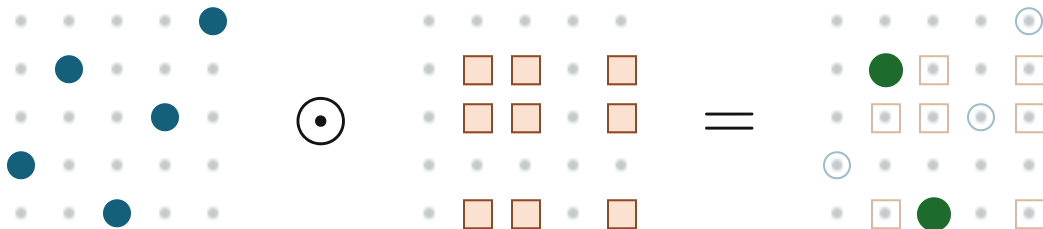


Global structure, local constraints, and quantum advantage

Multiplying quantum states to solve classical problems

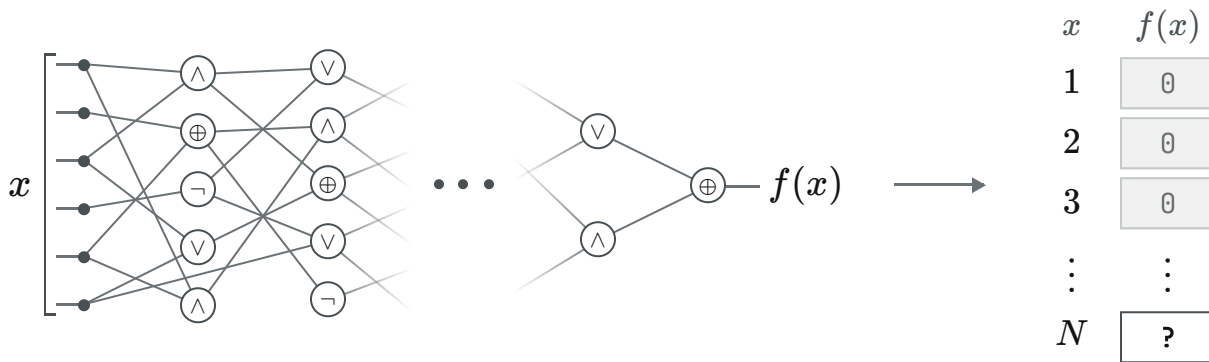
Kewen Wu Caltech



Known quantum speedups

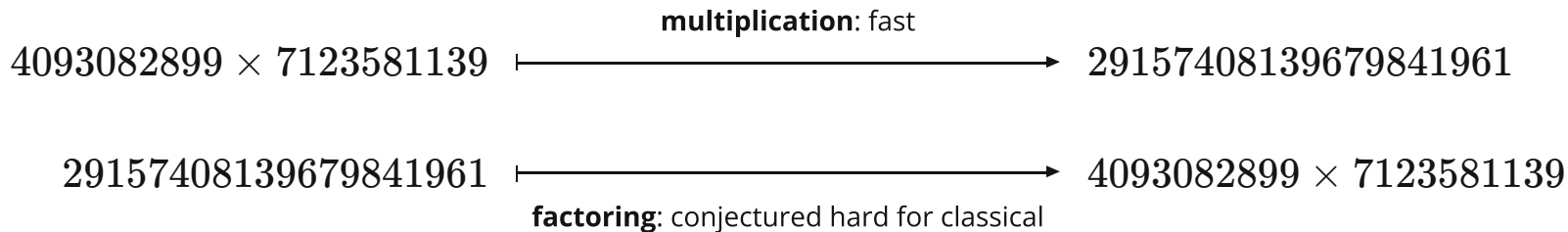
	BEST KNOWN CLASSICAL	QUANTUM	SPEEDUP
Unstructured search among N items	$\Theta(N)$	$\Theta(\sqrt{N})$ [Grover'96, BBBV'97]	quadratic

Given $f : \{1, 2, \dots, N\} \rightarrow \{0, 1\}$, output a location of 1, if f is nonzero



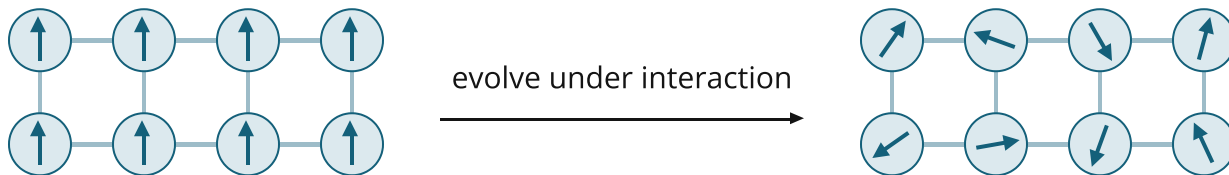
Known quantum speedups

	BEST KNOWN CLASSICAL	QUANTUM	SPEEDUP
Unstructured search among N items	$\Theta(N)$	$\Theta(\sqrt{N})$	quadratic
Factoring an integer with n bits	$\exp(O(n^{1/3} \log^{2/3} n))$ [LLMP'93]	$\text{poly}(n)$ [Shor'94]	exponential



Known quantum speedups

	BEST KNOWN CLASSICAL	QUANTUM	SPEEDUP
Unstructured search among N items	$\Theta(N)$	$\Theta(\sqrt{N})$	quadratic
Factoring an integer with n bits	$\exp(O(n^{1/3} \log^{2/3} n))$	$\text{poly}(n)$	exponential
Quantum simulation n qubits	$2^{O(n)}$ brute-force	$\text{poly}(n)$ [Lloyd'96]	exponential



Known quantum speedups

	BEST KNOWN CLASSICAL	QUANTUM	SPEEDUP
Unstructured search among N items	$\Theta(N)$	$\Theta(\sqrt{N})$	quadratic
Factoring an integer with n bits	$\exp(O(n^{1/3} \log^{2/3} n))$	$\text{poly}(n)$	exponential
Quantum simulation n qubits	$2^{O(n)}$	$\text{poly}(n)$	exponential

