

Theoretical Foundations of Quantum Advantage for Quantum Algorithms

François Le Gall

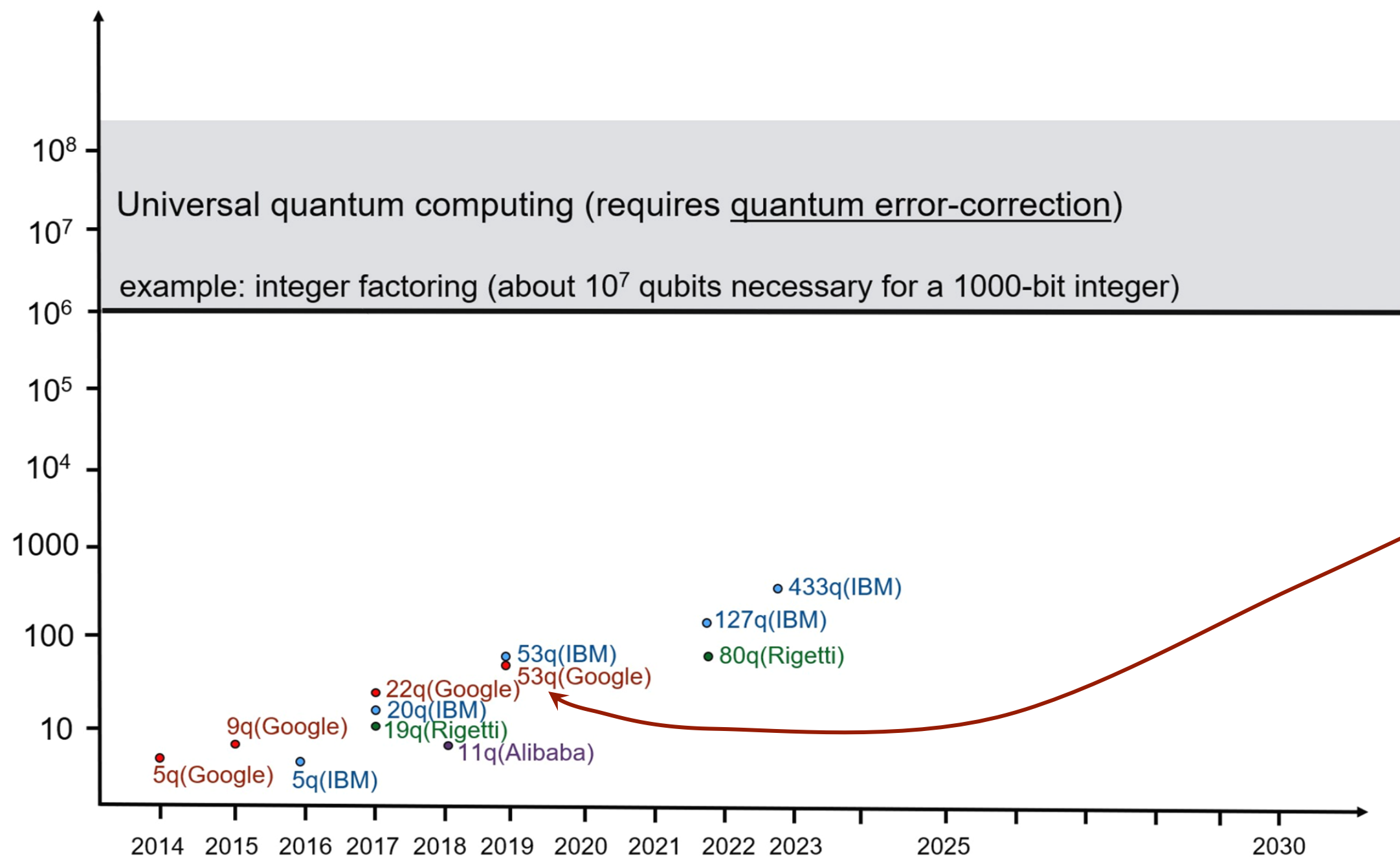
Graduate School of Mathematics, Nagoya University

Foundations and Developments of Quantum Information Theory

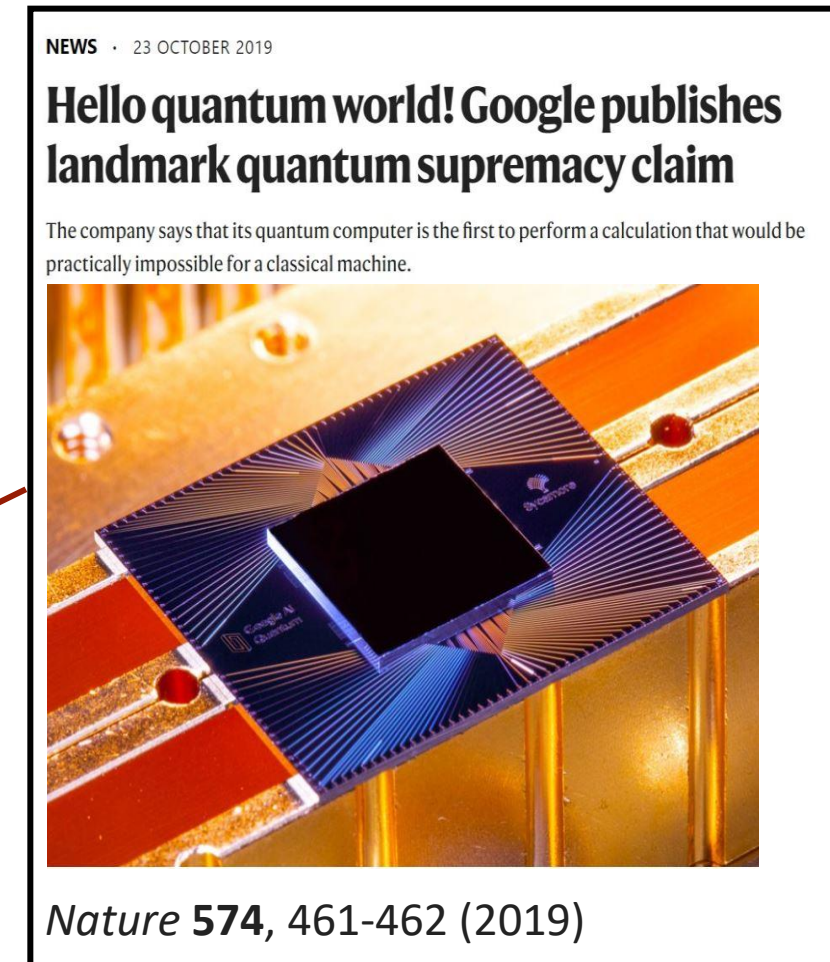
20 September 2023

Quantum “Supremacy”

number of quantum bits (qubits)



We know that such small-scale quantum computers can already perform computation faster than classical computers (e.g., random quantum circuits, boson sampling, ...)



