Asiacrypt 2023

ANTRAG: Annular NTRU Trapdoor Generation

Making Mitaka as secure as Falcon

Thomas Espitau, **Thi Thu Quyen Nguyen**, Chao Sun,

Mehdi Tibouchi, Alexandre Wallet

NIST
National Institute of Standards and Technology
Center of Excellence

Post-quantum Hash-and-Sign over lattices

Falcon (*NIST 2017*) 

Post-quantum Hash-and-Sign over lattices

Falcon (*NIST 2017*) 

- Fast
- Short signature
- Security NIST I,V

Post-quantum Hash-and-Sign over lattices

Falcon (*NIST 2017*) 

- Restricted parameter choices
- Hard implementation
- Fast
- Short signature
- Security NIST I,V

Post-quantum Hash-and-Sign over lattices

Falcon (*NIST 2017*)

- Restricted parameter choices
- Hard implementation
- Fast
- Short signature
- Security NIST I,V

Mitaka (*Eurocrypt 2022*)

- More parameter choices
- Simpler implementation
- Fast

Post-quantum Hash-and-Sign over lattices

Falcon (*NIST 2017*)

- Restricted parameter choices
- Hard implementation
- Fast
- Short signature
- Security NIST I,V

Mitaka (*Eurocrypt 2022*)

- More parameter choices
- Simpler implementation
- Fast
- Signature 15% larger
- Lower security

Post-quantum Hash-and-Sign over lattices

Falcon (*NIST 2017*)

- Restricted parameter choices
- Hard implementation
- Fast
- Short signature
- Security NIST I,V

Mitaka (*Eurocrypt 2022*)

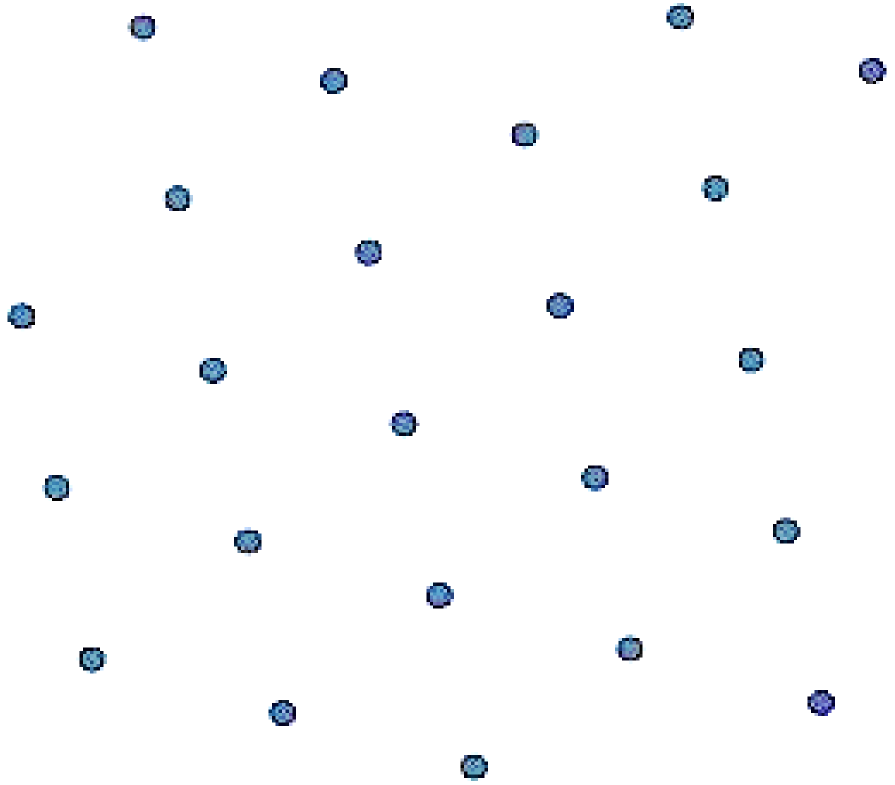
- More parameter choices
- Simpler implementation
- Fast
- Signature 15% larger
- Lower security

ANTRAG: Make Mitaka as secure as Falcon

Hash-and-Sign over lattices

Hash-and-Sign over lattices

Sign($\mathbf{m}$, $\mathbf{sk}_{\Lambda}$, γ):

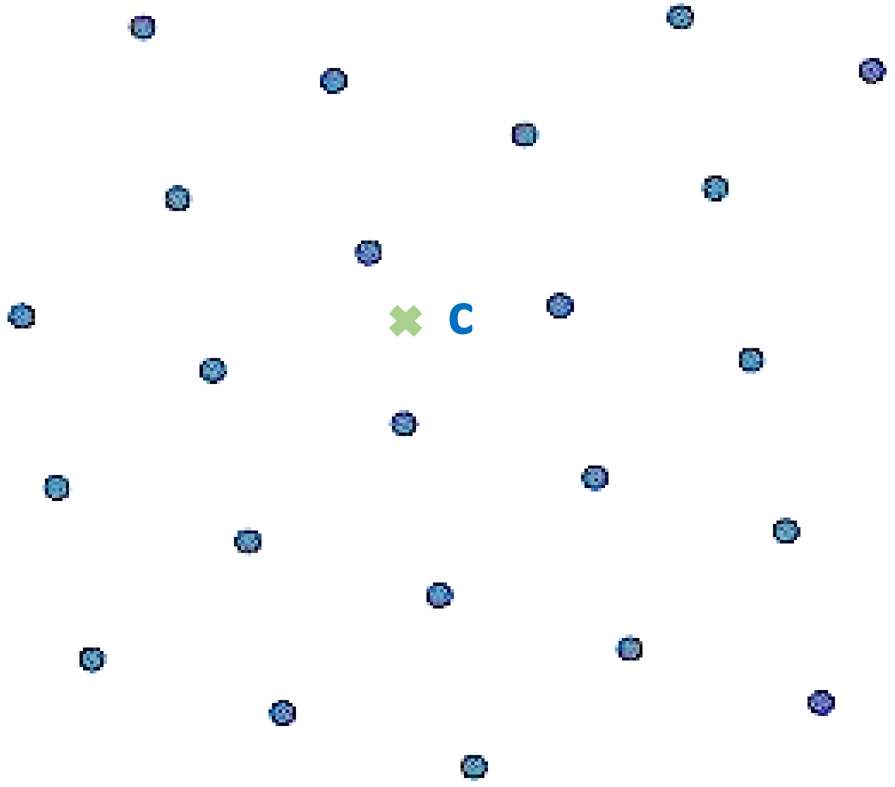


$\Lambda \subset \mathbb{R}^d$

Hash-and-Sign over lattices

Sign$(\mathbf{m}, \mathbf{sk}_\Lambda, \gamma)$:

- $\mathbf{c} := H(\mathbf{m})$

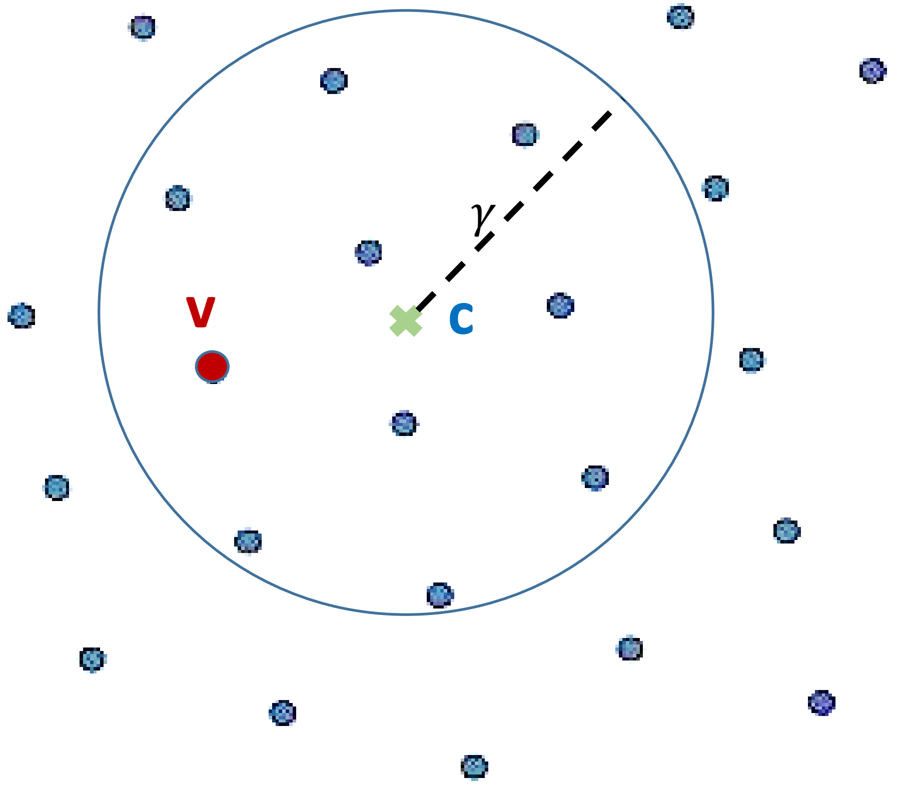


$\Lambda \subset \mathbb{R}^d$

Hash-and-Sign over lattices

Sign$(\mathbf{m}, \mathbf{sk}_\Lambda, \gamma)$:

- $\mathbf{c} \coloneqq H(\mathbf{m})$
- $\mathbf{v} \leftarrow \text{CloseVector}_{\Lambda,\gamma}(\mathbf{c})$

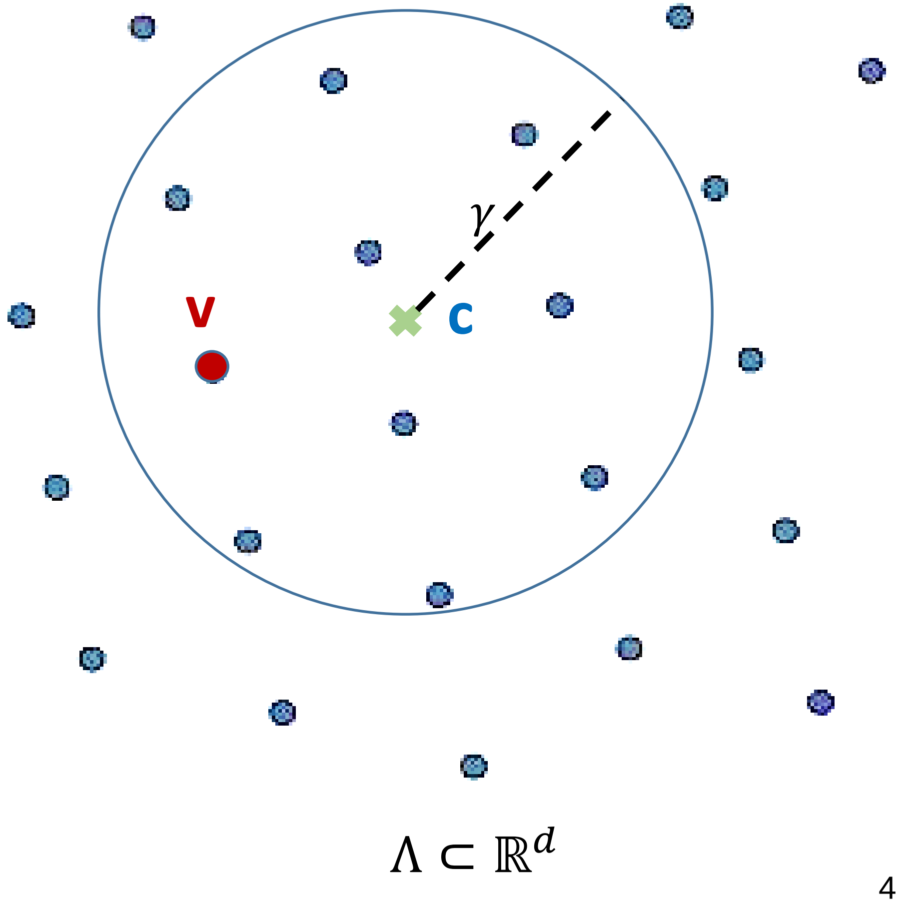


$\Lambda \subset \mathbb{R}^d$

Hash-and-Sign over lattices

Sign($\mathbf{m}, \mathbf{sk}_\Lambda, \gamma$):

- $\mathbf{c} \coloneqq H(\mathbf{m})$
- $\mathbf{v} \leftarrow \mathrm{CloseVector}_{\Lambda,\gamma}(\mathbf{c})$
- $\mathbf{s} \coloneqq \mathbf{c} - \mathbf{v}$
- Return $\mathbf{sig} \coloneqq \mathbf{s}$.