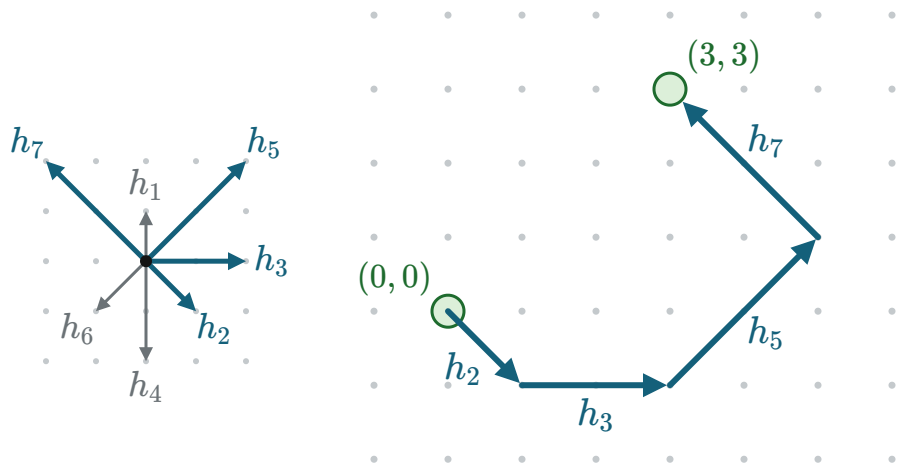
How to find more quantum advantages?
Exponential speedup and different algorithmic ideas?

Multiply two structured quantum states [Regev'05]

$$|\text{solutions}\rangle = |\text{global structure}\rangle \odot |\text{local constraints}\rangle$$

Vector subset sum modulo 3

Pick some of the input length- n vectors so they sum to zero modulo 3



$$h_2 + h_3 + h_5 + h_7 = (3, 3) \equiv (0, 0) \pmod{3}$$

Global structure

$$x_1 h_1 + x_2 h_2 + \cdots + x_m h_m \equiv \vec{0} \pmod{3}$$

x is not $\vec{0}$

Local constraints

$$\text{each } x_i \in \{0, 1\}$$

Classical

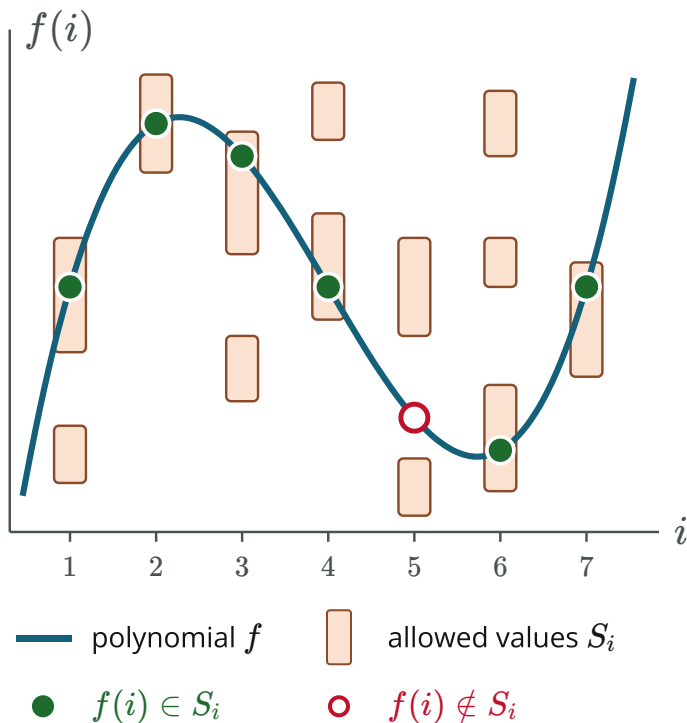
no efficient algorithm known if $m < 0.1n^2$

Quantum

polynomial time for any $m = \varepsilon n^2$ [KMOSW'26]

Polynomial fitting

Find a low-degree polynomial that takes allowed values at most points



Global structure

polynomial f has small degree

Local constraints

each $f(i) \in S_i$ and satisfy as many as possible

Classical

no efficient algorithm known to satisfy **76%** constraints

Quantum

polynomial time to satisfy **99%** constraints [JSW+'24, SW'26]
also has advantage in memory and data loading [WY'26]

Two types of quantum advantage

Candidate advantage

better than *every known* classical algorithm

Factoring [Shor'94]

Quantum simulation [Lloyd'96]

Polynomial fitting [JSW+'24, SW'26]

Vector subset sum modulo 3 [CLZ'22, KOW'25, KMOSW'26]

[CLZ'22] gave a primitive quantum algorithm

[KOW'25] designed a matching classical algorithm

[KMOSW'26] improved the quantum algorithm further

Part 1: vector subset sum

Proved advantage

better than *all possible* classical algorithms

Cloud computing

query model

[GTW'21, GSTW'24, LWWZ'25]

Parallel computing

circuit model

[KOW'24, GKMOW'26, GMW'26]

Distributed computing

communication model

[GSTW'23]

Big data

streaming model

[WY'26]

Part 2: polynomial fitting in big data

Common quantum idea $|\text{solutions}\rangle = |\text{global structure}\rangle \odot |\text{local constraints}\rangle$ [Regev'05]

PART 1

Vector subset sum modulo 3

Joint work with Robin Kothari, Tony Metger, Ryan O'Donnell, and Noah Shetty

Problem setup

Input $H = [h_1 \cdots h_m] \in \mathbb{F}_3^{n \times m}$

Output a nonzero $x \in \{0, 1\}^m$ satisfying $Hx = \vec{0}$

Result: a polynomial-time quantum algorithm for random H with $m = \varepsilon n^2$

In fact, we will sample a uniformly random $x \in \{0, 1\}^m$ satisfying $Hx = \vec{0}$

EXAMPLE: $n = 2, m = 4$

h_1	h_2	h_3	h_4
1	0	1	1
0	1	1	2
1	0	1	1

$$\underbrace{\begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \end{pmatrix}}_H \underbrace{\begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}}_x = \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ over } \mathbb{F}_3$$

Quantum basics

$$|\psi\rangle = \sum_{x \in \mathbb{F}_3^m} \underbrace{\alpha_x}_{\text{amplitude}} \underbrace{|x\rangle}_{\text{basis}}$$

Amplitudes as a function

a truth table of length 3^m , with value α_x at each x

x	0 ... 00	0 ... 01	0 ... 02	...	2 ... 22
α_x	$\alpha_{0\dots00}$	$\alpha_{0\dots01}$	$\alpha_{0\dots02}$	...	$\alpha_{2\dots22}$

