LWE with Quantum Amplitudes: Algorithm, Hardness, and Oblivious Sampling

Yilei Chen — Tsinghua
Zihan Hu — EPFL
Qipeng Liu — UCSD
Han Luo — Tsinghua
Yaxin Tu — Princeton

https://arxiv.org/abs/2310.00644

Lattice problems that are conjectured hard against quantum computers:

- Short vector problems (SVP)
- Short integer solution (SIS)
- Learning with errors (LWE)

Are they really hard for quantum computers?

In this talk

- Intro to learning with errors (LWE) and its quantum variant “Solve |LWE>” (S|LWE>).
- S|LWE> for Gaussian amplitudes: hardness and algorithm.
- S|LWE> for specific amplitudes: algorithm and application to oblivious sampling.

In this talk

- Intro to learning with errors (LWE) and its quantum variant “Solve |LWE>” (S|LWE>).
- S|LWE> for Gaussian amplitudes: hardness and algorithm.
- S|LWE> for specific amplitudes: algorithm and application to oblivious sampling.

What is learning with errors (LWE)?

$s = [\ s_1\ ,\ s_2\ ,\ s_3\ ,\ s_4\ ]$ is the secret vector.

Given an oracle O_s( ). Over one click, returns a random linear combination of the secret, plus a small amount of noise.

What is learning with errors (LWE)?

$s = [\, s_1 , s_2 , s_3 , s_4 \,]$ is the secret vector.

Given an oracle O_s( ). Over one click, returns a random linear combination of the secret, plus a small amount of noise.

(think of ≈ as + or - a small number)

$$
\begin{aligned}
34\, s_1 + 12\, s_2 + 39\, s_3 + 16\, s_4 &\approx 38 \\
63\, s_1 + 29\, s_2 + 17\, s_3 + 7\, s_4 &\approx 22 \\
9\, s_1 + 31\, s_2 + 52\, s_3 + 14\, s_4 &\approx 1 \\
54\, s_1 + 18\, s_2 + 43\, s_3 + 61\, s_4 &\approx 59 \\
19\, s_1 + 27\, s_2 + 53\, s_3 + 13\, s_4 &\approx 15 \\
&\dots \\
24\, s_1 + 50\, s_2 + 3\, s_3 + 36\, s_4 &\approx 58
\end{aligned}
\qquad \text{mod } 67
$$

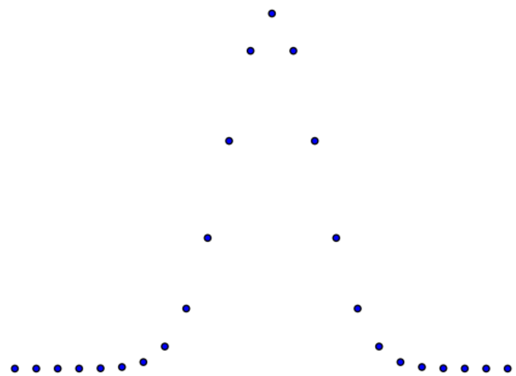
LWE: given the coefficients, the answers, find the secret vector.

Learning with errors (formal)

$s = [\, s_1, s_2, \dots, s_n \,]$ is the secret vector.

Given samples of the form

$$a_1, \quad y_1 = \langle s, a_1 \rangle + e_1 \mod q$$
$$a_2, \quad y_2 = \langle s, a_2 \rangle + e_2 \mod q$$
$$\dots$$
$$a_m, \quad y_m = \langle s, a_m \rangle + e_m \mod q$$

