Computations with Greater Quantum Depth Are Strictly More Powerful (Relative to an Oracle)

Matthew Coudron

NIST/QuICS, University of Maryland

Sanketh Menda

University of Waterloo

The Plan

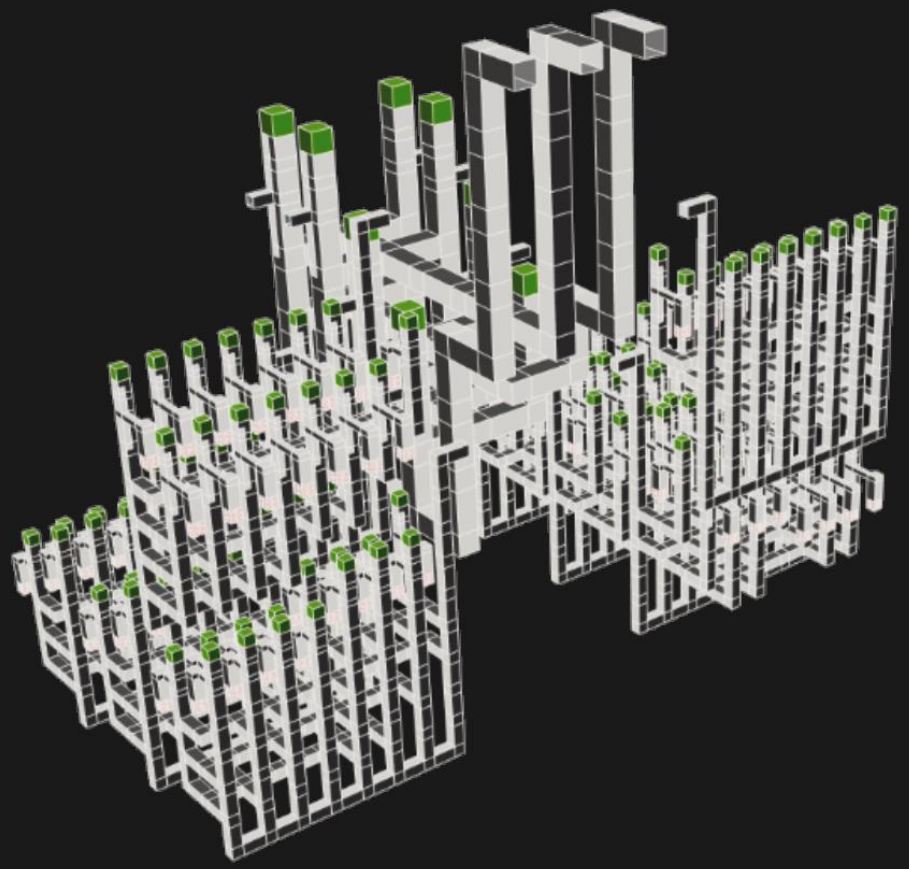
1. Why Hybrid Quantum Computation?
2. Conjectures of Aaronson and Jozsa
3. Main Result
4. The Welded Tree Problem
5. Proof Sketch
6. Open Problems

Before We Get Started

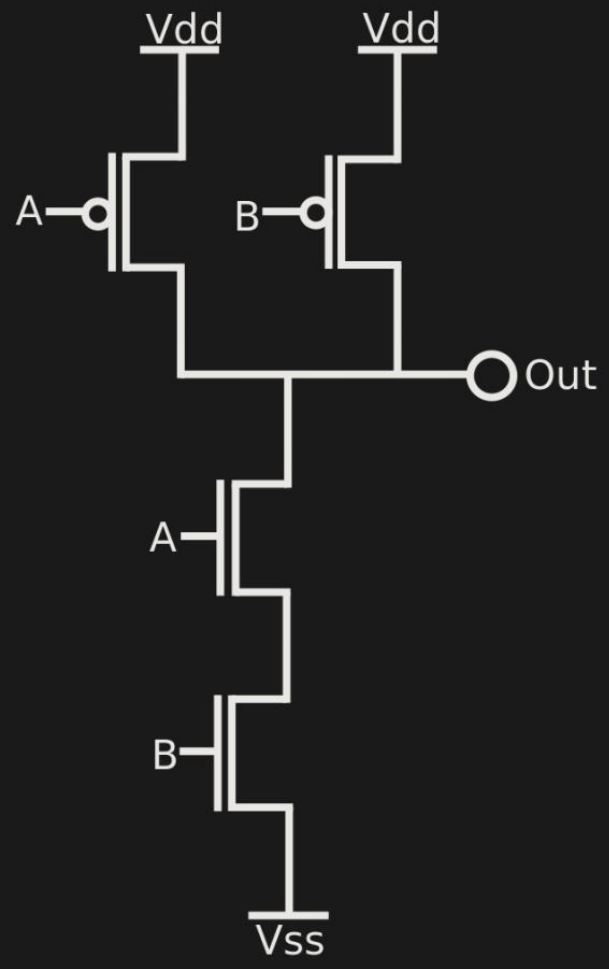
- Feel free to ask clarifying questions during the talk.
- But please reserve comments till the end.
- Keep the un-mute button handy, imma ask questions.
- There is a recorded version of essentially this talk by my co-author Matt, from STOC, so if you want to polish your QIP submission, go for it.

Why Care about Hybrid Models?

Fa ng is



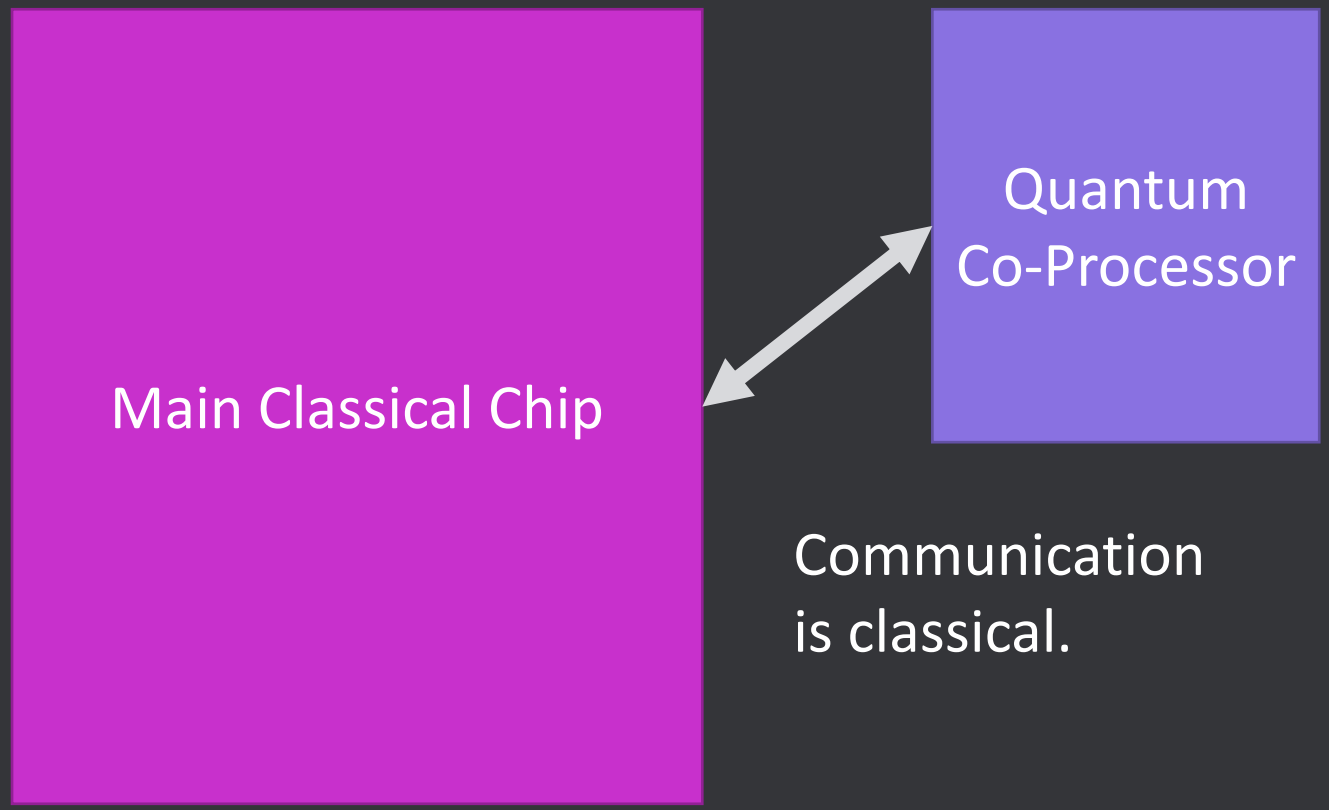
(a) "Quantum NAND"
> 10 qubitseconds



(b) "Classical NAND"
$< 10^{-9}$ transistorseconds

Babbush et al. Focus beyond quadratic speedups for error-corrected quantum advantage. arXiv:2011.04149

Quantum Co-Processor Architecture

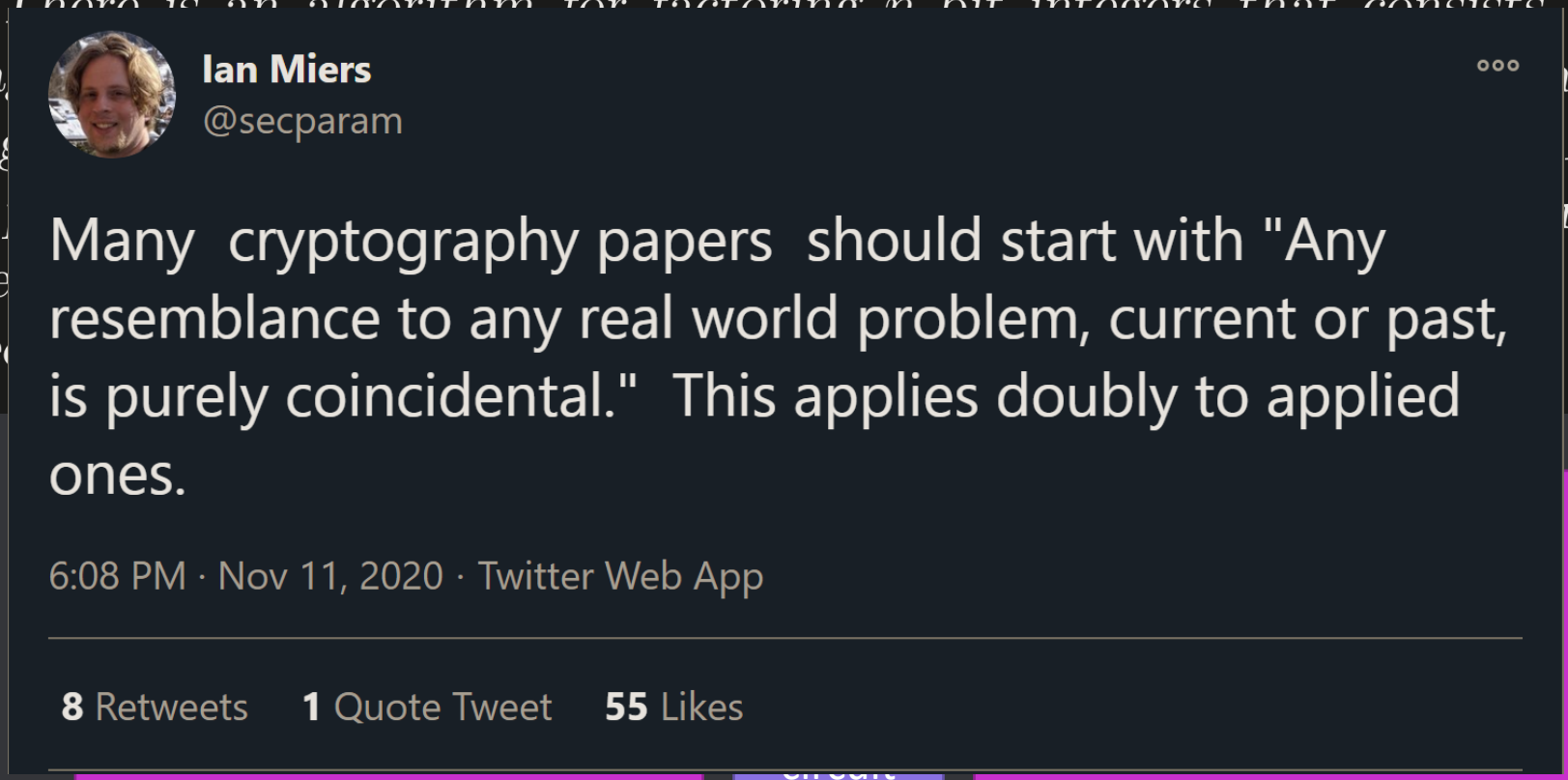


No persistent quantum RAM.