Task: compute the output of a random quantum circuit acting on 53 qubits

Can quantum computers perform **useful** computations faster for classical computers?

not really “useful” computational tasks...

Quantum Advantage for useful tasks

Provable advantage

- ✓ Integer Factoring (Shor algorithm)
- ✓ Quantum search (Grover algorithm)
- ✓ Quantum distributed computing
- ✓ Problems with quantum inputs
(most problems in quantum information theory)
- ...

merit: theoretical guarantees of the quantum advantage

demerit: generally requires quantum error-correction
and thus large-scale quantum computers

Heuristic

- ✓ Quantum annealing
- ✓ Adiabatic algorithms
- ✓ QAOA
- ✓ VQA
- ✓ Quantum machine learning
- ...

merit: can be implemented on NISQ devices

demerit: generally few theoretical guarantees
(performance needs to be analyzed on real data)

This talk: a survey of other examples of quantum algorithms with provable advantage (including several of my favorite examples)

Outline of the Talk

Quantum algorithms with polynomial advantage

- Matrix multiplication
- Quantum string algorithms
- Quantum optimization

Quantum algorithms with potential exponential advantage

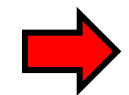
- Systems of linear equations (HHL algorithm)
- Quantum machine learning and dequantization

Quantum algorithms with exponential advantage

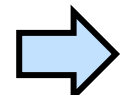
- Space-efficient quantum algorithms
- Quantum algorithms for quantum chemistry

Outline of the Talk

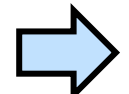
Quantum algorithms with polynomial advantage



Matrix multiplication

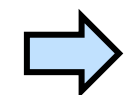


Quantum string algorithms

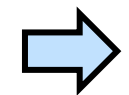


Quantum optimization

Quantum algorithms with potential exponential advantage

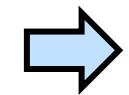


Systems of linear equations (HHL algorithm)

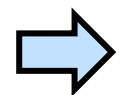


Quantum machine learning and dequantization

Quantum algorithms with exponential advantage



Space-efficient quantum algorithms



Quantum algorithms for quantum chemistry

Matrix Multiplication

Compute the product of two $n \times n$ matrices A and B

$$\begin{matrix} n \\ \updownarrow \end{matrix} \left[\begin{matrix} a_{ij} \end{matrix} \right] \begin{matrix} \xleftarrow{n} \end{matrix} \times \begin{matrix} \xleftarrow{n} \end{matrix} \left[\begin{matrix} b_{ij} \end{matrix} \right] \begin{matrix} \updownarrow n \end{matrix} = \left[\begin{matrix} c_{ij} \end{matrix} \right] \begin{matrix} \updownarrow n \end{matrix} \begin{matrix} \xleftarrow{n} \end{matrix}$$

One of the most fundamental computational tasks in science and engineering

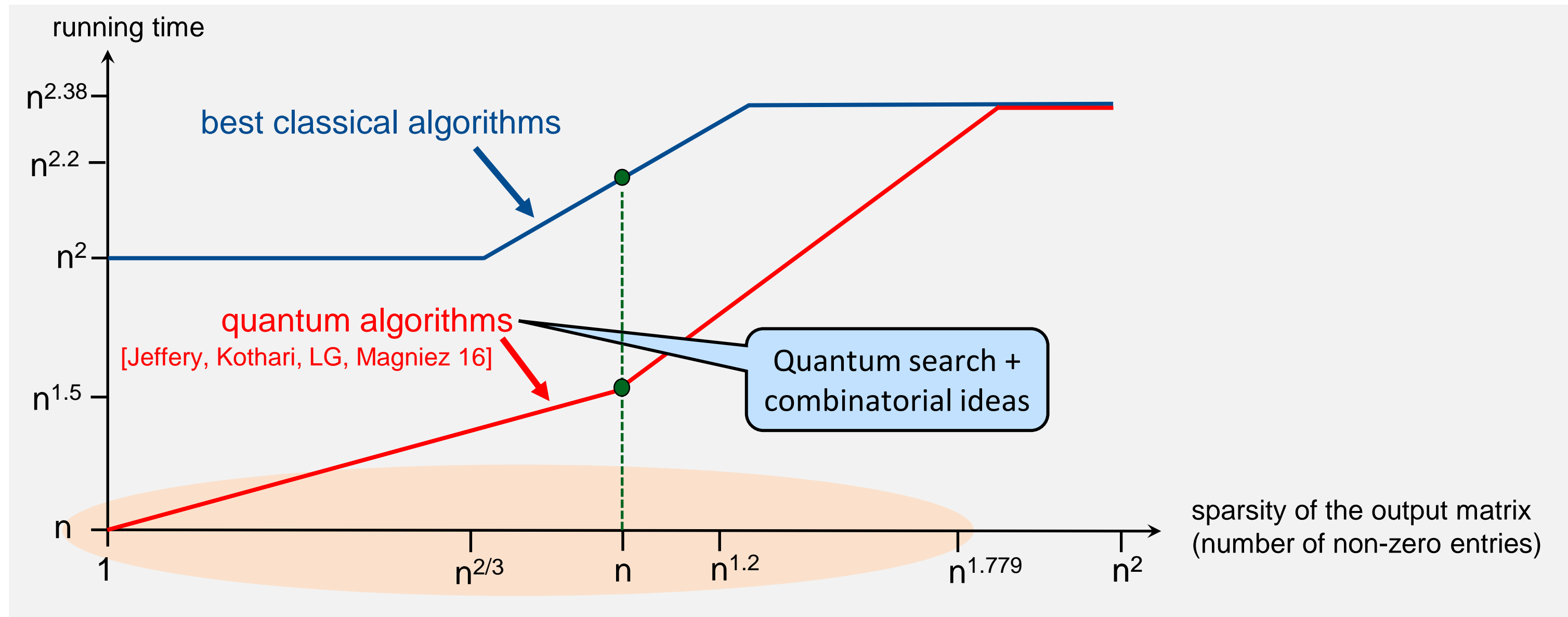
Trivial classical algorithm: $O(n^3)$ time

$$c_{ij} = \sum_{k=1}^n a_{ik} b_{kj} \quad \text{for all } 1 \leq i \leq n \text{ and } 1 \leq j \leq n$$

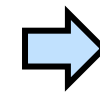
Best known classical algorithm: $O(n^{2.38})$ time [Coppersmith and Winograd 87]

Quantum Algorithms for Matrix Multiplication

[LG 12] [Jeffery, Kothari, LG, Magniez 16] [Jeffery, LG 16]



Quantum advantage for large sparse matrices!

