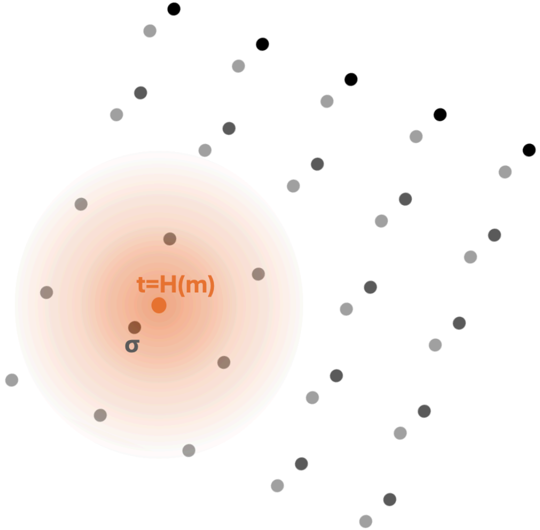
IMPERIAL

# On Gaussian Sampling for q-ary Lattices and Linear Codes with Lee Weight

Maiara Bollauf, **Maja Lie**, and Cong Ling
August 20, Crypto 2025

# In Cryptography...

- Protocols: digital signature schemes, encryption, etc...
- Reductions (worst-to-average case)
- SVP/CVP solvers

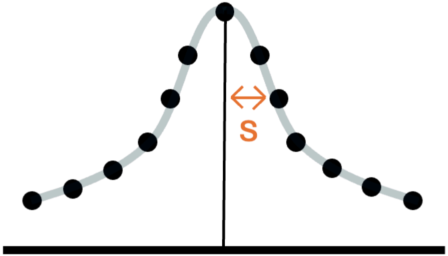


# What is lattice Gaussian sampling?

The Gaussian function is given by $\rho_s(\mathbf{x}) = e^{-\pi\|\mathbf{x}\|^2/s^2}$.
Over a discrete set S, we get $\rho_s(S) = \sum_{\mathbf{x}\in S} \rho_s(\mathbf{x})$.

## Discrete Gaussian distribution:

$$\mathcal{D}_{\Lambda+\mathbf{t},s}(\mathbf{y}) \triangleq \frac{\rho_s(\mathbf{y})}{\rho_s(\Lambda+\mathbf{t})}$$

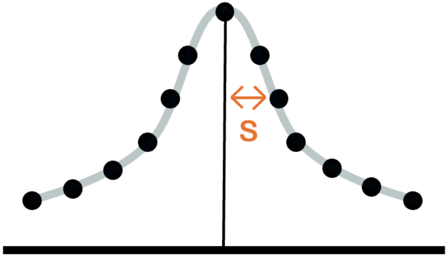


# What is lattice Gaussian sampling?

The Gaussian function is given by $\rho_s(\mathbf{x}) = e^{-\pi\|\mathbf{x}\|^2/s^2}$.
Over a discrete set S, we get $\rho_s(S) = \sum_{\mathbf{x}\in S} \rho_s(\mathbf{x})$.

Discrete Gaussian distribution:

$$\mathcal{D}_{\Lambda+\mathbf{t},s}(\mathbf{y}) \triangleq \frac{\rho_s(\mathbf{y})}{\rho_s(\Lambda+\mathbf{t})}$$



Two main goals in cryptography:

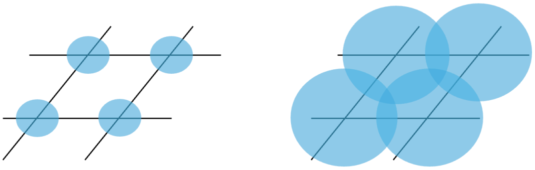
1. Security (secret information stays secret)
2. Efficiency (are the methods actually usable?)

Two main goals in cryptography:

1. Security (secret information stays secret)
2. Efficiency (are the methods actually usable?)

## Smoothing parameter

Minimum amount of noise that when added to the lattice makes the distribution uniform