Communication is classical.

Low-Depth Quantum + Classical Interleaving is Powerful

Theorem 3 *There is an algorithm for factoring n bit integers that consists of: a classical pre-processing sta… quantum informa-tion processing… m circuit[1]; followed by a classical … rcuit. Furthermore, the size of the… particular, the size can be reduce…*



Ian Miers
@secparam

Many cryptography papers should start with "Any resemblance to any real world problem, current or past, is purely coincidental." This applies doubly to applied ones.

6:08 PM · Nov 11, 2020 · Twitter Web App

8 Retweets **1** Quote Tweet **55** Likes

Cleve and Watrous. Fast parallel circuits for the quantum Fourier transform. arXiv:quant-ph/0006004

Conjectures of Aaronson and Jozsa

The Main Question

- How much can we parallelize quantum computation (with classical pre- and post-processing)? Can they be generically parallelized?
- **Conjecture** (Jozsa, 2005)**.** Any polynomial time quantum algorithm can be implemented with only $O(\log n)$ quantum layers interspersed with polynomial time classical computations.
- **Conjecture** (Aaronson, 2005)**.** There exists an oracle A such that $\mathrm{BQP}^{\mathrm{A}} \not\subseteq \left(\mathrm{BPP}^{\mathrm{BQNC}}\right)^{\mathrm{A}}$. (Where BQNC is the class of problems that can be solved with polylogarithmic-depth quantum circuits.)

Jozsa. An introduction to measurement based quantum computation. arXiv:quant-ph/0508124
Aaronson. Ten Semi-Grand Challenges for Quantum Computing Theory. https://www.scottaaronson.com/writings/qchallenge.html

Quicky, some Notation

Quantum Tier

polylog-depth, poly-size quantum circuit

Classical Tier

poly-size classical circuit

Moreover, each quantum tier is composed of polylog, depth-1 quantum circuits called “quantum layers”.

Aaronson's Conjecture*

Conjecture (Aaronson, 2005)**.** There exists an oracle A such that $\mathrm{BQP}^A \not\subseteq \left(\mathrm{BPP}^{\mathrm{BQNC}}\right)^A$.

Define the complexity class HQC to capture circuits of the form:

Aaronson. Ten Semi-Grand Challenges for Quantum Computing Theory. https://www.scottaaronson.com/writings/qchallenge.html

Jozsa's Conjecture

Conjecture (Jozsa, 2005). Any polynomial time quantum algorithm can be implemented with only $O(\log n)$ quantum layers interspersed with polynomial time classical computations.

Define the complexity class JQC to capture circuits of the form:

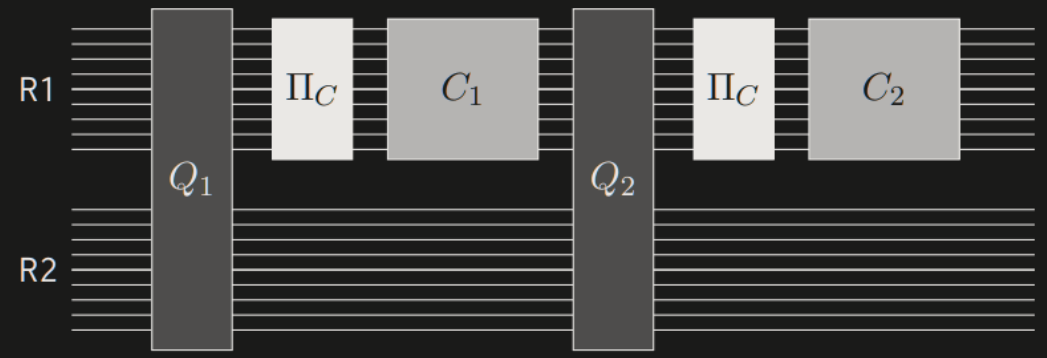


Figure 2: An illustration of an $(n, 2, q, c)$-jozsa-quantum circuit. The light boxes represent quantum circuits, the black boxes represent classical basis measurements, and the dark boxes represent classical circuits. The width of the circuit $g(n)$ is split into two registers R1 and R2 of equal size.

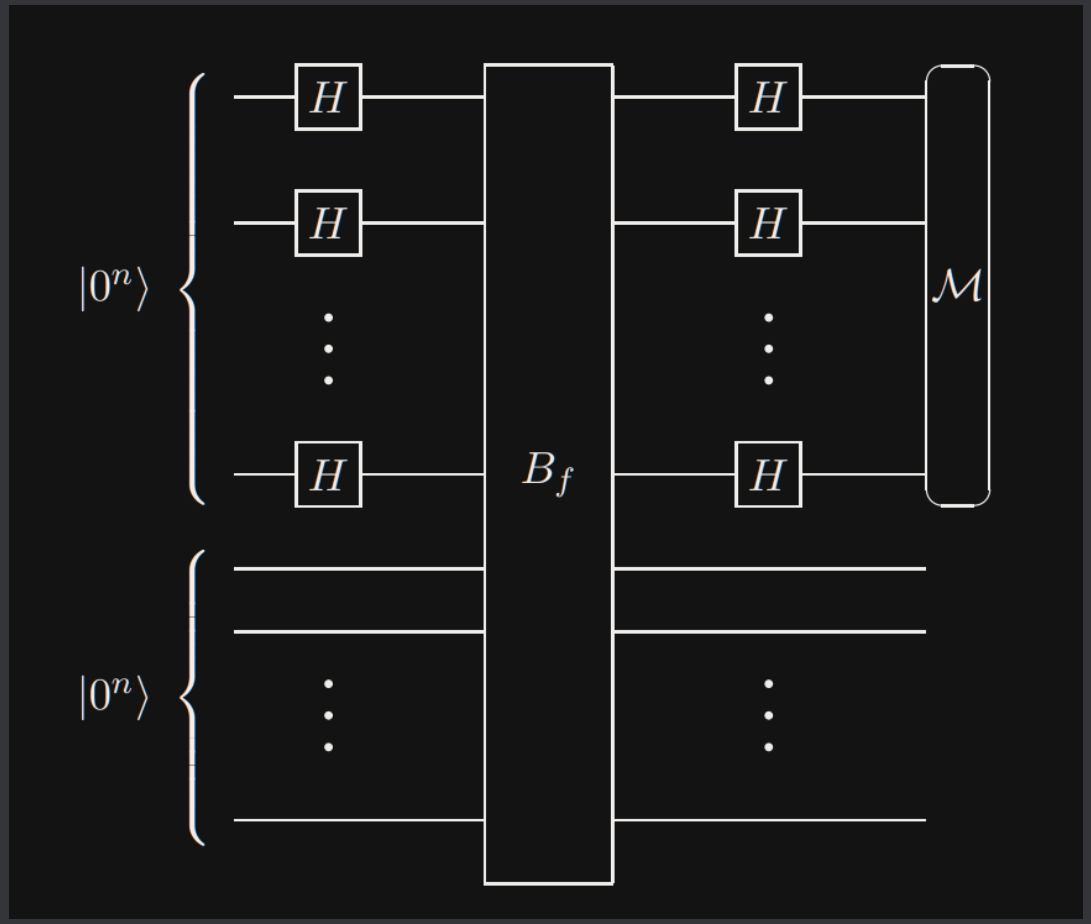
Jozsa. An introduction to measurement based quantum computation. arXiv:quant-ph/0508124

Some Context

- Since $\mathrm{P} \subseteq \mathrm{HQC} \subseteq \mathrm{BQP} \subseteq \mathrm{PSPACE}$, we cannot unconditionally separate HQC from BQP without proving $\mathrm{P} \neq \mathrm{PSPACE}$.
- So, let's try to aim for an oracle separation.

Problem	Speedup over BPP	Quantum Algorithm	Quantum Depth
Bernstein-Vazirani	Super-Polynomial	poly(n)	O(1)
Deutsch-Jozsa	None (over BPP) Exponential (over P)	poly(n)	O(1)
Simon	Exponential	poly(n)	O(1)
Forrelation	Exponential	poly(n)	O(1)

Quick Refresher: The Quantum Part of Simon's Algorithm



Watrous. Lecture 6: Simon's algorithm. https://cs.uwaterloo.ca/~watrous/QC-notes/QC-notes.06.pdf

I Lied

- There is another oracle problem that we could consider.

Problem	Speedup over BPP	Quantum Algorithm	Quantum Depth
Bernstein-Vazirani	Super-Polynomial	poly(n)	O(1)
Deutsch-Jozsa	None (over BPP) Exponential (over P)	poly(n)	O(1)
Simon	Exponential	poly(n)	O(1)
Forrelation	Exponential	poly(n)	O(1)
Welded Tree	**Exponential**	**poly(n)**	**?**

Childs et al. Exponential algorithmic speedup by quantum walk. arXiv:quant-ph/0209131

Main Result

Main Result

Theorem (Informal)**.** There exists an oracle T such that $\mathrm{WeldedTreeProblem}(T) \notin \mathrm{HQC}^T$.

And since we know from Childs et al. that the Welded Tree Oracle problem is in BQP^T for all oracles T, we get:

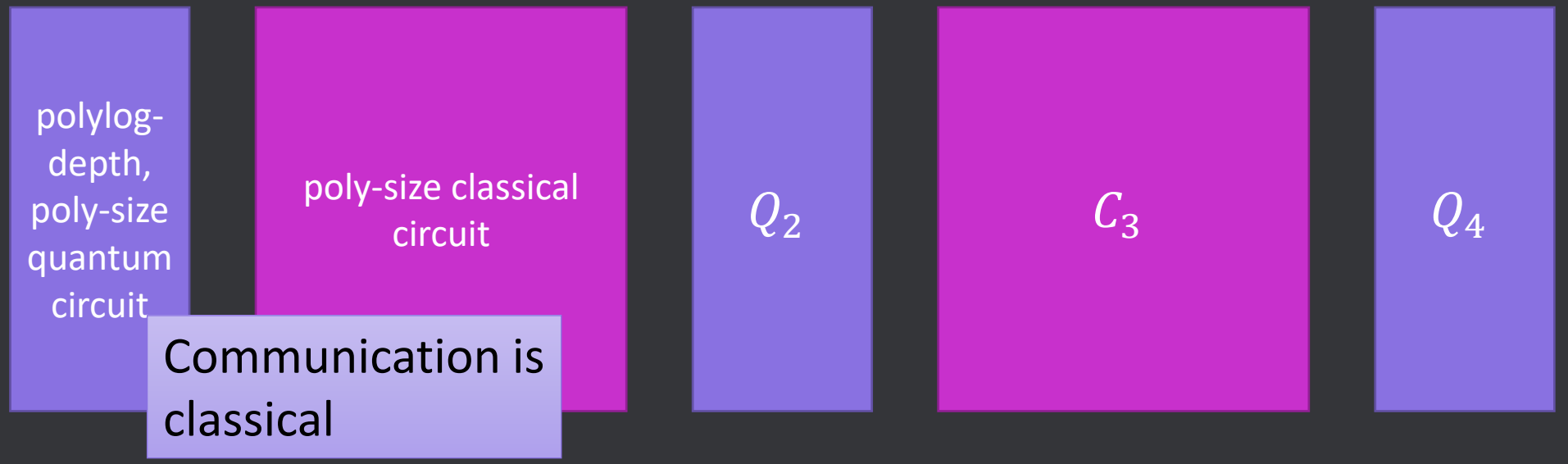
Corollary. There exists an oracle A such that $\mathrm{BQP}^A \not\subseteq \mathrm{HQC}^A$.

This result was independently and concurrently obtained by Nai-Hui Chia, Kai-Min Chung, and Ching-Yi Lai, in *On the Need for Large Quantum Depth* (STOC 2020; arXiv:1909.10303).