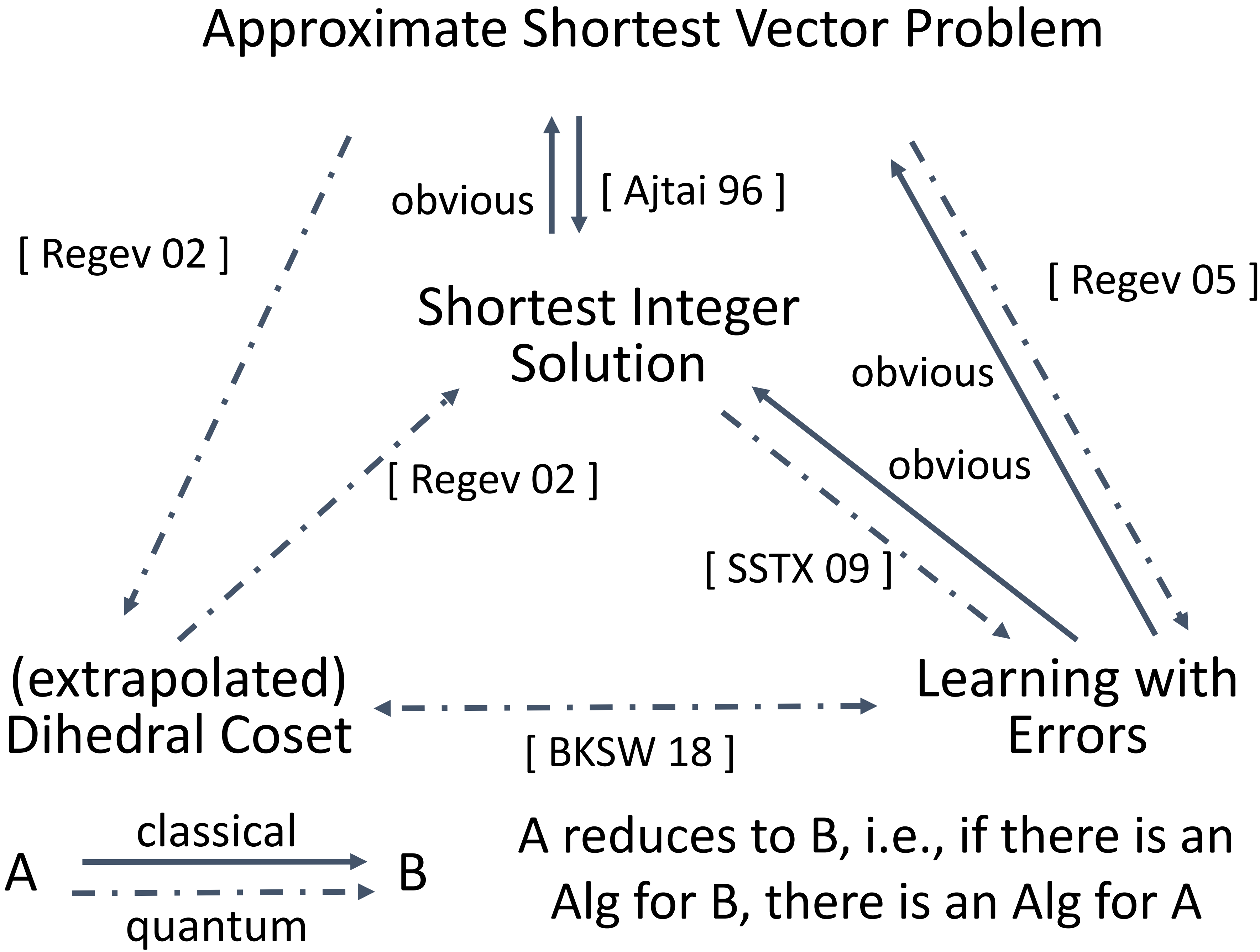


Typical error distribution: Gaussian

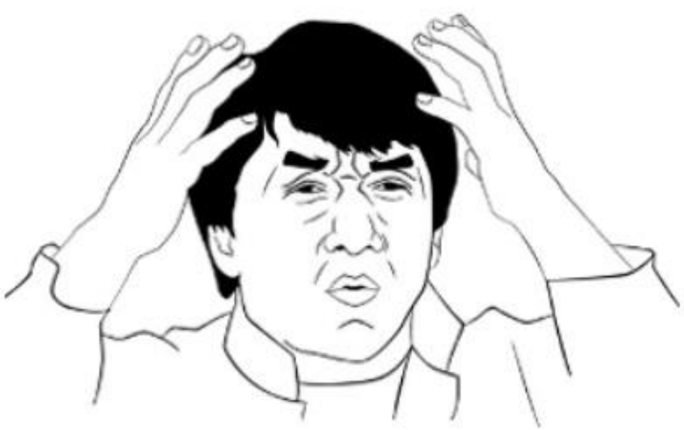
Goal: find the secret vector (or the error vector).

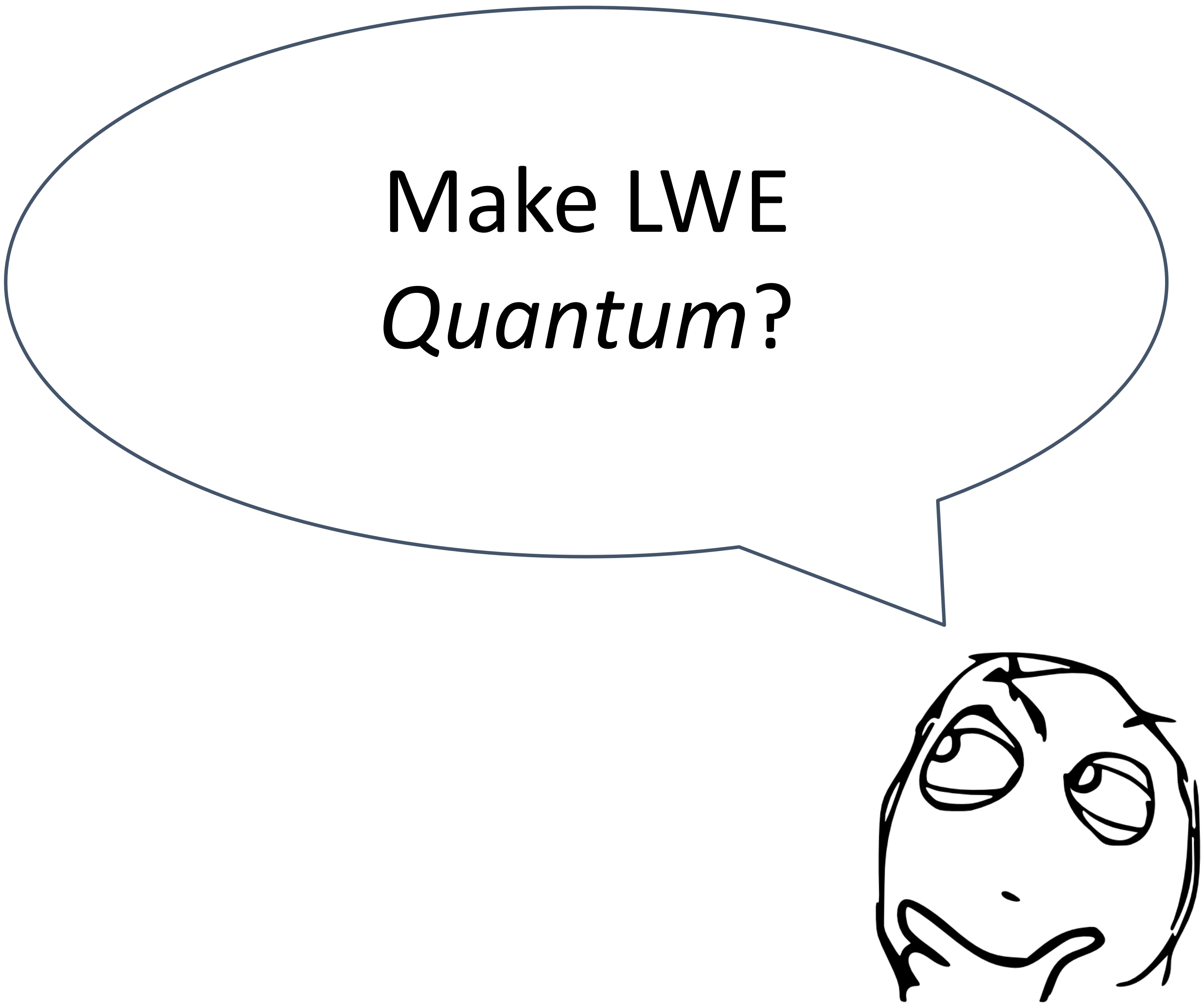
Typical parameters: $q = O(n^2)$, $m = \mathrm{poly}(n)$, $|e| < n$.

