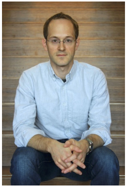
Symmetries, graph properties, and quantum speedups

Daochen Wang (Maryland)

FOCS 2020, full version: arXiv: 2006.12760

Shalev Ben-David
(Waterloo)



Andrew Childs
(Maryland)

András Gilyén
(Caltech)

William Kretschmer
(UT Austin)

Supartha Podder
(Ottawa)

Quantum speedups in query complexity

Query complexity

Let Σ be a finite alphabet and let $f : \mathcal{D} \subset \Sigma^n \to \{0,1\}$ be a known function.

- How many positions of input $x \in \mathcal{D}$ do you need to query to compute $f(x)$ with high probability in the worst case?
- Answer denoted $R(f)$ and $Q(f)$ in the classical and quantum cases respectively. Quantumly, can query x in superposition.
- We want to know when $R(f) = Q(f)^{\omega(1)}$ (large speedup) and when $R(f) = Q(f)^{O(1)}$ (small speedup).
- Interesting facts:
  1. When $\mathcal{D} = \Sigma^n$, there can only be small speedups[1].
  2. Large speedups exist! For example, in 1997, Simon exhibited an f with $R(f) = \Theta(\sqrt{n})$ and $Q(f) = \Theta(\log(n))$.

[1] Beals, Buhrman, Cleve, Mosca, and de Wolf (2001); Aaronson, Ben-David, Kothari, and Tal (2020).

Characterization of quantum speedups for symmetric functions: “must be small for hypergraph-based symmetries, else can be large”

Symmetric functions

Definition

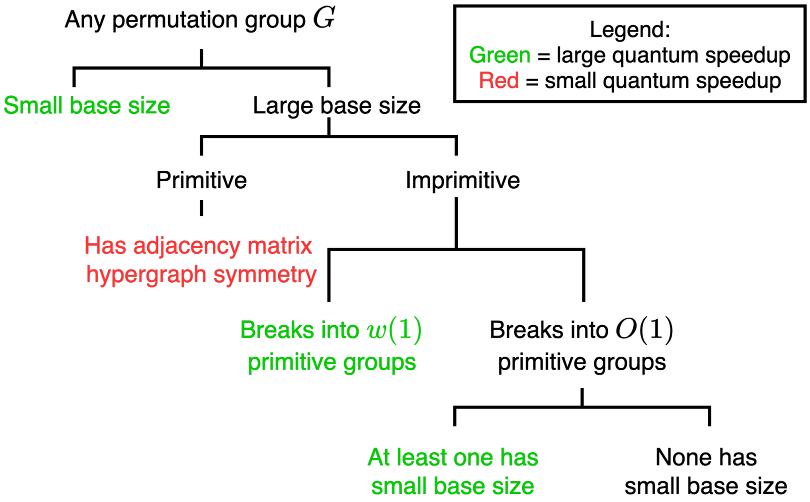
Let $f : \mathcal{D} \subset \Sigma^n \to \{0,1\}$ be a function. f is symmetric under a permutation group G on $\{1,\ldots,n\}$ if, for all $\pi \in G$, we have:

1. $x = (x_1,\ldots,x_n) \in \mathcal{D} \implies x \circ \pi := (x_{\pi(1)},\ldots,x_{\pi(n)}) \in \mathcal{D}$.
2. $f(x) = f(x \circ \pi)$ for all $x \in \mathcal{D}$.

Near-complete characterization theorem

Prior art[2]: small quantum speedup for f symmetric under $G = S_n$.

Our theorem:

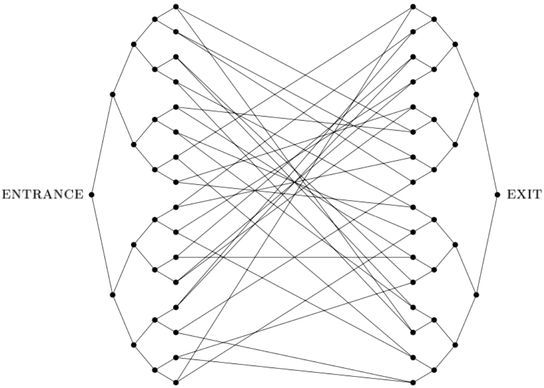


[2] Aaronson and Ambainis (2009); Chailloux (2018).

There exists an exponential quantum speedup for graph property testing in the adjacency list model

The glued trees problem

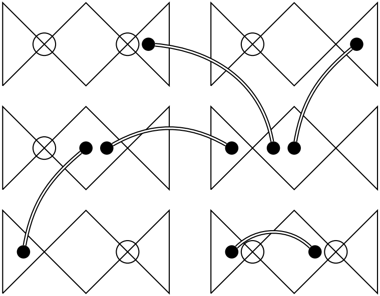
Given access to the adjacency list of a glued trees graph and the label of ENTRANCE, a quantum algorithm can find the label of EXIT exponentially faster than any classical algorithm[3].



[3] Childs, Cleve, Deotto, Farhi, Gutmann, and Spielman (2003).

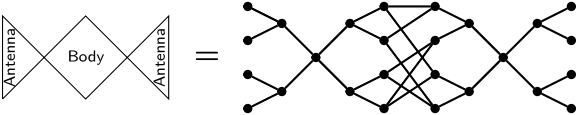
Use glued trees to construct a property testing problem with exponential quantum speedup

The graph property:



1. Can *classically* test the *entire* glued-trees if we mark the leaves of the two trees that are glued.
2. Mark the leaves in a way that can only be read efficiently by a quantum computer but not a classical computer – use further copies of the glued-trees problem.

where



In particular: quantum speedups of computing graph properties depend significantly on the input model!

Adjacency list: an exponential quantum speedup exists even for graph property testing.

Adjacency matrix: there can be at most polynomial quantum speedup, $R(f) = O(Q(f)^6)$.

These results resolve an open question of Ambainis, Childs, and Liu (2010) and Montanaro and de Wolf (2013).

Thank you for your attention!

Open problems

Thank you for your attention! Here are some of our open problems:

1. We showed $R(f) = O(Q(f)^{3p})$ for p-uniform hypergraph properties f in the adjacency matrix model as part of our characterization theorem. How tight is this?
2. Can we complete our characterization theorem?
3. Is there a *useful* graph property testing problem in the adjacency list model with super-polynomial quantum speedup?