Childs et al. Exponential algorithmic speedup by quantum walk. arXiv:quant-ph/0209131

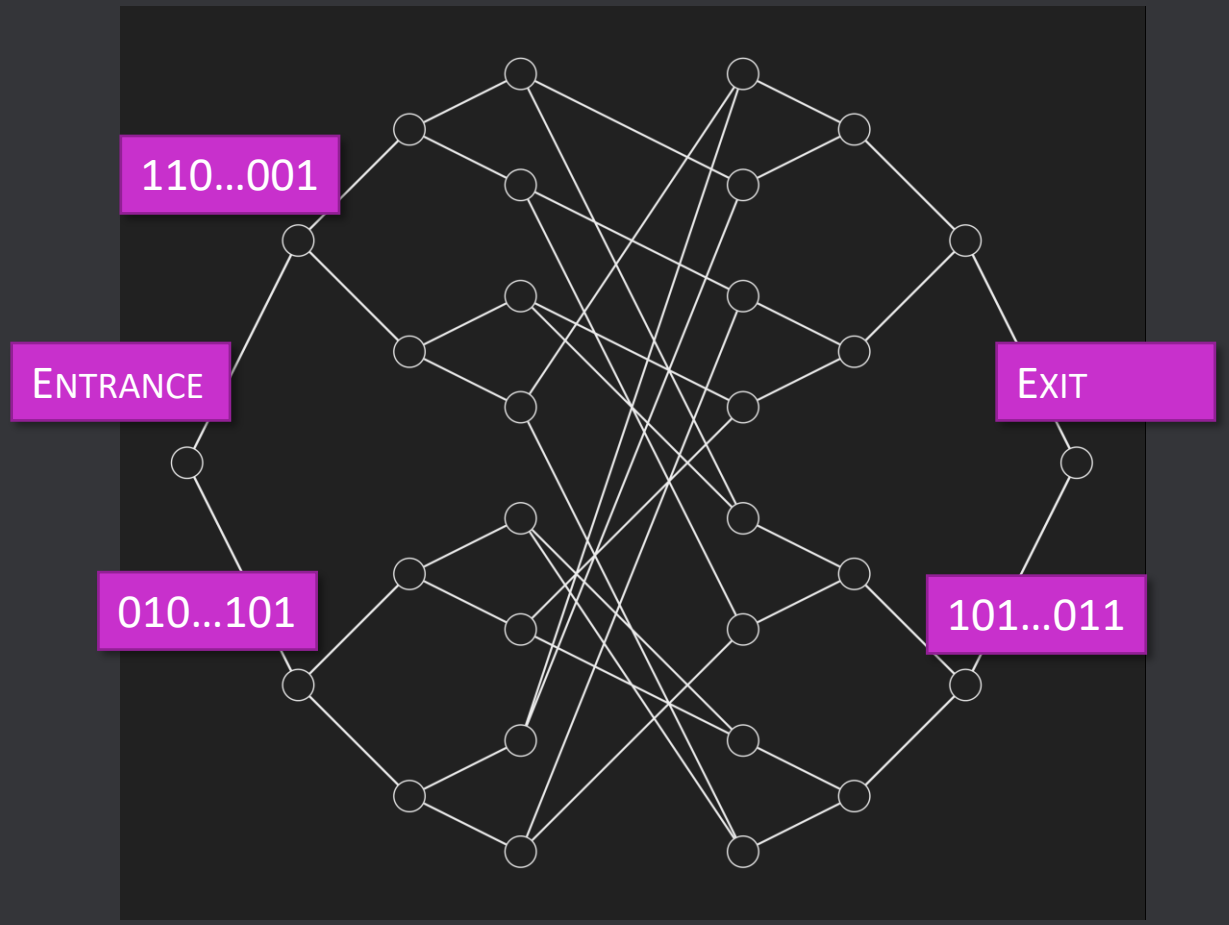
Main Result

Theorem (informal)**.** No quantum algorithm with oracle access to the random welded tree oracle and using only $O(\mathrm{polylog}(n))$-depth quantum circuits, alternated with polynomial time classical computations, polynomially many times, can solved the Welded Tree Oracle problem with probability greater than $O(2^{-\Omega(n)})$.



The Welded Tree Problem

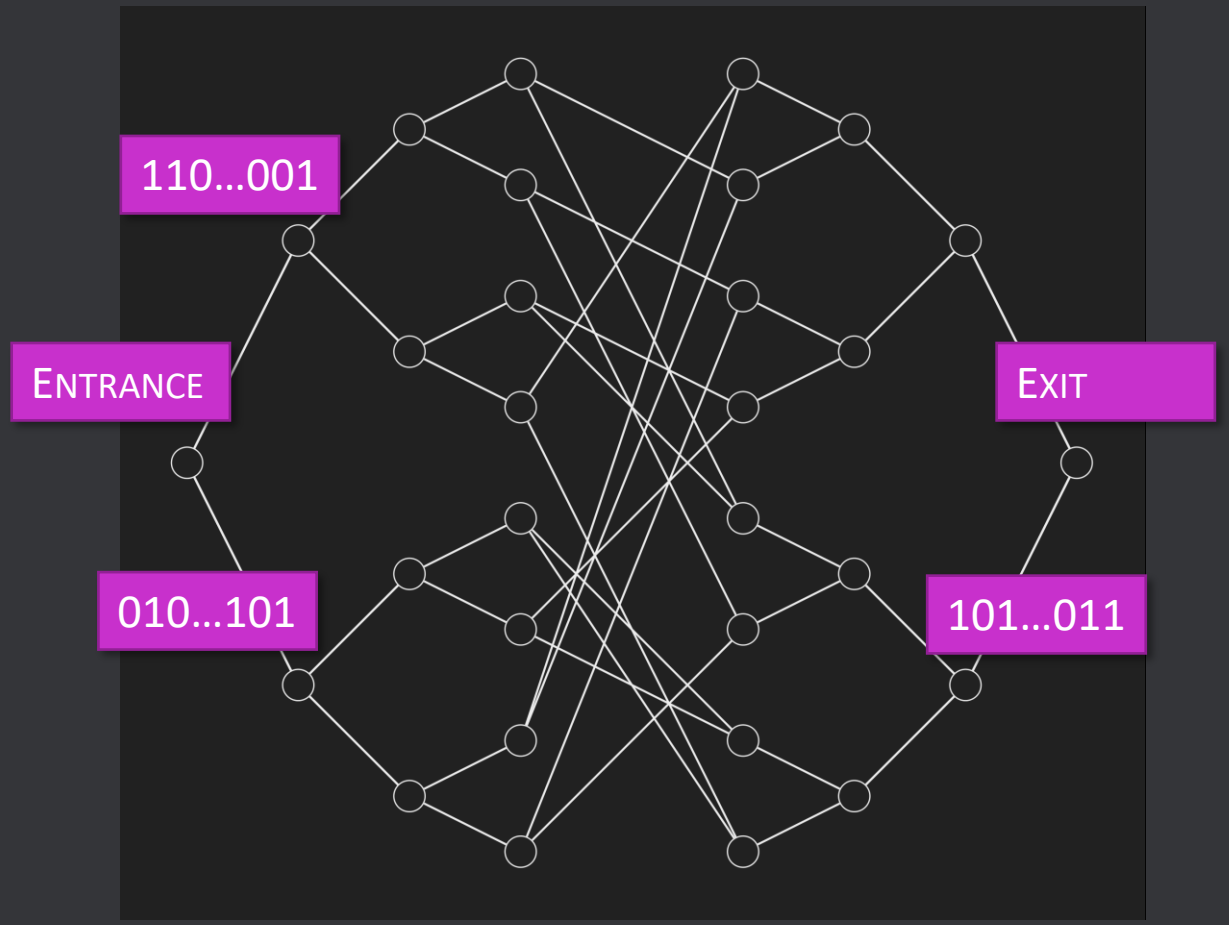
The Welded Tree Problem



- Take two trees of height n and then "weld" them together with a random cycle.
- The roots of the trees are called ENTRANCE and EXIT, respectively. Notice that all other nodes have degree-3.
- Each vertex has a distinct, random, 2n-bit label. To emphasize the randomness let's call these "random welded / blackbox trees".

Childs et al. Exponential algorithmic speedup by quantum walk. arXiv:quant-ph/0209131

The Welded Tree Problem



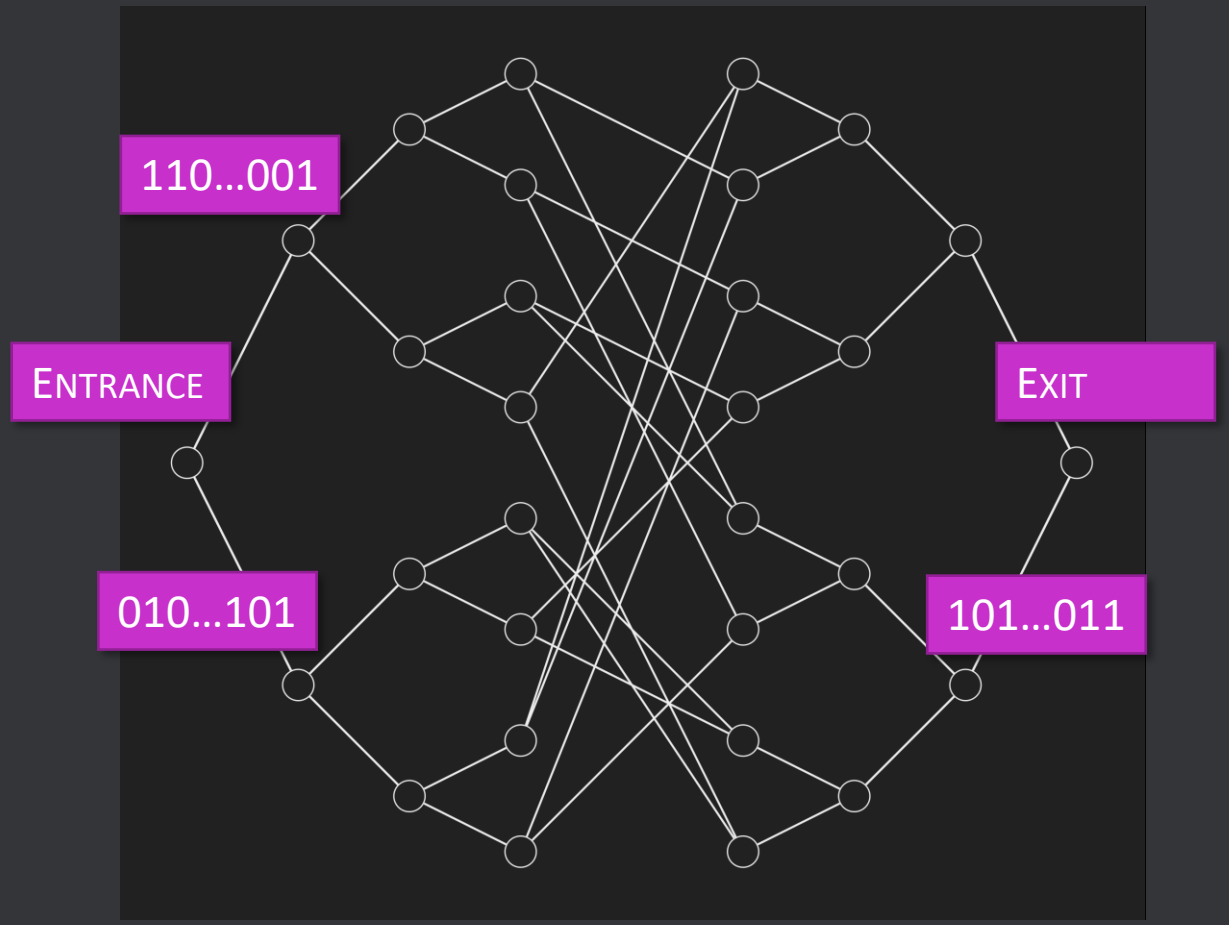
- We can only access this graph through a black-box function:

$$K_T(x, c) = \begin{cases} c \text{ neighbour of } x, & x \text{ is a valid vertex} \\ \text{INVALID}, & \text{otherwise} \end{cases}$$

- $\text{INVALID} = 111 \ldots 111$ (we don't use this for vertex labels.)
- c here is an edge color. This is just for mathematical convenience; we can pick any coloring. In our paper, following Childs et al., we pick a 9-colouring. Since the number of colors is constant, we can WLOG, assume that a query gives us all the neighbours.