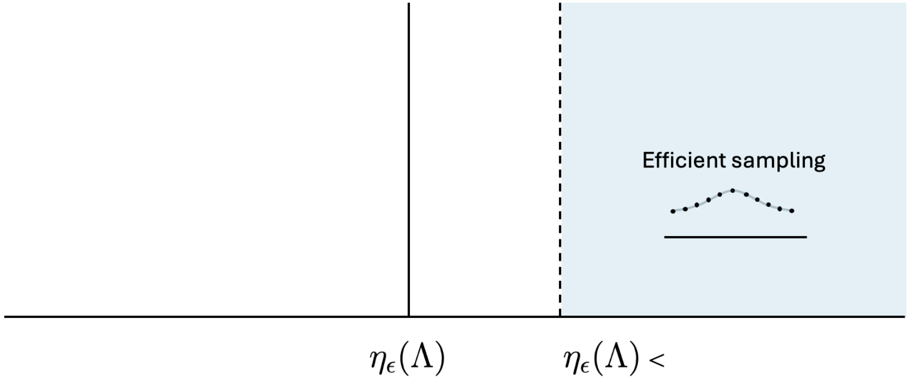


Two main goals in cryptography:

1. **Security** (secret information stays secret)
2. **Efficiency** (are the methods actually usable?)

Smoothing parameter is $\eta_\epsilon(\Lambda)$ such that

$$\Theta_{\Lambda^*}(\mathrm{i}\eta_\epsilon(\Lambda)^2) - 1 = \epsilon$$

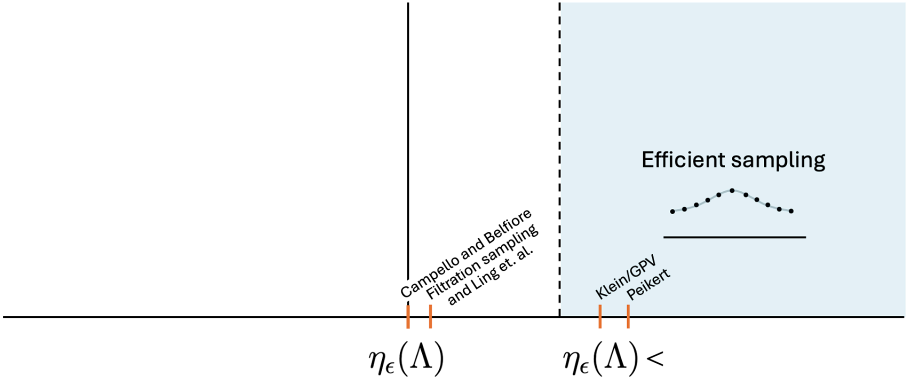


Two main goals in cryptography:

1. Security (secret information stays secret)
2. Efficiency (are the methods actually usable?)

Smoothing parameter is $\eta_\epsilon(\Lambda)$ such that

$$\Theta_{\Lambda^*}(\mathrm{i}\eta_\epsilon(\Lambda)^2) - 1 = \epsilon$$

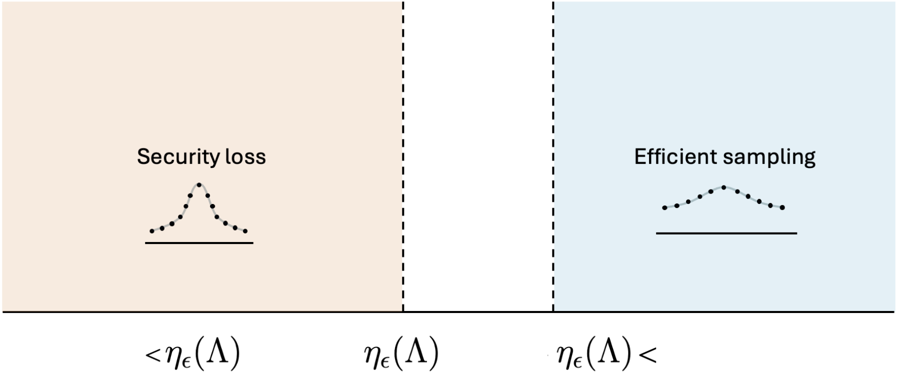


Two main goals in cryptography:

1. **Security** (secret information stays secret)
2. **Efficiency** (are the methods actually usable?)