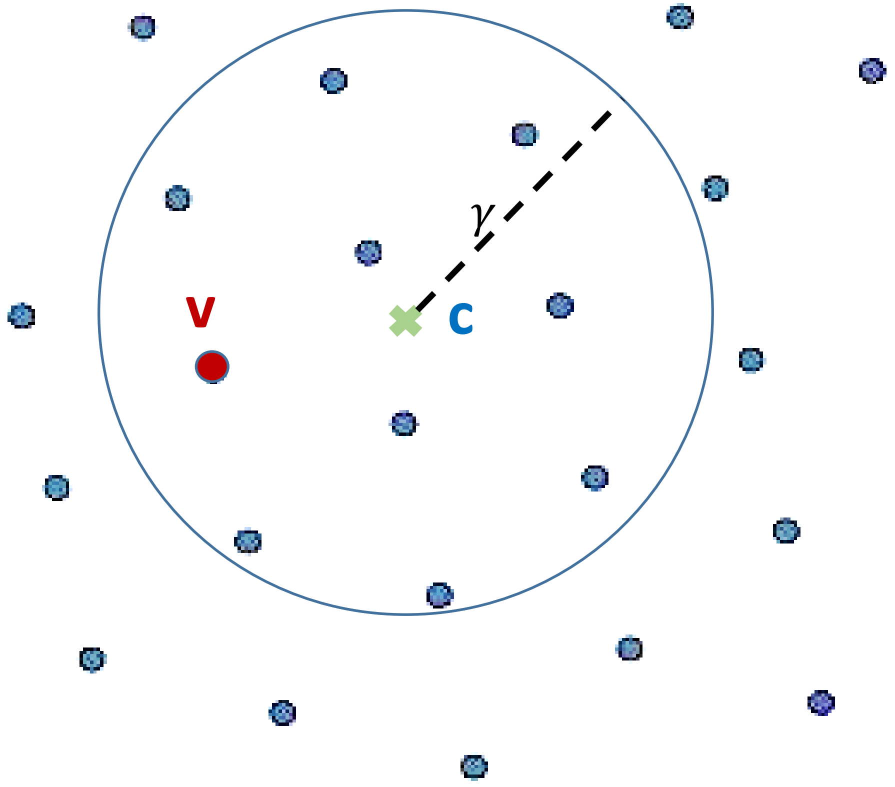
Hash-and-Sign over lattices

Sign$(\mathbf{m}, \mathbf{sk}_\Lambda, \gamma)$:

- $\mathbf{c} \coloneqq H(\mathbf{m})$
- $\mathbf{v} \leftarrow \mathrm{CloseVector}_{\Lambda,\gamma}(\mathbf{c})$
- $\mathbf{s} \coloneqq \mathbf{c} - \mathbf{v}$
- Return $\mathbf{sig} \coloneqq \mathbf{s}$.

Verify$(\mathbf{m}, \mathbf{sig}, \mathbf{pk}_\Lambda, \gamma)$:

- Accept iff $\|\mathbf{sig}\| \leq \gamma$ and $H(\mathbf{m}) - \mathbf{sig} \in \Lambda$.



$\Lambda \subset \mathbb{R}^d$

Hash-and-Sign over lattices

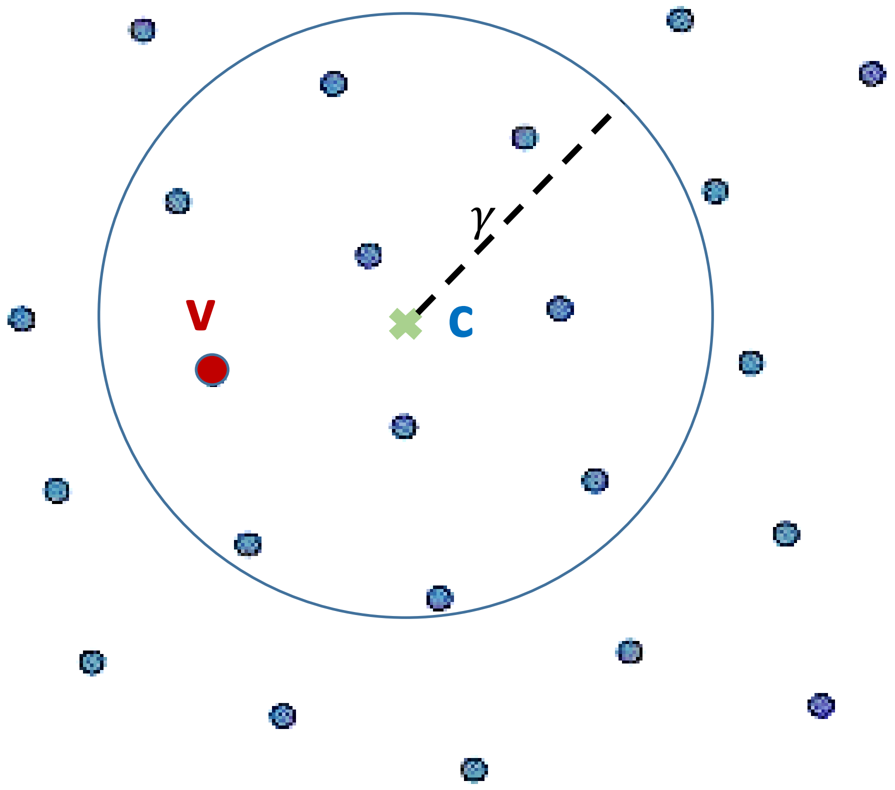
Sign$(\mathbf{m}, \mathbf{sk}_\Lambda, \gamma)$:

- $\mathbf{c} := H(\mathbf{m})$
- $\mathbf{v} \leftarrow \text{DiscreteGaussianSampler}(\mathbf{sk}_\Lambda, \mathbf{c})$
- $\mathbf{s} := \mathbf{c} - \mathbf{v}$
- Return $\mathbf{sig} := \mathbf{s}$.

Verify$(\mathbf{m}, \mathbf{sig}, \mathbf{pk}_\Lambda, \gamma)$:

- Accept iff $\|\mathbf{sig}\| \leq \gamma$ and $H(\mathbf{m}) - \mathbf{sig} \in \Lambda$.

CVP_γ

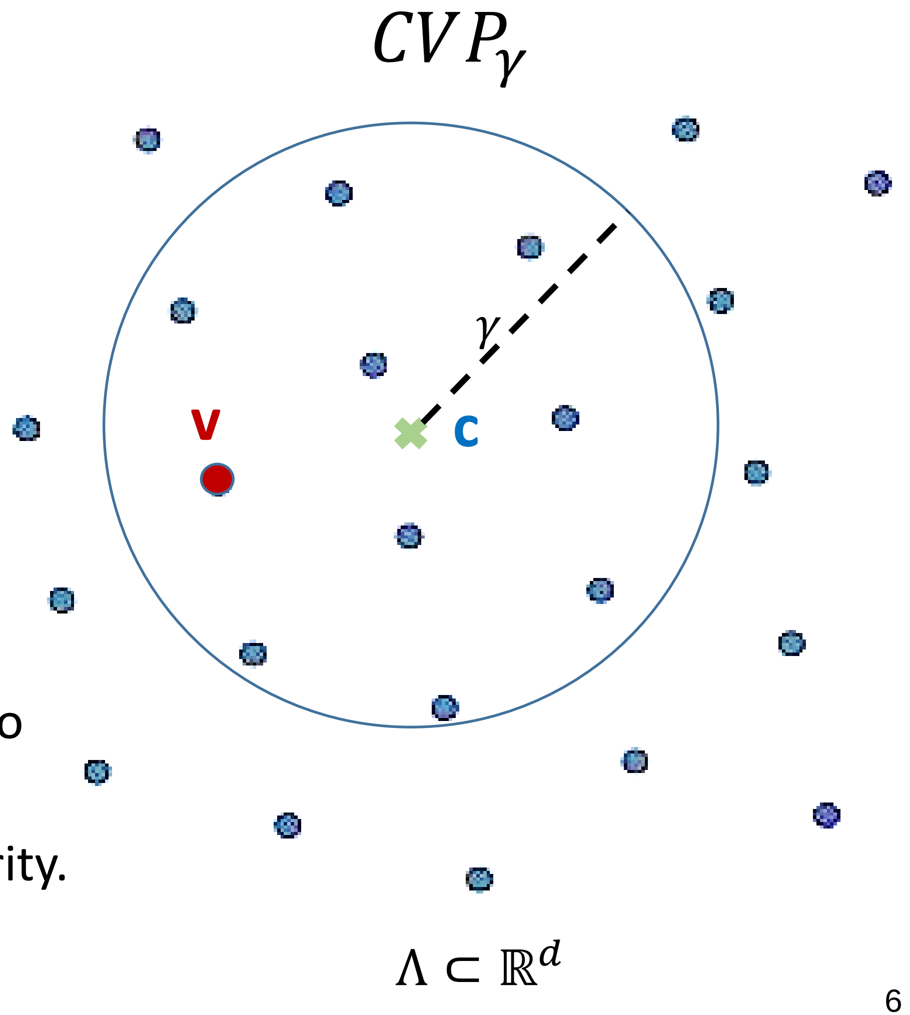


$\Lambda \subset \mathbb{R}^d$

Hash-and-Sign over lattices

Sign$(\mathbf{m}, \mathbf{sk}_\Lambda, \gamma)$:

- $\mathbf{c} \coloneqq H(\mathbf{m})$
- $\mathbf{v} \leftarrow \mathrm{DiscreteGaussianSampler}(\mathbf{sk}_\Lambda, \mathbf{c})$
- $\mathbf{s} \coloneqq \mathbf{c} - \mathbf{v}$
- Return $\mathbf{sig} \coloneqq \mathbf{s}$.



Remarks:

- **Security** : related to Close Vector Problem (CVP) hard to solve without $\mathbf{sk}$.
- Smaller $\mathrm{DiscreteGaussianSampler}(\mathbf{sk},\cdot)$: better security.

→ need $\mathbf{sk}$ of « good quality ».

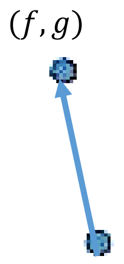
NTRU lattices

NTRU lattices

- $\mathcal{K} = \mathbb{Z}[X]/(X^n + 1) \approx \mathbb{Z}^n, n = 512$ and q is a prime

NTRU lattices

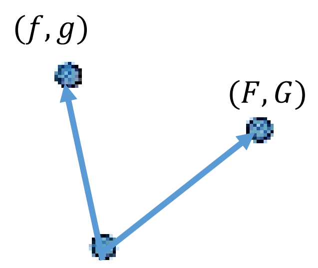
- $\mathcal{K} = \mathbb{Z}[X]/(X^n + 1) \approx \mathbb{Z}^n, n = 512$ and q is a prime
- Small polynomials $f, g \in \mathcal{K}$



$\mathcal{K}^2$

NTRU lattices

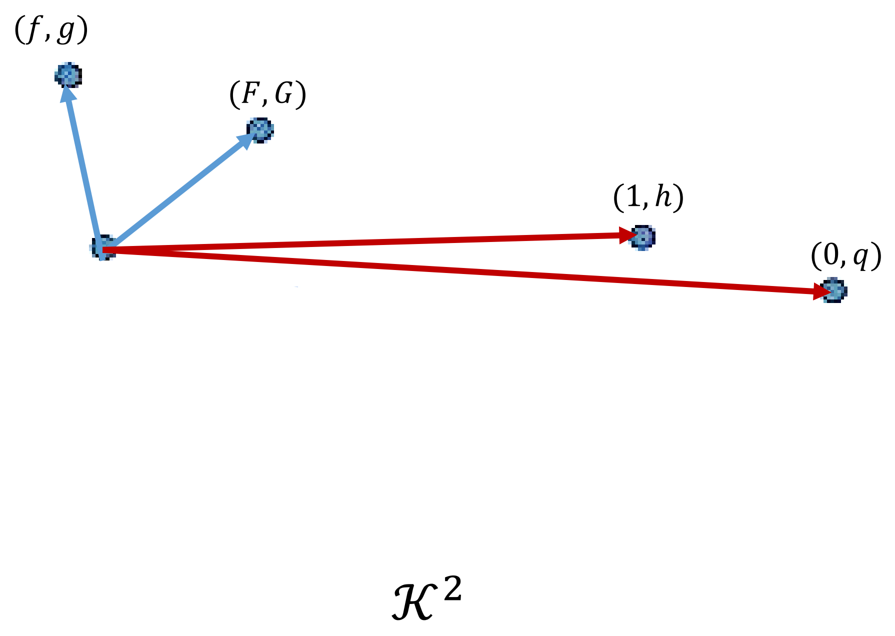
- $\mathcal{K} = \mathbb{Z}[X]/(X^n + 1) \approx \mathbb{Z}^n, n = 512$ and q is a prime
- Small polynomials $f, g \in \mathcal{K}$
- Small $F, G \in \mathcal{K}$ such that $fG - gF = q$