Childs et al. Exponential algorithmic speedup by quantum walk. arXiv:quant-ph/0209131

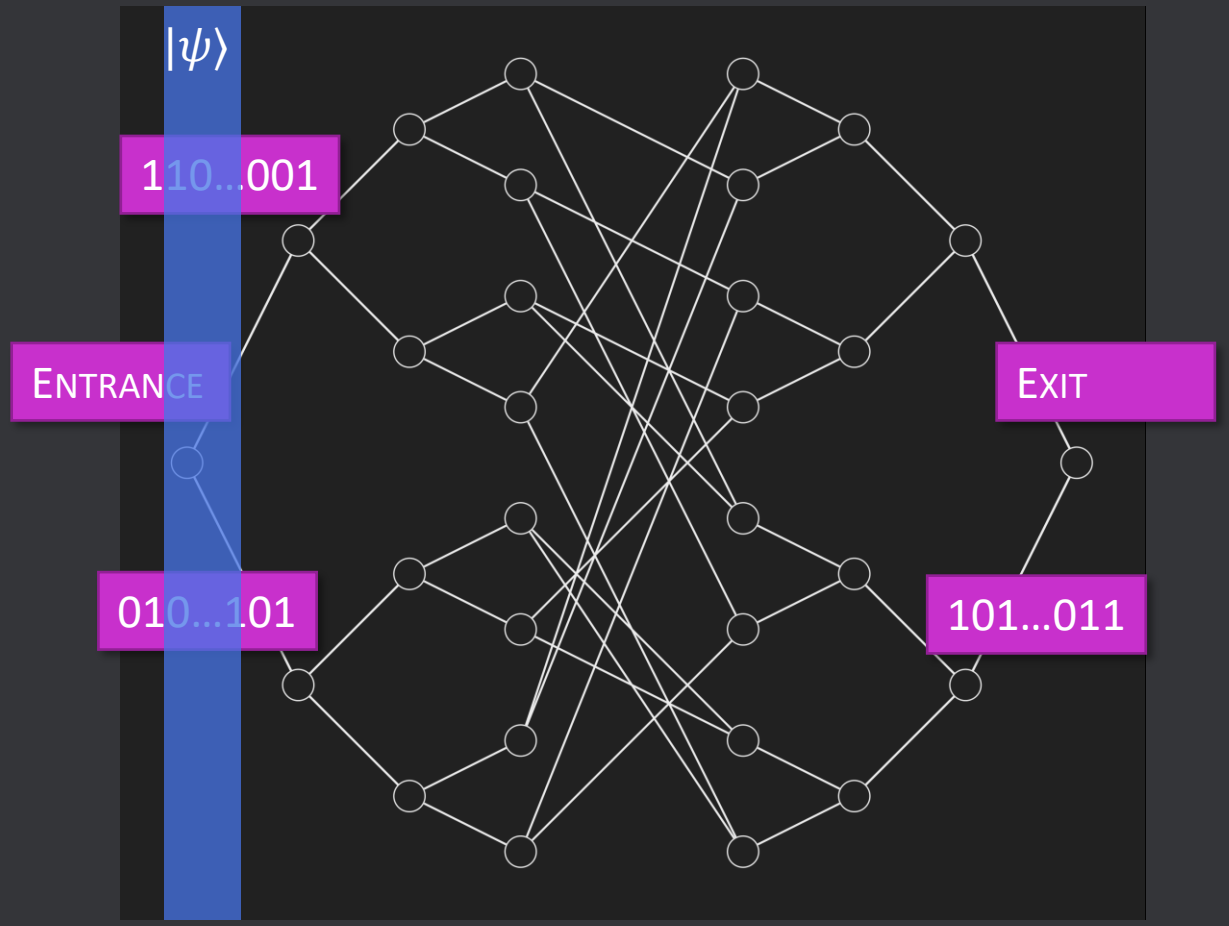
The Welded Tree Problem



- Quantumly, the black box is a unitary
  $$K_T|x\rangle|c\rangle|0^{2n}\rangle \mapsto |x\rangle|c\rangle|y\rangle$$
- In the Welded Tree Problem, you get the label of the ENTRANCE vertex and blackbox access to K_T, and the goal is to find the label of the EXIT vertex (you can tell that given label is the EXIT by the fact that the corresponding node has degree 2 and not the ENTRANCE.)

Childs et al. Exponential algorithmic speedup by quantum walk. arXiv:quant-ph/0209131

Exponential Speedup by Quantum Walks



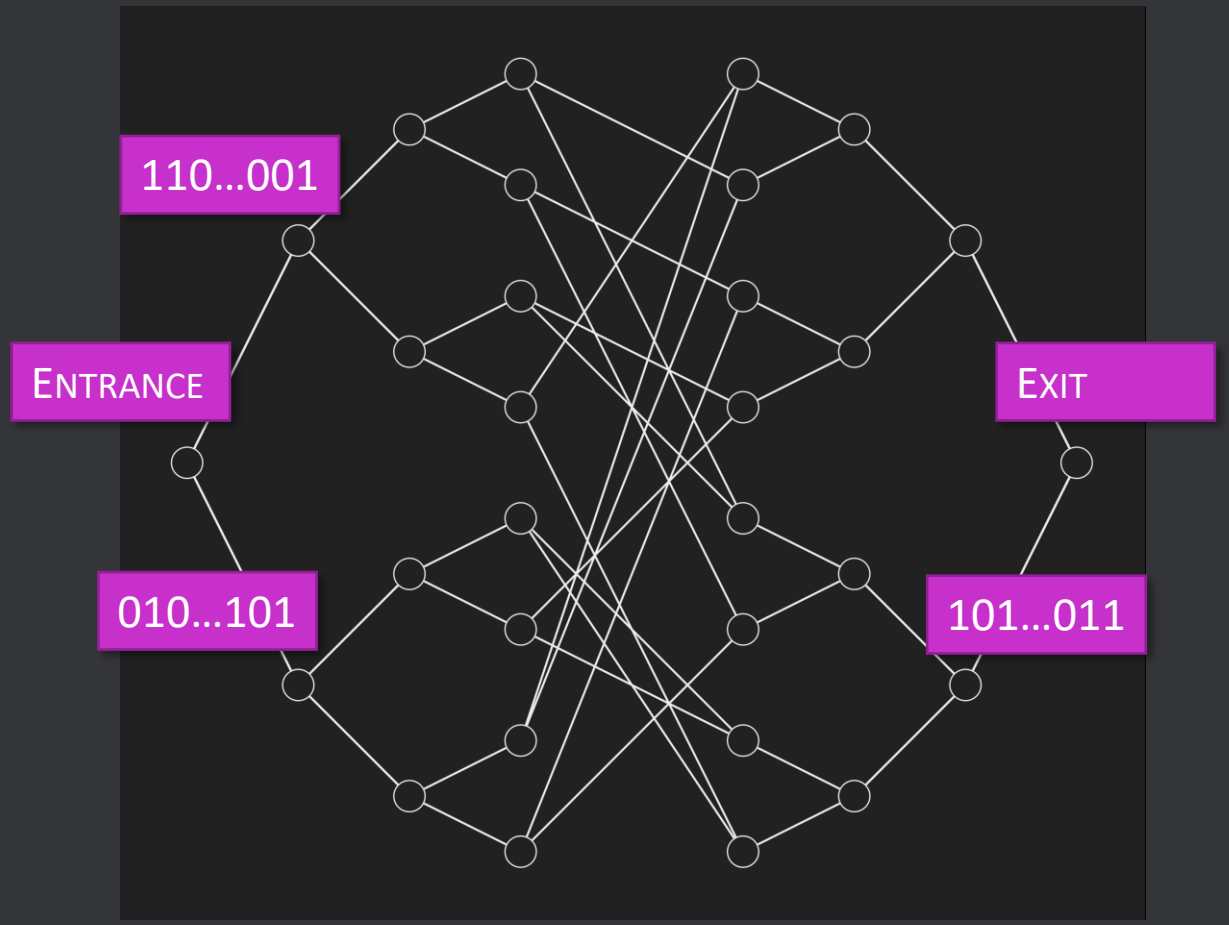
Theorem (Childs et al.). $\mathrm{WeldedTreeProblem}(T) \in \mathrm{BQP}^T$ for all oracles T.

The BQP algorithm is a carefully orchestrated Quantum Walk.

Theorem (Childs et al.). $\mathrm{WeldedTreeProblem}(T) \notin \mathrm{P}^T$ for a random blackbox tree T.

Childs et al. Exponential algorithmic speedup by quantum walk. arXiv:quant-ph/0209131

The Classical Lower Bound

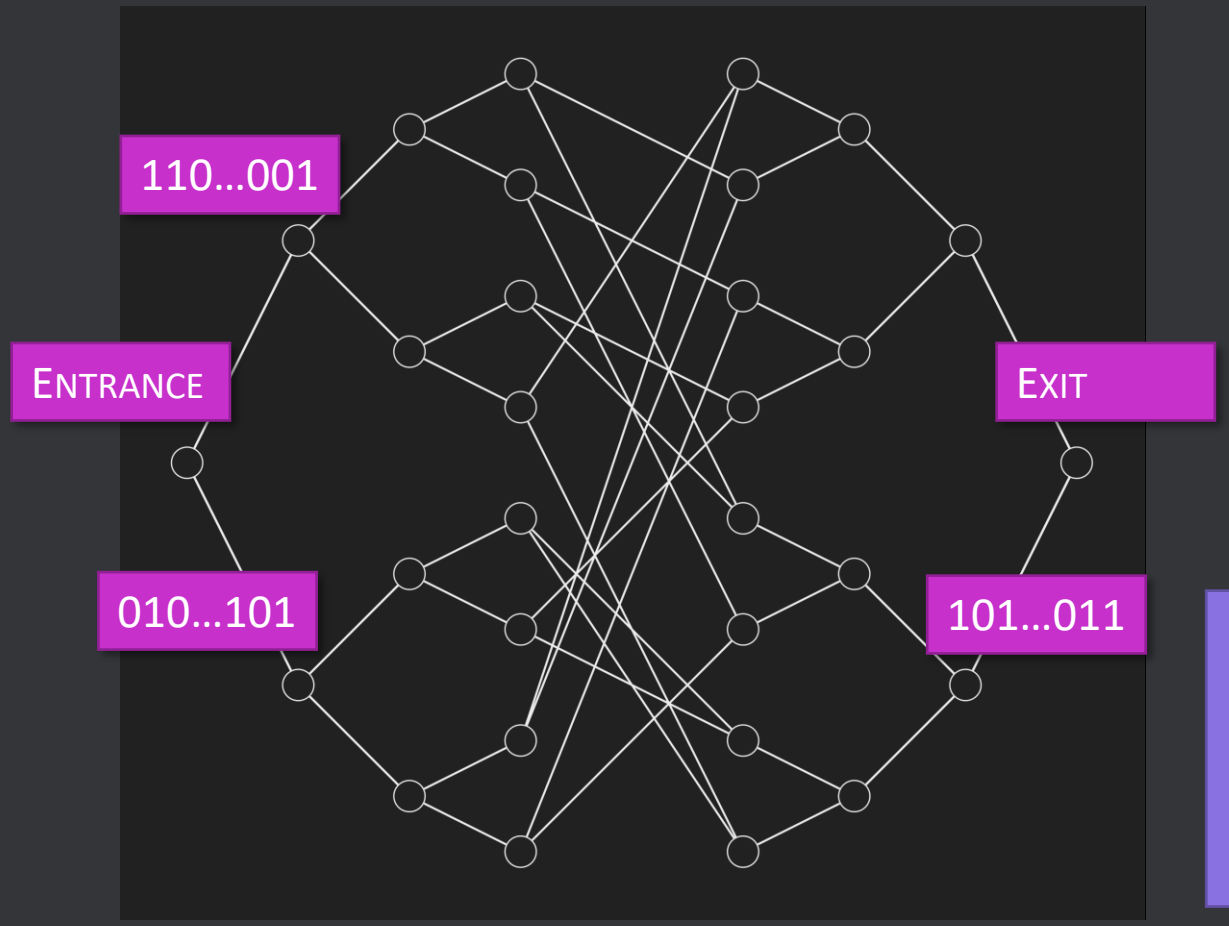


Theorem (Childs et al. with improvements by Fenner and Zhang, informal). Any classical algorithm for the $\mathrm{WeldedTreeProblem}(T)$ with a random welded tree T that makes at most $2^{n/3}$ queries outputs the correct answer (the label of the exit vertex) with probability at most $O(n2^{-n/3})$.