Smoothing parameter is $\eta_\epsilon(\Lambda)$ such that

$$\Theta_{\Lambda^*}(i\eta_\epsilon(\Lambda)^2) - 1 = \epsilon$$

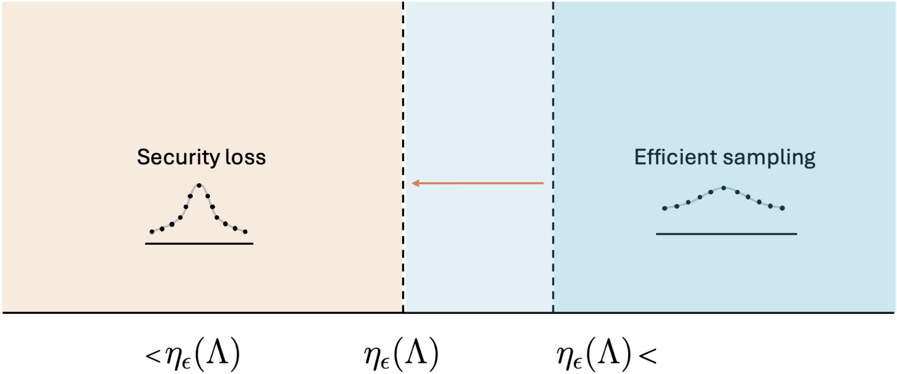


Two main goals in cryptography:

1. **Security** (secret information stays secret)
2. **Efficiency** (are the methods actually usable?)

Smoothing parameter is $\eta_\epsilon(\Lambda)$ such that

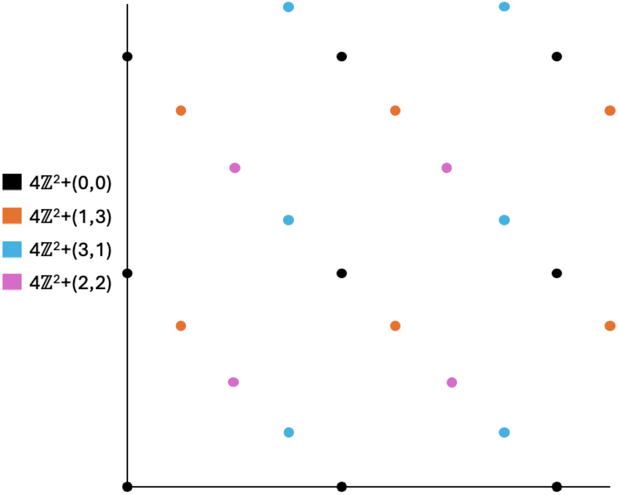
$$\Theta_{\Lambda^*}(\mathrm{i}\eta_\epsilon(\Lambda)^2) - 1 = \epsilon$$



A q-ary lattice $\Lambda$ is such that

$$q\mathbb{Z}^n \subseteq \Lambda \subseteq \mathbb{Z}^n, \qquad q \in \mathbb{N}.$$

One-to-one with linear codes over $\mathbb{Z}_q$

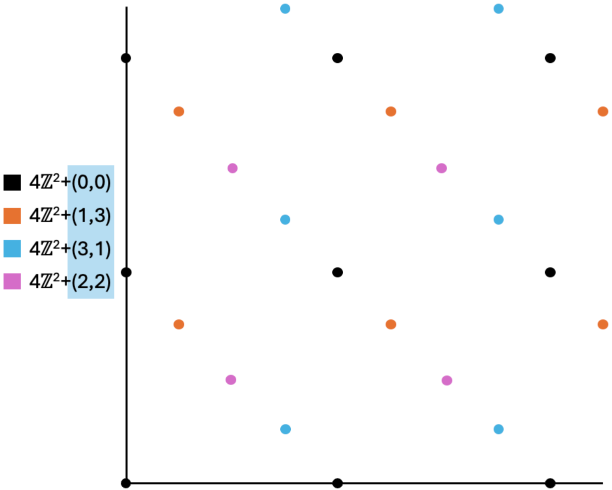


A q-ary lattice $\Lambda$ is such that

$$q\mathbb{Z}^n \subseteq \Lambda \subseteq \mathbb{Z}^n, \qquad q \in \mathbb{N}.$$

One-to-one with linear codes over
$\mathbb{Z}_q \to q\mathbb{Z}^n + \mathcal{C}$

$$\mathcal{C} = \{(0,0), (1,3), (3,1), (2,2)\}$$



## Theta series of a lattice

$$\Theta_\Lambda(z) = \sum_{\mathbf{x}\in\Lambda} e^{\pi i z\|\mathbf{x}\|^2}, \qquad \mathrm{Im}(z) > 0$$

## Weight enumerator of a code

$$\mathrm{swe}_\mathcal{C}(x_0, \dots, x_\ell) = \sum_{\mathbf{c}\in\mathcal{C}} x_0^{n_0(\mathbf{c})} \dots x_\ell^{n_\ell(\mathbf{c})}$$

where $\ell = \lceil q/2 \rceil$.