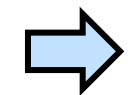


Quantum advantage for many graph problems

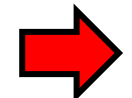
Issue: **in practice**, is an $O(n^{1.5})$ -time quantum algorithm faster than an $O(n^{2.2})$ -time classical algorithm?
(this depends on the architecture and the constant hidden in the big- O notation)

Outline of the Talk

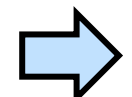
Quantum algorithms with polynomial advantage



Matrix multiplication

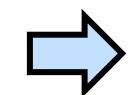


Quantum string algorithms

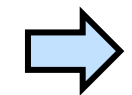


Quantum optimization

Quantum algorithms with potential exponential advantage

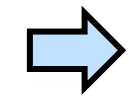


Systems of linear equations (HHL algorithm)



Quantum machine learning and dequantization

Quantum algorithms with exponential advantage



Space-efficient quantum algorithms



Quantum algorithms for quantum chemistry

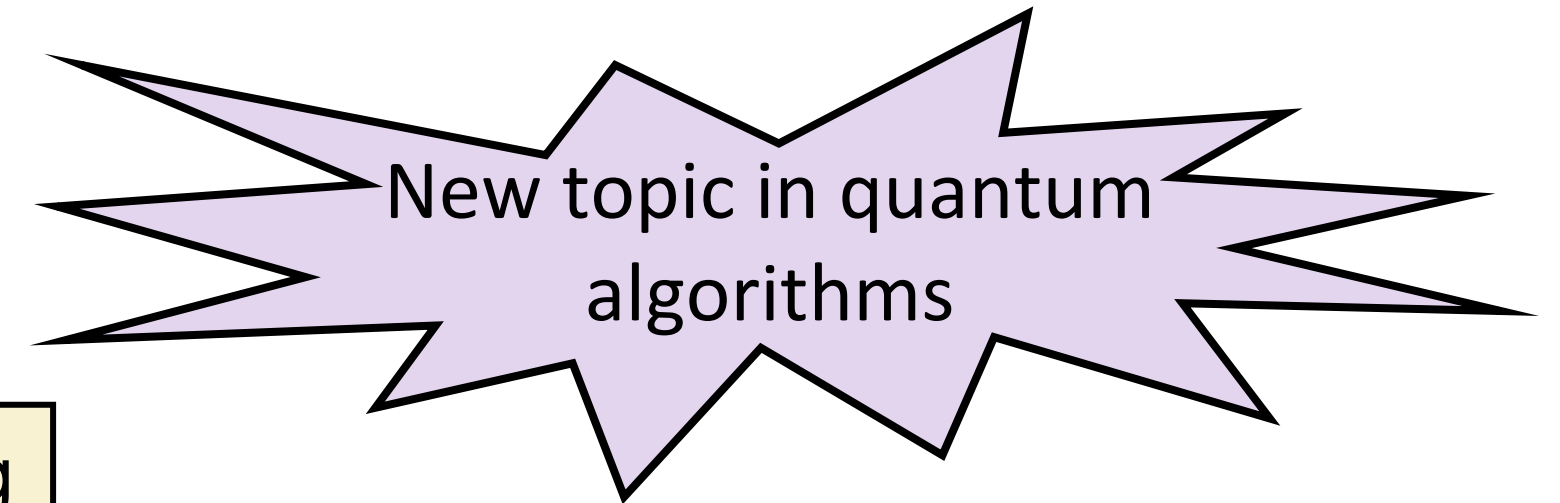
Quantum String Algorithms

Given two strings X and Y of length n , compute their similarity

example: file comparison ← length: several MB
DNA comparison ← length: 3 billion

How to define the similarity?

one standard definition length of the longest common substring



Best known classical algorithm: $O(n)$ time (optimal)

Quantum algorithms: $O(n^{5/6})$ time [LG and Seddighin 21] $O(n^{2/3})$ time [Akmal and Jin 22]

Grover search, amplitude amplification, quantum walks

Issue: **in practice**, is an $O(n^{2/3})$ -time quantum algorithm faster than an $O(n)$ -time classical algorithm?
(also depends on the implementation of Quantum Random Access Memory)

Outline of the Talk

Quantum algorithms with polynomial advantage

- ➡ Matrix multiplication
- ➡ Quantum string algorithms
- ➡ Quantum optimization

Quantum algorithms with potential exponential advantage

- ➡ Systems of linear equations (HHL algorithm)
- ➡ Quantum machine learning and dequantization

Quantum algorithms with exponential advantage

- ➡ Space-efficient quantum algorithms
- ➡ Quantum algorithms for quantum chemistry

Quantum Optimization

Provable advantage

- ✓ Quantum search (Grover algorithm)
- ✓ Quantum walks
- ✓ Backtracking
- ...

Heuristic

- ✓ Quantum annealing
- ✓ Adiabatic algorithms
- ✓ QAOA
- ✓ VQA
- ✓ Quantum machine learning
- ...