$\mathcal{K}^2$

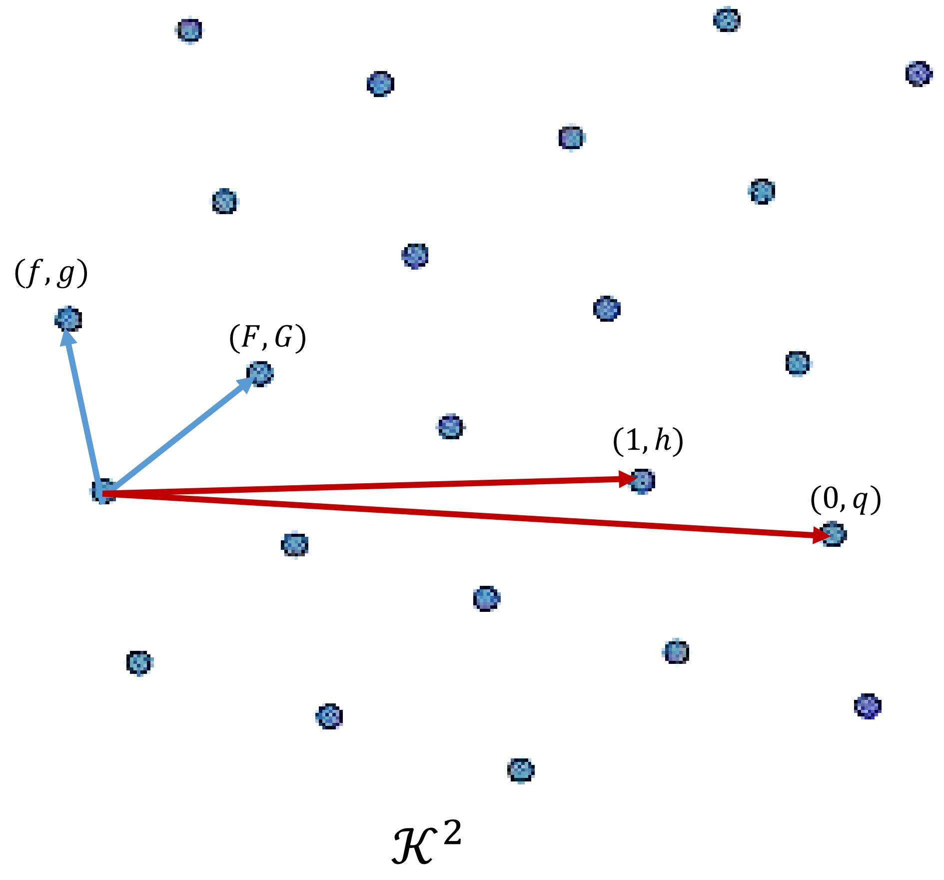
NTRU lattices

- $\mathcal{K} = \mathbb{Z}[X]/(X^n + 1) \approx \mathbb{Z}^n, n = 512$ and q is a prime
- Small polynomials $f, g \in \mathcal{K}$
- Small $F, G \in \mathcal{K}$ such that $fG - gF = q$
- Large $h := f^{-1} g \bmod q$



NTRU lattices

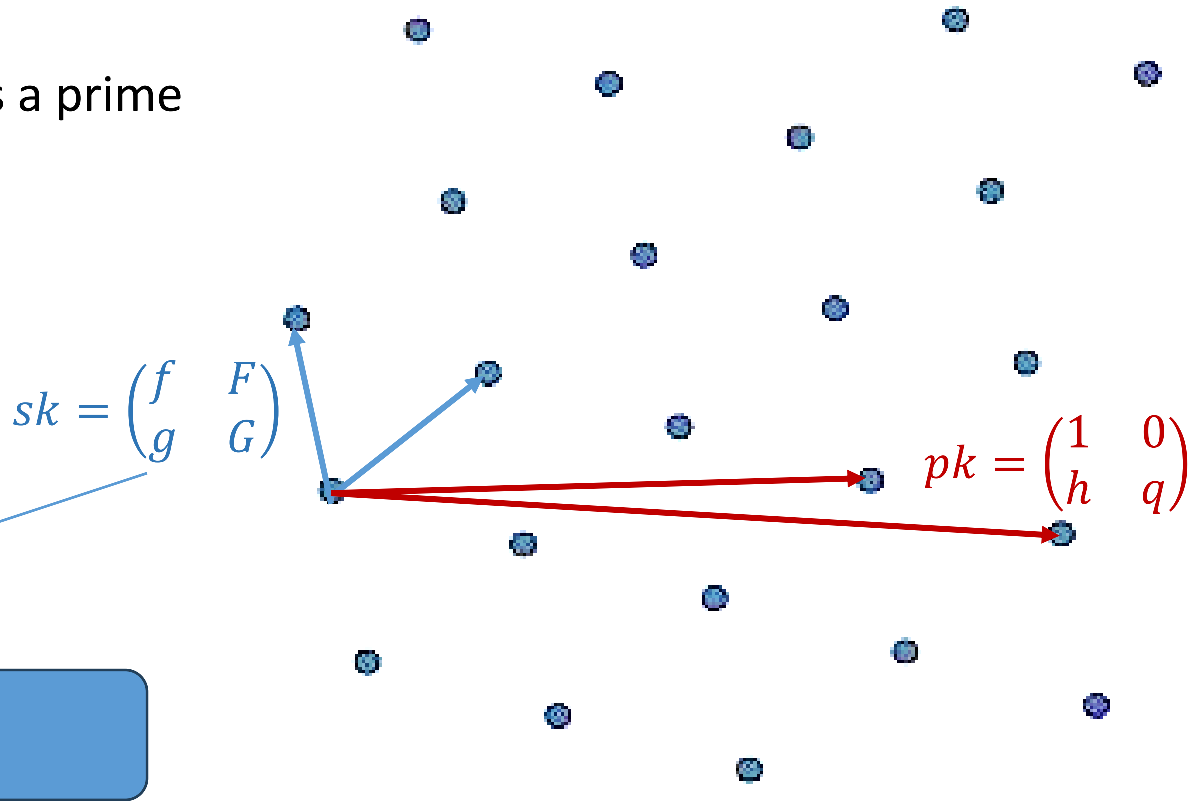
- $\mathcal{K} = \mathbb{Z}[X]/(X^n + 1) \approx \mathbb{Z}^n, n = 512$ and q is a prime
- Small polynomials $f, g \in \mathcal{K}$
- Small $F, G \in \mathcal{K}$ such that $fG - gF = q$
- Large $h \coloneqq f^{-1} g \bmod q$
- $\Lambda_{NTRU} \coloneqq \{(u, v) \in \mathcal{K}^2 | v = uh \bmod q\}$



NTRU lattices

- $\mathcal{K} = \mathbb{Z}[X]/(X^n + 1) \approx \mathbb{Z}^n, n = 512$ and q is a prime
- Small polynomials $f, g \in \mathcal{K}$
- Small $F, G \in \mathcal{K}$ such that $fG - gF = q$
- Large $h := f^{-1} g \bmod q$
- $\Lambda_{NTRU} := \{(u, v) \in \mathcal{K}^2 | v = uh \bmod q\}$
- The secret key sk is the trapdoor.

NTRU ***Trapdoor*** generation

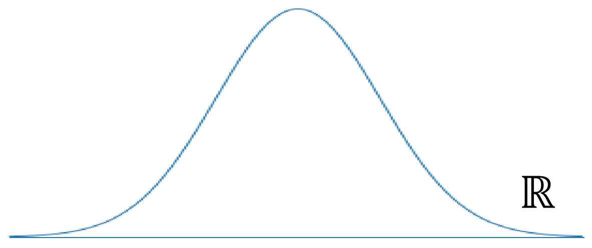


$\Lambda_{NTRU} \subset \mathbb{Z}^{2n}$

Gaussian Distributions

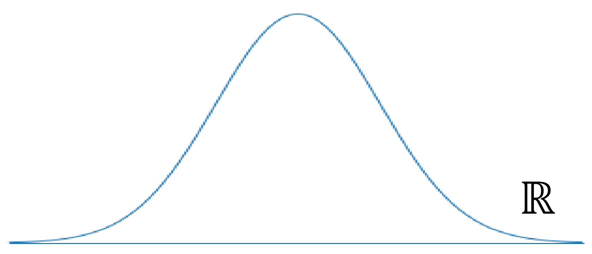
Gaussian Distributions

- Gaussian Distribution $\mathcal{N}_{\mathbb{R},c,\sigma}$

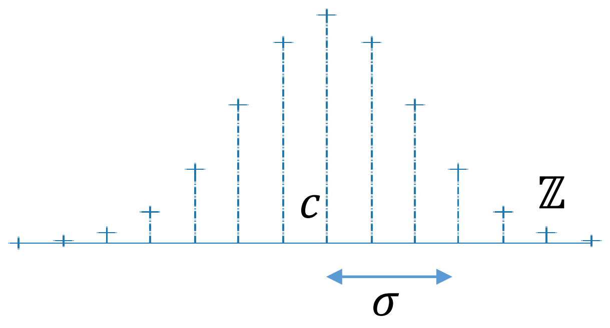


Gaussian Distributions

- Gaussian Distribution $\mathcal{N}_{\mathbb{R},c,\sigma}$

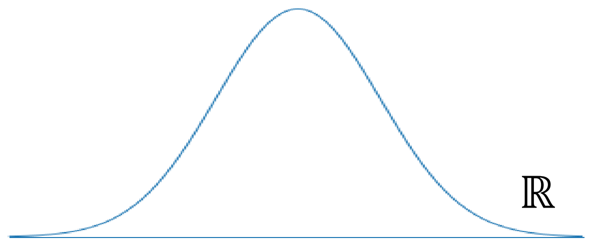


- Discrete Gaussian Distribution on $\mathbb{Z}$: $D_{\mathbb{Z},c,\sigma}$

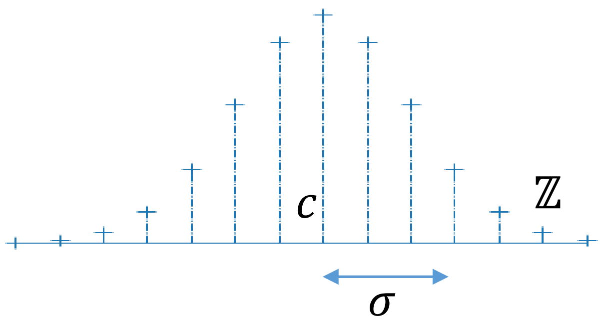


Gaussian Distributions

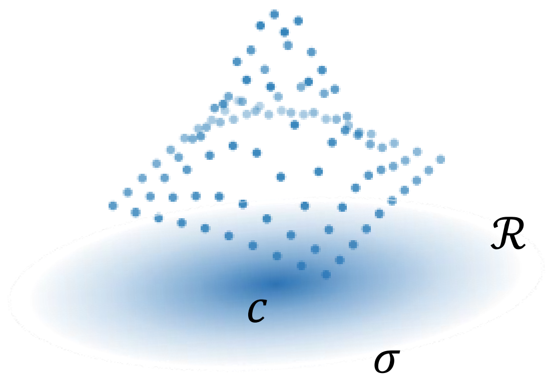
- Gaussian Distribution $\mathcal{N}_{\mathbb{R},c,\sigma}$