## Theta series of a lattice

$$\Theta_\Lambda(z) = \sum_{\mathbf{x}\in\Lambda} e^{\pi i z \|\mathbf{x}\|^2}, \qquad \mathrm{Im}(z) > 0$$

## Weight enumerator of a code

$$\mathrm{swe}_{\mathcal{C}}(x_0, \ldots, x_\ell) = \sum_{\mathbf{c}\in\mathcal{C}} x_0^{n_0(\mathbf{c})} \ldots x_\ell^{n_\ell(\mathbf{c})}$$

where $\ell = \lceil q/2 \rceil$.

Lee weight of $c \in \mathbb{Z}_q$ is $w_{\text{Lee}}(c) = \min\{c, q-c\}$

(q=4)

$$\mathbf{c} = (1, 1, 2, 0, 3, 2, 1, 2)$$

## Theta series of a lattice

$$\Theta_\Lambda(z) = \sum_{\mathbf{x}\in\Lambda} e^{\pi i z \|\mathbf{x}\|^2}, \qquad \mathrm{Im}(z) > 0$$

## Weight enumerator of a code

$$\mathrm{swe}_{\mathcal{C}}(x_0, \ldots, x_\ell) = \sum_{\mathbf{c}\in\mathcal{C}} x_0^{n_0(\mathbf{c})} \ldots x_\ell^{n_\ell(\mathbf{c})}$$

where $\ell = \lceil q/2 \rceil$.

Lee weight of $c \in \mathbb{Z}_q$ is $w_{\text{Lee}}(c) = \min\{c, q-c\}$

(q=4)

$$\mathbf{c} = (1, 1, 2, 0, 3, 2, 1, 2)$$

Lee Weight Profile: Tracks the Lee weights of each coordinate of the codeword

$$[n_0(\mathbf{c}), n_1(\mathbf{c}), \ldots, n_\ell(\mathbf{c})], \quad \ell = \lceil q/2 \rceil$$

where $n_w(\mathbf{c}) = \#\{i \colon w_{\text{Lee}}(c_i) = w\}$, i.e., the number of coordinates of $\mathbf{c}$ that are $\pm w$.

## How does the theta series help with sampling?

Recall:

$$\rho_s(\Lambda + \mathbf{t}) = \sum_{\mathbf{x} \in \Lambda + \mathbf{t}} e^{-\pi \|\mathbf{x}\|^2 / s^2} = \sum_{\mathbf{x} \in \Lambda + \mathbf{t}} e^{\pi i z \|\mathbf{x}\|^2} = \Theta_{\Lambda + \mathbf{t}}(z)$$

⋆ Set $z = i/s^2$

# Tool 1: Theta Series

We can show that for a Construction A lattice with $q \geq 2$ and some shift $\mathbf{t} \in \mathbb{Z}_q^n$,

$$\Theta_{\Lambda_A(\mathcal{C})+\mathbf{t}}(z) = \mathrm{swe}_{\mathcal{C}+\mathbf{t}}(\Theta_{\mathbb{Z}}(q^2 z), \ldots, \Theta_{\mathbb{Z}+\frac{\ell}{q}}(q^2 z)).$$

## Lemma (Key observation)

Sample q-ary lattice $\Lambda_A + \mathbf{t}$ ⟷ Sample a codeword in $\mathcal{C} + \mathbf{t}$ with respect to Lee weight profiles

Consider a vector in the coset $4\mathbb{Z}^8 + (1, 1, 2, 0, 3, 1, 2)$.

$$4z_j + \nu_j = \begin{cases} 4z_j, & \text{if } \nu_j = 0 \\ 4z_j + \textcircled{1}, & \text{if } \nu_j = 1 \\ 4z_j + 2, & \text{if } \nu_j = 2 \\ 4z_j + \textcircled{3}, & \text{if } \nu_j = 3. \end{cases}$$

Derive a corresponding theta series for each one-dimensional lattice coset $4\mathbb{Z} + \nu_j$. Notice

$$\Theta_{q\mathbb{Z}+j}(z) = \Theta_{q\mathbb{Z}+q-j}(z), \quad j = 1, 2, \ldots, q - 1.$$

$$4\mathbb{Z}^8 + (1, 1, 2, 0, 3, 1, 2) \sim 4\mathbb{Z} \oplus (4\mathbb{Z}^4 + (1, 1, 1, 1)) \oplus (4\mathbb{Z}^2 + (2, 2))$$

$$\downarrow \qquad\qquad \downarrow \qquad\qquad \downarrow$$

$$\Theta_{4\mathbb{Z}}(z) \quad \times \quad \Theta_{4\mathbb{Z}+1}(z)^4 \quad \times \quad \Theta_{4\mathbb{Z}+2}(z)^2$$