Measuring $|\psi\rangle$ returns x with probability proportional to $|\alpha_x|^2$

Quantum operations are (simple) unitary transforms

Fourier transform on $\mathbb{F}_3$, with $\omega = e^{2\pi i/3}$

$$F_3|0\rangle = \frac{1}{\sqrt{3}}(|0\rangle + |1\rangle + |2\rangle)$$

$$F_3|1\rangle = \frac{1}{\sqrt{3}}(|0\rangle + \omega|1\rangle + \omega^2|2\rangle)$$

$$F_3|2\rangle = \frac{1}{\sqrt{3}}(|0\rangle + \omega^2|1\rangle + \omega|2\rangle)$$

Constraints as states [Regev'05, CLZ'22]

Recall: we want to sample $x \in \{0, 1\}^m \subset \mathbb{F}_3^m$ satisfying $Hx = \vec{0}$

Global

$$|C\rangle = \sum_{x: Hx=\vec{0}} |x\rangle$$

all vectors satisfying the linear equations

Local

$$|B\rangle = (|0\rangle + |1\rangle)^{\otimes m}$$

every coordinate is independently 0 or 1

$$|0\rangle + |1\rangle \otimes |0\rangle + |1\rangle \otimes |0\rangle + |1\rangle \otimes \dots$$

$$\underbrace{|C\rangle}_{Hx=\vec{0}} \odot \underbrace{|B\rangle}_{x_i \in \{0,1\}} = \underbrace{|S\rangle}_{S=C \cap B} = \sum_{x \in \{0,1\}^m, Hx=\vec{0}} |x\rangle$$

Entrywise multiply the amplitudes

basis label	0000	1000	2210	1011
$ C\rangle$	1	0	1	1
$ B\rangle$	1	1	0	1
$ S\rangle$	1	0	0	1

Entrywise product and convolution

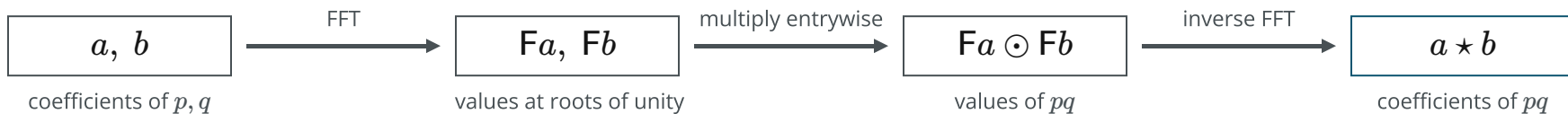
Entrywise product

$$(\alpha \odot \beta)(x) = \alpha(x) \beta(x)$$

Convolution

$$(\alpha \star \beta)(w) = \sum_{u+v=w} \alpha(u) \beta(v)$$

Example: Fast Fourier transform multiplies polynomials p and q



$$|C\rangle \odot |B\rangle = |S\rangle \quad \Leftrightarrow \quad (F|C\rangle) \star (F|B\rangle) = F|S\rangle$$

Fourier transform
 $F = F_3^{\otimes m}$

Why convolution?

Assume we have states $|\phi_1\rangle = \sum_x \alpha_x |x\rangle$ and $|\phi_2\rangle = \sum_x \beta_x |x\rangle$

Then we put them together and have

$$|\phi_1\rangle \otimes |\phi_2\rangle = \sum_{x,y} \alpha_x \beta_y |x\rangle |y\rangle$$

Then we can apply reversible addition to obtain

$$\sum_{x,y} \alpha_x \beta_y |x\rangle |x+y\rangle = \sum_{x,z} \alpha_x \beta_{z-x} |x\rangle |z\rangle = \sum_x \alpha_x \left(\sum_z \beta_{z-x} |z\rangle \right) |x\rangle$$

Now **if we could erase x given $\sum_z \beta_{z-x} |z\rangle$** , then we would obtain

$$\sum_{x,z} \alpha_x \beta_{z-x} |z\rangle = \sum_z \left(\sum_x \alpha_x \beta_{z-x} \right) |z\rangle = |\phi_1\rangle \star |\phi_2\rangle$$

from convolution to a quantum decoding problem

Back to our case

$$(F|C\rangle) \quad \boxed{\star} \quad (F|B\rangle)$$

?

$$|C\rangle = \sum_{x \in \mathbb{F}_3^m, Hx = \vec{0}} |x\rangle$$

$$|B\rangle = \sum_{x \in \{0,1\}^m} |x\rangle = (|0\rangle + |1\rangle)^{\otimes m}$$

$$F|C\rangle = \sum_{s \in \mathbb{F}_3^n} |H^\top s\rangle$$

superposition of linear combinations of rows of H

$$H = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \end{pmatrix}$$

$$F|C\rangle = \sum_{s \in \mathbb{F}_3^2} |s_1(1, 0, 1, 1) + s_2(0, 1, 1, 2)\rangle$$

Easy to prepare

$$F|B\rangle = \boxed{|\hat{0}\rangle + |\hat{1}\rangle} \otimes \boxed{|\hat{0}\rangle + |\hat{1}\rangle} \otimes \boxed{|\hat{0}\rangle + |\hat{1}\rangle} \otimes \dots$$

where

$$|\hat{0}\rangle = F_3|0\rangle = \frac{1}{\sqrt{3}}(|0\rangle + |1\rangle + |2\rangle)$$

$$|\hat{1}\rangle = F_3|1\rangle = \frac{1}{\sqrt{3}}(|0\rangle + \omega|1\rangle + \omega^2|2\rangle)$$

$$\omega = e^{2\pi i/3}$$

Easy to prepare

Back to our case

$$|C\rangle \odot |B\rangle = |S\rangle \Leftrightarrow (F|C\rangle) \star (F|B\rangle) = F|S\rangle$$

In our case, $|\phi_1\rangle = F|C\rangle = \sum_{s \in \mathbb{F}_3^n} |H^\top s\rangle$ and $|\phi_2\rangle = F|B\rangle = \sum_{y \in \mathbb{F}_3^m} \beta_y |y\rangle$

Putting them together, we have

$$F|C\rangle \otimes F|B\rangle = \sum_{s,y} \beta_y |H^\top s\rangle |y\rangle$$

