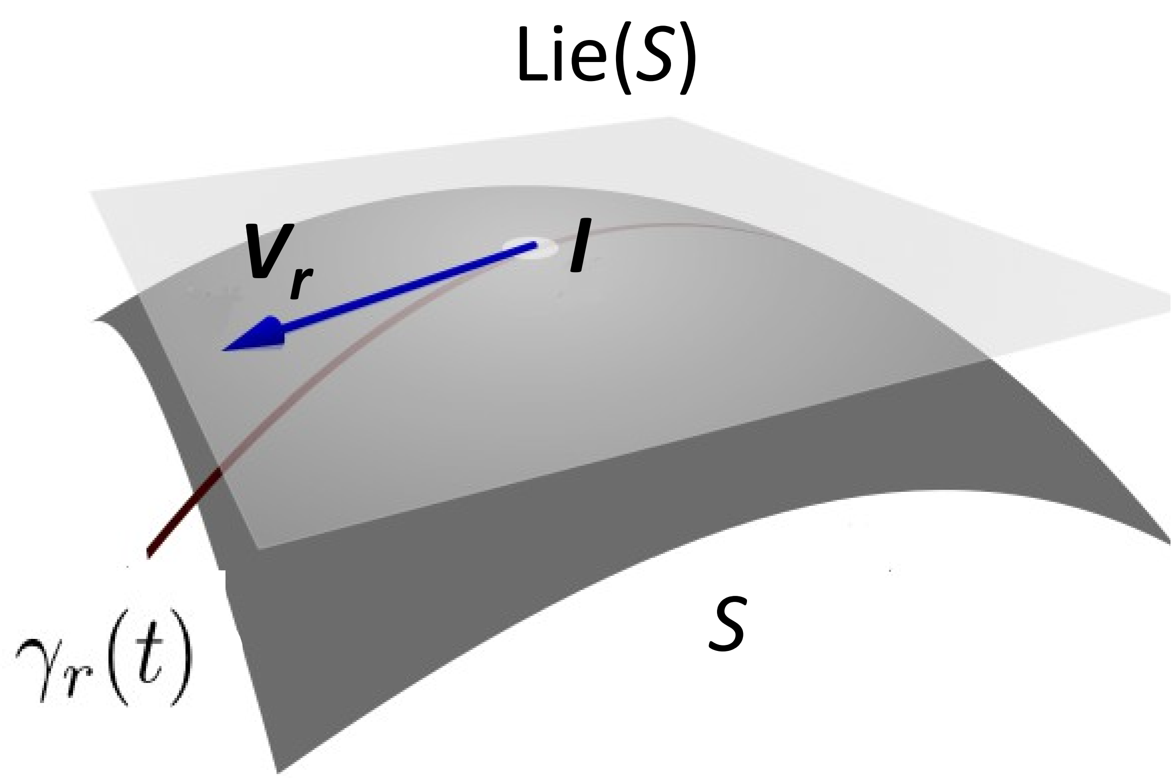
$$\left.\frac{\partial \gamma_r}{\partial t}\right|_{t=r} =: V_r \in \mathrm{Lie}(S)$$

Lie algebra of SL(2,C)

sl(2,C) = all traceless 2x2 matrices (6 dimensions)

Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*



- Let $\gamma_r(t) := L(t)L^{-1}(r)$.
- Then $\gamma_r(r) = I$ and so for any $t \in \mathbb{R}$

$$\left.\frac{\partial \gamma_r}{\partial t}\right|_{t=r} =: V_r \in \mathrm{Lie}(S)$$