# Tool 2: Coset Decomposition

$$\Lambda = \bigcup_{\boldsymbol{\nu} \in \mathcal{C} + \mathbf{t}} \Lambda' + \boldsymbol{\nu}$$

Coset decomposition

1. Sample $\boldsymbol{\nu} \in \mathcal{C} + \mathbf{t}$ with probability $\mathcal{D}_{\Lambda + \mathbf{t}, s}(\Lambda' + \boldsymbol{\nu})$,
2. Sample a lattice vector $\mathbf{x}' \in \Lambda'$ with probability $\mathcal{D}_{\Lambda' + \boldsymbol{\nu}, s}(\mathbf{x}' + \boldsymbol{\nu})$

# Algorithms for sampling
## Construction A for $q \geq 2$

Set $\Lambda' := q\mathbb{Z}^n$ and sample a representative from the shifted code $\mathcal{C} + \mathbf{t}$.

1. Sample a coset representative $\boldsymbol{\nu}$ from $\mathcal{C} + \mathbf{t}$ with probability depending only on the Lee weight profile.

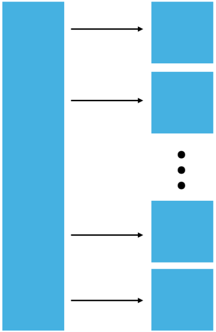
# Algorithms for sampling
## Construction A for $q \geq 2$

Set $\Lambda' := q\mathbb{Z}^n$ and sample a representative from the shifted code $\mathcal{C} + \mathbf{t}$.

1. Sample a coset representative $\boldsymbol{\nu}$ from $\mathcal{C} + \mathbf{t}$ with probability depending only on the Lee weight profile.

Recall that $\oplus_i q z_i + \nu_i \simeq q\mathbf{z} + \boldsymbol{\nu}$.

2. Apply n $\mathbb{Z}$-samplers, i.e. sample $\mathbb{Z} + \nu_i/q$ many times.



# Algorithms for sampling
## Construction A for q $\geq 2$

Set $\Lambda' := q\mathbb{Z}^n$ and sample a representative from the shifted code $\mathcal{C} + \mathbf{t}$.

1. Sample a coset representative $\boldsymbol{\nu}$ from $\mathcal{C} + \mathbf{t}$ with probability depending only on the Lee weight profile.

Recall that $\oplus_i q z_i + \nu_i \simeq q\mathbf{z} + \boldsymbol{\nu}$.

2. Apply n $\mathbb{Z}$-samplers, i.e. sample $\mathbb{Z} + \nu_i/q$ many times.
3. Multiply by q.

## Code symmetries

- For the binary case $q = 2$, we can utilise the fact that if a code contains the $\mathbf{1}$ word, then it is symmetric
  - Sample/store half of the codewords

### Example ($E_8$)

$E_8 = 2\mathbb{Z}^8 + \mathrm{RM}(1,3)$

$$W_{\mathrm{RM}(1,3)}(x,y) = x^8 + 14x^4y^4 + y^8$$

- $y^8 \implies \mathbf{1} \in \mathrm{RM}(1,3)$
- Only need 7 codewords to recover the entire code

## Code symmetries

- For the binary case $q = 2$, we can utilise the fact that if a code contains the $\mathbf{1}$ word, then it is symmetric
  - Sample/store half of the codewords

### Example ($E_8$)

$E_8 = 2\mathbb{Z}^8 + \mathrm{RM}(1,3)$

$$W_{\mathrm{RM}(1,3)}(x,y) = x^8 + 14x^4y^4 + y^8$$

- $y^8 \implies \mathbf{1} \in \mathrm{RM}(1,3)$
- Only need 7 codewords to recover the entire code

## Code structure

- Concentration of codewords of certain weights
  - Rejection sampling with few iterations
- Predictable codeword structure

### Example ($D_n, n \geq 1$)

$D_n = 2\mathbb{Z}^n + \mathcal{P}_n$ where $\mathcal{P}_n$ is the even weight code