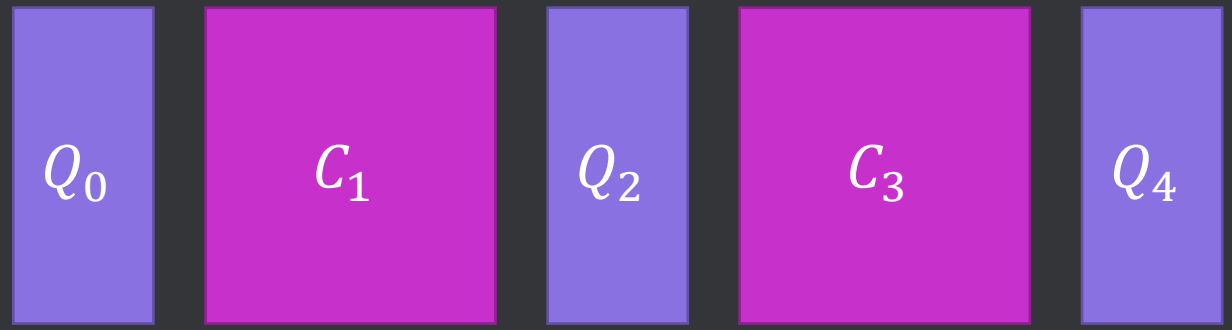
Childs et al. Exponential algorithmic speedup by quantum walk. arXiv:quant-ph/0209131
Fenner and Zhang. A note on the classical lower bound for a quantum walk algorithm. arXiv:quant-ph/0312230

Proof Sketch

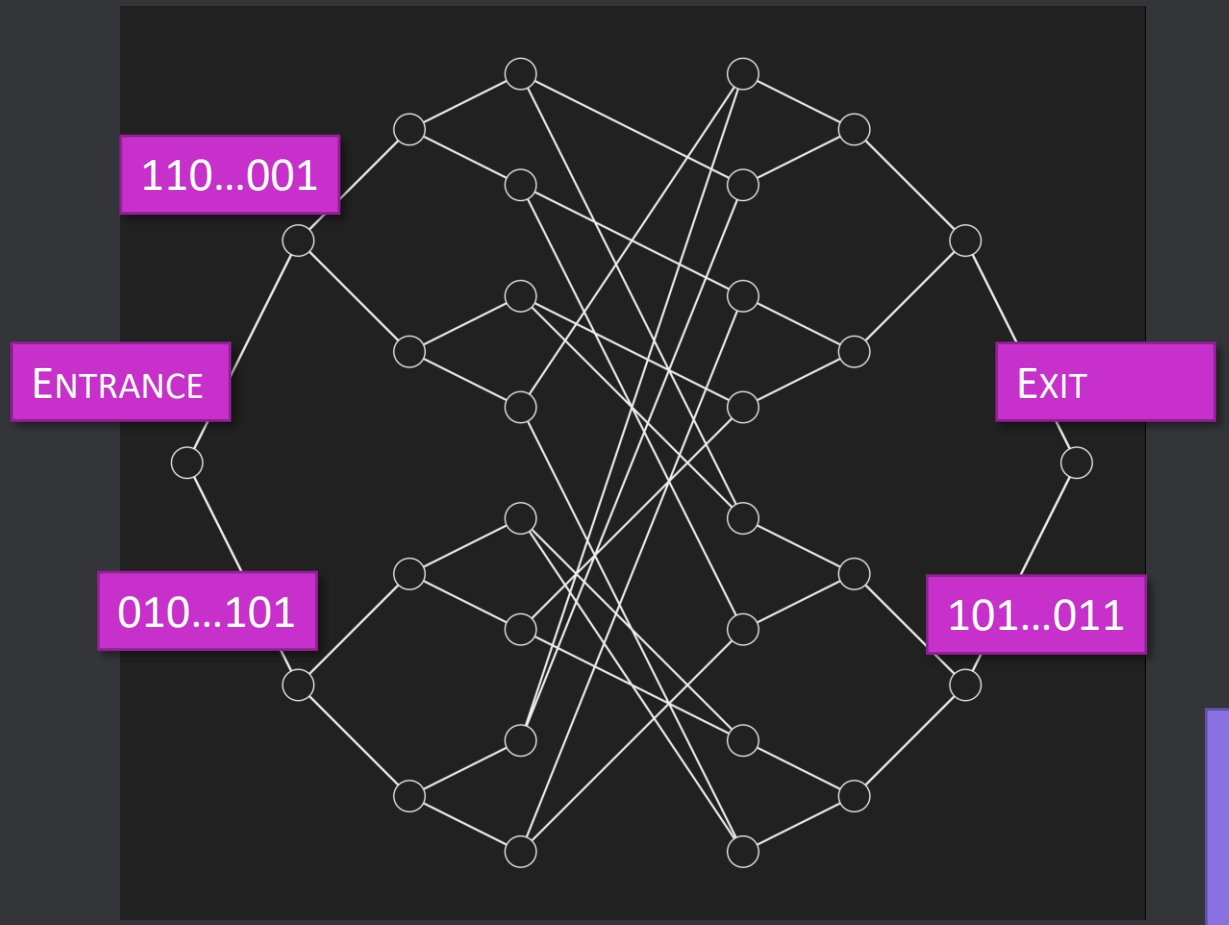
The Main Result



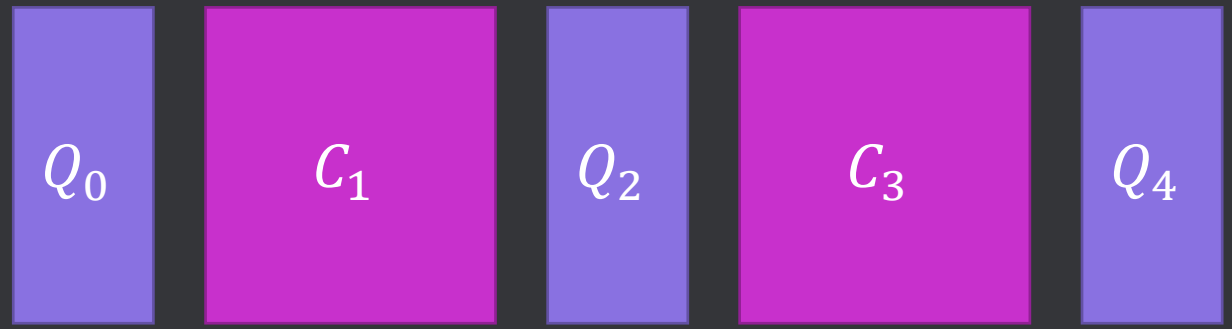
Theorem (Coudron and M, informal). For a random blackbox tree T,
$\mathrm{WeldedTreeProblem}(T) \notin \mathrm{HQC}^T$.



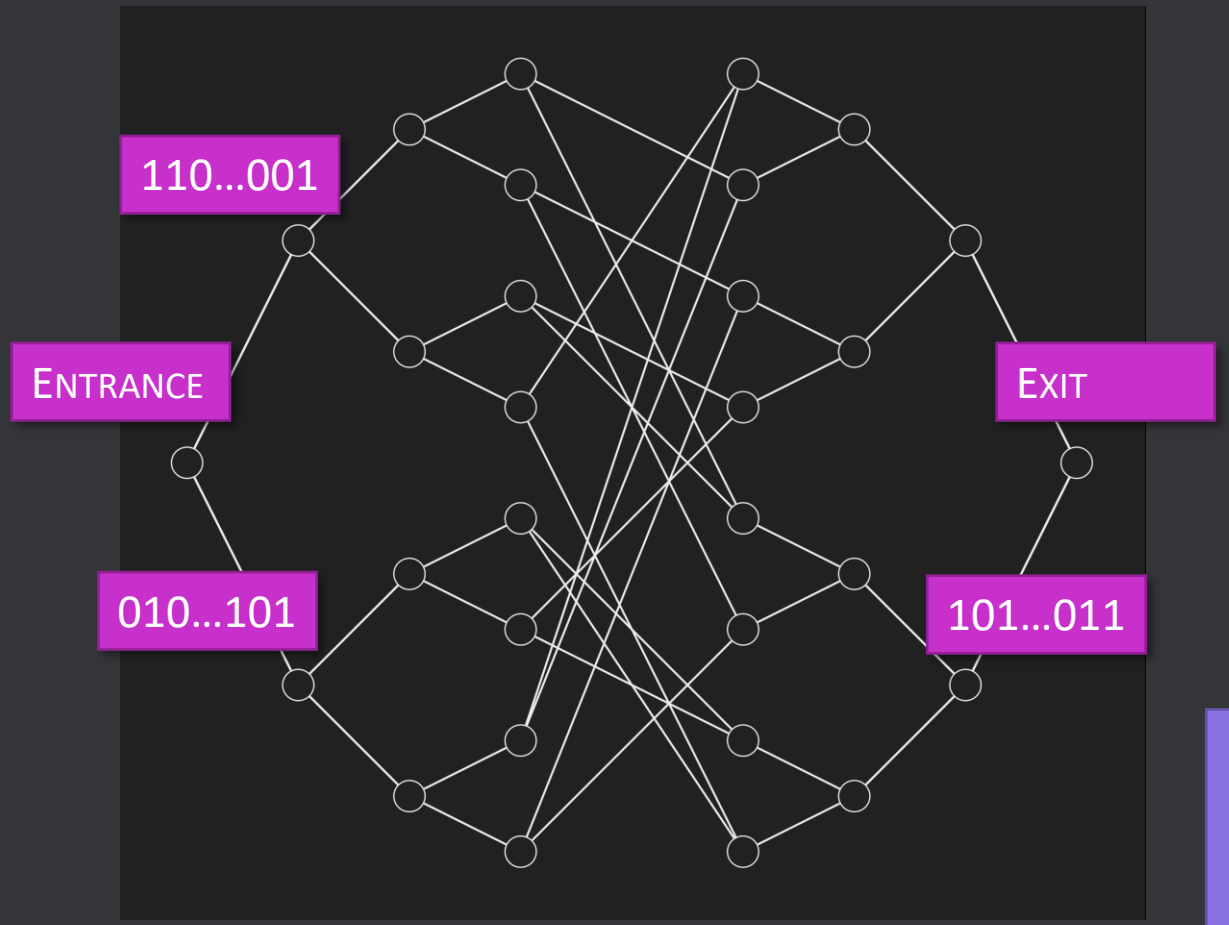
The Main Result: The Idea



- Assume towards contradiction that the Welded Tree Problem is in HQC^T, that is, there is a hybrid quantum algorithm.
- Classically simulate this hybrid quantum algorithm with subexponentially many classical queries.
- Apply the existing classical lower bound, to get a contradiction.