- Discrete Gaussian Distribution on $\mathbb{Z}$: $D_{\mathbb{Z},c,\sigma}$



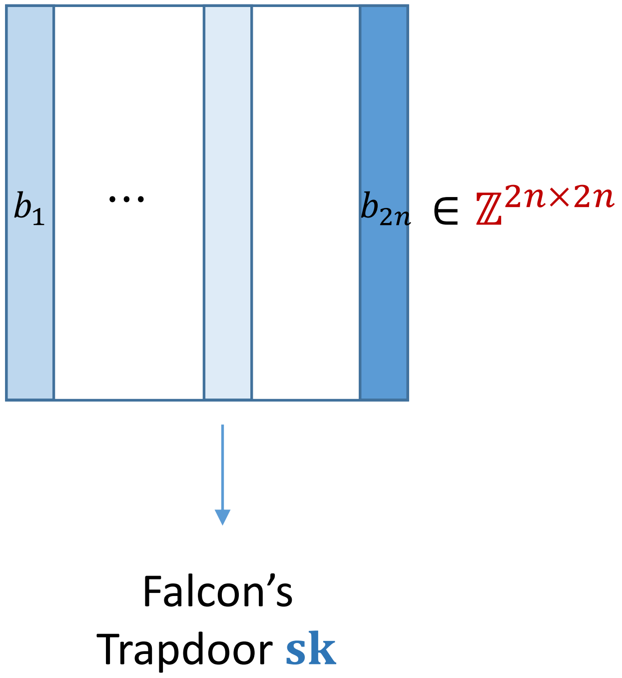
- Discrete Gaussian Distribution on Ring $\mathcal{R}$: $D_{\mathcal{R},c,\sigma}$



DiscreteGaussianSampler$(\mathbf{sk}_\Lambda, \mathbf{c}) \to \mathbf{v}$

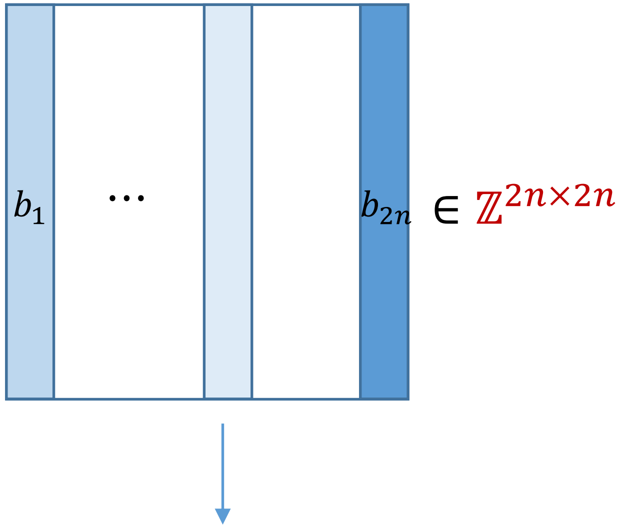
DiscreteGaussianSampler($\mathbf{sk}_\Lambda, \mathbf{c}$) $\rightarrow \mathbf{v}$

KGPV sampler
[Kle00,GPV08]



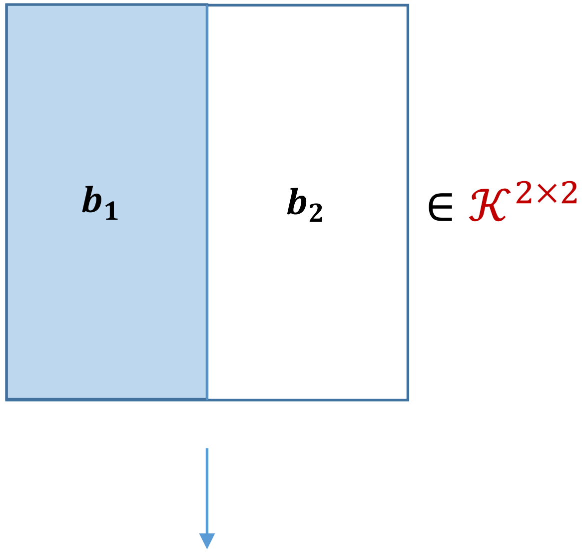
DiscreteGaussianSampler$(\mathbf{sk}_\Lambda, \mathbf{c}) \rightarrow \mathbf{v}$

KGPV sampler
[Kle00,GPV08]



Falcon’s
Trapdoor **sk**

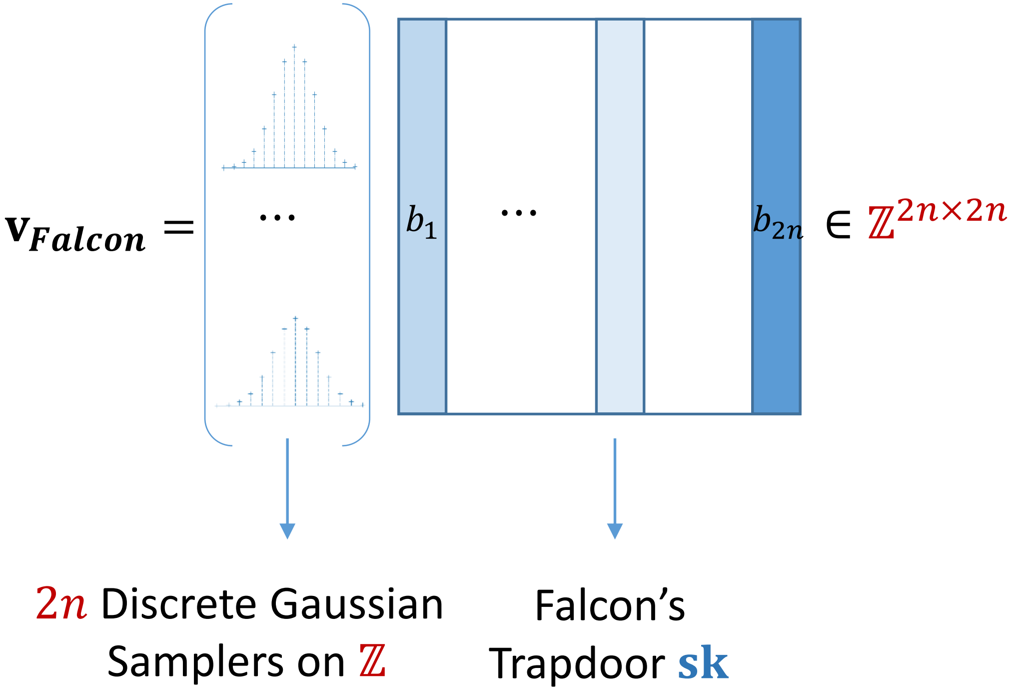
Hybrid sampler
[Pre15]



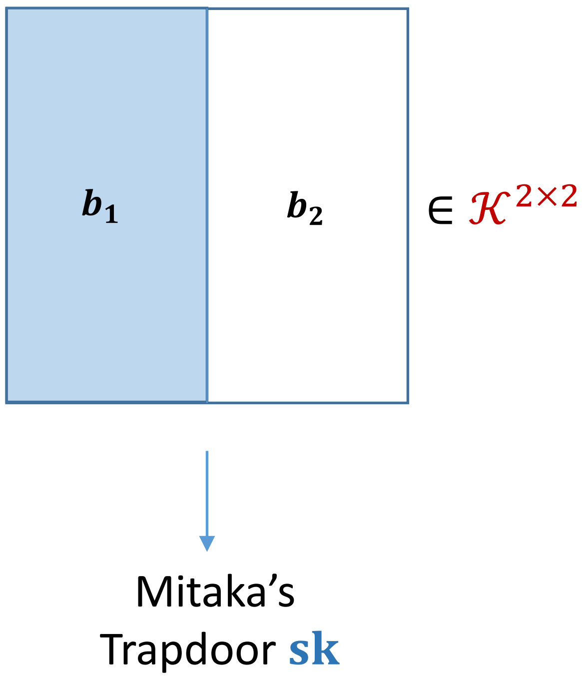
Mitaka’s
Trapdoor **sk**

DiscreteGaussianSampler($\mathbf{sk}_\Lambda, \mathbf{c}$) $\rightarrow \mathbf{v}$

KGPV sampler
[Kle00,GPV08]

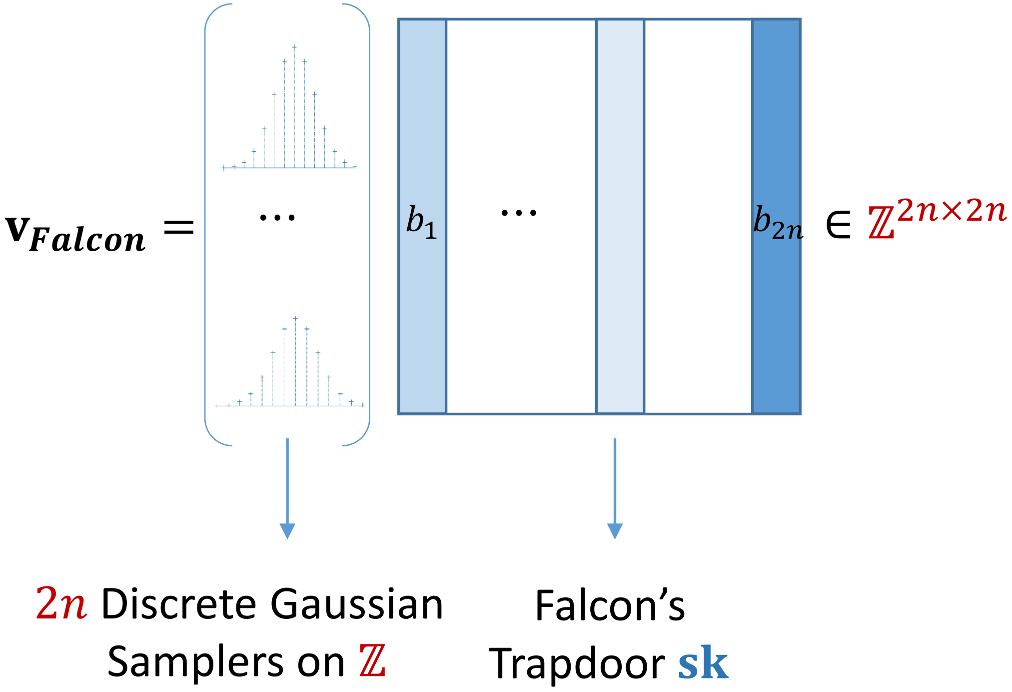


Hybrid sampler
[Pre15]

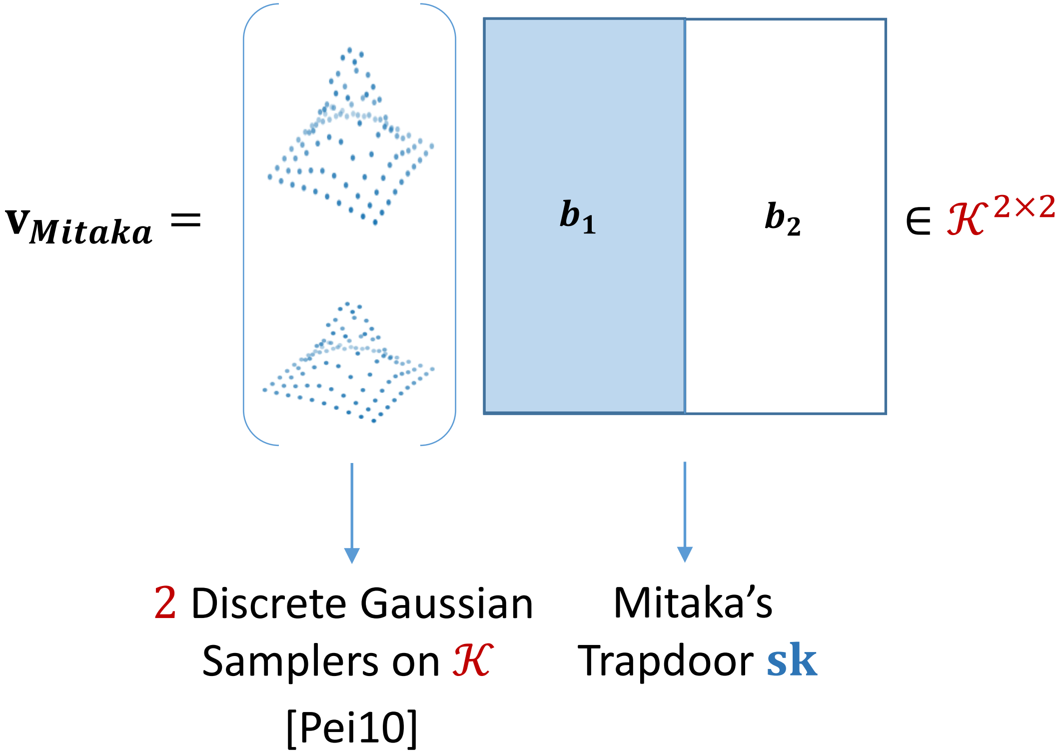


$\text{DiscreteGaussianSampler}(\mathbf{sk}_\Lambda, \mathbf{c}) \to \mathbf{v}$

KGPV sampler
[Kle00,GPV08]