Applying reversible addition, we obtain

$$\sum_{s,y} \beta_y |H^\top s\rangle |H^\top s + y\rangle = \sum_{s,z} \beta_{z-H^\top s} |H^\top s\rangle |z\rangle = \sum_s \left(\sum_z \beta_{z-H^\top s} |z\rangle \right) |H^\top s\rangle$$

Now **if we could compute s given $\sum_z \beta_{z-H^\top s} |z\rangle$** , then we can erase $H^\top s$ and obtain

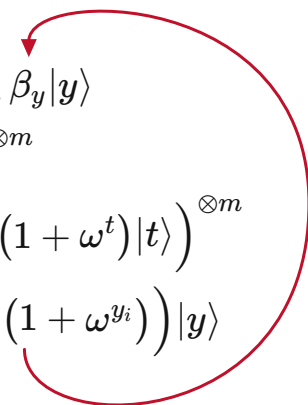
$$\sum_{s,z} \beta_{z-H^\top s} |z\rangle = \sum_z \left(\sum_s \beta_{z-H^\top s} \right) |z\rangle = F|C\rangle \star F|B\rangle$$

The quantum decoding problem

Known $H \in \mathbb{F}_3^{n \times m}$

Input one quantum state $|\Psi_s\rangle = \sum_z \beta_{z-H^\top s} |z\rangle$, for an unknown $s \in \mathbb{F}_3^n$

Goal recover s with high probability


$$\begin{aligned} |\phi_2\rangle &= \mathbf{F}|B\rangle = \sum_y \beta_y |y\rangle \\ &= (|\hat{0}\rangle + |\hat{1}\rangle)^{\otimes m} \\ &= \left(\frac{1}{\sqrt{3}} \sum_{t \in \mathbb{F}_3} (1 + \omega^t) |t\rangle \right)^{\otimes m} \\ &= \sum_y \left(\prod_i \frac{1}{\sqrt{3}} (1 + \omega^{y_i}) \right) |y\rangle \end{aligned}$$

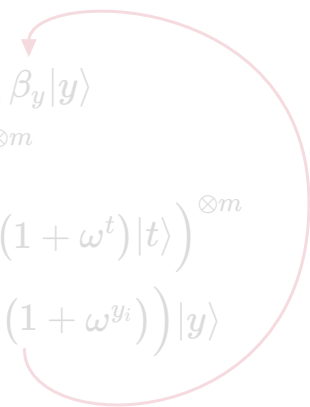
$$\begin{aligned} |\Psi_s\rangle &= \sum_z \beta_{z-H^\top s} |z\rangle \\ &= \sum_z \left(\prod_i \frac{1}{\sqrt{3}} (1 + \omega^{z_i - (H^\top s)_i}) \right) |z\rangle \\ &= \bigotimes_{i=1}^m \frac{1}{\sqrt{3}} \sum_{t \in \mathbb{F}_3} (1 + \omega^{t - (H^\top s)_i}) |t\rangle \\ &= \bigotimes_{i=1}^m (|\hat{0}\rangle + \omega^{-(H^\top s)_i} |\hat{1}\rangle) \end{aligned}$$

The quantum decoding problem

Known $H \in \mathbb{F}_3^{n \times m}$

Input one quantum state $|\Psi_s\rangle = \sum_z \beta_{z-H^\top s} |z\rangle$, for an unknown $s \in \mathbb{F}_3^n$

Goal recover s with high probability


$$\begin{aligned} |\phi_2\rangle &= \mathbf{F}|B\rangle = \sum_y \beta_y |y\rangle \\ &= (|\hat{0}\rangle + |\hat{1}\rangle)^{\otimes m} \\ &= \left(\frac{1}{\sqrt{3}} \sum_{t \in \mathbb{F}_3} (1 + \omega^t) |t\rangle \right)^{\otimes m} \\ &= \sum_y \left(\prod_i \frac{1}{\sqrt{3}} (1 + \omega^{y_i}) \right) |y\rangle \end{aligned}$$

$$\begin{aligned} |\Psi_s\rangle &= \sum_z \beta_{z-H^\top s} |z\rangle \\ &= \sum_z \left(\prod_i \frac{1}{\sqrt{3}} (1 + \omega^{z_i - (H^\top s)_i}) \right) |z\rangle \\ &= \bigotimes_{i=1}^m \frac{1}{\sqrt{3}} \sum_{t \in \mathbb{F}_3} (1 + \omega^{t - (H^\top s)_i}) |t\rangle \\ &= \bigotimes_{i=1}^m (|\hat{0}\rangle + \omega^{-(H^\top s)_i} |\hat{1}\rangle) \end{aligned}$$

$$|\psi_{(H^\top s)_1}\rangle \otimes |\psi_{(H^\top s)_2}\rangle \otimes \cdots \otimes |\psi_{(H^\top s)_i}\rangle \otimes \cdots \otimes |\psi_{(H^\top s)_m}\rangle$$

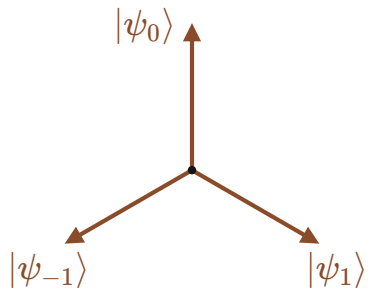
focus on one coordinate: $a = (H^\top s)_i \in \mathbb{F}_3$

What can one coordinate $|\psi_{(H^\top s)_i}\rangle$ tell us about its value $a = (H^\top s)_i$?

