Do not overstretch NTRU-like problems

Alexandre Wallet
Inria, Centre de Rennes Bretagne-Atlantique

PQ Cryptanalysis Workshop, Birmingham, 21-25 March 2022

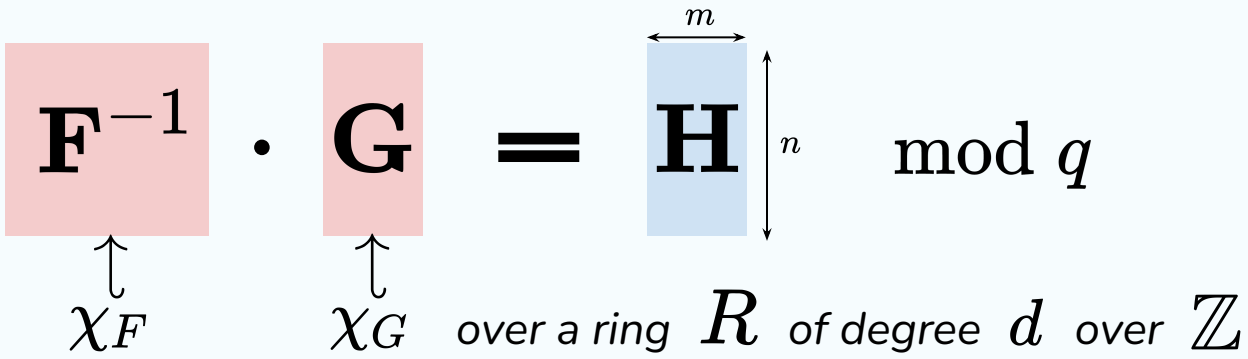
What’s “NTRU-like” like?

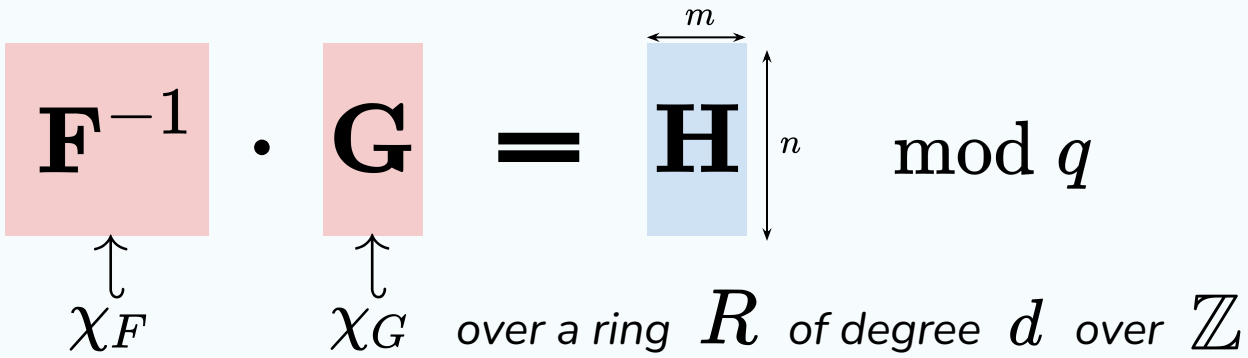
Crypto with NTRU-lattices

Chronology of overstretched cryptanalysis

Open problems & questions about the NTRU design

What’s “NTRU-like” like?


m
$\mathbf{F}^{-1} \cdot \mathbf{G} = \mathbf{H}$
n
$\mod q$
χ_F
χ_G
over a ring R of degree d over $\mathbb{Z}$