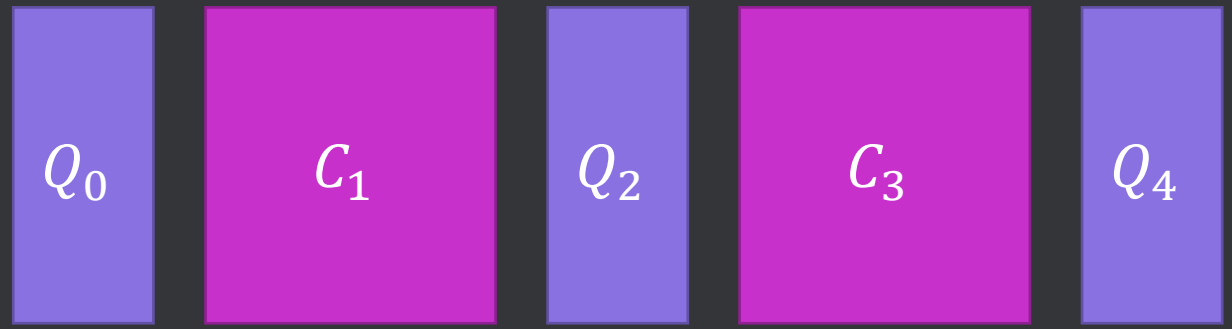


Classically Simulating Hybrid Quantum Circuits

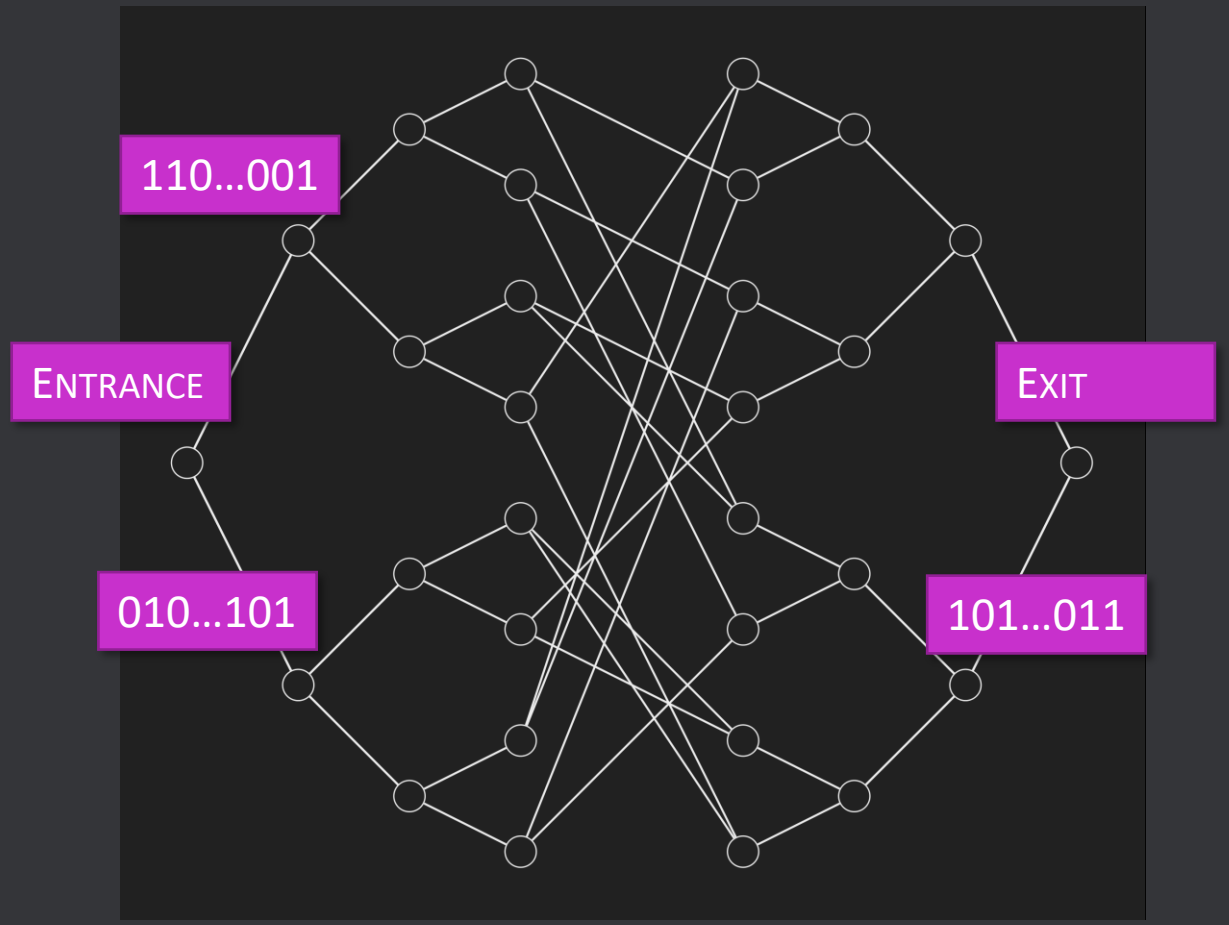


- While simulating a quantum tier, we can keep track of the (exponentially large) classical description of the quantum state and simulate all quantum gates accurately.
- But we still need a way to simulate superposition queries.

$$K_T \sum_{x,c} a_{x,c}|x\rangle|c\rangle|0^{2n}\rangle \mapsto \sum_{x,c} a_{x,c}|x\rangle|c\rangle|y\rangle$$

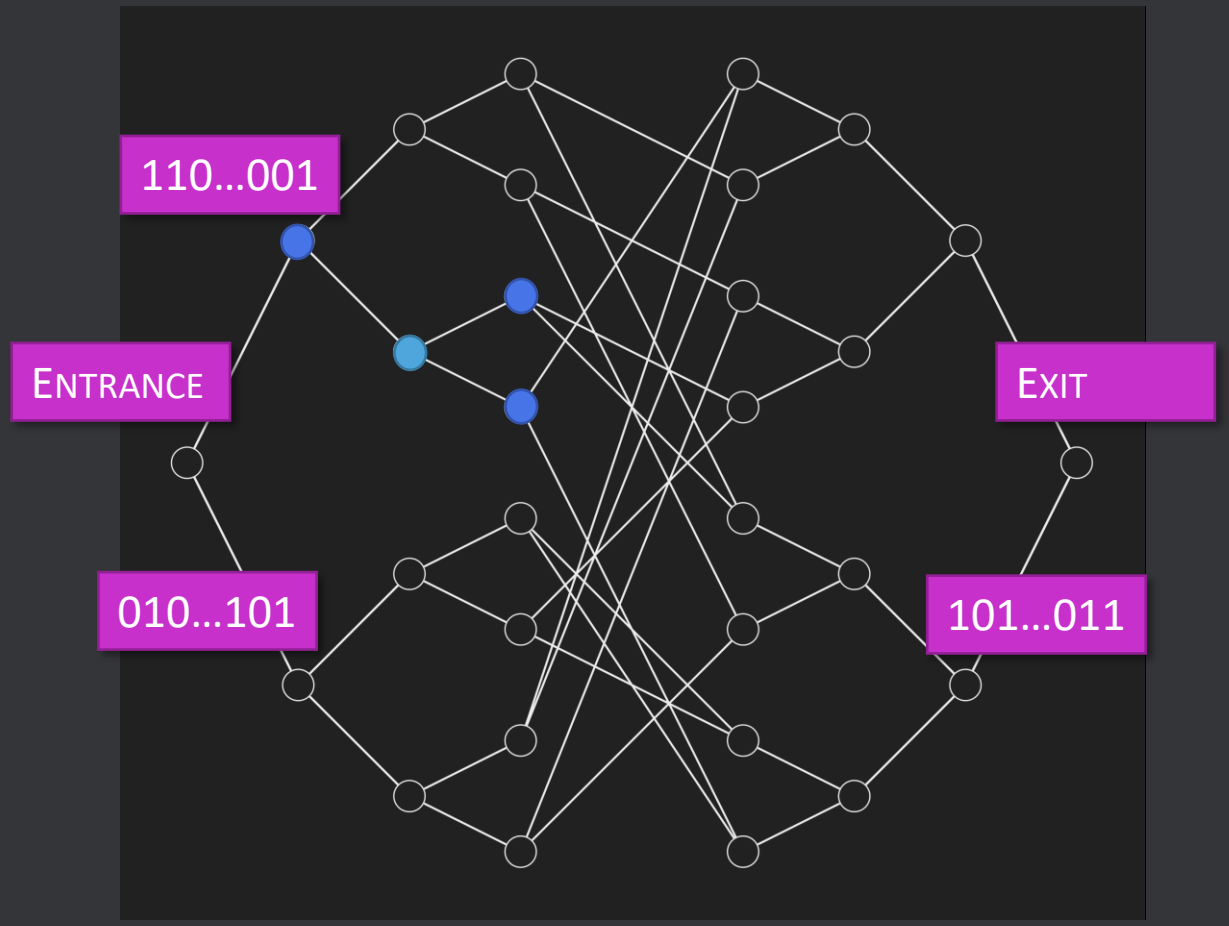


Classically Simulating Hybrid Quantum Circuits



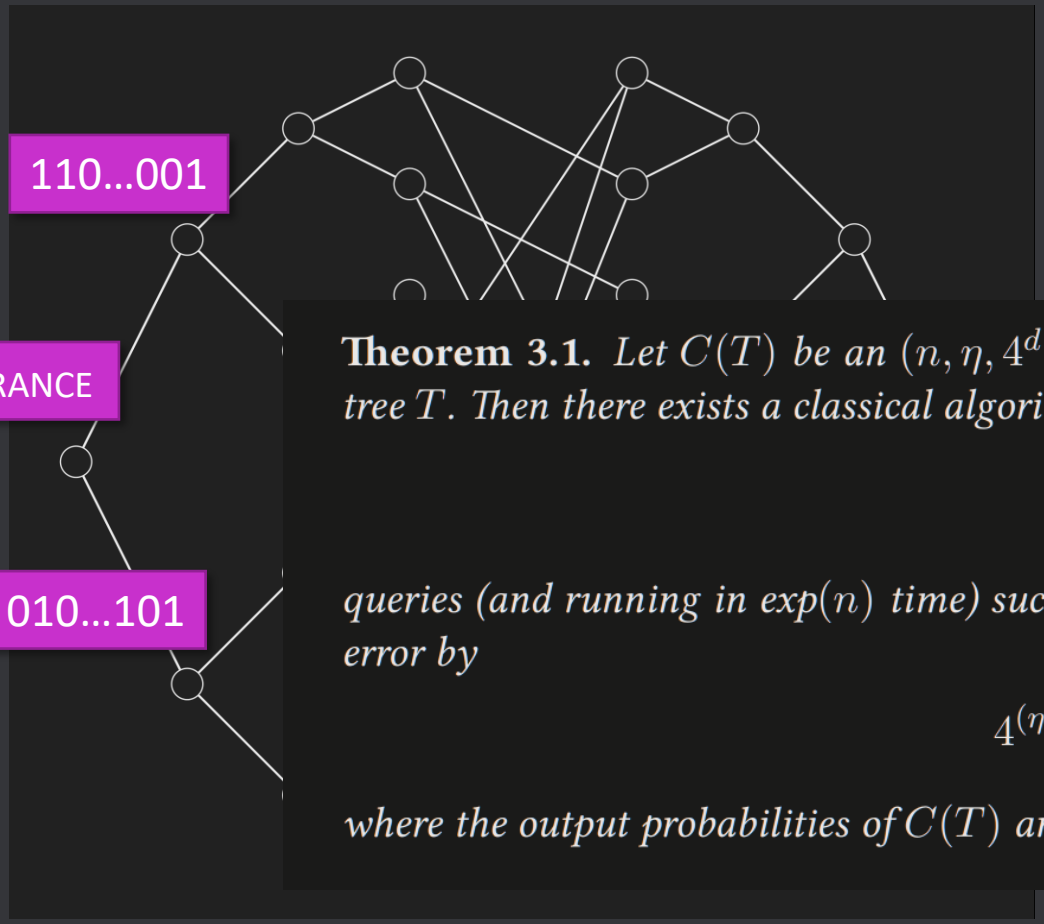
- **Key Insight 1:** the chance that the circuit can guess the label of a vertex that is not "known" (not the result of a previous query) is exponentially small.
- So, we keep track of the set of vertices (V_{known}) that were returned by the blackbox K_T and only execute queries on those vertices. For the rest, we assume that the result is INVALID.
- This simulation produces a state which is exponentially close to the true state outputted by the quantum circuit.

Algorithm Based on Key Insight 1



- Suppose at the beginning of quantum tier i, we have some V_{known}.
- Towards the worst case, suppose that every vertex in V_{known} is queried in the 1st quantum layer. So $V_{known}^{new} = 4V_{known}$ since each vertex adds at most 3 new vertices to the known set.
- Since each quantum tier has polylog(n) quantum layers, each quantum tier increases the number of known vertices by a factor of $4^{\text{polylog}(n)}$.

Algorithm Based on Key Insight 1



This approach allows our classical query algorithm to simulate a single quantum tier, or even $\sqrt{n}$ tiers.

Theorem 3.1. *Let $C(T)$ be an $(n, \eta, 4^d, d, g(n))$-hybrid-quantum circuit that queries a random n-welded tree T. Then there exists a classical algorithm $\mathcal{A}(T)$ making*

$$4^{\eta(d+1)}(g(n) \cdot d) \tag{20}$$

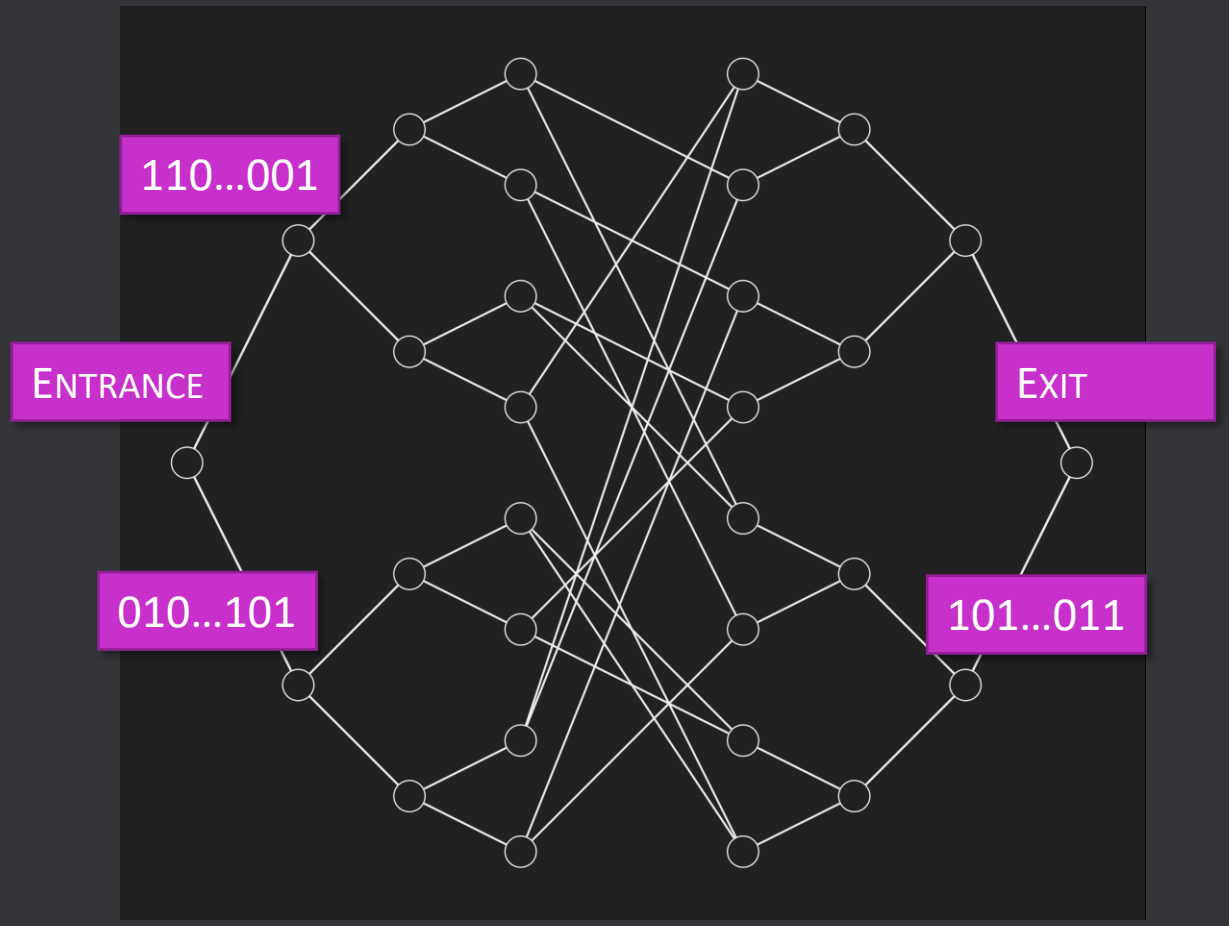
Number of queries is exponential in number of tiers η