The state distinguishing problem

Input one quantum state $|\psi_a\rangle = |\hat{0}\rangle + \omega^{-a}|\hat{1}\rangle$, for an unknown $a \in \{0, 1, -1\}$

Goal get useful information about a with constant probability, we learn a value that a is not

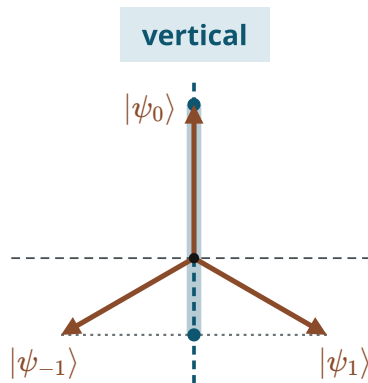


Three candidates on the same plane

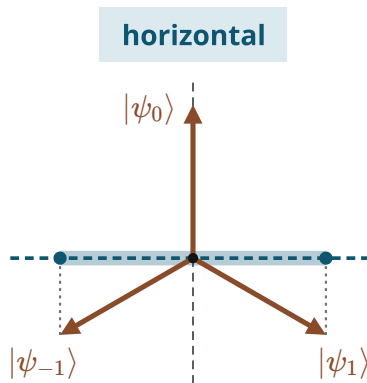
Perfect identification is impossible

only orthogonal states can be distinguished apart

Instead, test whether a could be 0



every candidate is possible



$|\psi_0\rangle$ has no horizontal component

$a \neq 0$

Learning from linear disequations

Known $H \in \mathbb{F}_3^{n \times m}$ with columns h_i and values c_i for $i = 1, 2, \dots, m$

Input disequations $\langle h_i, s \rangle \neq c_i$, for an unknown $s \in \mathbb{F}_3^n$

Goal recover s

linear disequation

$$\langle h_i, s \rangle \neq c_i \quad \text{in } \mathbb{F}_3$$

$$\Leftrightarrow \langle h_i, s \rangle - c_i \in \{+1, -1\} \quad \text{in } \mathbb{F}_3$$

$$\Leftrightarrow (\langle h_i, s \rangle - c_i)^2 - 1 = 0 \quad \text{in } \mathbb{F}_3$$

quadratic equation

Example: $H = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 2 & 0 \end{pmatrix}, \quad c = (0, 2, 1, 1, 2)$

$$s_1 \neq 0 \quad \Leftrightarrow s_1^2 - 1 = 0$$

$$s_2 \neq 2 \quad \Leftrightarrow (s_2 - 2)^2 - 1 = 0$$

$$s_1 + s_2 \neq 1 \quad \Leftrightarrow (s_1 + s_2 - 1)^2 - 1 = 0$$

$$s_1 + 2s_2 \neq 1 \quad \Leftrightarrow (s_1 + 2s_2 - 1)^2 - 1 = 0$$

$$s_1 \neq 2 \quad \Leftrightarrow (s_1 - 2)^2 - 1 = 0$$

How do we solve this?

The linearization algorithm

- 1 Expand quadratic equation and treat monomials as new unknowns
- 2 Solve the linear system and read off the secret

linear disequation

$$s_1 + s_2 \neq 1$$

quadratic equation

$$(s_1 + s_2 - 1)^2 - 1 = 0$$

expand

$$s_1 + s_2 + s_1^2 + 2s_1s_2 + s_2^2 = 0$$

vector view

s_1	s_2	s_1^2	s_1s_2	s_2^2	
1	1	1	2	1	= 0

Nonlinear in s , but linear in its monomials

Example from last page:

	s_1	s_2	s_1^2	s_1s_2	s_2^2	
$s_1 \neq 0$	0	0	1	0	0	= 1
$s_2 \neq 2$	0	2	0	0	1	= 0
$s_1 + s_2 \neq 1$	1	1	1	2	1	= 0
$s_1 + 2s_2 \neq 1$	1	2	1	1	1	= 0
$s_1 \neq 2$	2	0	1	0	0	= 0
solution	1	1	1	1	1	

$s = (1, 1)$

What if the solution is not unique?

High-order linearization

When direct linearization has no **unique solution**, create more equations

- 1 Multiply each quadratic equation by low-degree monomials
- 2 Linearize the larger system as before

A zero formula stays zero for any multiple
 $q(s) = 0 \Rightarrow \mu(s) q(s) = 0$

quadratic equation

$$s_1 + s_2 + s_1^2 + 2s_1s_2 + s_2^2 = 0$$

multiply by s_1 and expand $s_1q(s)$

$$s_1^2 + s_1s_2 + s_1^3 + 2s_1^2s_2 + s_1s_2^2 = 0$$

replace $s_1^3 = s_1$ over $\mathbb{F}_3$

$$s_1^2 + s_1s_2 + s_1 + 2s_1^2s_2 + s_1s_2^2 = 0$$

vector view

	s_1	s_2	s_1^2	s_1s_2	s_2^2	new unknowns of degree 3		
						$s_1^2s_2$	$s_1s_2^2$	
q	1	1	1	2	1	0	0	= 0
s_1q	1	0	1	1	0	2	1	= 0
s_2q	0	1	0	1	1	1	2	= 0
$\vdots$								$\vdots$

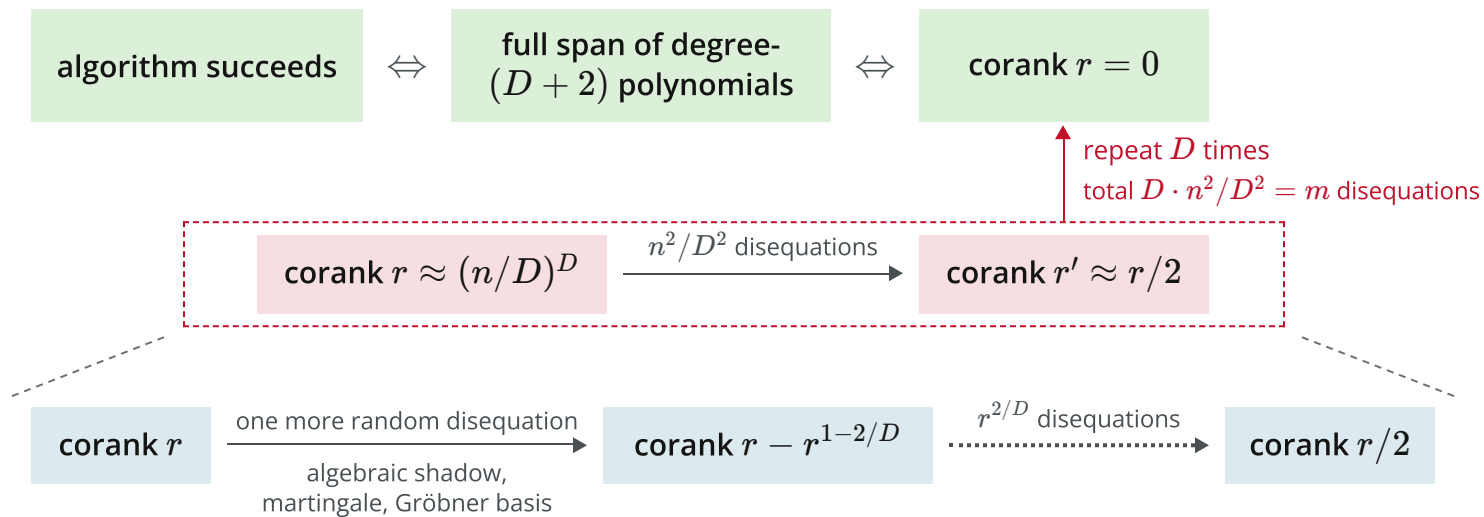
Multiply each equation by every monomial of degree $\leq D$,
then linearize: one linear system in the monomials of degree $\leq D + 2$

Provable guarantee

Theorem [KMOSW'26]

Assume we are given $m = \varepsilon \cdot n^2$ random disequations. Let $D \approx 1/\varepsilon$.