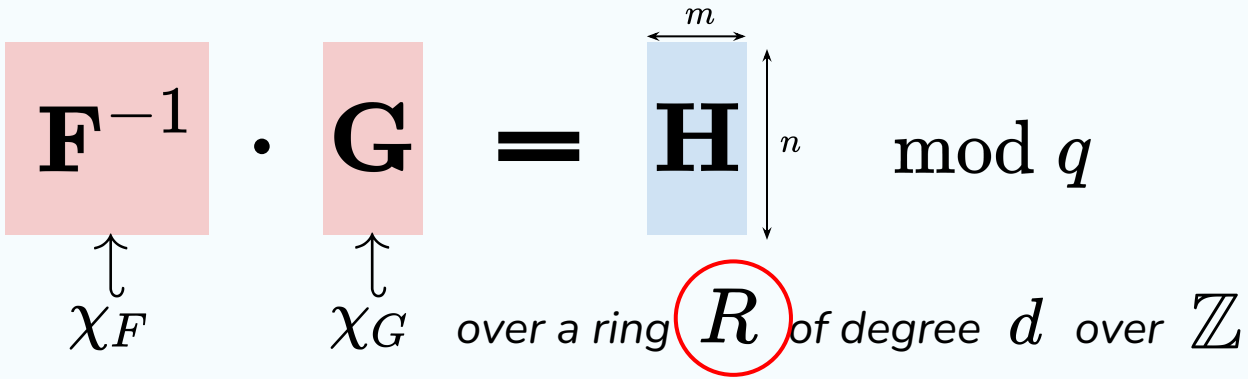


Search version

Given $\mathbf{H}$, compute $\mathbf{F}, \mathbf{G}$

Decision version

Distinguish $\mathbf{H}$ from $\mathcal{U}(R/qR)$



“Vintage” ring

$\mathbb{Z}[x]/(x^d - 1)$

(out-of-fashion because attacks using eval at 1)

Popular ring

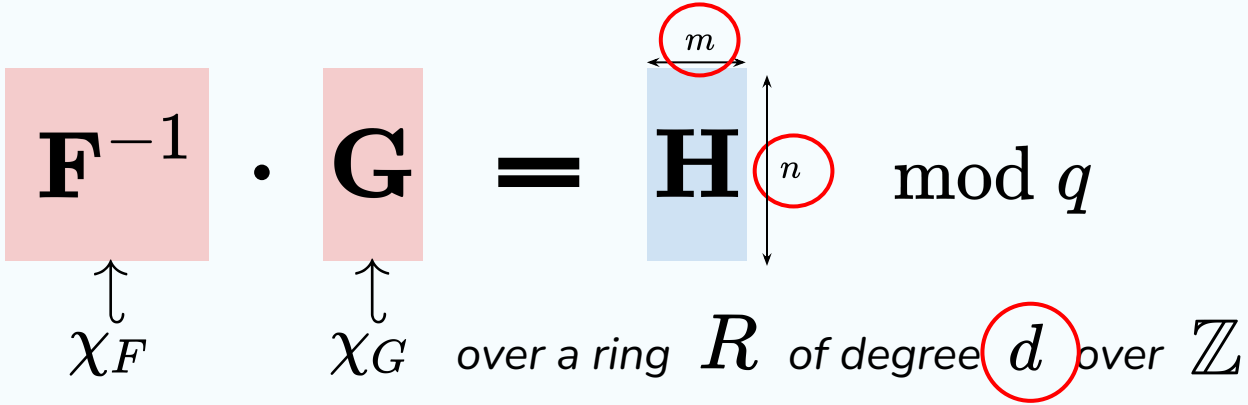
$\mathbb{Z}[x]/(x^{2^k} + 1)$

Power-of-two cyclotomic

“NTRU Prime”

$\mathbb{Z}[x]/(x^p \pm x \pm 1)$

“Unstructured” ring



“original” NTRU

d large

$n = m = 1$

“Module” NTRU

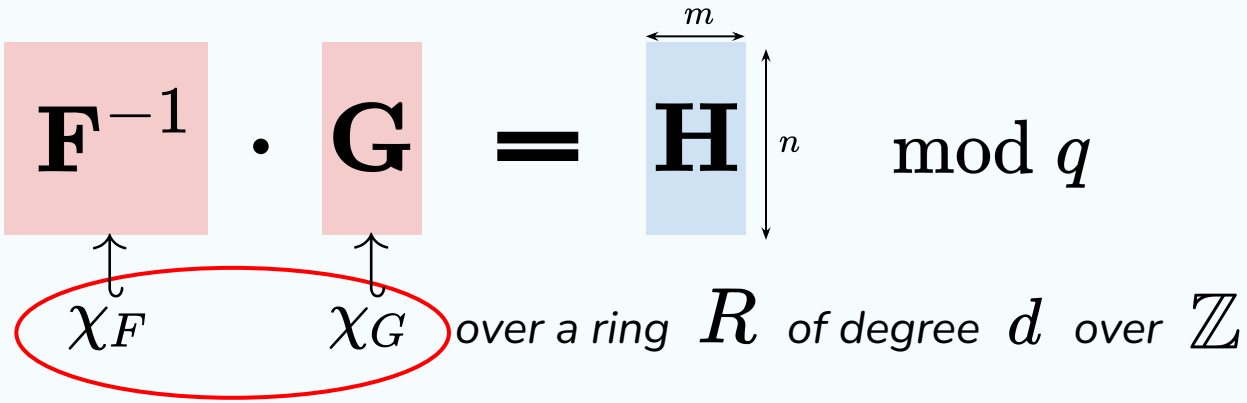
d less large

$n \geq m \geq 1$

“Matrix” NTRU

$d = 1$

$n \geq m \geq 1$



For practical efficiency:

- binary, (sparse) ternary
- short Gaussians
- short-ish Gaussians

Regimes for proofs:

- large-ish Gaussians
- ???