queries (and running in $\exp(n)$ time) such that the output probabilities of $C(T)$ and $\mathcal{A}(T)$ differ in 1-norm error by

$$4^{(\eta+2)(d+2)}(g(n))^2 \cdot \frac{2^{n+2}-2}{2^{2n}}, \tag{21}$$

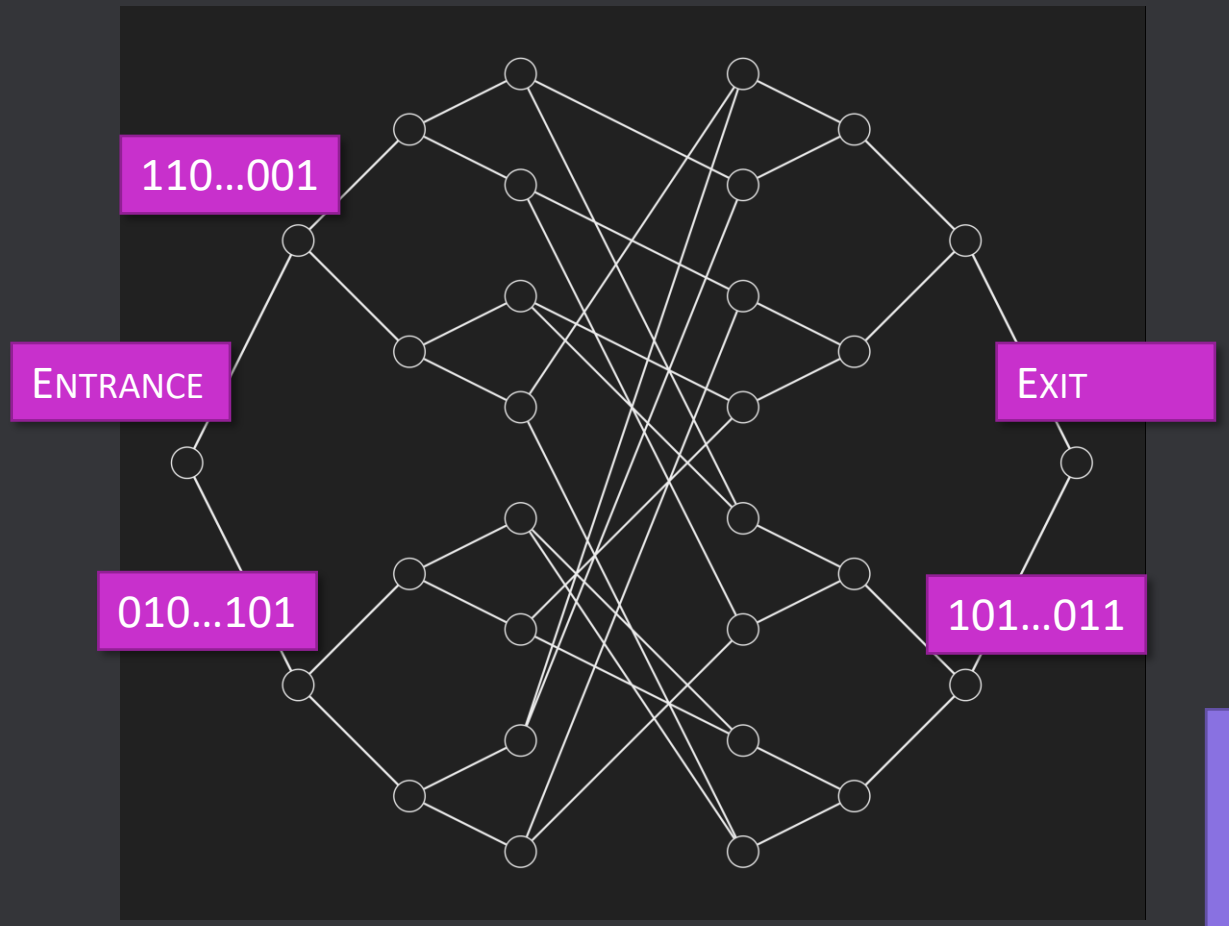
where the output probabilities of $C(T)$ and $\mathcal{A}(T)$ are defined over all possible labellings of the tree T.

Classically Simulating Hybrid Quantum Circuits

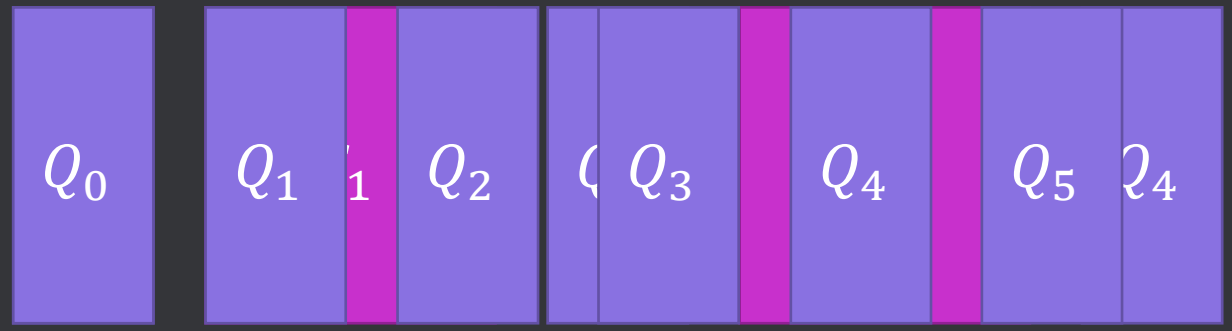


- The problem with this approach is that each new quantum tier we simulate increases the set of explored vertices by a multiplicative factor of $4^{\text{polylog}(n)}$.
- The number of explored vertices quick becomes exponential and trivalizes our bound for super-linear number of tiers.
- We need another idea to get over this.

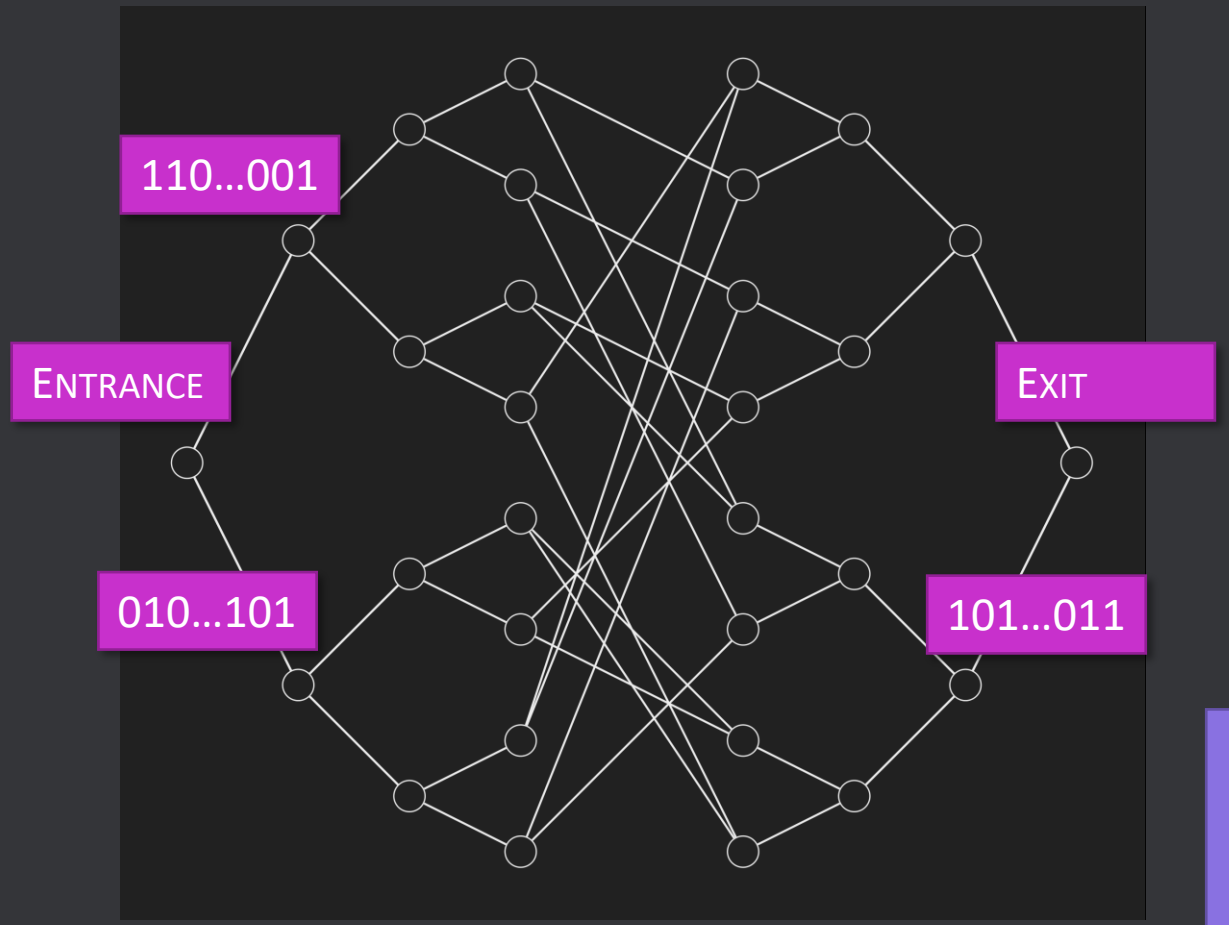
Classically Simulating Hybrid Quantum Circuits



In the previous argument, after $\sqrt{n}$ tiers the number of known vertices becomes exponential. But notice that the tiers don't have any persistent state and the communication is limited to a polynomial-length classical bitstring. You cannot fit an exponential number of 2n-bit labels in there, right?



Classically Simulating Hybrid Quantum Circuits



Key Insight: Each tier can only tell the next tier about a polynomial number of vertices. Moreover, these vertices can be inferred from the classical bitstring outputted by the tier and the set of vertices we choose to “remember”.

Q_0 Q_1 Q_2 Q_3 Q_4 Q_5

Algorithm Based on Key Insights 1 and 2

Suppose that, at tier i our simulation algorithm produces

$$x, V_{knowninit}, V_{known}^{hist} \leftarrow \mathcal{M}_r^i(C(T), T)$$

We want to simulate the $(i+1)$st tier using only the "effectively known" vertices discovered in $\mathcal{M}_r^i(C(T), T)$ rather than all of V_{known}^{hist}.