With high probability, the degree- D linearization algorithm works. Moreover, the runtime is $n^{O(D)}$.

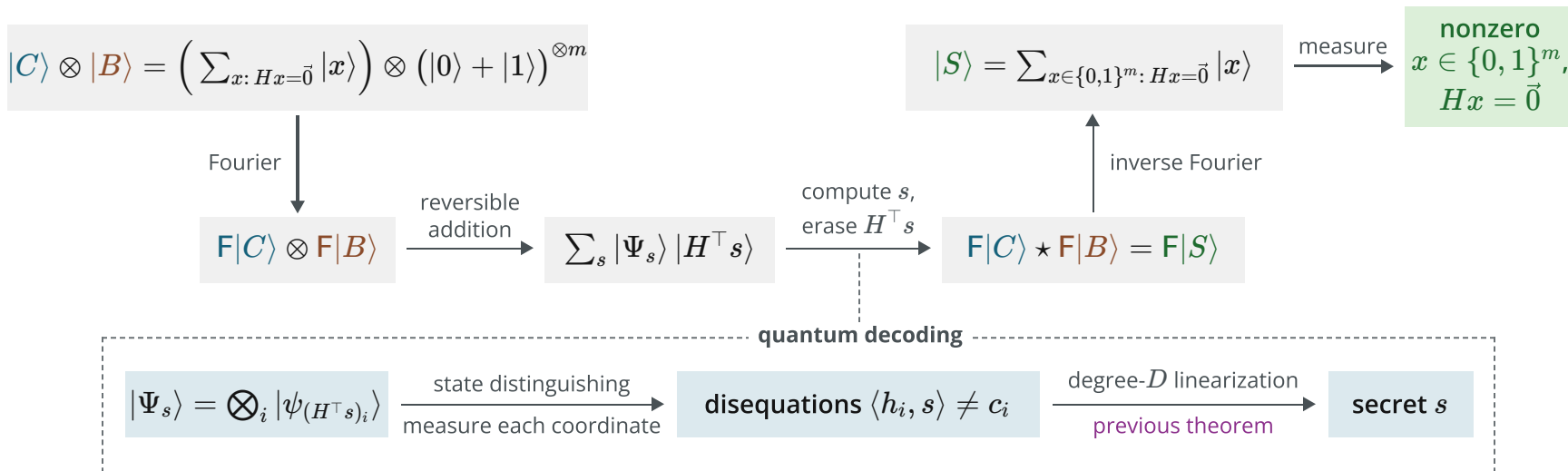


Back to vector subset sum modulo 3

Theorem [KMOSW'26]

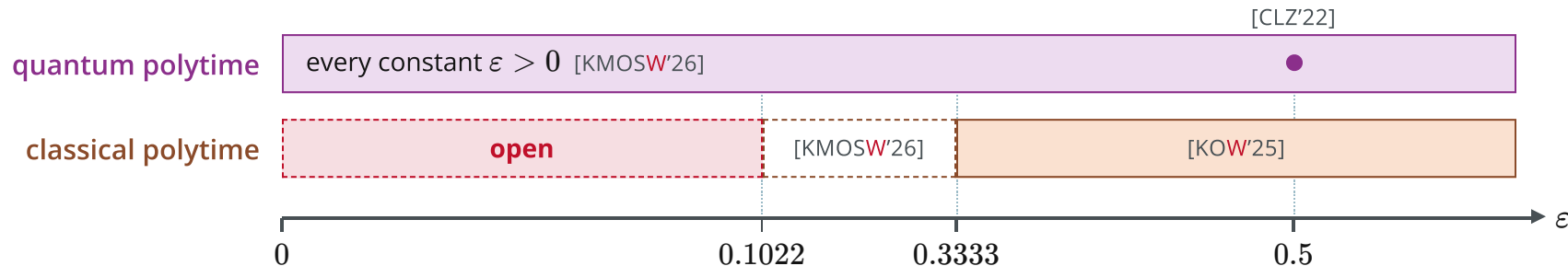
Assume $m = \varepsilon \cdot n^2$ where ε is a constant. Let $h_1, \dots, h_m \in \mathbb{F}_3^n$ be random vectors.

A **polynomial-time quantum** algorithm finds a nonempty subset summing to $\vec{0}$ with high probability.



Classical algorithms?

The quantum algorithm runs in polynomial time with $m = \varepsilon \cdot n^2$ for **every constant** $\varepsilon > 0$
No known classical algorithm matches this



Is there a classical polynomial-time algorithm for every constant $\varepsilon > 0$?

In fact, no quasipolynomial- or even $\exp(n^{0.99})$ -time algorithm is known.