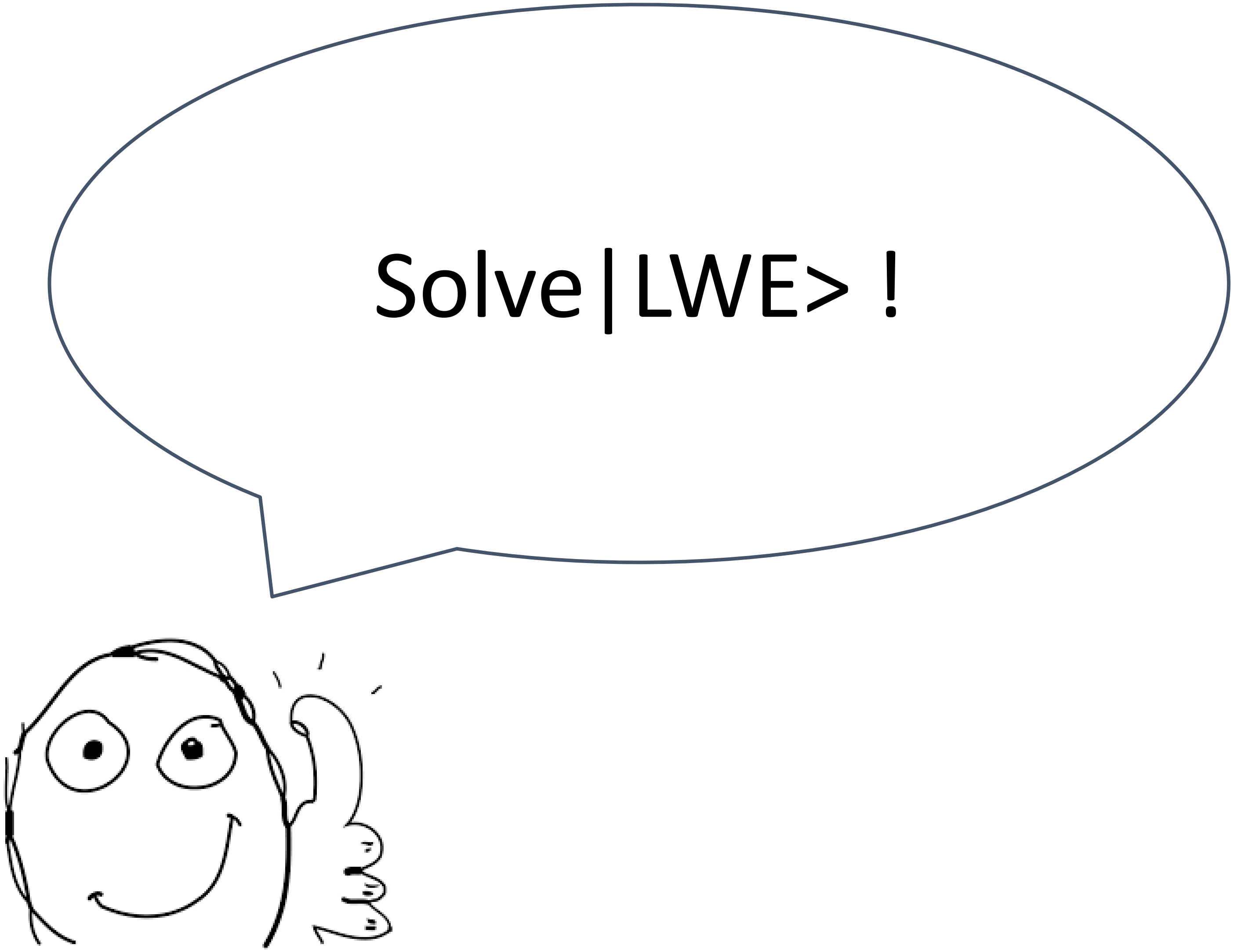
If you quantumly solve the LWE problem, you quantumly solve Approximate SIVP, SIS, EDCP problems, etc.


Approximate Shortest Vector Problem
obvious
[Ajtai 96]
[Regev 02]
Shortest Integer Solution
[Regev 05]
obvious
obvious
[Regev 02]
[SSTX 09]
(extrapolated) Dihedral Coset
Learning with Errors
[BKSW 18]
classical
A
B
quantum
A reduces to B, i.e., if there is an Alg for B, there is an Alg for A

How to use *Quantum*
to solve LWE?




Make LWE
Quantum?

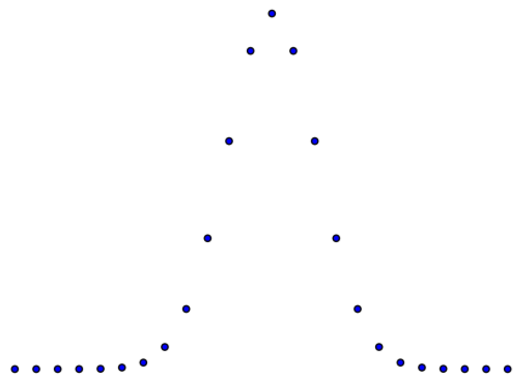

Solve|LWE> !

Learning with errors (formal)

$s = [\, s_1 , s_2 , \dots , s_n \,]$ is the secret vector.

Given samples of the form

$$a_1 ,\quad y_1 = \langle s, a_1 \rangle + e_1 \mod q$$
$$a_2 ,\quad y_2 = \langle s, a_2 \rangle + e_2 \mod q$$
$$\dots$$
$$a_m ,\quad y_m = \langle s, a_m \rangle + e_m \mod q$$



Typical error distribution: Gaussian

Goal: find the secret vector (or the error vector).