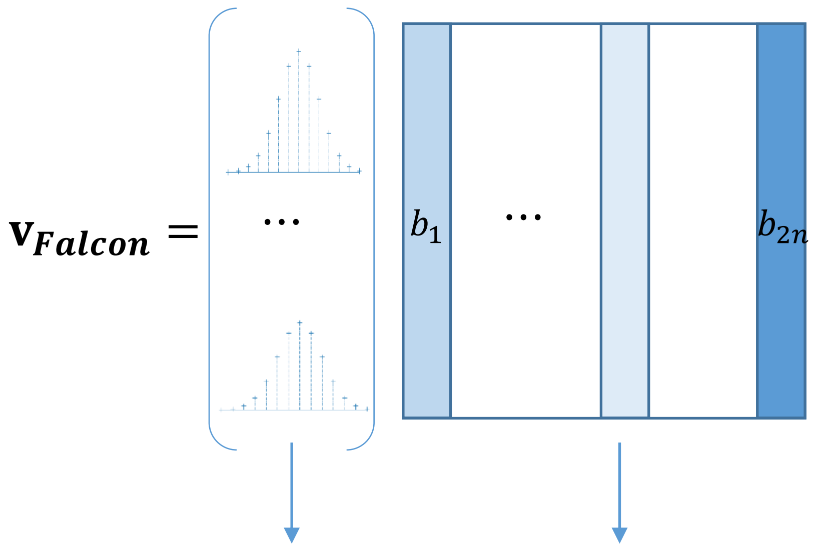
Hybrid sampler
[Pre15]



DiscreteGaussianSampler$(\mathbf{sk}_{\Lambda}, \mathbf{c}) \rightarrow \mathbf{v}$

KGPV sampler

Quadratic

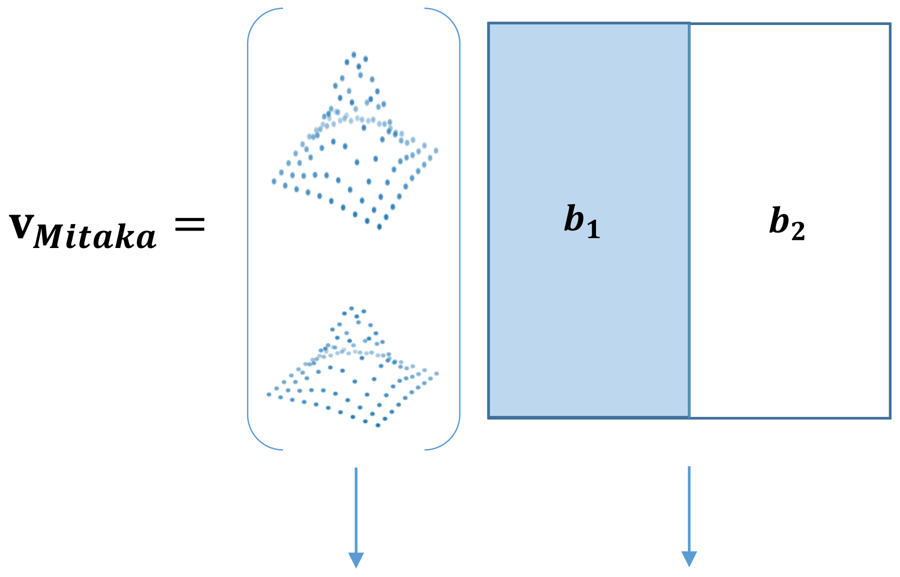


$2n$ Discrete Gaussian Samplers on $\mathbb{Z}$

Falcon's Trapdoor $\mathbf{sk}$

Hybrid sampler

Quasi-linear



2 Discrete Gaussian Samplers on $\mathcal{K}$ [Pei10]

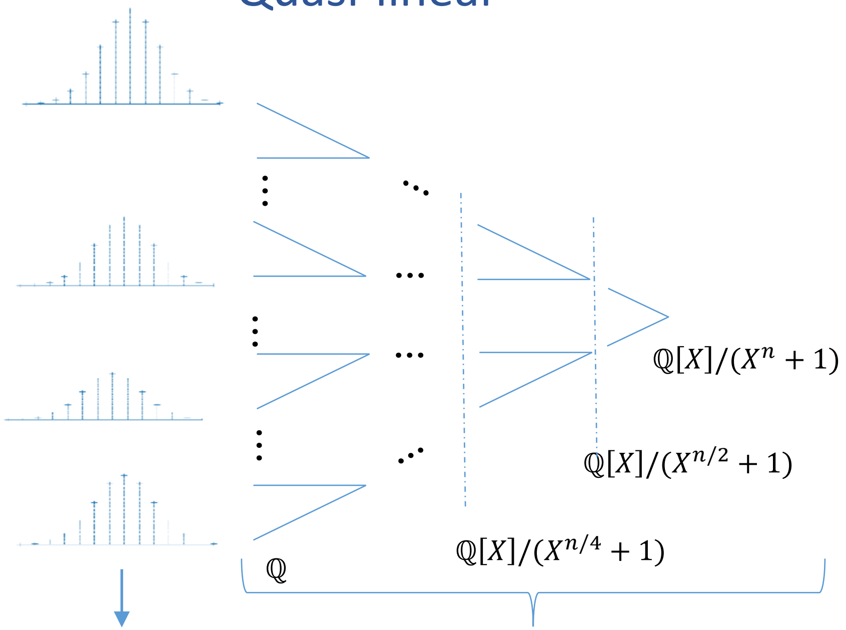
Mitaka's Trapdoor $\mathbf{sk}$

DiscreteGaussianSampler($\mathbf{sk}_\Lambda, \mathbf{c}$) $\rightarrow \mathbf{v}$

FFO sampler [DP16]

Quasi-linear

$\mathbf{v}_{Falcon} =$



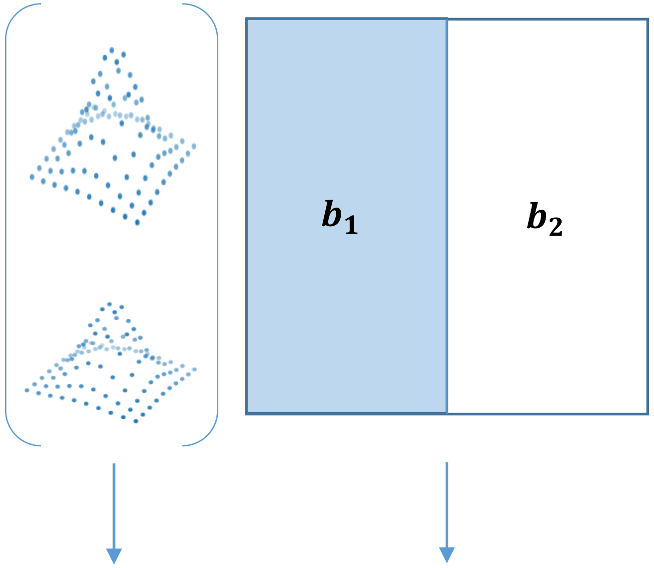
$2n$ Discrete Gaussian Samplers on $\mathbb{Z}$