PART 2

Polynomial fitting in a data stream

Joint work with Guangxu Yang

Polynomial fitting

Input sets of allowed values $S_i \subseteq \mathbb{F}_p$ for each $i \in \mathbb{F}_p$

Output a univariate polynomial f of degree $\leq K$ over $\mathbb{F}_p$
with $\text{score}(f)$ close to OPT

$$\text{score}(f) = \#\{i \in \mathbb{F}_p : f(i) \in S_i\}$$

$$\text{OPT} = \max_{\deg(f) \leq K} \text{score}(f)$$

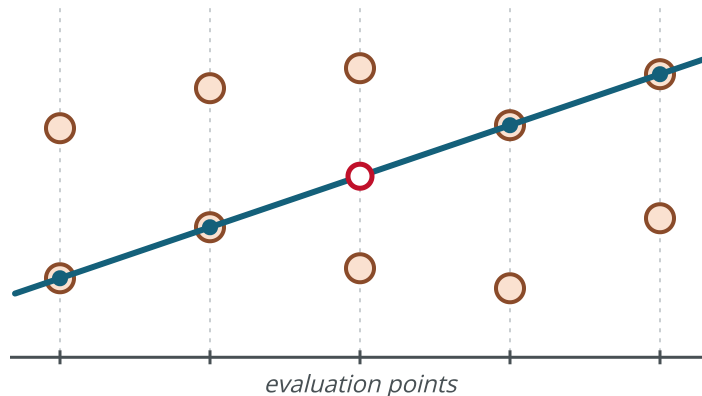
$K = 1$: linear regression over $\mathbb{F}_p$

Find a straight line fitting as many data points as possible

○ data points

● $f(i) \in S_i$: counts toward the score

○ $f(i) \notin S_i$: does not count



Polynomial fitting

Input sets of allowed values $S_i \subseteq \mathbb{F}_p$ for each $i \in \mathbb{F}_p$

Output a univariate polynomial f of degree $\leq K$ over $\mathbb{F}_p$
with $\text{score}(f)$ close to OPT

$$\text{score}(f) = \#\{i \in \mathbb{F}_p : f(i) \in S_i\}$$

$$\text{OPT} = \max_{\deg(f) \leq K} \text{score}(f)$$

$$\begin{aligned} |S_i| &\approx p/2 \\ \text{and} \\ K &\approx p/2 \end{aligned}$$

Theorem A polynomial-time **quantum** algorithm satisfies 99% of all constraints
[JSW+'24, SW'26] No **known** polynomial-time **classical** algorithm can satisfy even 76%

Potential quantum advantages

The $K = 1$ linear regression case is classically easy; the potential quantum advantage appears in high degree

Polynomial fitting in a stream

Input sets of allowed values $S_i \subseteq \mathbb{F}_p$ for each $i \in \mathbb{F}_p$
as a stream of entries $(i, v, \text{yes/no } v \in S_i)$ arriving one at a time



$K = 1$: streaming
linear regression

Output a univariate polynomial f of degree $\leq K$ over $\mathbb{F}_p$
with $\text{score}(f)$ close to OPT

$$\text{score}(f) = \#\{i \in \mathbb{F}_p : f(i) \in S_i\}$$

$$\text{OPT} = \max_{\deg(f) \leq K} \text{score}(f)$$

Theorem [WY'26] A polynomial-time **quantum** algorithm can reach 99% with $N^{0.5}$ qubits of memory and by going through the stream just once

Any classical algorithm reaching even 76% needs $N^{0.51}$ bits of memory, even if it can go through the stream $N^{0.01}$ times and regardless of computation time

**Provable
quantum
advantages**

Higher-order polynomial fitting in a stream

Input sets of allowed values $S_i \subseteq \mathbb{F}_p^\ell$ for each $i \in \mathbb{F}_p$
as a stream of entries $(i, v, \text{yes/no } v \in S_i)$ arriving one at a time

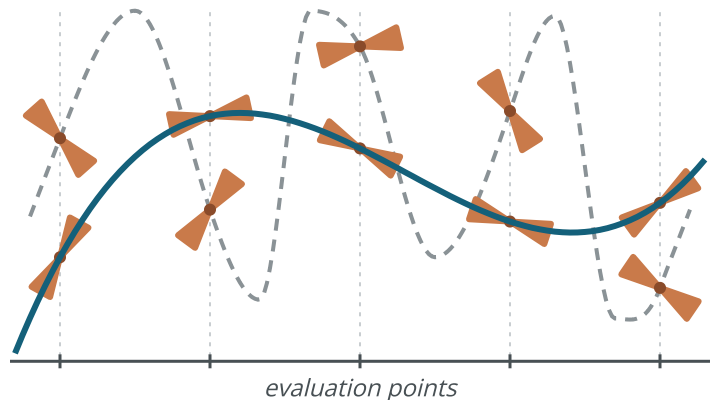
Output a univariate polynomial f of degree $\leq K$ over $\mathbb{F}_p$
with $\text{score}(f)$ close to OPT

$$\text{score}(f) = \#\{i \in \mathbb{F}_p : (f(i), f'(i), \dots, f^{(\ell-1)}(i)) \in S_i\}$$

$$\text{OPT} = \max_{\deg(f) \leq K} \text{score}(f)$$

EXAMPLE: $\ell = 2$

 Allowed S_i constraining $f(i)$ and $f'(i)$



Higher-order polynomial fitting in a stream

Input sets of allowed values $S_i \subseteq \mathbb{F}_p^\ell$ for each $i \in \mathbb{F}_p$
as a stream of entries $(i, v, \text{yes/no } v \in S_i)$ arriving one at a time

Output a univariate polynomial f of degree $\leq K$ over $\mathbb{F}_p$
with $\text{score}(f)$ close to OPT

$$\text{score}(f) = \#\{i \in \mathbb{F}_p : (f(i), f'(i), \dots, f^{(\ell-1)}(i)) \in S_i\}$$