What about convex optimization?

convex optimization, especially linear programs (LP) and semidefinite programs (SDPs), has a wide range of applications, rigorous guarantees, and can be solved efficiently by classical solvers

Convex Optimization

n : number of variables
 ϵ : precision of the solution

SDP: Semidefinite Programs

Best classical algorithm for SDPs

$$O(n^{2.5} (\log(1/\epsilon))^5) \text{ [Jiang et al. 20]}$$


Quantum algorithms for SDPs

$$O(n (1/\epsilon)^{18}) \text{ [Brandao and Svore 16]}$$

$$O(n (1/\epsilon)^8) \text{ [van Aepferdoon et al. 17]}$$

$$O(\sqrt{n} (1/\epsilon)^{12}) \text{ [Brandao et al. 18]}$$

$$O(\sqrt{n} (1/\epsilon)^5) \text{ [van Aepferdoon and Gilyen 18]}$$



Potentially wide
impact

Significant improvement if we
only need low precision

For huge systems (millions of variables) this can be the only way to get a rough approximation of the solution in reasonable time

Outline of the Talk

Quantum algorithms with polynomial advantage

→ Matrix multiplication

→ Quantum string algorithms

→ Quantum optimization

Potential very wide impact, but need better understanding of quantum architectures to estimate the running time in practice

Quantum algorithms with potential exponential advantage

→ Systems of linear equations (HHL algorithm)

→ Quantum machine learning and dequantization

Quantum algorithms with exponential advantage

→ Space-efficient quantum algorithms

→ Quantum algorithms for quantum chemistry

Outline of the Talk

Quantum algorithms with polynomial advantage

- Matrix multiplication
- Quantum string algorithms
- Quantum optimization

Quantum algorithms with potential exponential advantage

- Systems of linear equations (HHL algorithm)
- Quantum machine learning and dequantization

Quantum algorithms with exponential advantage

- Space-efficient quantum algorithms
- Quantum algorithms for quantum chemistry

HHL Algorithm for System of Equations [Harrow, Hassidim, Lloyd 09]

Input :

- ✓ A sparse and well-conditioned $n \times n$ matrix A
- ✓ A unit-norm vector $b \in \mathbb{C}^n$ given as a quantum state $|b\rangle$

write $x = A^{-1}b$ and $\bar{x} = \frac{A^{-1}b}{\|A^{-1}b\|}$

solution of: $Ax = b$

sparse: at most $O(\log n)$ non-zero entries per row and column

well-conditioned: eigenvalues of AA^\dagger in $[-1, -\text{poly}(1/\log n)] \cup [\text{poly}(1/\log n), 1]$

Output:

An approximation of the quantum state $|\bar{x}\rangle$

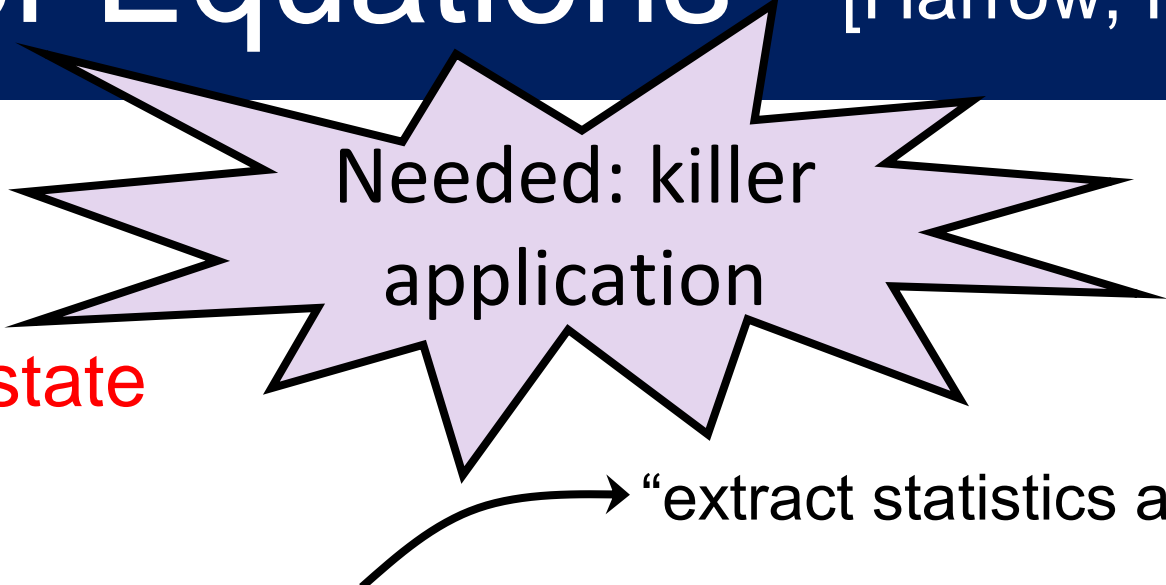
Theorem ([Harrow, Hassidim, Lloyd 09])

There is a quantum algorithm that computes a good approximation of $|\bar{x}\rangle$ in time polynomial in $\log(n)$

exponentially faster

Classically, the best known algorithm solving sparse and well-conditioned systems of linear equations uses $O(n)$ time

HHL Algorithm for System of Equations [Harrow, Hassidim, Lloyd 09]



Needed: killer application

- ✓ Main issue: the solution is output as a **quantum state**
- ✓ Possible applications of the HHL Algorithm: estimate $\langle \bar{x} | M | \bar{x} \rangle$ for some operator M

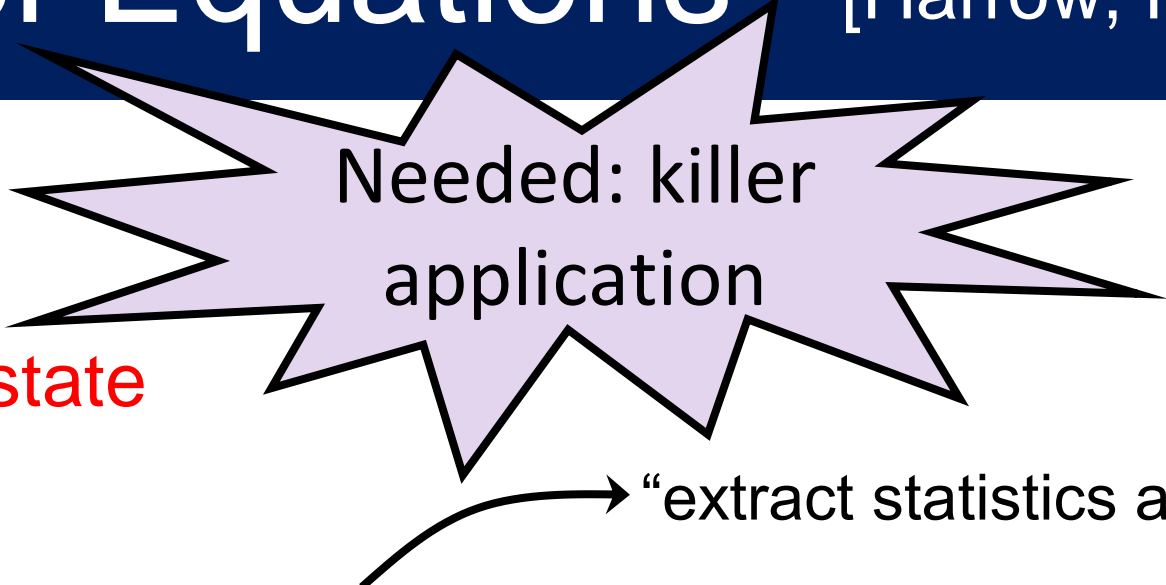
Theorem [Harrow, Hassidim, Lloyd 09]:

Estimating $\langle \bar{x} | M | \bar{x} \rangle$ is BQP-hard.
(i.e., as hard as simulating an arbitrary quantum circuit)

Applications to quantum machine learning ?

very promising but hard to analyze the performance

HHL Algorithm for System of Equations [Harrow, Hassidim, Lloyd 09]



Needed: killer application

- ✓ Main issue: the solution is output as a **quantum state**
- ✓ Possible applications of the HHL Algorithm: estimate $\langle \bar{x} | M | \bar{x} \rangle$ for some operator M

Theorem [Harrow, Hassidim, Lloyd 09]:

Estimating $\langle \bar{x} | M | \bar{x} \rangle$ is BQP-hard.
(i.e., as hard as simulating an arbitrary quantum circuit)

Applications to quantum machine learning ?

very promising but hard to analyze the performance

- performance on (large) real-data: need a large quantum computer
- theoretical investigations: there exist a few quantum machine learning algorithms with rigorous analysis ... but most of them have been “dequantized”
[Tang 19] [Chia, Gilyen, Li, Lin, Tang, and Wang 2020]

Outline of the Talk

Quantum algorithms with polynomial advantage

- Matrix multiplication
- Quantum string algorithms
- Quantum optimization

Quantum algorithms with potential exponential advantage

- Systems of linear equations (HHL algorithm)
- Quantum machine learning and dequantization

Quantum algorithms with exponential advantage

- Space-efficient quantum algorithms
- Quantum algorithms for quantum chemistry

Quantum Machine Learning

[Gilyen, Su, Low, Wiebe 2020]

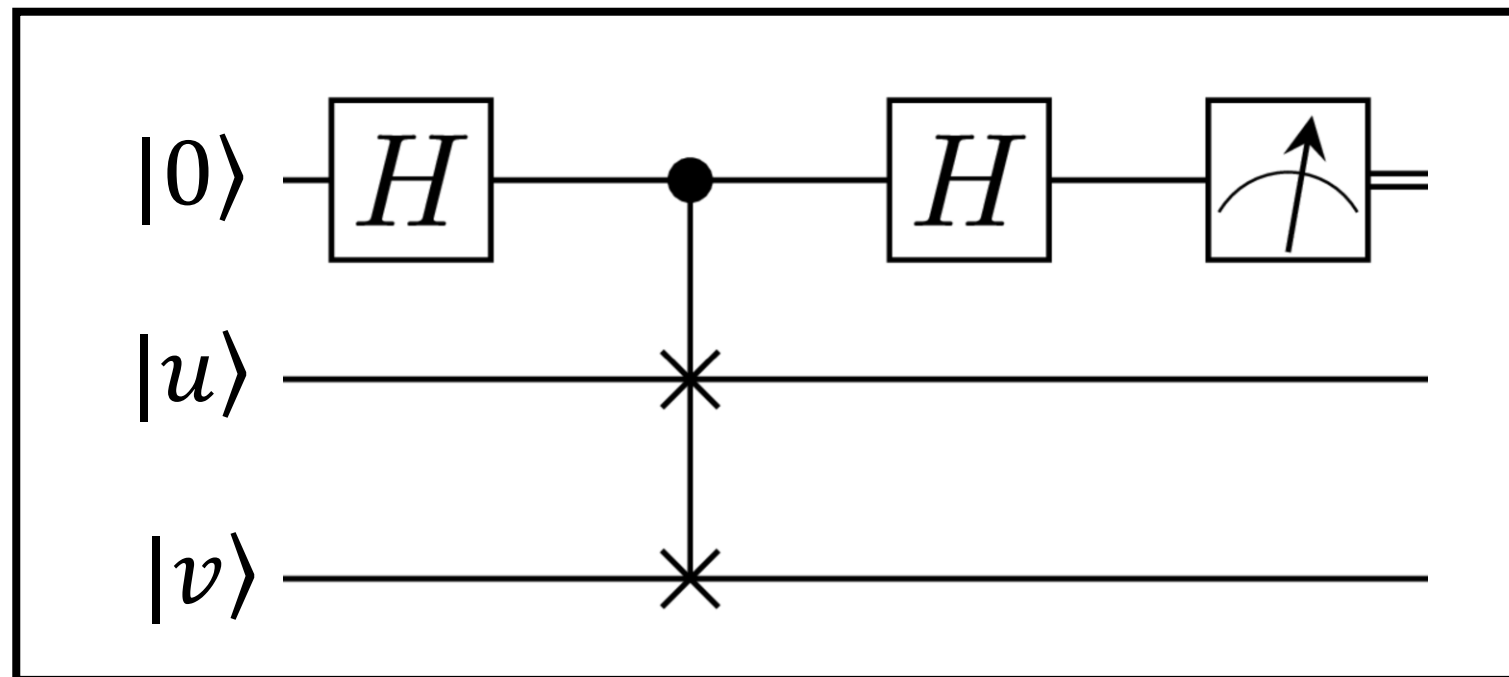
Gilyen, Su, Low, Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. 2019

Many quantum machine learning algorithms solve the following task (either as a main routine or a subroutine):

Given two unit-norm vectors $u, v \in \mathbb{R}^n$, compute an approximation of their inner product.

For several machine learning problems (e.g., recommendation systems or supervised clustering) the states can be created efficiently from the data using the Quantum Singular Value Transformation [Gilyen, Su, Low, Wiebe 2020]

If the vectors are given as quantum states, then we can use the SWAP test.



The probability of measuring 1 is $\frac{1}{2} - \frac{1}{2} |\langle u|v \rangle|^2$

Repeating the SWAP test $O(1/\epsilon^2)$ times gives an estimate of the (absolute value of the) inner product with additive error $O(\epsilon)$

independent of n !