“Cryptographers love it!
All crypto with this very simple trick”

$$\begin{bmatrix} \mathbf{F} & \mathbf{G} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{H} \\ -\mathbf{I}_m \end{bmatrix} = 0 \bmod q$$

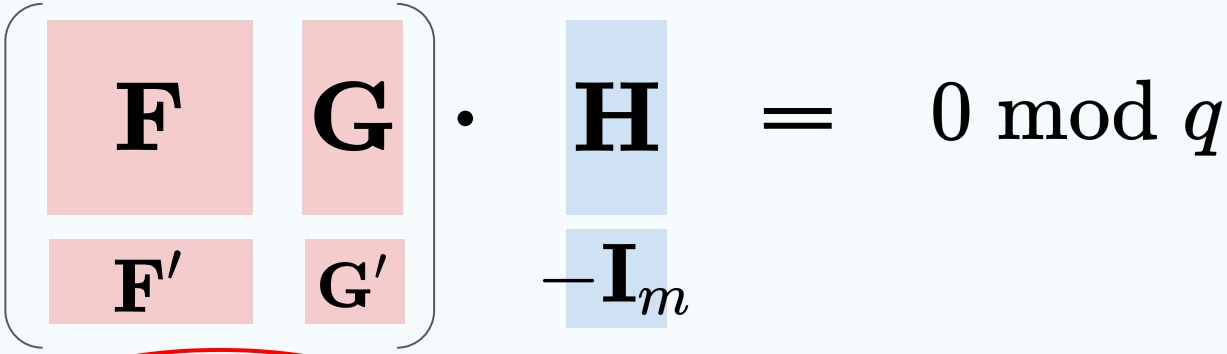
$$\begin{bmatrix} \mathbf{F} & \mathbf{G} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{H} \\ -\mathbf{I}_m \end{bmatrix} = 0 \bmod q$$

Encryption schemes!
[HPS98, HRSS16, BCLV16, CPS*X20, ...]

$\Rightarrow$

Homomorphic encryption schemes!
[G09, LTV12, BLLN13, GGHLM19, BIPPS22, ...]

Signature schemes!
[DDLL13] (BLISS)



Encryption schemes!
[HPS98, HRSS16, BCLV16, CPS*X20, ...]

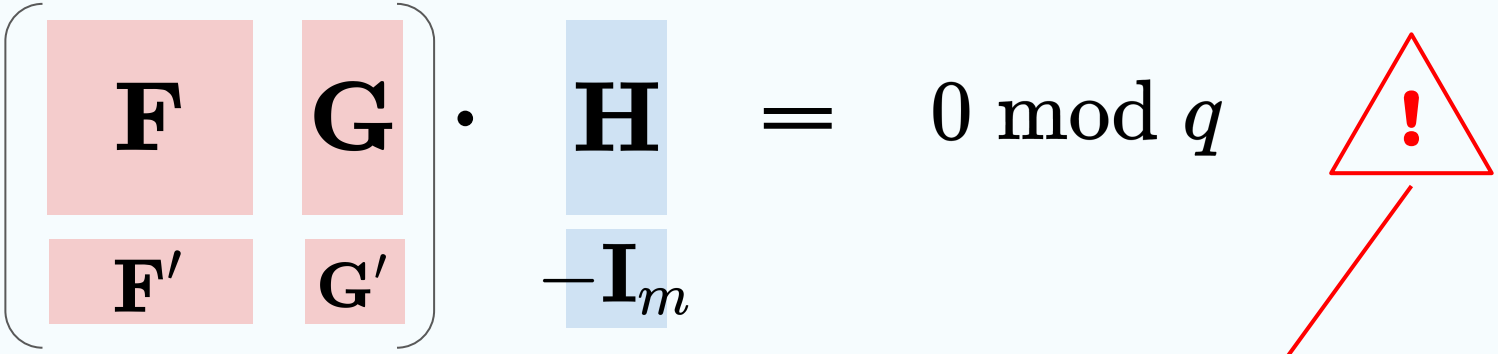
$\Rightarrow$

Homomorphic encryption schemes!
[G09, LTV12, BLLN13, GGHLM19, BIPPS22, ...]

Signature schemes!
[DDLL13] (BLISS)

$\Rightarrow$ Trapdoor sampling!

MORE signature schemes and IBE!
[HPSS00, DLP14, FKLPSWZ17 (Falcon), CPS*X20, CKKS20, EFGRTT*Y22,...]