$$\text{OPT} = \max_{\deg(f) \leq K} \text{score}(f)$$

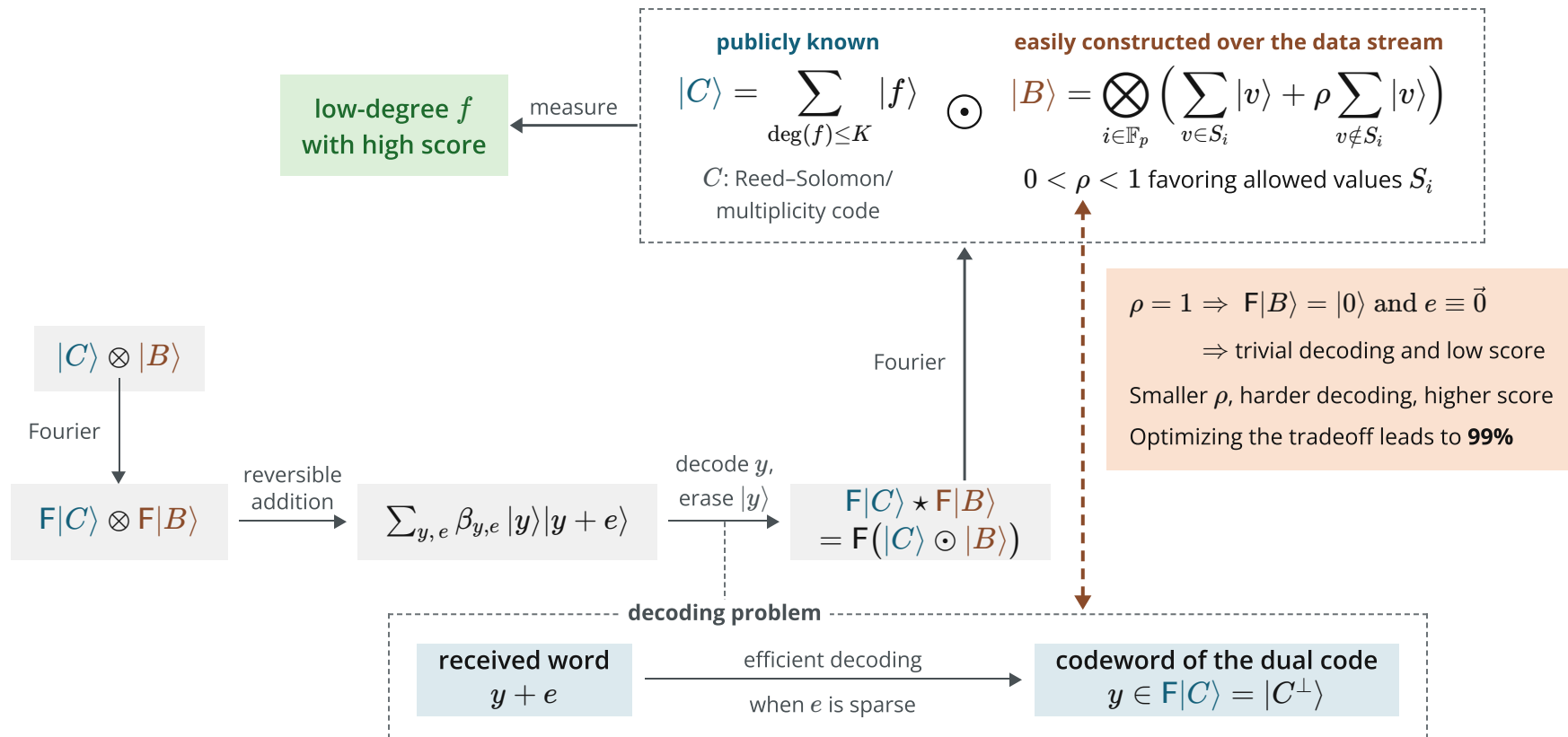
Theorem ^[WY'26] A **quantum** algorithm can reach 99% with $\text{polylog}(N)$ qubits of memory ,
by going through the stream just once , and using $\text{polylog}(N)$ time per stream entry

Any classical algorithm reaching even 76% needs $N^{0.01}$ bits of memory ,
even if it can go through the stream $N^{0.01}$ times and regardless of computation time

**Stronger
quantum
advantages**

for some parameter choices of p, ℓ and K

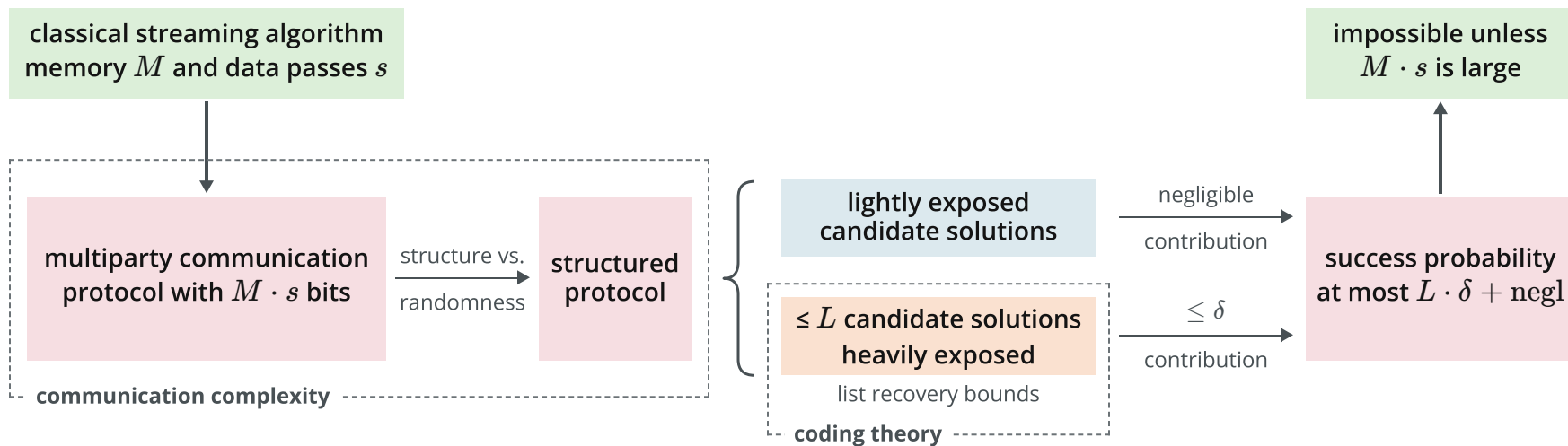
The quantum algorithm



The classical lower bound

Theorem [WY'26]

Assume a classical streaming algorithm uses M bits of memory, s data passes, and reaches **76% OPT**
Then $M \cdot s \geq N^{0.1}$ where N is the length of a random input stream



Summary

Quantum algorithmic ideas

Period finding / phase estimation

[Shor'94, Kitaev'95]

factoring, discrete logarithm, ground-state energies

Grover search [Grover'96]

unstructured search

Hamiltonian simulation [Lloyd'96]

quantum dynamics

Adiabatic / variational optimization

[FGGS'00, FGG'14]

combinatorial optimization (max-cut)

Multiplying quantum states [Regev'05]

|global structure⟩ ⊙ |local constraints⟩

vector subset sum modulo 3 [CLZ'22, KOW'25, KMOSW'26]

polynomial fitting [JSW+'24, SW'26, WY'26]

Solve more problems with these techniques
Develop other quantum algorithmic ideas

Thank you!