Do we have quantum advantage?

Dequantization of the Inner Product Part [Tang 2019]

Given two unit-norm vectors $u, v \in \mathbb{R}^n$, compute an approximation of their inner product.

Assumption in the classical setting: length-squared access to u

- ✓ for $i \in \{1, \dots, n\}$, we can obtain u_i in $O(\log n)$ time
- ✓ we can in $O(\log n)$ time sample one index $i \in \{1, \dots, n\}$ from the following distribution p_u

concept introduced in the 1990s in works on “randomized linear algebra”

$$p_u: \{1, \dots, n\} \rightarrow [0, 1] \quad p_u(i) = |u_i|^2 \text{ for all } i \in \{1, \dots, n\}$$

same distribution as when measuring $|u\rangle = \sum_{i=1}^n u_i |i\rangle$

Tang’s paradigm: if we assume access to $|u\rangle$ (and $|v\rangle$) in the quantum setting, we should assume length-squared access to u (and v) in the classical setting

Dequantization of the Inner Product Part [Tang 2019]

Given two unit-norm vectors $u, v \in \mathbb{R}^n$, compute an approximation of their inner product.


Assumption in the classical setting: length-squared access to u (and also to v):

- ✓ for $i \in \{1, \dots, n\}$, we can obtain u_i in $O(\log n)$ time
- ✓ we can in $O(\log n)$ time sample one index $i \in \{1, \dots, n\}$ from the following distribution p_u

$$p_u: \{1, \dots, n\} \rightarrow [0, 1] \quad p_u(i) = |u_i|^2 \text{ for all } i \in \{1, \dots, n\}$$

Dequantized algorithm: sample an index $i \in \{1, \dots, n\}$ according to p_u and output the value v_i/u_i

Expectation of the output:
$$\sum_{i=1}^n p_u(i) \frac{v_i}{u_i} = \sum_{i=1}^n |u_i|^2 \frac{v_i}{u_i} = \sum_{i=1}^n u_i v_i = \langle u | v \rangle$$

Variance: small  repeating a small number of times and taking the mean gives a good estimate of $\langle u | v \rangle$

No quantum advantage!

Quantum Machine Learning

[Gilyen, Su, Low, Wiebe 2020]

Assume quantum access to the data

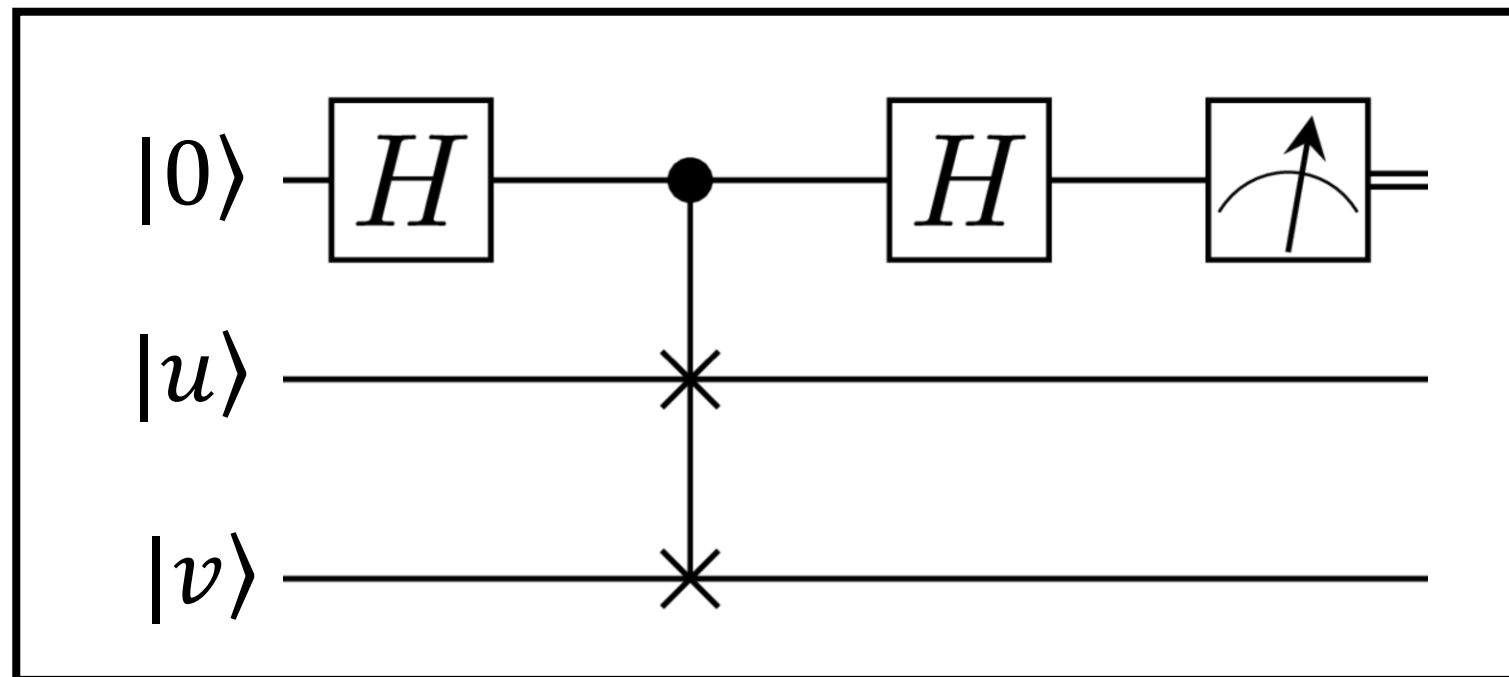
ation and beyond: exponential improvements

Many quantum machine learning algorithms solve the following task (either as a main routine or a subroutine):

Given two unit-norm vectors $u, v \in \mathbb{R}^n$, compute an approximation of their inner product.

For several machine learning problems (e.g., recommendation systems or supervised clustering) the states can be created efficiently from the data using the Quantum Singular Value Transformation.

If the vectors are given as quantum states, then we can use the SWAP test.



The probability of measuring 1 is $\frac{1}{2} - \frac{1}{2} |\langle u | v \rangle|^2$

Repeating the SWAP test $O(1/\epsilon^2)$ times gives an estimate of the inner product with additive error $O(\epsilon)$

Do we have quantum advantage?