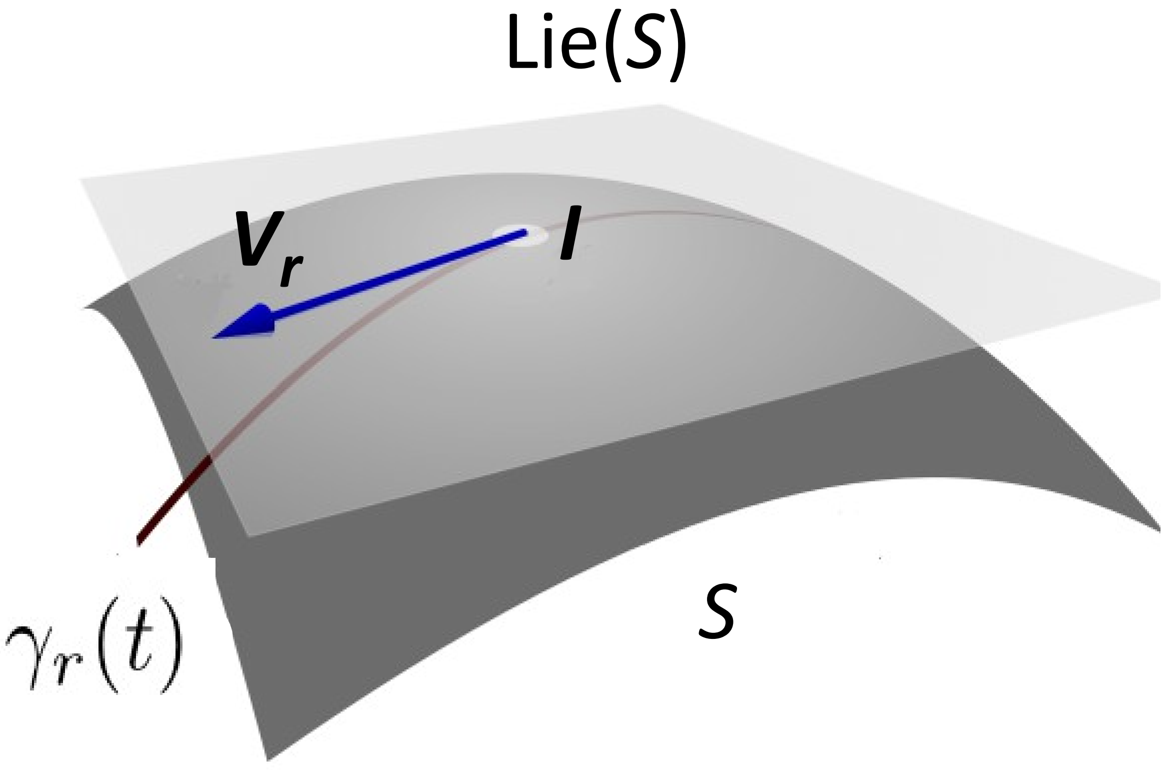
Lie algebra of SL(2,C)

sl(2,C) = all traceless 2x2 matrices (6 dimensions)

- We can take commutators and real linear combinations of the V_r's
- Find a basis of the 6-dimensional space sl(2,C)

Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*



- Let $\gamma_r(t) := L(t)L^{-1}(r)$.
- Then $\gamma_r(r) = I$ and so for any $t \in \mathbb{R}$

$$\left.\frac{\partial \gamma_r}{\partial t}\right|_{t=r} =: V_r \in \mathrm{Lie}(S)$$