Falcon's tree: complex Trapdoor **sk**

Hybrid sampler

Quasi-linear

$\mathbf{v}_{Mitaka} =$

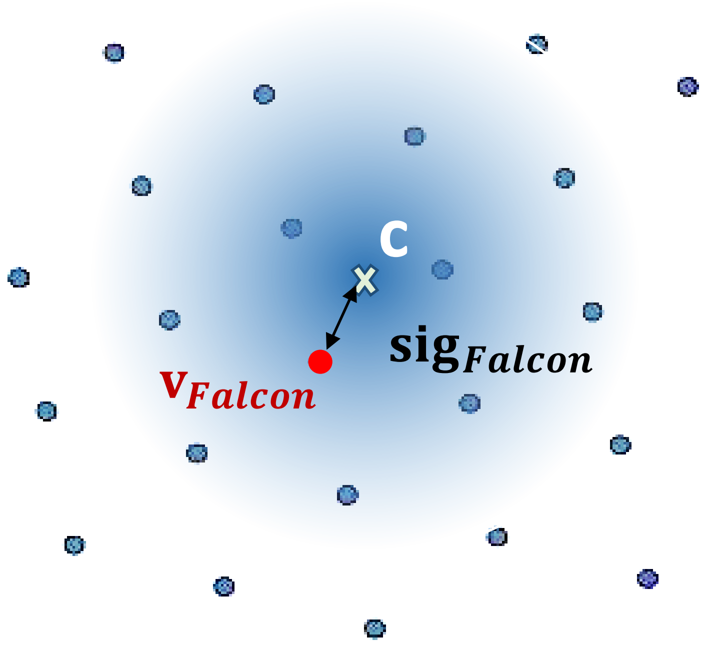


2 Discrete Gaussian Samplers on $\mathcal{K}$ [Pei10]

Mitaka's Trapdoor **sk**

Sampler/Signature's size

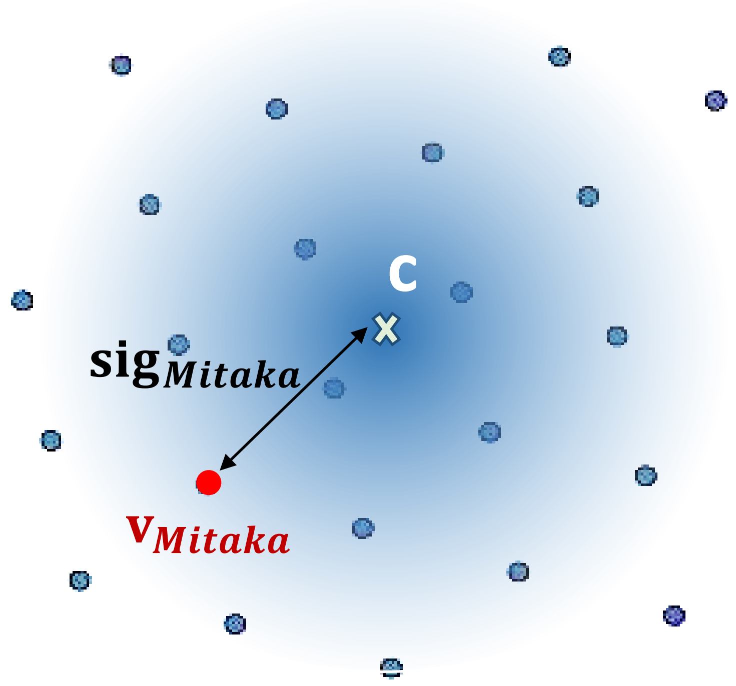
Falcon



$$\|\mathbf{sig}_F\| \propto \|\mathbf{sk}\|_{Klein} \approx 1.17\sqrt{q}$$

$<$

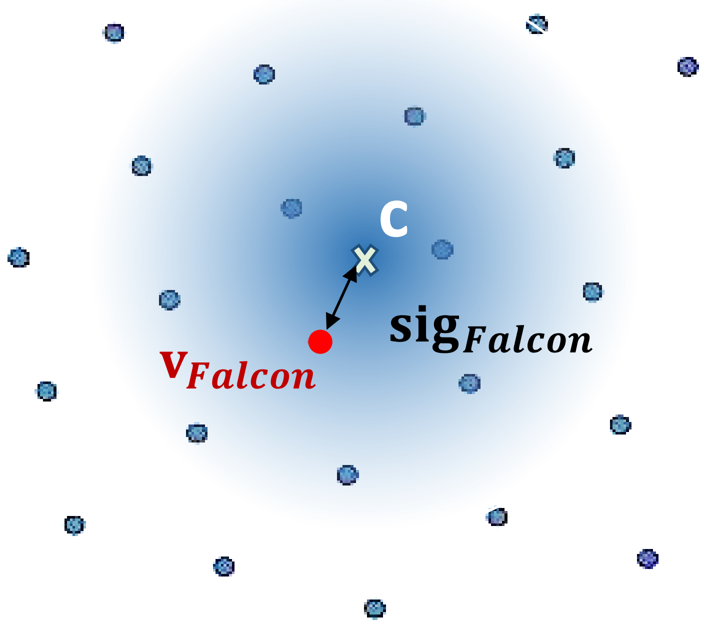
Mitaka



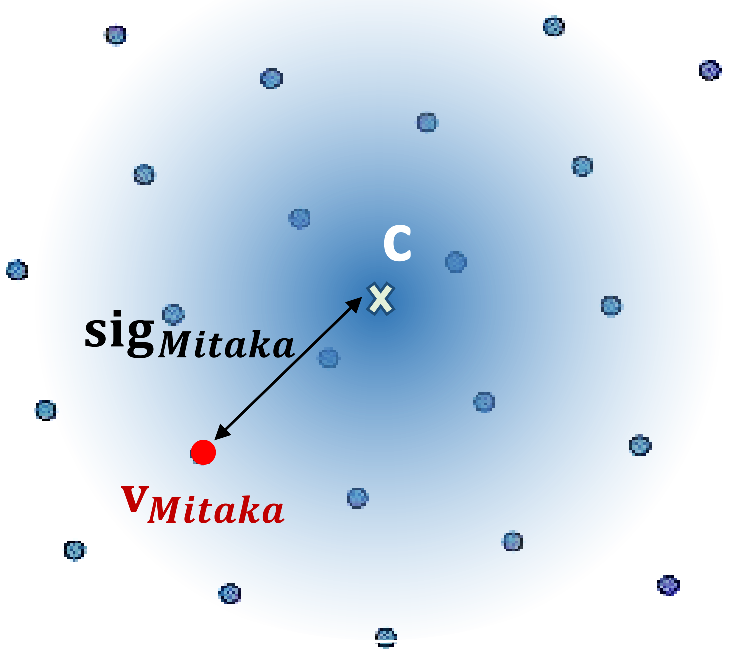
$$2.04\sqrt{q} \approx \|\mathbf{sk}\|_{hybrid} \propto \|\mathbf{sig}_M\|$$

Sampler/Signature's size

Falcon



Mitaka



Falcon < Mitaka

$\|\mathbf{sig}_{\boldsymbol{F}}\| \propto \|\mathbf{sk}\|_{Klein} \approx 1.17\sqrt{q}$

$2.04\sqrt{q} \approx \|\mathbf{sk}\|_{hybrid} \propto \|\mathbf{sig}_{\boldsymbol{M}}\|$

Quality α

Quality α and Trapdoor Generation

The security of the scheme depends on the quality α of the **trapdoor**

$$\alpha = \frac{\|\mathbf{sk}\|}{\sqrt{q}} = \frac{1}{\sqrt{q}} \left\| \begin{pmatrix} f & F \\ g & G \end{pmatrix} \right\|$$

with $\|\cdot\|$ defined by the **sampler** .

Goal: minimize α.

Quality α and Trapdoor Generation

The security of the scheme depends on the quality α of the **trapdoor**

$$\alpha = \frac{\|\mathbf{sk}\|}{\sqrt{q}} = \frac{1}{\sqrt{q}} \left\| \begin{pmatrix} f & F \\ g & G \end{pmatrix} \right\|$$

with $\|\cdot\|$ defined by the **sampler** .

Goal: minimize α.

› Observation: α only depends on f, g.

Quality α and Trapdoor Generation

The security of the scheme depends on the quality α of the **trapdoor**

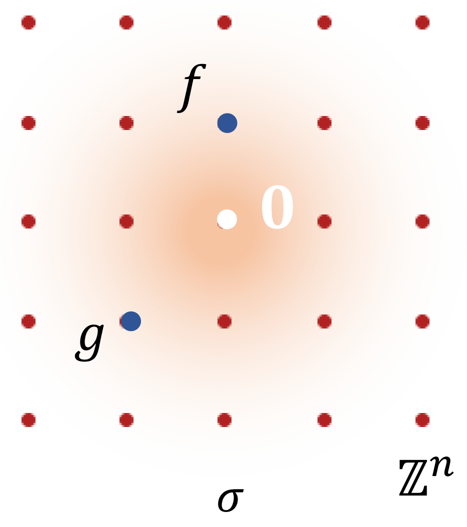
$$\alpha = \frac{\|\mathbf{sk}\|}{\sqrt{q}} = \frac{1}{\sqrt{q}} \left\| \begin{pmatrix} f & F \\ g & G \end{pmatrix} \right\|$$

with $\|\cdot\|$ defined by the **sampler** .

Goal: minimize α.

› Observation: α only depends on f, g.

› Previous method: Sample f, g from a small $D_{\mathbb{Z}^n,0,\sigma}$



Quality α and Trapdoor Generation

The security of the scheme depends on the quality α of the **trapdoor**

$$\alpha = \frac{\|\mathbf{sk}\|}{\sqrt{q}} = \frac{1}{\sqrt{q}} \left\| \begin{pmatrix} f & F \\ g & G \end{pmatrix} \right\|$$

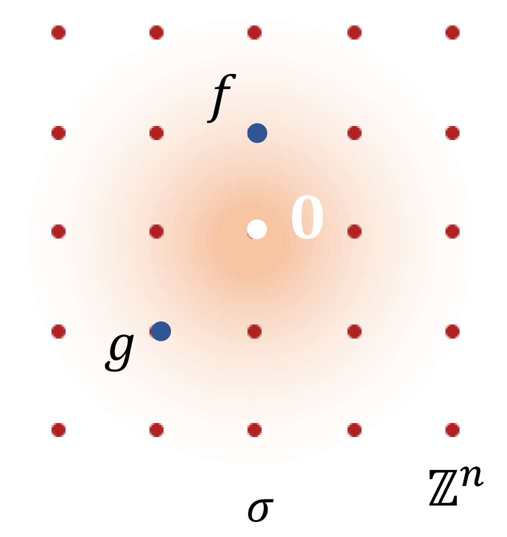
with $\|\cdot\|$ defined by the **sampler** .

Goal: minimize α.

› Observation: α only depends on f, g.

› Previous method: Sample f, g from a small $D_{\mathbb{Z}^n,0,\sigma}$

With a reasonable number of repetitions
we can find f, g with $\|\mathbf{sk}\| \leq \alpha(\sigma)\sqrt{q}$.



Quality α and Trapdoor Generation

The security of the scheme depends on the quality α of the **trapdoor**

$$\alpha = \frac{\|\mathbf{sk}\|}{\sqrt{q}} = \frac{1}{\sqrt{q}} \left\| \begin{pmatrix} f & F \\ g & G \end{pmatrix} \right\|$$

with $\|\cdot\|$ defined by the **sampler** .

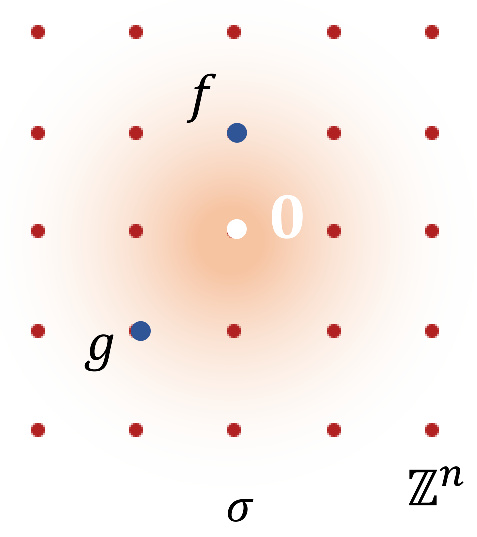
Goal: minimize α.

› Observation: α only depends on f, g.

› Previous method: Sample f, g from a small $D_{\mathbb{Z}^n,0,\sigma}$

With a reasonable number of repetitions
we can find f, g with $\|\mathbf{sk}\| \leq \alpha(\sigma)\sqrt{q}$.

› Our method:



ANTRAG: Annular Trapdoor Generation for Mitaka
$\alpha_{Mitaka} = 1.15$

ANTRAG: Annular NTRU Trapdoor Generation

$$\mathbb{Z}^n \approx \mathcal{K} \ni \sum_n f_i x^i = f \xrightarrow{\text{DFT}} \big(f(\zeta_1), \cdots, f(\zeta_n)\big) \in \mathbb{C}^n$$

ANTRAG: Annular NTRU Trapdoor Generation

$$\mathbb{Z}^n \approx \mathcal{K} \ni \sum_n f_i x^i = f \xrightarrow{\text{DFT}} \big(f(\zeta_1), \cdots, f(\zeta_n)\big) \in \mathbb{C}^n$$

- For fixed $\alpha_{Mitaka} = \alpha$, we want to find f, g such that for $\forall i \leq n$

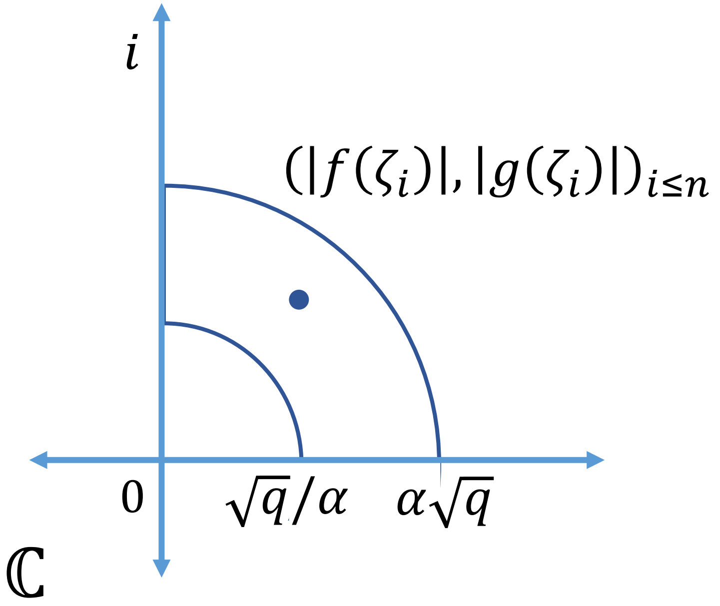
$$\frac{q}{\alpha^2} \leq |f(\zeta_i)|^2 + |g(\zeta_i)|^2 \leq \alpha^2 q$$

ANTRAG: Annular NTRU Trapdoor Generation

$$\mathbb{Z}^n \approx \mathcal{K} \ni \sum_n f_i x^i = f \xrightarrow{\text{DFT}} \left(f(\zeta_1), \cdots, f(\zeta_n)\right) \in \mathbb{C}^n$$