Full Dequantization

[Tang 2019]

[Chia, Gilyen, Li, Lin, Tang, and Wang 2020]

~~Assume quantum access to the data~~

ation and beyond: exponential improvements

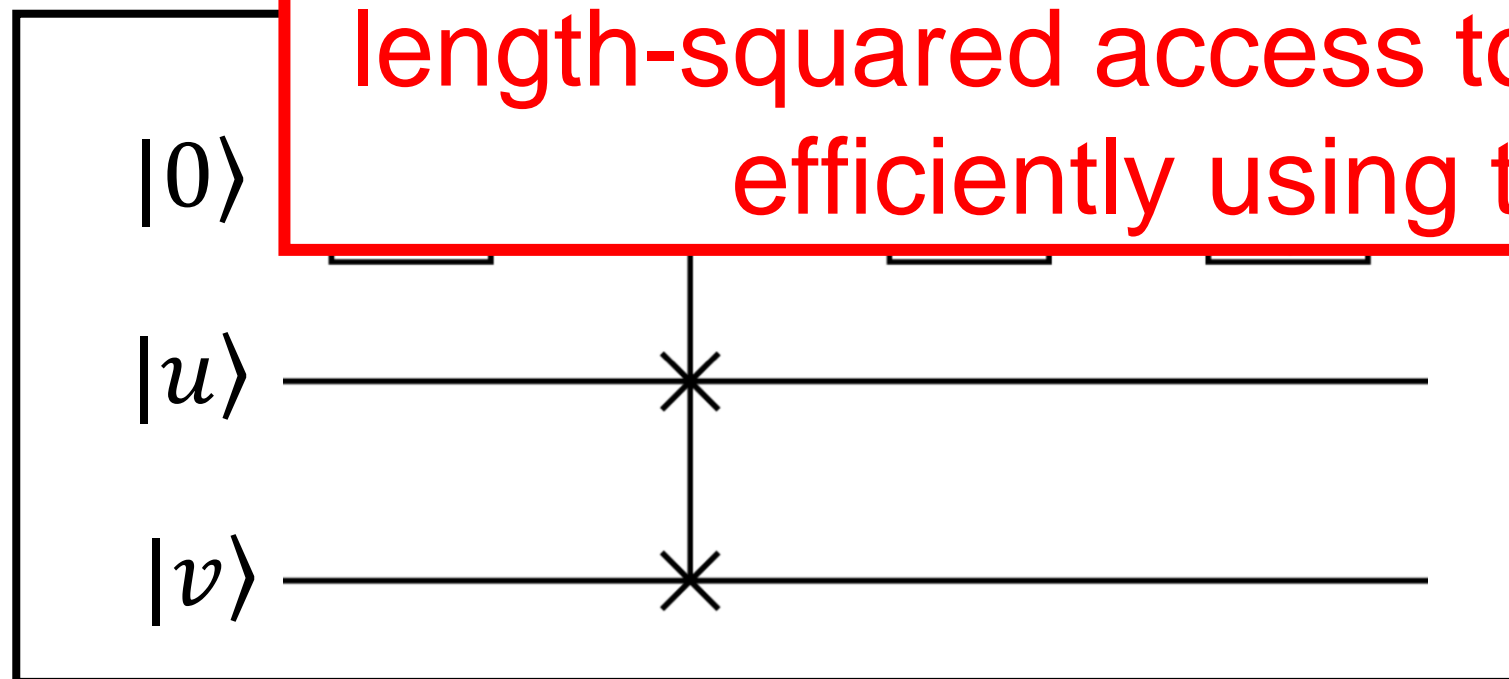
Ma **Assume we have length-squared access to the data** as a main routine or a subroutine):

Given two unit-norm vectors $u, v \in \mathbb{R}^n$, compute an approximation of their inner product.

For several machine learning problems (e.g., recommendation systems or supervised clustering) the states can be created efficiently from the data using the Quantum Singular Value Transformation.

If the vectors are given as quantum states, then we can use the SWAP test.

length-squared access to those vectors can be implemented efficiently using the methods from the 1990s



Repeating the SWAP test $O(1/\epsilon^2)$ times gives an estimate of the inner product with additive error $O(\epsilon)$

no quantum advantage here

Do we have quantum advantage?

Given two unit-norm vectors $u, v \in \mathbb{R}^n$, compute an approximation of their inner product.

Assumption in the classical setting: **approximate** length-squared access to u

- ✓ for $i \in \{1, \dots, n\}$, we can obtain u_i in $O(\log n)$ time
- ✓ we can in $O(\log n)$ time sample $i \in \{1, \dots, n\}$ from a distribution \tilde{p}_u close to p_u in total variation distance:

$$\sum_{i=1}^n |\tilde{p}_u(i) - p_u(i)| < 10^{-6}$$

Tentative algorithm: sample an index $i \in \{1, \dots, n\}$ according to \tilde{p}_u and output the value v_i/u_i

Expectation of the output:

$$\sum_{i=1}^n \tilde{p}_u(i) \frac{v_i}{u_i} \not\approx \sum_{i=1}^n \overset{|u_i|^2}{p_u(i)} \frac{v_i}{u_i} = \langle u | v \rangle$$

doesn't work anymore!
(problem when u_i small but $\tilde{p}_u(i)$ large)

On the other, the SWAP test on $|\tilde{u}\rangle$ and $|v\rangle$ where $\| |u\rangle - |\tilde{u}\rangle \| < 10^{-6}$ gives an estimate of $|\langle \tilde{u} | v \rangle| \approx |\langle u | v \rangle|$

Do we have a quantum advantage?