Encryption schemes!
[HPS98, HRSS16, BCLV16, CPS*X20, ...]

$\Rightarrow$

Homomorphic encryption schemes!
[G09, LTV12, BLLN13, GGHLM19, BIPPS22, ...]

Signature schemes!
[DDLL13] (BLISS)

$\Rightarrow$
Trapdoor sampling!

MORE signature schemes and IBE!
[HPSS00, DLP14, FKLPSWZ17 (Falcon), CPS*X20, CKKS20, EFGRTT*Y22,...]

NTRU Lattices in a nutshell

$$\begin{matrix}\mathbf{u} & \mathbf{v}\end{matrix} \cdot \begin{matrix}\mathbf{H} \\ -\mathbf{I}_m\end{matrix} = 0 \bmod q$$

$$\mathcal{L}_{\text{NTRU}} = \{(u, v) \in R^{n+m} \ : \ uH - v = 0 \bmod q\}$$

public basis

$$\begin{matrix}\mathbf{I}_n & \mathbf{H} \\ \mathbf{0} & q\mathbf{I}_m\end{matrix}$$

rank $(n+m)d$

volume q^{md}

if “random enough”

$$\lambda_1(\mathcal{L}) \approx q^{\frac{m}{n+m}}$$

How random these lattices look depends on

$$\chi_f = \chi_g = \chi$$

(say, $\text{Var}(\chi) = \sigma^2$)

NTRU Lattices in a nutshell

$$\begin{matrix}\mathbf{u} & \mathbf{v}\end{matrix} \cdot \begin{matrix}\mathbf{H} \\ -\mathbf{I}_m\end{matrix} = 0 \bmod q$$

$$\mathcal{L}_{\text{NTRU}} = \{(u, v) \in R^{n+m} \ : \ uH - v = 0 \bmod q\}$$

public basis

$$\begin{matrix}\mathbf{I}_n & \mathbf{H} \\ \mathbf{0} & q\mathbf{I}_m\end{matrix}$$

$\lambda_1(\mathcal{L}) \approx q^{\frac{m}{n+m}}$ VS. $\|\text{row}\| \approx \sigma\sqrt{n+m}$

secret key

$$\begin{matrix}\mathbf{F} & \mathbf{G}\end{matrix} \quad n$$

Quick sum-up, and overstretching

- secret key defines a sublattice $\mathcal{L}_{\mathrm{sk}}$, of possibly large rank
- smaller $\sigma \Rightarrow$ denser $\mathcal{L}_{\mathrm{sk}}$
- yet, volume is fixed! smaller $\sigma \Rightarrow$ larger gap between minima $\Rightarrow$ less “random” lattice

“Overstretched regime”: $\frac{q}{\sigma}$ so large that it makes lattices too far from random

Quick sum-up, and overstretching

- secret key defines a sublattice $\mathcal{L}_{\mathrm{sk}}$, of possibly large rank
- smaller $\sigma \Rightarrow$ denser $\mathcal{L}_{\mathrm{sk}}$
- yet, volume is fixed! smaller $\sigma \Rightarrow$ larger gap between minima $\Rightarrow$ less "random" lattice

"Overstretched regime": $\frac{q}{\sigma}$ so large that it makes lattices too far from random

Why would q be so large??	To cover for noise growth in FHE schemes
How small could σ be?	As small as possible! (noise grows slower, smaller ciphertexts)
And how large is rank $\mathcal{L}_{\mathrm{sk}}$?	Application dependent, usually $\frac{\mathrm{rank}\,\mathcal{L}}{2}$

A concrete example

[GGHLM19] smaller parameters set: $n = m = 1024, \chi = \text{ternary}, \quad q = 2^{42}$

$$\text{Vol}\left(\mathcal{L}_{\text{sk}}\right) = \left(\frac{n}{2}\right)^{\frac{n}{2}} \qquad \text{Vol}\left(\mathcal{L}\right) = q^m$$

VS.

$$\text{Vol}\left(\mathcal{L}_{\text{sk}}\right)^{\frac{1}{n}} \approx 22 \qquad \text{Vol}(\mathcal{L})^{\frac{1}{2n}} = 2^{21}$$

(hand waving a bit)

$$\frac{\lambda_{n+1}}{\lambda_n} \approx 2^{34} \text{ !!}$$

It corresponds to a unique-SVP instance with a gigantic gap.