Typical parameters: $q = O(n^2)$, $m = \mathrm{poly}(n)$, $|e| < n$.

Solve |Learning with errors> (S|LWE>)

$s = [\, s_1 , s_2 , \ldots , s_n \,]$ is the secret vector.

Given quantum samples of the form

Solve |Learning with errors> (S|LWE>)

$s = [s_1 , s_2 , \ldots , s_n]$ is the secret vector.

Given quantum samples of the form

$$a_1 , \quad |y_1\rangle = \sum_{e\_1 \in [0, \ldots, q-1]} f(e_1) \; | \langle s, a_1\rangle + e_1 \bmod q \rangle$$

$$a_2 , \quad |y_2\rangle = \sum_{e\_2 \in [0, \ldots, q-1]} f(e_2) \; | \langle s, a_2\rangle + e_2 \bmod q \rangle$$

$$\ldots$$

$$a_m , \quad |y_m\rangle = \sum_{e\_m \in [0, \ldots, q-1]} f(e_m) \; | \langle s, a_m\rangle + e_m \bmod q \rangle$$

f: the error amplitude!

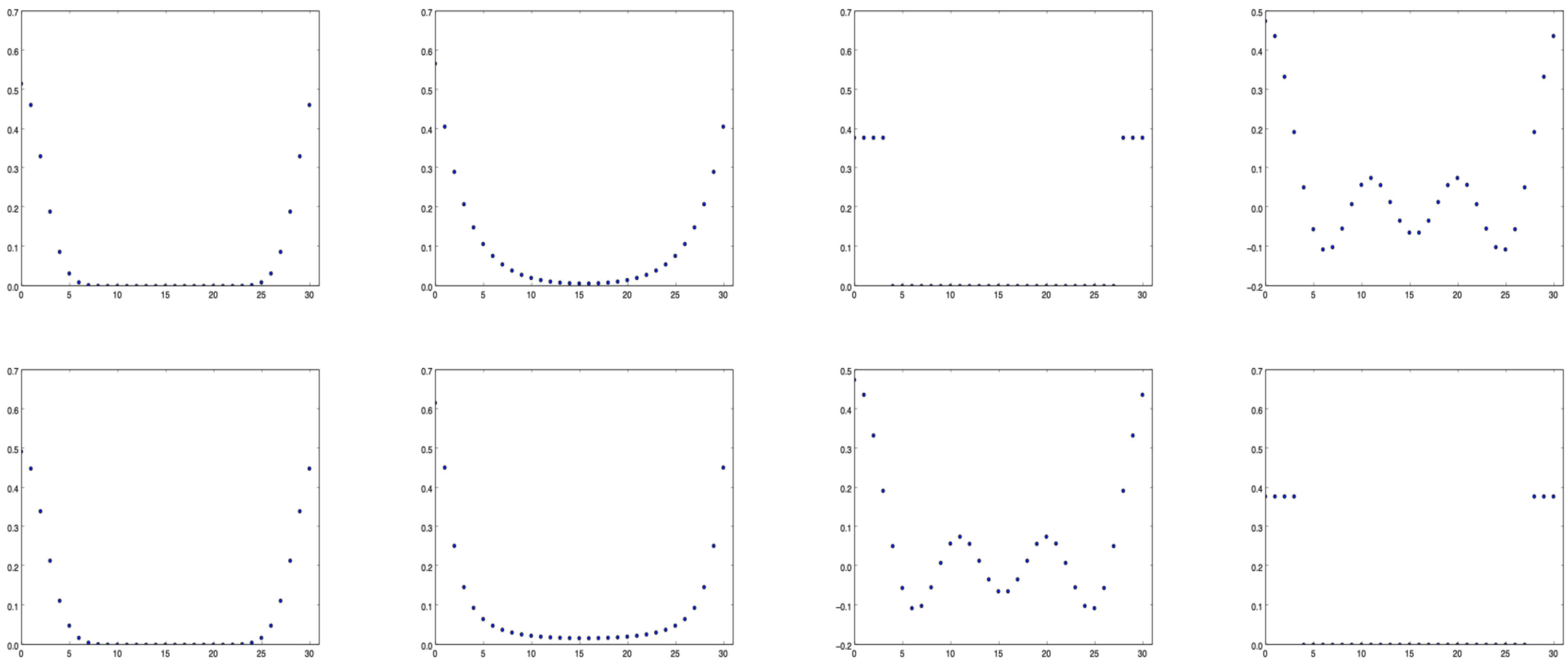
Goal: find the secret vector (or the error vector).

Reference [CLZ22].

Solve |Learning with errors> (S|LWE>)