4 Our goal is to build V_{known} into a set satisfying $V_{\text{known}}^{current} \subseteq V_{\text{known}} \subseteq V_{\text{known}}^{hist}$, and;

5

$$\forall b \in \{0,1\}^{2n} \text{ such that } b \text{ does not appear in } V_{\text{known}}:$$
$$\mathbb{P}_{P \in \mathcal{T}^i_{V_{\text{known}}, x, r_{\leq i}}} [b \text{ is a valid label in P}] \leq 2^{-n/100}$$

Key Invariant

Definition 5.4. For any n, dictionary V, and bitstring $s \in \{0,1\}^{g(n)}$, let

$$\mathcal{T}^i_{V,s,r_{\leq i}} := \{\text{random } n\text{-welded black-box trees P such that}$$
$$s = \mathcal{M}^i_{r_{\leq i}}(C(P), P)[1]$$
$$\text{and P is consistent with } V\}.$$

Computing "Effectively Known" Vertices

Algorithm 5.3: Bottleneck

Input : Index i of current quantum tier, bit string x, dictionaries $V_{\text{known}}^{current} \subseteq V_{\text{known}}^{hist}$ of initially known and finally known vertices respectively

Output: Dictionary V_{known} of "effectively known" vertices, satisfying $V_{\text{known}}^{current} \subseteq V_{\text{known}} \subseteq V_{\text{known}}^{hist}$

if $\left|\mathcal{T}^{i}_{V_{known}^{current},x,r_{\leq i}}\right| < 2^{-n(g(n)+|r|)}\left|\mathcal{T}_{V_{known}^{current}}\right|$ **then**

  ABORT and guess a random label for the EXIT vertex of the entire welded tree problem on T;

Initialize $V_{\text{known}} \leftarrow V_{\text{known}}^{current}$;

Our goal is to build V_{known} into a set satisfying $V_{\text{known}}^{current} \subseteq V_{\text{known}} \subseteq V_{\text{known}}^{hist}$, and;

$$\forall b \in \{0,1\}^{2n} \text{ such that } b \text{ does not appear in } V_{\text{known}}:$$
$$\mathbb{P}_{P \in \mathcal{T}^{i}_{V_{\text{known}},x,r_{\leq i}}}[b \text{ is a valid label in P}] \leq 2^{-n/100}$$

/* No
qu
Eq
alg

whil

(

i

  ABORT and guess a random label for the EXIT vertex of the entire welded tree problem on T;

  If b' does appear in V_{known}^{hist}, then add b' and its children, and edge colors in V_{known}^{hist} to the dictionary V_{known}, and continue;

/* After the above while loop terminates we conclude the subroutine with the following clean-up step. */

Let $V_{\text{known}}^{\text{complete}} \subseteq V_{\text{known}}^{hist}$ be the minimum size subtree (rooted at ENTRANCE) of V_{known}^{hist} which contains V_{known}.;

/* Since $V_{\text{known}} \subseteq V_{\text{known}}^{hist}$ and V_{known}^{hist} is a tree rooted at ENTRANCE we can compute $V_{\text{known}}^{\text{complete}}$ without any queries to T, only look-ups to V_{known}^{hist}. We will see in the analysis that this does not adversely increase the size of V_{known}. */

$V_{\text{known}} \leftarrow V_{\text{known}}^{\text{complete}}$;

return V_{known}

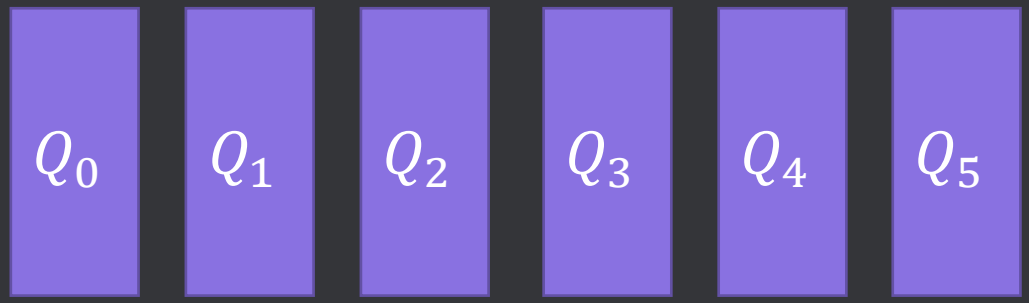
Lemma C.6. *In the classical simulation algorithm $\mathcal{M}^k$ which simulates the first k tiers of $C(T)$, the set of all encountered vertices after k tiers $V_{known}^{hist} = \mathcal{M}^k(T)[3]$ has size*

$$|V_{known}^{hist}| \leq kq(n)2^{q(n)}2n(g(n)+|r|)$$

Number of queries is polynomial in number of tiers k

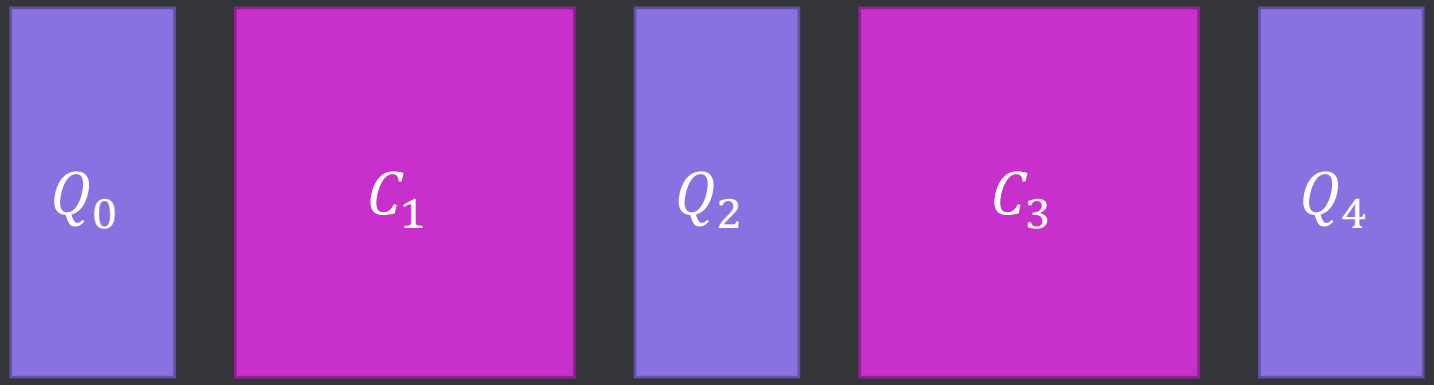
h that

and P is consistent with V}.



Conclusion

- We showed that the $\mathrm{WeldedTreeProblem}(T) \notin \mathrm{HQC}^T$. This resolves Aaronson's conjecture that there exists an oracle A such that $\mathrm{BQP}^A \not\subseteq \left(\mathrm{BPP}^{\mathrm{BQNC}}\right)^A$.
- Using similar ideas, we also resolve Jozsa's conjecture.

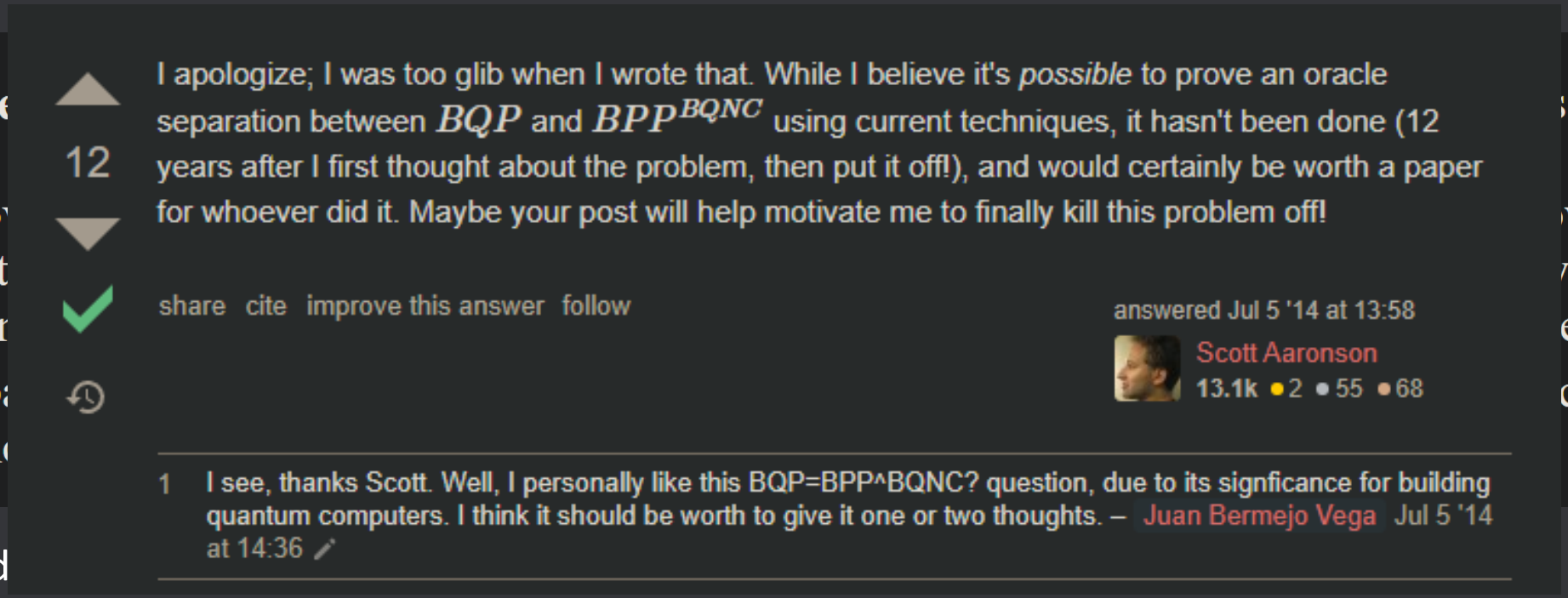


Open Problem 1: Revisiting Aaronson's Conjecture

9. The
the
pro
be t
be r
sepa
func

, can

wn to
would
e
crete



12

I apologize; I was too glib when I wrote that. While I believe it's *possible* to prove an oracle separation between BQP and BPP^{BQNC} using current techniques, it hasn't been done (12 years after I first thought about the problem, then put it off!), and would certainly be worth a paper for whoever did it. Maybe your post will help motivate me to finally kill this problem off!

share cite improve this answer follow

answered Jul 5 '14 at 13:58

Scott Aaronson

13.1k ●2 ●55 ●68

1 I see, thanks Scott. Well, I personally like this BQP=BPP^BQNC? question, due to its signficance for building quantum computers. I think it should be worth to give it one or two thoughts. – Juan Bermejo Vega Jul 5 '14 at 14:36

- "Incid
- […]
- but the question is whether there's any concrete function "instantiating" such an oracle."

Aaronson. Ten Semi-Grand Challenges for Quantum Computing Theory. https://www.scottaaronson.com/writings/qchallenge.html

Open Problem 1: Revisiting Aaronson's Conjecture

- **Open Problem 1.** Assuming post-quantum classical indistinguishability obfuscation, is it possible to produce a family of random welded tree blackboxes to instantiate our separation?

- This doesn't seem trivial, but I would start by understanding the results in Amit Sahai and Brent Waters. *How to Use Indistinguishability Obfuscation: Deniable Encryption, and More*. ia.cr/2013/454

Open Problem 2: Hybrid Quantum Attacks on Crypto

We showed that hybrid algorithms aren't great at simulating quantum walk algorithms. One can show a similar result for Grover's algorithm. Can we use these results to improve cryptanalysis?

- **Open Problem 2a.** Is there a non-trivial hybrid generic pre-image attack on AES?
- **Open Problem 2b.** Tani's (2007) algorithm, which is used for generic claw-finding attacks, is based on quantum walks which seem to have the same parallelization difficulties as Grover's algorithm. (See Section 5.6 in Jaques and Schanck (2019).) Is there a non-trivial hybrid generic claw-finding attack on SIKE?

For context and references, see https://up.c1own.com/2019/two-open-problems-hybrid-quantum-attacks-on-crypto/

Open Problem 3: Offloading Quantum Computation

Going back to the original motivation, we solved the simplest problem by showing that a generic quantum computation cannot be solved in a hybrid fashion. But there might still be many interesting problems that have hybrid algorithms.

Jaques and Gidney start looking into ways of offloading quantum computation in their recent *Offloading Quantum Computation by Superposition Masking.* arXiv:2008.04577

More generally, it would be great to have more hybrid quantum algorithms---especially in the non-asymptotic world.

The End

Open Problems

1. Can we instantiate our separation using post-quantum classical indistinguishability obfuscation?
2. Can we use this new model to do better cryptanalysis?
3. Design new hybrid algorithms and techniques for offloading quantum computations.

Thanks

- Richard Cleve, Aram Harrow, John Watrous, and Umesh Vazirani for helpful comments and discussions.
- IQC (where this work was done)

More

- Twitter: @__sanketh
- Web: snkth.com

Slides with clickable links at https://snkth.com/talks

MC was supported at the IQC by Canada's NSERC and the Canadian Institute for Advanced Research (CIFAR), and through funding provided to IQC by the Government of Canada and the Province of Ontario.