- If n is even, then $\mathcal{P}_n$ is symmetric
- k-out-of-n choosing procedure samples a codeword of weight k

## Multilevel sampling

$$2^L \mathbb{Z}^n + \mathcal{C}$$
$$\downarrow$$
$$\mathcal{C} \triangleq 2^{L-1}\mathcal{C}_L + \ldots + 2\mathcal{C}_2 + \mathcal{C}_1$$

$$2^L \mathbb{Z}^n + 2^{L-1}\mathcal{C}_L + 2^{L-2}\mathcal{C}_{L-1} + \ldots + 2\mathcal{C}_2 + \mathcal{C}_1$$

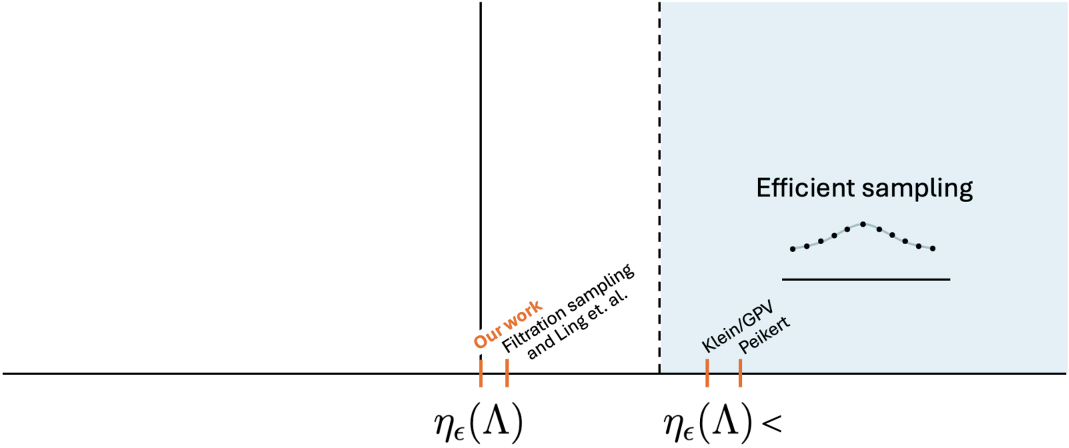
$$2^L \mathbb{Z}^n + 2^{L-1}\mathbf{c}_L + 2^{L-2}\mathcal{C}_{L-1} + \ldots + 2\mathcal{C}_2 + \mathcal{C}_1$$

$$\vdots$$

$$2^L \mathbb{Z}^n + 2^{L-1}\mathbf{c}_L + 2^{L-2}\mathbf{c}_{L-1} + \ldots + 2\mathbf{c}_2 + \mathbf{c}_1$$

## Schur product

- Compute the Schur product of codewords to get number of positions of 1's in each codeword
- Together with the Hamming weights, we can solve a system of linear equations for the $n_j(\mathbf{c})$


Efficient sampling
Our work
Filtration sampling
and Ling et. al.
Klein/GPV
Peikert
$\eta_\epsilon(\Lambda)$
$\eta_\epsilon(\Lambda) <$

| Lattice | Speed-up (Simulation) | Sampling width (Our work) | Sampling width ([EWY23]) |
|---|---|---|---|
| $A_2$ | $32\times$ | $= \eta_\epsilon(A_2)$ | $\approx \eta_\epsilon(A_2)$ |
| $E_8$ | $25\times$ | $= \eta_\epsilon(E_8)$ | $\approx \eta_\epsilon(E_8)$ |
| $D_n$ | $2\times$ | $= \eta_\epsilon(D_n)$ | $\approx \eta_\epsilon(D_n)$ |
| $BW_{16}$ | $9.5\times$ | $= \eta_\epsilon(BW_{16})$ | $> \eta_\epsilon(BW_{16})$ |

Table: Results for 100,000 samples with $\epsilon = 2^{-36}$ and shift $\mathbf{t} = \mathbf{0}$.

We compared the efficiency of our algorithms with the work done in [EWY23][1].

[1]Espitau, T., Wallet, A., Yu, Y.: On Gaussian sampling, smoothing parameter and application to lattice signatures. In: Guo, J., Steinfeld, R. (eds.) Advances in Cryptology – ASIACRYPT 2023. pp. 65–97. Springer Nature, Singapore (2023)

IMPERIAL

# Thank you.
# Questions?

On Gaussian Sampling for q-ary Lattices and Linear Codes with Lee Weight
August 20, Crypto 2025