f

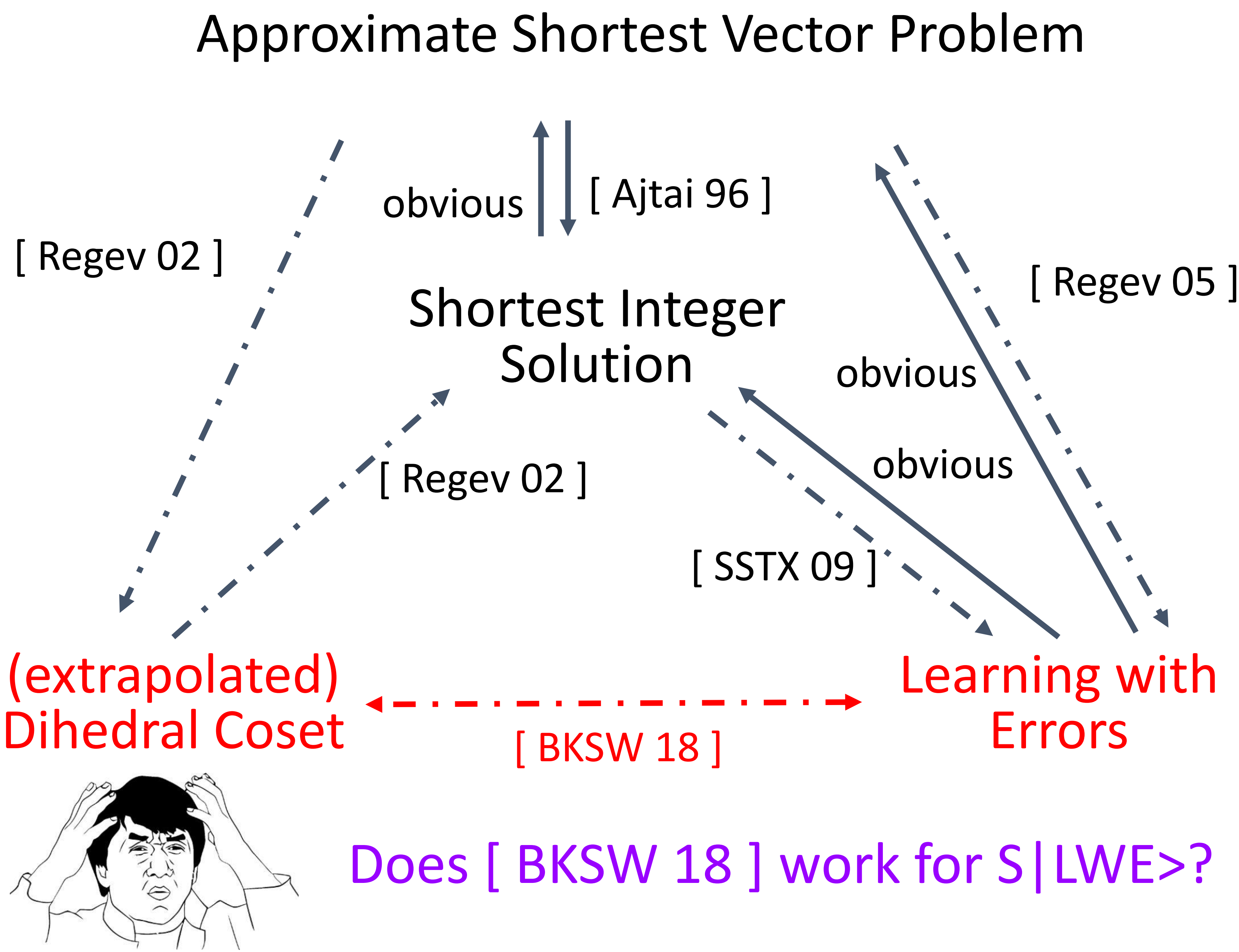
DFT(f)



Gaussian Laplacian Bounded uniform sin(x)/x

In this talk

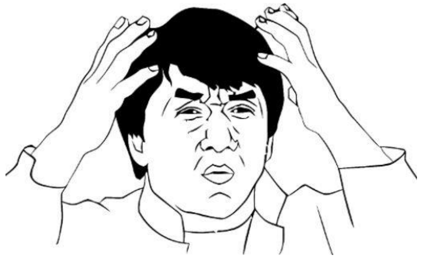
- Intro to learning with errors (LWE) and its quantum variant “Solve |LWE>” (S|LWE>).
- S|LWE> for Gaussian amplitudes: hardness and algorithm.
- S|LWE> for specific amplitudes: algorithm and application to oblivious sampling.


Approximate Shortest Vector Problem
obvious
[Ajtai 96]
[Regev 02]
[Regev 05]
Shortest Integer Solution
obvious
obvious
[Regev 02]
[SSTX 09]
(extrapolated) Dihedral Coset
Learning with Errors
[BKSW 18]
Does [BKSW 18] work for S|LWE>?

Problem: the reduction in [BKSW 18] only reduces (extrapolated) Dihedral Coset Problem to the classical LWE problem.

(extrapolated) Dihedral Coset ←- - - - - - - → Learning with Errors

[BKSW 18]



Does [BKSW 18] work for S|LWE>?

Reduction from LWE to S|LWE>

Step 1. Prepare

$\sum_{j \in [0, ..., q-1]} \rho(j) \ |j\rangle \sum_{v \in [0, ..., q-1]^n} \rho(v) \ |v\rangle \sum_{x \in [0, ..., q-1]^m} \rho(x) \ |x\rangle$

Reduction from LWE to S|LWE>

[BKSW 18] uses uniform distribution over a bounded sphere here.

Step 1. Prepare

$\sum_{j \in [0, \ldots, q-1]} \rho(j) \; |j\rangle \sum_{v \in [0, \ldots, q-1]^n} \rho(v) \; |v\rangle \sum_{x \in [0, \ldots, q-1]^m} \rho(x) \; |x\rangle$

Reduction from LWE to S|LWE>

[BKSW 18] uses uniform distribution over a bounded sphere here.

Step 1. Prepare

$$\sum_{j \in [0, \ldots, q-1]} \rho(j) \,|j\rangle \sum_{v \in [0, \ldots, q-1]^n} \rho(v) \,|v\rangle \sum_{x \in [0, \ldots, q-1]^m} \rho(x) \,|x\rangle$$

-> For classical LWE instance $y = A^T s + e$, add $A^T v - j \cdot y$ to $|x\rangle$, get

$$\sum_{v, x} \left(\sum_j \rho(j) \, \rho(x + j \cdot e) \,|j\rangle \,|v + j \cdot s\rangle \right) |\, A^T v + x \,\rangle$$

Reduction from LWE to S|LWE>

[BKSW 18] uses uniform distribution over a bounded sphere here.

Step 1. Prepare

$\sum_{j \in [0, ..., q-1]} \rho(j) \, |j\rangle \sum_{v \in [0, ..., q-1]^n} \rho(v) \, |v\rangle \sum_{x \in [0, ..., q-1]^m} \rho(x) \, |x\rangle$

-> For classical LWE instance $y = A^T s + e$, add $A^T v - j \cdot y$ to $|x\rangle$, get

$\sum_{v, x} \left(\sum_j \rho(j) \, \rho(x + j \cdot e) \, |j\rangle \, |v + j \cdot s\rangle \right) | \, A^T v + x \rangle$

-> Measure $A^T v + x$, get EDCP state with unknown center

$\sum_j \rho(j - c) \, |j\rangle \, |v + j \cdot s\rangle$

The success probability 1 – exp(-n), while in [BKSW 18] it is 1 – 1/poly(n).