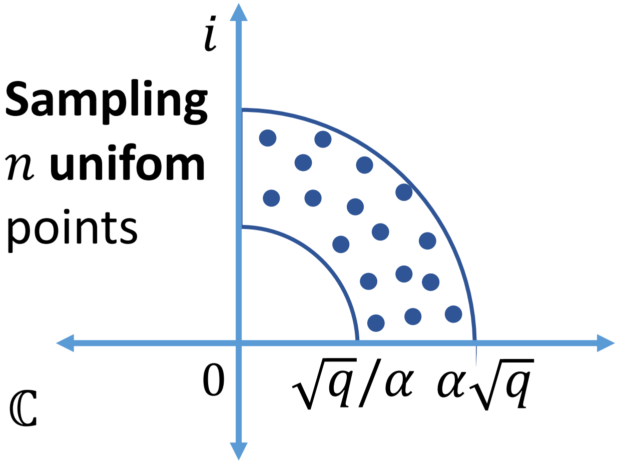
- For fixed $\alpha_{Mitaka} = \alpha$, we want to find f, g such that for $\forall i \leq n$

$$\frac{q}{\alpha^2} \leq |f(\zeta_i)|^2 + |g(\zeta_i)|^2 \leq \alpha^2 q$$

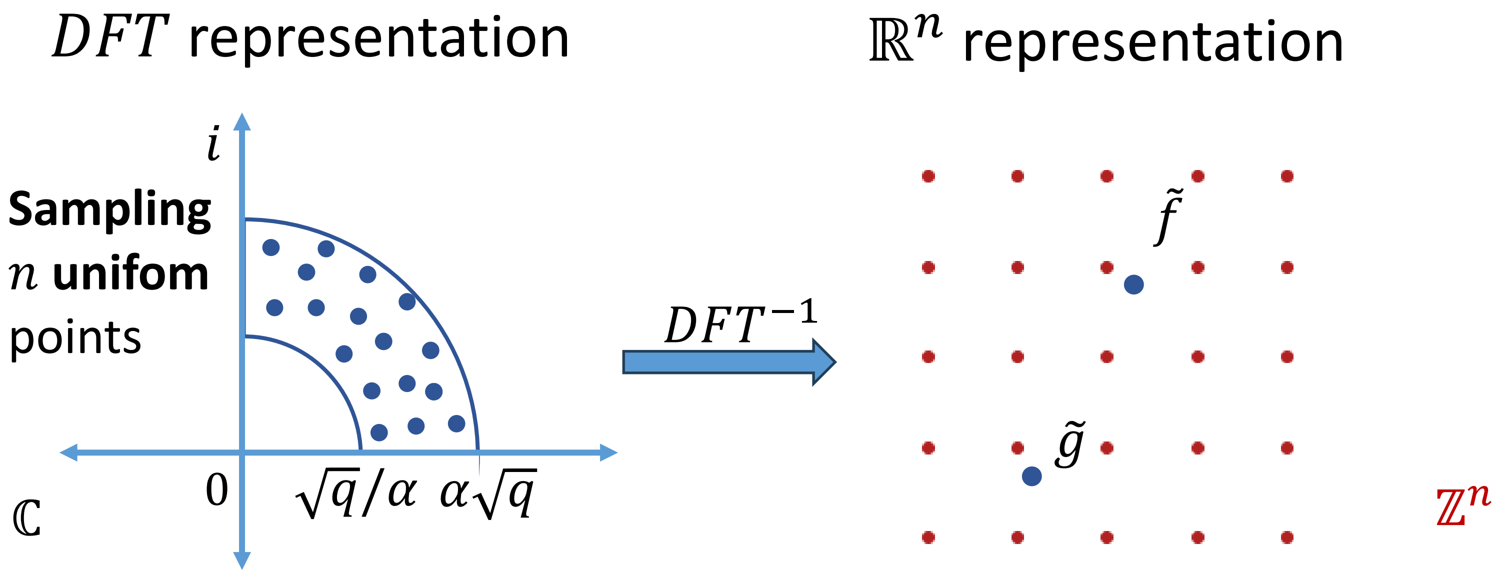


ANTRAG: Annular NTRU Trapdoor Generation (1)

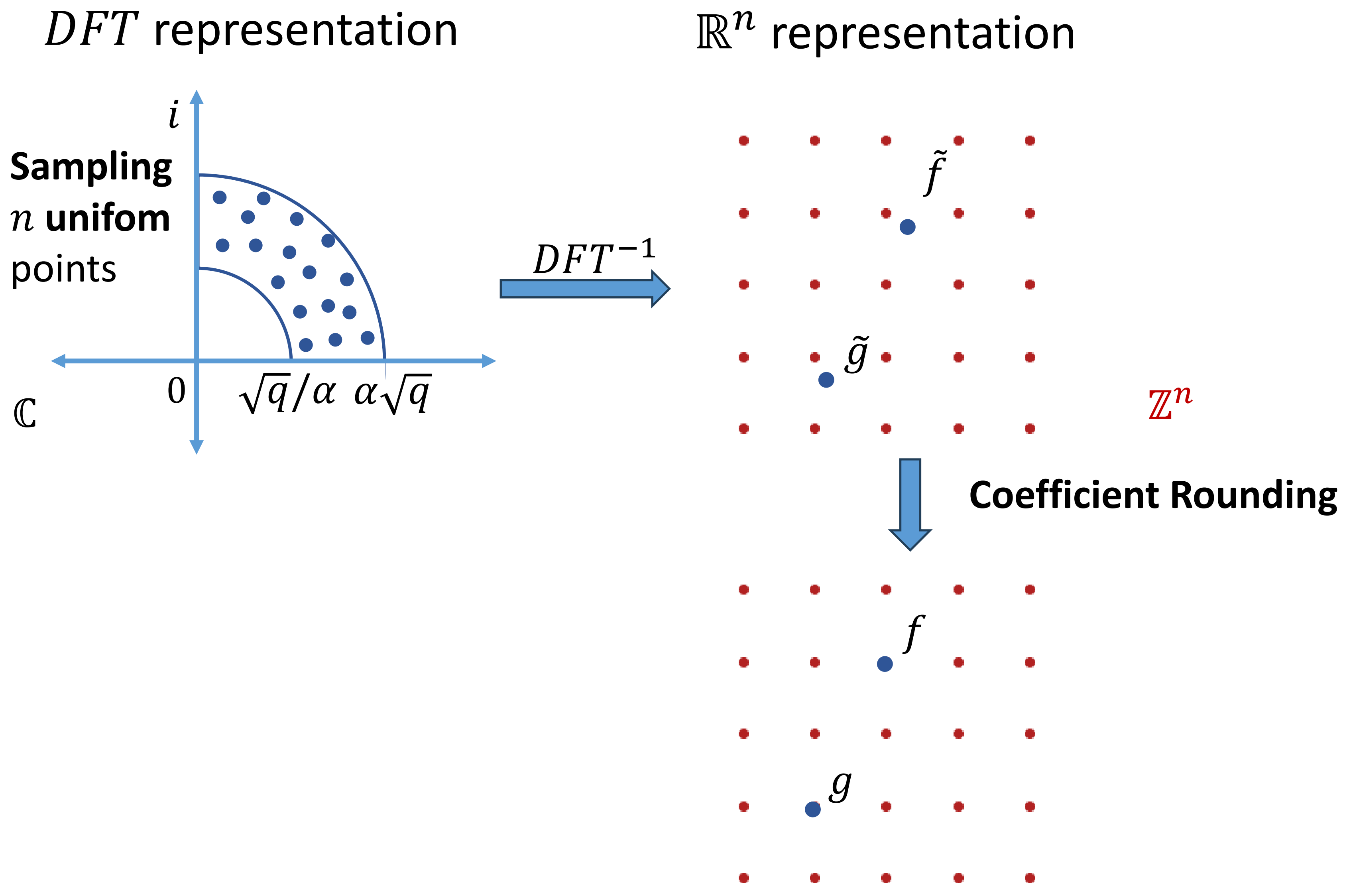
DFT representation



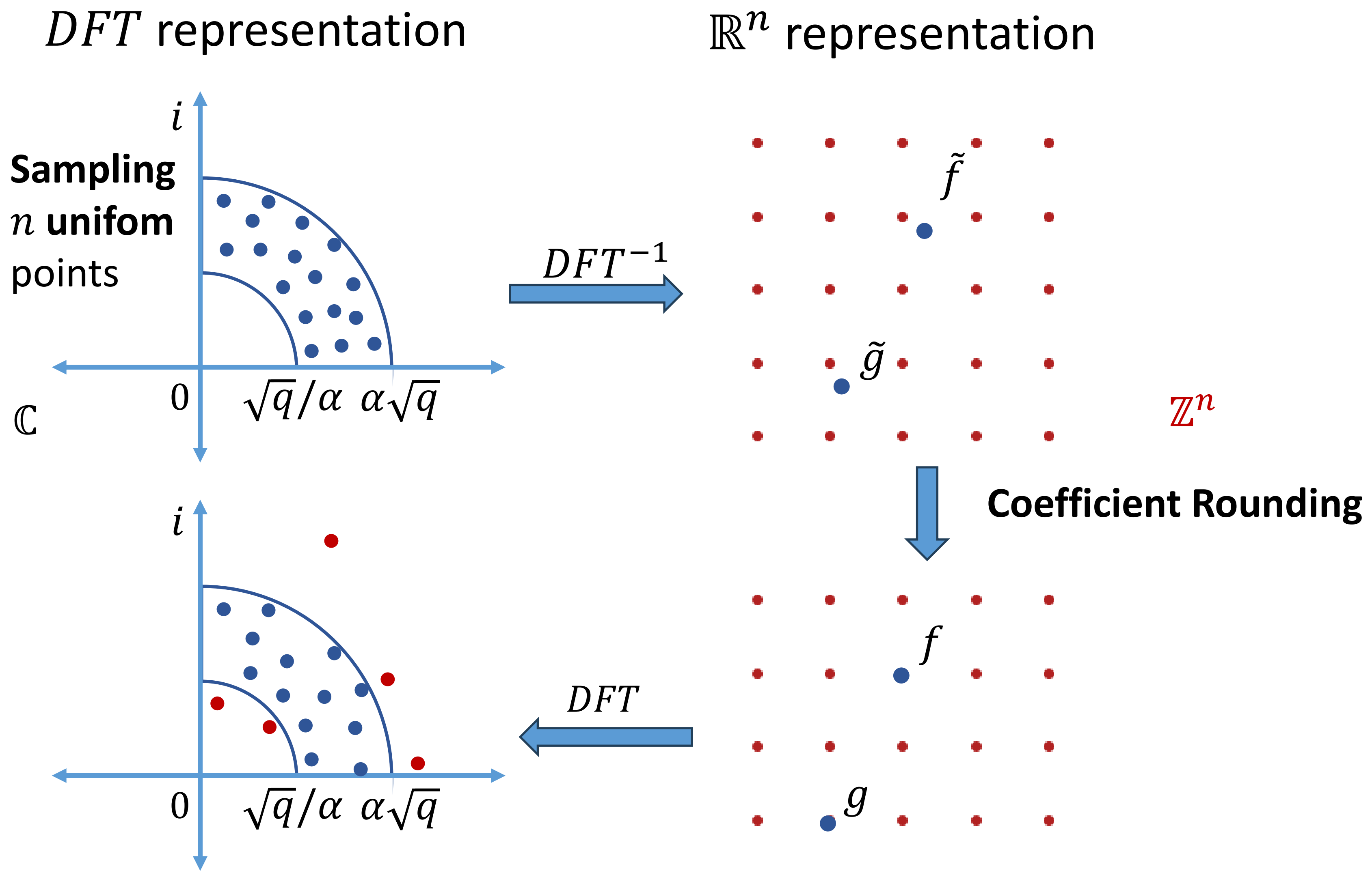
ANTRAG: Annular NTRU Trapdoor Generation (1)



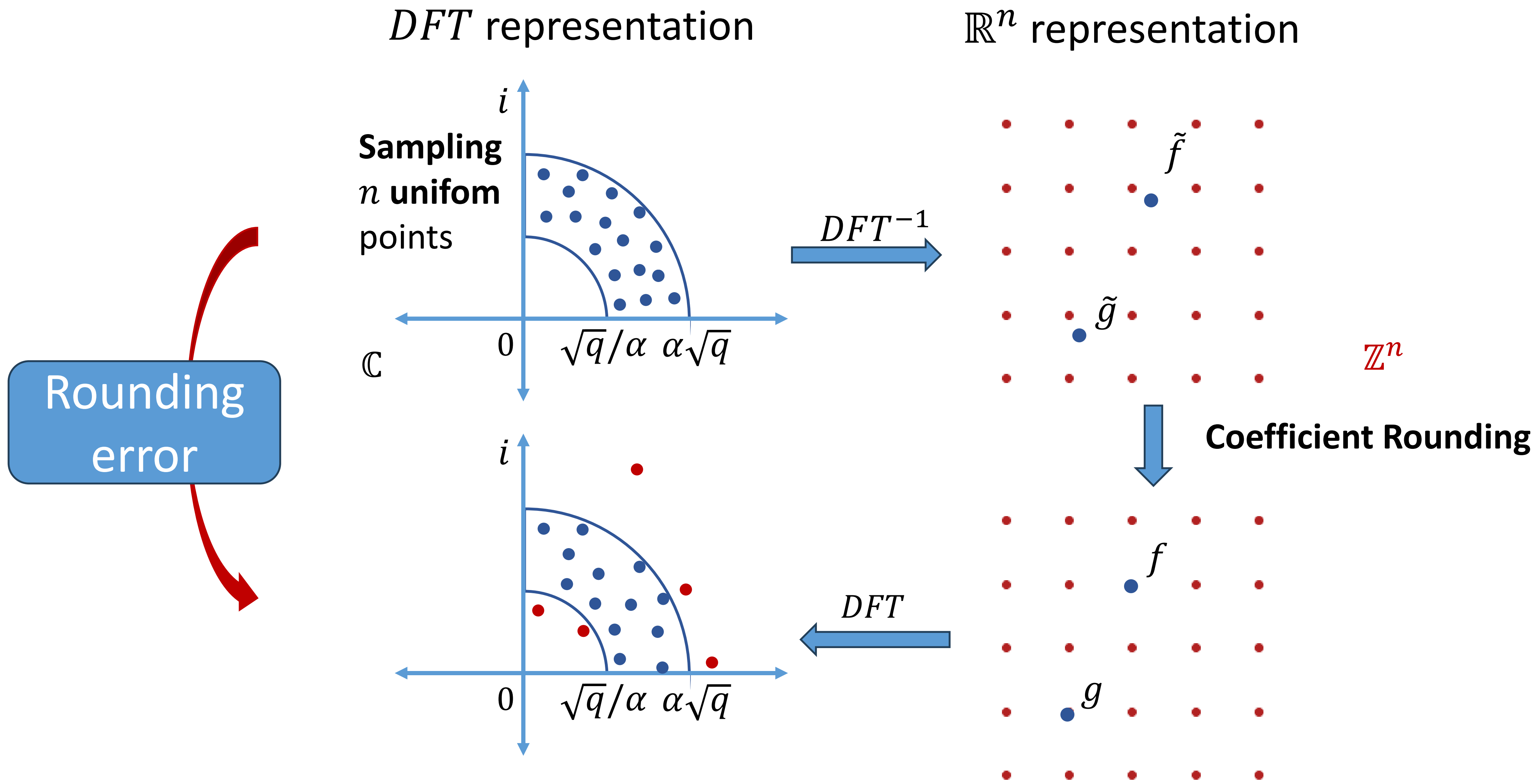
ANTRAG: Annular NTRU Trapdoor Generation (1)