No, all known quantum machine learning algorithms can be dequantized in this setting as well [LG 2023]

Outline of the Talk

Quantum algorithms with polynomial advantage

- Matrix multiplication
- Quantum string algorithms
- Quantum optimization

Quantum algorithms with potential exponential advantage

- Systems of linear equations (HHL algorithm)
- Quantum machine learning and dequantization

} Need convincing applications

Quantum algorithms with exponential advantage

- Space-efficient quantum algorithms
- Quantum algorithms for quantum chemistry

Outline of the Talk

Quantum algorithms with polynomial advantage

- Matrix multiplication
- Quantum string algorithms
- Quantum optimization

Quantum algorithms with potential exponential advantage

- Systems of linear equations (HHL algorithm)
- Quantum machine learning and dequantization

Quantum algorithms with exponential advantage

- Space-efficient quantum algorithms
- Quantum algorithms for quantum chemistry

Space-efficient Equation Solving

[Ta-Shma 2013]

Input :
✓ a sparse and well-conditioned $n \times n$ matrix A
✓ a unit-norm vector $b \in \mathbb{C}^n$
✓ an index $i \in \{1, \dots, n\}$

Output : a good approximation of the i -th coordinate of the vector $x = A^{-1}b$

This problem is complete for
logspace quantum
computation
[Fefferman-Lin 2016]

$$\bar{x} = \frac{A^{-1}b}{\|A^{-1}b\|}$$

The HHL algorithm enables us to approximate the quantum state $|\bar{x}\rangle$ in time **polynomial in $\log n$**

Theorem ([Ta-Shma 2013]):

There exists a quantum algorithm that solves the above problem using **$O(\log n)$ space** (i.e., $O(\log n)$ bits and $O(\log n)$ qubits of memory) and **$\text{poly}(n)$ time**.

proof idea

repeat **$\text{poly}(n)$** times:

apply HHL and measure its output (which is close to $|\bar{x}\rangle$) in some appropriate basis \rightarrow gives a random coordinate of x

No **$o(n)$ -space** $\text{poly}(n)$ -time classical algorithm is known

**exponentially improvement in
the space requirements!**

Outline of the Talk

Quantum algorithms with polynomial advantage

- Matrix multiplication
- Quantum string algorithms
- Quantum optimization

Quantum algorithms with potential exponential advantage

- Systems of linear equations (HHL algorithm)
- Quantum machine learning and dequantization

Quantum algorithms with exponential advantage

- Space-efficient quantum algorithms
- Quantum algorithms for quantum chemistry

Computational Quantum Chemistry

Computing the ground energy of a quantum system is hard even for quantum computers

“The local Hamiltonian problem is QMA-hard” [Kempe, Kitaev and Regev 2004]

Given a rough approximation of the ground state (e.g., using Hartree–Fock in quantum chemistry), the ground energy can be estimated **with high precision** efficiently with a quantum computer

Very promising application

[Gharibian and LG 2022] [Cade, Folkertsma, Gharibian, Hayakawa, LG, Morimae and Weggemans 2023]

result #1: Given a rough approximation of the ground state, computing the ground energy with **high precision** is hard for classical computers

➡ This shows the superiority of quantum algorithms

result #2: Given a rough approximation of the ground state, computing the ground energy with **constant precision** can be done efficiently classically

➡ This shows the quantum advantage comes from the improved precision

Formalization: the Guided Local Hamiltonian Problem

Computing the ground energy of a quantum system is hard even for quantum computers

“The local Hamiltonian problem is QMA-hard” [Kempe, Kitaev and Regev 2004]

Given a rough approximation of the ground state (e.g., using Hartree–Fock in quantum chemistry), the ground energy can be estimated **with high precision** efficiently with a quantum computer

GLH(k, ϵ, δ) “Guided local Hamiltonian problem”

$k \geq 1$: locality parameter
 $\delta \in (0, 1]$: overlap parameter
 $\epsilon \in (0, 1]$: precision parameter

input: ① an k -local Hamiltonian H acting on n qubits such that $\|H\| \leq 1$
② an n -qubit quantum state $|u\rangle$

promise: $|u\rangle$ has overlap at least δ with the ground state of H

output: an estimate $\tilde{\lambda}$ such that $|\tilde{\lambda} - \lambda_H| \leq \epsilon$

λ_H : ground energy (i.e., smallest eigenvalue) of H

Formalizes the main computational task solved by quantum algorithms for quantum chemistry

Theorem (prior works):

For any $k \leq \log(n)$ any $\delta \geq 1/\text{poly}(n)$ and any $\epsilon \geq 1/\text{poly}(n)$, the problem GLH(k, ϵ, δ) can be solved in $\text{poly}(n)$ -time with a quantum computer.

GHL: our Results

n : number of qubits
(H : $2^n \times 2^n$ matrix)

$k \geq 1$: nb. of qubits on which each term of H acts
 $\delta \in (0,1]$: overlap between $|u\rangle$ and the ground state
 $\varepsilon \in (0,1]$: precision parameter

result #1: Given a rough approximation of the ground state, computing the ground energy with **high precision** is hard for classical computers

Formal statement:

[Gharibian and LG 2022]
[Cade, Folkertsma, Gharibian,
Hayakawa, LG, Morimae and
Weggemans 2023]

The problem $GLH(k,\varepsilon,\delta)$ is **BQP-hard** ← “as hard as simulating an arbitrary quantum circuit”
for $k = 2$, **$\varepsilon = 1/\text{poly}(n)$** and $\delta \approx 1$.

➡ This shows the superiority of quantum algorithms

result #2: Given a rough approximation of the ground state, computing the ground energy with **constant precision** can be done efficiently classically

Formal statement:

[Gharibian and LG 2022]

For any $k \leq \log(n)$ and any constant δ and **any constant $\varepsilon > 0$** , the problem $GLH(s,\varepsilon,\delta)$ can be solved in $\text{poly}(n)$ -time with a **classical** computer.

➡ This shows the quantum advantage comes from the improved precision

Theorem (prior works):

For any $k \leq \log(n)$ any $\delta \geq 1/\text{poly}(n)$ and any **$\varepsilon \geq 1/\text{poly}(n)$** , the problem $GLH(k,\varepsilon,\delta)$ can be solved in $\text{poly}(n)$ -time with a quantum computer.

Summary of the Talk

Quantum algorithms with polynomial advantage

- ➔ Matrix multiplication
- ➔ Quantum string algorithms
- ➔ Quantum optimization

Potential very wide impact, but need better understanding of quantum architectures to estimate the running time in practice

Quantum algorithms with potential exponential advantage

- ➔ Systems of linear equations (HHL algorithm)
- ➔ Quantum machine learning and dequantization

Need more convincing applications

Quantum algorithms with exponential advantage

- ➔ Space-efficient quantum algorithms
- ➔ Quantum algorithms for quantum chemistry

Some of the most convincing examples of quantum advantage

Perspectives

Most pressing questions:

- ✓ Find more applications of quantum computers, and especially more provable exponential speedups
- ✓ Build theoretical foundations for the advantage of “quantum heuristic algorithms”

Thank you for your attention!