Reduction from LWE to S|LWE>

Step 2.

-> Measure $A^T v + x$, get EDCP state with unknown center

$$\sum_j \rho(j - c) \ |j\rangle \ |v + j \cdot s\rangle$$

Reduction from LWE to S|LWE>

Step 2.

-> Measure $A^T v + x$, get EDCP state with unknown center

$$\sum_j \rho(j - c) \, |j\rangle \, |v + j \cdot s\rangle$$

-> QFT on 2nd register, get

$$\sum_a \sum_j e^{2\pi i \langle a,\, v + j \cdot s\rangle / q} \rho(j - c) \, |j\rangle \, |a\rangle$$

Reduction from LWE to S|LWE>

Step 2.

-> Measure A^T v + x , get EDCP state with unknown center

$$\sum_j \rho(j - c)\ |j\rangle\ |v + j \cdot s\rangle$$

-> QFT on 2^{nd} register, get

$$\sum_a \sum_j e^{2\pi i \langle a,\, v + j \cdot s\rangle / q}\ \rho(j - c)\ |j\rangle\ |a\rangle$$

-> Measure a, then QFT on 1^{st} register, get

$$\sum_e \rho(e)\ \exp(\ 2\pi i\ ce/q\ )\ |\langle s, -a\rangle + e \bmod q\rangle$$

S|LWE> state with unknown phase.

Approximate Shortest Vector Problem

We also provide another reduction from LWE to S|LWE> with unknown phase, quantizing [Regev 05]

See Appendix, or Section 6 in our full version.

[Regev 05]

Learning with Errors

The *unknown phase* turns out to be crucial because…

Subexponential time algo for S|LWE>

$s = [\, s_1 , s_2 , \dots , s_n \,]$ is the secret vector.

Given subexponential many samples of the form

$$a_j , \quad |y_j\rangle = \sum_{e\_j \,\in\, [0, \dots, q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \,\rangle$$

Our work. A subexponential time quantum algorithm for solving S|LWE> with *completely known* amplitudes.

(the amplitude f can be anything as long as DFT(f) has more than one non-negligible points, including Gaussian and Gaussian with known phase)

Subexponential time algo for S|LWE>

$s = [\, s_1 , s_2 , \ldots , s_n \,]$ is the secret vector.

Given subexponential many samples of the form

$$a_j , \quad |y_j> = \sum_{e\_j \in [0, \ldots, q-1]} f(e_j) \;\; | <s, a_j> + e_j \bmod q >$$

Idea. Apply QFT on the S|LWE> samples

$$\to \sum_k \mathrm{DFT}(f)(k)\, e^{2\pi\, i\, k<s,\, a>/q}\, |k>$$

Subexponential time algo for S|LWE>

$s = [\, s_1 , s_2 , \ldots , s_n \,]$ is the secret vector.

Given subexponential many samples of the form

$$a_j , \quad |y_j\rangle = \sum_{e\_j \in [0, \ldots, q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \rangle$$

Idea. Apply QFT on the S|LWE> samples

-> $\sum_k \mathrm{DFT}(f)(k)\, e^{2\pi i k \langle s, a \rangle / q} |k\rangle$

-> Apply quantum rejection sampling to get

$$|0\rangle + e^{2\pi i \langle s, a \rangle / q} |1\rangle$$

Subexponential time algo for S|LWE>

$s = [\, s_1, s_2, \ldots, s_n \,]$ is the secret vector.

Given subexponential many samples of the form

$$a_j, \quad |y_j\rangle = \sum_{e\_j \in [0, \ldots, q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \rangle$$

Idea. Apply QFT on the S|LWE> samples

-> $\sum_k \mathrm{DFT}(f)(k)\, e^{2\pi i k \langle s, a \rangle / q} |k\rangle$

-> Apply quantum rejection sampling to get

$$|0\rangle + e^{2\pi i \langle s, a \rangle / q} |1\rangle$$

-> Use Kuperberg sieve: given a, $|0\rangle + e^{2\pi i \langle s, a \rangle / q} |1\rangle$, find s.

(needs $2^{O(\sqrt{n \cdot \log q})}$ many samples, time $2^{O(\sqrt{n \cdot \log q})}$)

In this talk

- Intro to learning with errors (LWE) and its quantum variant “Solve |LWE>” (S|LWE>).
- S|LWE> for Gaussian amplitudes: hardness and algorithm.
- S|LWE> for specific amplitudes: algorithm and application to oblivious sampling.

Efficient algorithm for S|LWE>

$s = [s_1 , s_2 , \ldots , s_n]$ is the secret vector.

Given sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e\_j \in [0, \ldots, q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \rangle$$

Efficient algorithm for S|LWE>

$s = [\, s_1, s_2, \ldots, s_n \,]$ is the secret vector.

Given sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e\_j \in [0, \ldots, q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \rangle$$

Our work. A poly(n, log q) time quantum algorithm for solving S|LWE> using only $m = \tilde{O}(n)$ samples, for a specific amplitude f:

$$f(e) = \rho_\sigma(e) \cdot \exp(-\pi\, i\, e\^2/p)$$

Efficient algorithm for S|LWE>

$s = [\, s_1 , s_2 , \ldots , s_n \,]$ is the secret vector.

Given sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e\_j \in [0, \ldots, q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \,\rangle$$

Our work. A poly(n, log q) time quantum algorithm for solving S|LWE> using only $m = \tilde{O}(n)$ samples, for a specific amplitude f:

$$f(e) = \rho_\sigma(e) \cdot \exp(-\pi\, i\, e\hat{}2/p)$$

Gaussian of width σ,
satisfying some mild restrictions

Efficient algorithm for S|LWE>

$s = [\, s_1 , s_2 , \ldots , s_n \,]$ is the secret vector.