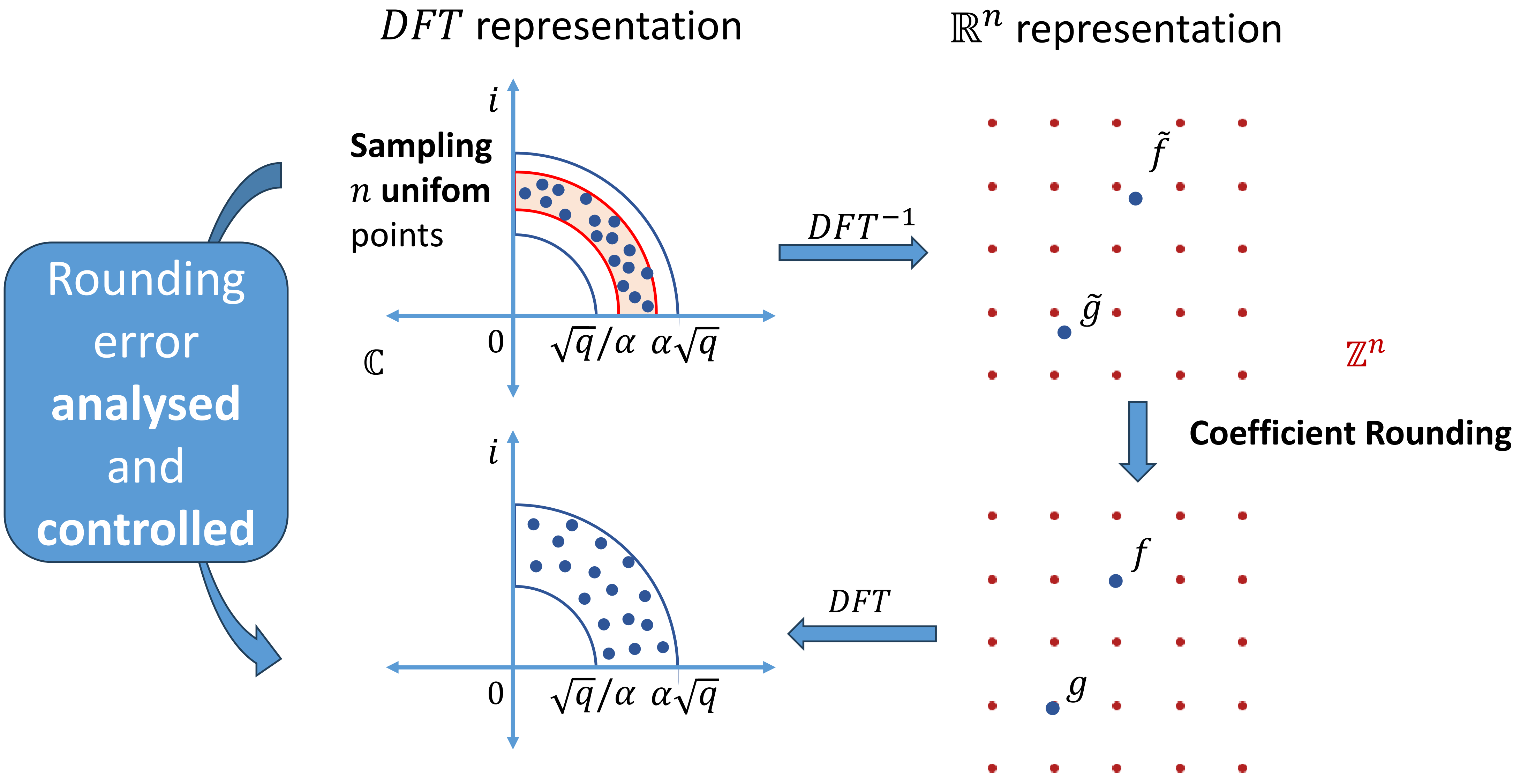
ANTRAG: Annular NTRU Trapdoor Generation (1)



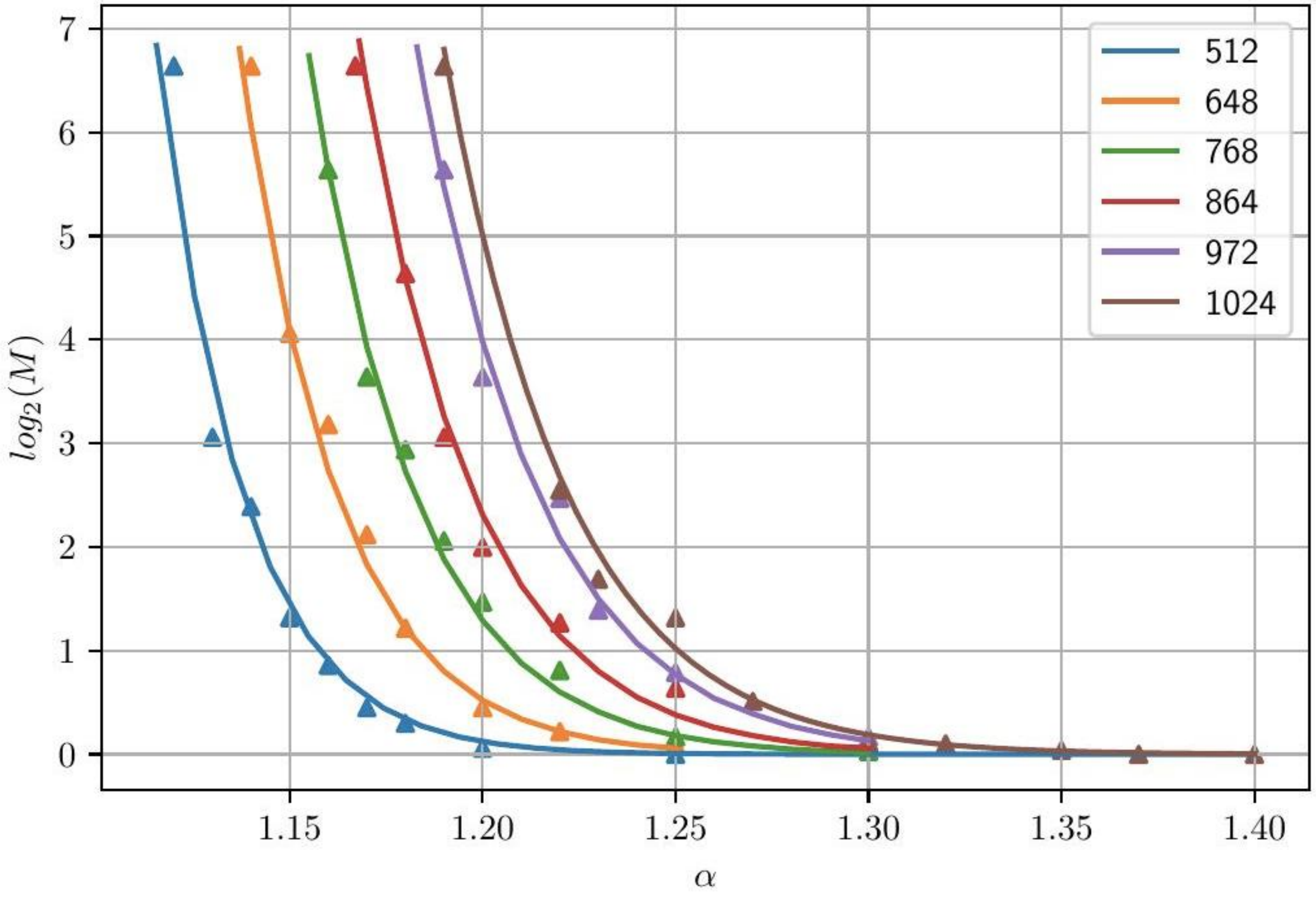
ANTRAG: Annular NTRU Trapdoor Generation (1)



ANTRAG: Annular NTRU Trapdoor Generation (2)



Quality/repetition in ANTRAG



Performance comparison with Mitaka and Falcon

	Antrag+Hybrid	
n	**512**	**1024**
α	**1.15**	**1.23***
Keygen repetitions	**3**	**4**
Classical security (bits)	**124**	**264**
Sign speed (μs)	**8**	**15**
Signature size (bytes)	**646**	**1260**

Performance comparison with Mitaka and Falcon

	Antrag+Hybrid		Mitaka ($D_{\mathbb{Z}^n,0}$+Hybrid)	
n	**512**	**1024**	512	1024
α	**1.15**	**1.23***	2.04	2.33
Keygen repetitions	**3**	**4**	-	-
Classical security (bits)	**124**	**264**	102	233
Sign speed (μs)	**8**	**15**	8	16
Signature size (bytes)	**646**	**1260**	713	1405

- No precise number is given but Mitaka is estimated to have many repetitions.

Performance comparison with Mitaka and Falcon

	Antrag+Hybrid		Mitaka ($D_{\mathbb{Z}^n,0}$+Hybrid)		Falcon ($D_{\mathbb{Z}^n,0}$+FFO)	
n	**512**	**1024**	512	1024	512	1024
α	**1.15**	**1.23***	2.04	2.33	1.17	1.17
Keygen repetitions	**3**	**4**	-	-	8	8
Classical security (bits)	**124**	**264**	102	233	123	284
Sign speed (μs)	**8**	**15**	8	16	18	36
Signature size (bytes)	**646**	**1260**	713	1405	666	1280

*We do not need too small α to obtain the level NIST V of security.

- No precise number is given but Mitaka is estimated to have many repetitions.

3-smooth dimensions

n	648 $(2^3 \cdot 3^4)$			768 $(2^8 \cdot 3)$			864 $(2^5 \cdot 3^3)$			972 $(2^2 \cdot 3^5)$		
q	**12289**	**3889**	**9721**	**12289**	**3329**	**18433**	**12289**	**3727**	**10369**	**12289**	**4373**	**17497**
α	1.17	1.32	1.19	1.19	1.39	1.16	1.21	1.40	1.23	1.22	1.40	1.18
Repetitions	4	4	4	3	4	3	3	4	3	4	4	4
Classical/Quantum Security (bits)	166/ 151	159/ 144	164/ 149	196/ 178	192/ 174	195/ 177	222/ 201	220/ 200	222/ 201	251/ 227	254/ 230	250/ 227
Signature size (bytes)	808	747	796	952	883	977	1069	1000	1058	1701	1580	1225

Versatility with security!

Perspectives

- Antrag is integrated in the signature Solmae submitted at KPQC (Solmae = Antrag + Hybrid Sampler) (ongoing)
- More optimizations in Antrag's design (ongoing)
  - Annulus -> Circle sampling?
  - Integrating new rejection sampling technique
  - Full-fledged implementation?

Thank you!