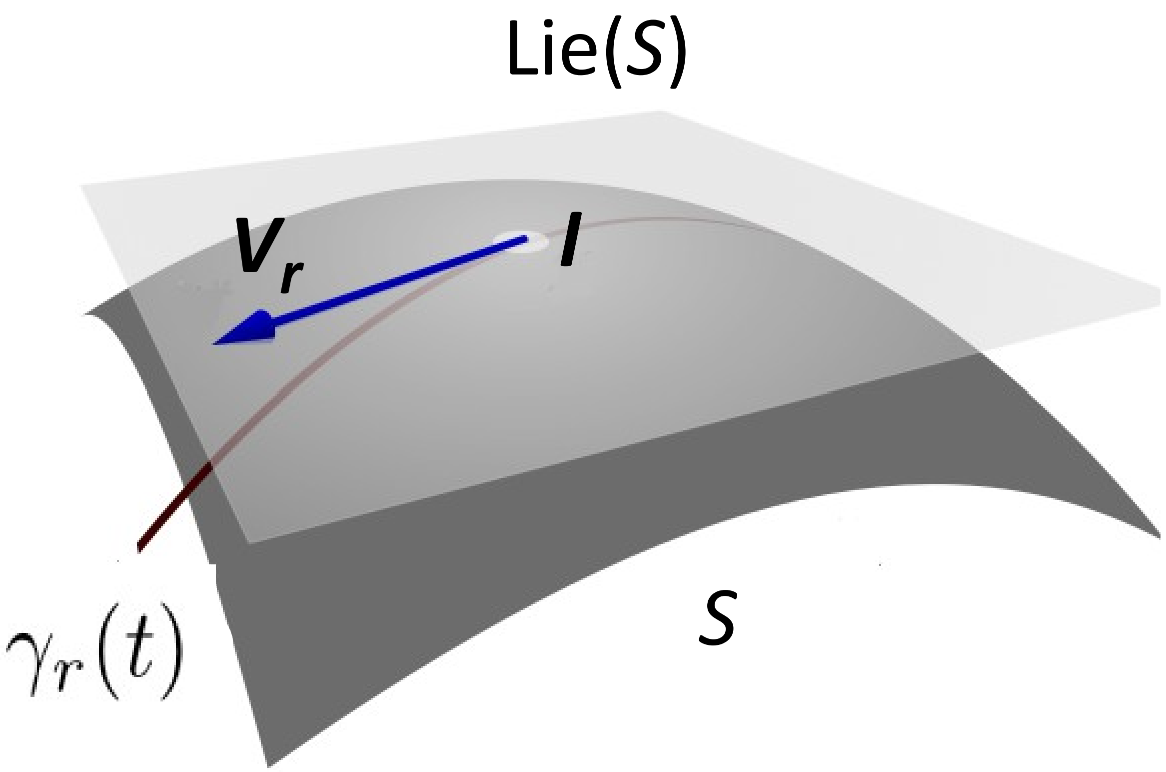
Lie algebra of SL(2,C)

sl(2,C) = all traceless 2x2 matrices (6 dimensions)

- We can take commutators and real linear combinations of the V_r's
- Find a basis of the 6-dimensional space sl(2,C)
- So Lie(S) = sl(2,C) and also S = SL(2,C)

Current goal: Simulate 1-qubit gates

by showing that SL(2,C) = *S*



- Let $\gamma_r(t) := L(t)L^{-1}(r)$.
- Then $\gamma_r(r) = I$ and so for any $t \in \mathbb{R}$

$$\left.\frac{\partial \gamma_r}{\partial t}\right|_{t=r} =: V_r \in \mathrm{Lie}(S)$$

Lie algebra of SL(2,C)

sl(2,C) = all traceless 2x2

- We can take commutators and real linear combinations of the V_r's
- Find a basis of the 6-dimensional space sl(2,C)
- So Lie(S) = sl(2,C) and also S = SL(2,C)

Summary

Main result.
A commuting 2-qubit Hamiltonian which can create entanglement from standard basis states gives rise to classically intractable probability distributions unless PH collapses to its third level.

Open problems

- Complete the classification of H-computations for noncommuting H

Open problems

- Complete the classification of *H*-computations for noncommuting *H*
- Extend the results to L1 norm

Open problems

- Complete the classification of *H*-computations for noncommuting *H*
- Extend the results to L1 norm
- Establish quantum advantage for a **natural** sampling or a **decision** problem

Open problems

- Complete the classification of *H*-computations for noncommuting *H*
- Extend the results to L1 norm
- Establish quantum advantage for a **natural** sampling or a **decision** problem
- Classify commuting gate sets

Open problems

- Complete the classification of *H*-computations for noncommuting *H*
- Extend the results to L1 norm
- Establish quantum advantage for a **natural** sampling or a **decision** problem
- Classify commuting gate sets

Caution:
The classification of subgroups of SU(8) and also SL(2,C) is currently incomplete

Open problems

- Complete the classification of *H*-computations for noncommuting *H*
- Extend the results to L1 norm
- Establish quantum advantage for a **natural** sampling or a **decision** problem
- Classify commuting gate sets

Caution:
The classification of subgroups of SU(8) and also SL(2,C) is currently incomplete

Thank you!