Given sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e\_j \,\in\, [0, \ldots, q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \, \rangle$$

Our work. A poly(n, log q) time quantum algorithm for solving S|LWE> using only $m = \tilde{O}(n)$ samples, for a specific amplitude f:

$$f(e) = \rho_\sigma(e) \cdot \exp(-\pi\, i\, e^2/p)$$

Gaussian of width σ,
satisfying some mild restrictions

Unit-length complex number,
Gaussian rotation

Efficient algorithm for S|LWE>

$s = [\, s_1 , s_2 , \dots , s_n \,]$ is the secret vector.

Given sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e\_j \in [0, \dots, q-1]} f(e_j) \mid \langle s, a_j\rangle + e_j \bmod q \,\rangle$$

Our work. A poly(n, log q) time quantum algorithm for solving S|LWE> using only $m = \tilde{O}(n)$ samples, for a specific amplitude f:

$$f(e) = \rho_\sigma(e) \cdot \exp(-\pi i\, e\text{^}2/p)$$

Gaussian of width σ,
satisfying some mild restrictions

Unit-length complex number,
Gaussian rotation

Carefully chosen p

Efficient algorithm for S|LWE>: Overview

center of $|y_j>$

Each sample of the form:

$$a_j, \quad |y_j> = \sum_{e \in [0, \dots, q-1]} f(e_j) \; |<s, a_j> + e_j \bmod q>$$

where $f(e_j) = \rho_\sigma(e_j) \cdot \exp(-\pi \, i \, e_j^2/p)$ for some number $p \mid q$.

Efficient algorithm for S|LWE>: Overview

Each sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e \in [0, \ldots, q-1]} f(e_j) \; |\langle s, a_j\rangle + e_j \bmod q \rangle$$

where $f(e_j) = \rho_\sigma(e_j) \cdot \exp(-\pi i \, e_j^2/p)$ for some number $p \mid q$.

Plan:

- extract $|y_j\rangle$'s center: $\langle s, a_j\rangle \bmod q$,
- solve $s \bmod q$ by Gaussian elimination.

Efficient algorithm for S|LWE>: Overview

Each sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e \in [0, \ldots, q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \rangle$$

where $f(e_j) = \rho_\sigma(e_j) \cdot \exp(-\pi i \, e_j^2/p)$ for some number $p \mid q$.

Plan:

- extract $|y_j\rangle$'s center: $\langle s, a_j \rangle \bmod q$,
- solve $s \bmod q$ by Gaussian elimination.

Key observation

For each $c \in [0, \ldots, p-1]$, define $|\psi_c\rangle = \sum_{e \in [0, \ldots, q-1]} f(e) \; | c + e \bmod q \rangle$.

When $q \gg p$, $\{|\psi_c\rangle\}_{c \in [0, \ldots, p-1]}$ are almost orthogonal.

Efficient algorithm for S|LWE>: Overview

Each sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e \in [0, \ldots, q-1]} f(e_j) \; |\langle s, a_j\rangle + e_j \bmod q\rangle$$

where $f(e_j) = \rho_\sigma(e_j) \cdot \exp(-\pi i \, e_j^2/p)$ for some number $p \mid q$.

Plan:

- extract $|y_j\rangle$'s center: $\langle s, a_j\rangle \bmod q$,
- solve $s \bmod q$ by Gaussian elimination.

Key observation

For each $c \in [0, \ldots, p-1]$, define $|\psi_c\rangle = \sum_{e \in [0, \ldots, q-1]} f(e) \; |c + e \bmod q\rangle$.

When $q \gg p$, $\{|\psi_c\rangle\}_{c \in [0, \ldots, p-1]}$ are almost orthogonal.

-> Measure $|y_j\rangle$ in appropriate basis extracts $\langle s, a_j\rangle \bmod p$ with high probability.

Efficient algorithm for S|LWE>: Overview

Each sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e \in [0, ..., q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \rangle$$

where $f(e_j) = \rho_\sigma(e_j) \cdot \exp(-\pi i e_j^2/p)$ for some number $p \mid q$.

Plan:

- extract $|y_j\rangle$'s center: $\langle s, a_j \rangle \bmod q$,
- solve $s \bmod q$ by Gaussian elimination.

<u>Key observation</u>

For each $c \in [0, ..., p-1]$, define $|\psi_c\rangle = \sum_{e \in [0, ..., q-1]} f(e) \; | c + e \bmod q \rangle$.
When <u>q >> p</u>, $\{|\psi_c\rangle\}_{c \in [0, ..., p-1]}$ are almost orthogonal.

-> Measure $|y_j\rangle$ in appropriate basis extracts $\langle s, a_j \rangle \bmod p$ with high probability.

-> Caveat: only works when $q \gg p$.

Efficient algorithm for S|LWE>: Overview

Each sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e \in [0, \ldots, q-1]} f(e_j) \; |\langle s, a_j\rangle + e_j \bmod q\rangle$$

where $f(e_j) = \rho_\sigma(e_j) \cdot \exp(-\pi i \, e_j^2/p)$ for some number $p \mid q$.

Plan:

- extract $|y_j\rangle$'s center: $\langle s, a_j\rangle \bmod q$,
- solve $s \bmod q$ by Gaussian elimination.

Key observation

For each $c \in [0, \ldots, p-1]$, define $|\psi_c\rangle = \sum_{e \in [0, \ldots, q-1]} f(e) \; |c + e \bmod q\rangle$.

When $q \gg p$, $\{|\psi_c\rangle\}_{c \in [0, \ldots, p-1]}$ are almost orthogonal.

-> Measure $|y_j\rangle$ in appropriate basis extracts $\langle s, a_j\rangle \bmod p$ with high probability.

-> Caveat: only works when $q \gg p$.

-> Resolved by restricting to a composite number $q = p_1 p_2 \ldots p_l$, and extracting $\langle s, a_j\rangle \bmod p_1, \ldots, \langle s, a_j\rangle \bmod p_l$ respectively.

Efficient algorithm for S|LWE>: Overview

Each sample of the form:

$$a_j, \quad |y_j\rangle = \sum_{e \in [0, \dots, q-1]} f(e_j) \; | \langle s, a_j \rangle + e_j \bmod q \rangle$$

where $f(e_j) = \rho_\sigma(e_j) \cdot \exp(-\pi i e_j^2/p)$ for some number $p \mid q$.

Plan:

- extract $|y_j\rangle$'s center: $\langle s, a_j \rangle \bmod q$,
- solve $s \bmod q$ by Gaussian elimination.

<u>Key observation</u>

For each $c \in [0, \dots, p-1]$, define $|\psi_c\rangle = \sum_{e \in [0, \dots, q-1]} f(e) \; | c + e \bmod q \rangle$.

When <u>$q \gg p$</u>, $\{|\psi_c\rangle\}_{c \in [0, \dots, p-1]}$ are almost orthogonal.

-> Measure $|y_j\rangle$ in appropriate basis extracts $\langle s, a_j \rangle \bmod p$ with high probability.

-> Caveat: only works when $q \gg p$.

-> Resolved by restricting to a composite number $q = p_1 p_2 \dots p_l$, and extracting $\langle s, a_j \rangle \bmod p_1$, … , $\langle s, a_j \rangle \bmod p_l$ respectively.

-> Modulus Switching technique in [BLP+13] can switch to any $q' < q$.

Oblivious LWE Sampling from S|LWE>

[Reg09, CLZ22, DFS24]

Oblivious LWE Sampling: Sample (A, sA + e) without knowing s.

Oblivious LWE Sampling from S|LWE>

[Reg09, CLZ22, DFS24]

Oblivious LWE Sampling: Sample (A, sA + e) without knowing s.

- We don't know any efficient classical oblivious LWE sampler.
- [DFS24] Efficient quantum oblivious LWE sampler.

Oblivious LWE Sampling from S|LWE>

[Reg09, CLZ22, DFS24]

Oblivious LWE Sampling: Sample (A, sA + e) without knowing s.

- We don’t know any efficient classical oblivious LWE sampler.
- [DFS24] Efficient quantum oblivious LWE sampler.

S|LWE> algorithm ⟶ Oblivious LWE Sampler

Oblivious LWE Sampling from S|LWE>

[Reg09, CLZ22, DFS24]

Oblivious LWE Sampling: Sample (A, sA + e) without knowing s.

- We don’t know any efficient classical oblivious LWE sampler.
- [DFS24] Efficient quantum oblivious LWE sampler.

S|LWE> algorithm ⟶ Oblivious LWE Sampler

$$\sum_s |s\rangle \otimes \sum_e f(e)\, |e\rangle$$

Oblivious LWE Sampling from S|LWE>

[Reg09, CLZ22, DFS24]

Oblivious LWE Sampling: Sample (A, sA + e) without knowing s.

- We don't know any efficient classical oblivious LWE sampler.
- [DFS24] Efficient quantum oblivious LWE sampler.

S|LWE> algorithm → Oblivious LWE Sampler

$$\sum_s |s\rangle \otimes \sum_e f(e)\, |e\rangle$$

$$\Downarrow$$

$$\sum_s |s\rangle \otimes \sum_e f(e)\, |sA + e\rangle$$

Oblivious LWE Sampling from S|LWE>

[Reg09, CLZ22, DFS24]

Oblivious LWE Sampling: Sample (A, sA + e) without knowing s.

- We don't know any efficient classical oblivious LWE sampler.
- [DFS24] Efficient quantum oblivious LWE sampler.

S|LWE> algorithm ⟶ Oblivious LWE Sampler

$$\sum_s |s\rangle \otimes \sum_e f(e)\, |e\rangle$$

⬇

$$\sum_s |s\rangle \otimes \sum_e f(e)\, |sA + e\rangle$$

⬇ Solve s from $\sum_e f(e)\, |sA + e\rangle$ via S|LWE> algorithm

$$\sum_s |0\rangle \otimes \sum_e f(e)\, |sA + e\rangle$$

Oblivious LWE Sampling from S|LWE>

[Reg09, CLZ22, DFS24]

Oblivious LWE Sampling: Sample (A, sA + e) without knowing s.

- We don't know any efficient classical oblivious LWE sampler.
- [DFS24] Efficient quantum oblivious LWE sampler.

S|LWE> algorithm → Oblivious LWE Sampler

$\sum_s |s\rangle \otimes \sum_e f(e)\, |e\rangle$

↓

$\sum_s |s\rangle \otimes \sum_e f(e)\, |sA + e\rangle$

↓ Solve s from $\sum_e f(e)\, |sA + e\rangle$ via S|LWE> algorithm

$\sum_s |0\rangle \otimes \sum_e f(e)\, |sA + e\rangle$

↓

Measure the second register to get sA + e, error distribution $e \sim |f|^2$.

Improved Oblivious LWE Sampler

Oblivious LWE Sampling: Sample (A, sA + e) without knowing s.

$s = [s_1, \dots, s_n], A = [a_1, \dots, a_m], e = [e_1, \dots, e_m].$

Improved S|LWE> ⟶ Improved Oblivious LWE Sampler

	Run time	Sample Complexity (m)
[DFS24]	poly(n, log q)	$\tilde{O}(n\sigma)$
Ours	poly(n, log q)	$\tilde{O}(n)$

LWE error:
Gaussian of width σ

Takeaways

- $2^{O(\sqrt{n \cdot \log q})}$-time **S|LWE> algorithm** for known amplitudes with >1 non-negligible point in DFT.
  - When q is a power-of-2, [BJKNY25] gives $2^{O(\log n \cdot \log q)}$-time algorithm.
- $\mathrm{poly}(n, \log q)$-time **S|LWE> algorithm** for a specific complex Gaussian amplitude, using Õ(n) samples.
- S|LWE> (Gaussian amplitude with a small unknown phase) is **as hard as LWE**.

Thanks for listening!

Questions are welcome!