[LM09] unique-SVP with gap γ ~~ BDD at distance $\frac{1}{\gamma}$

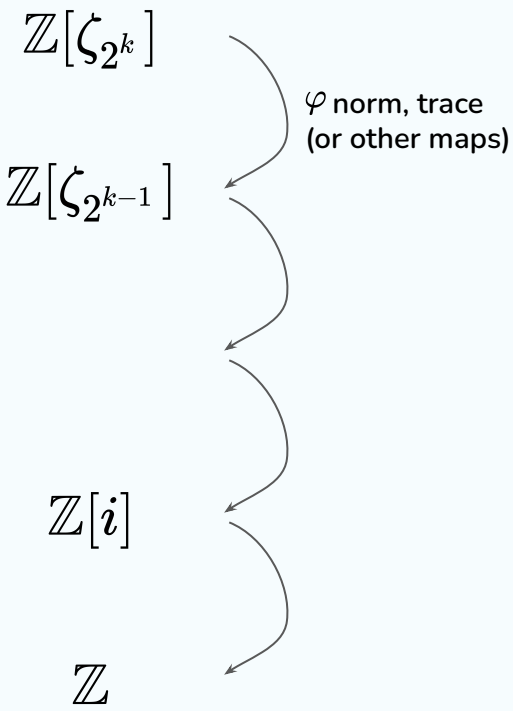
This looks like an easy problem... but we don't know $\mathcal{L}_{\text{sk}}$ (yet).

Attacks against overstretched regimes:

a tale of refinements.

Subfield attacks [ABD16,CJL16] < "LLL in the middle" [KF'17] = MiNTRU, without the ring [L*20] < NTRU Fatigue [DvW21]

Subfield attacks [ABD16,CJL16]



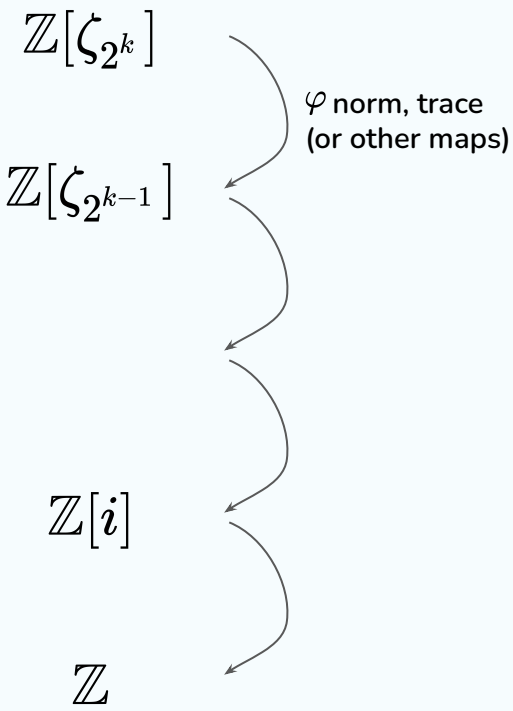
if x is very short, then $\varphi(x)$ remains short

but dimension is halved so lattice reduction is easier!

If moreover q is large, any short vector will be a multiple of $\varphi(\mathrm{sk})$

Then we lift it back to $\mathbb{Z}[\zeta_{2^k}]$

Subfield attacks [ABD16,CJL16]



if x is very short, then $\varphi(x)$ remains short

but dimension is halved so lattice reduction is easier!

If moreover q is large, any short vector will be a multiple of $\varphi(\mathrm{sk})$

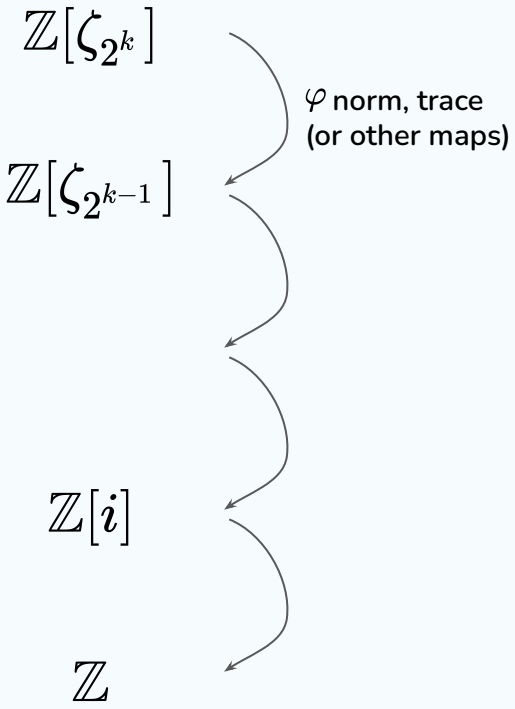
Then we lift it back to $\mathbb{Z}[\zeta_{2^k}]$

$\sigma = \mathrm{poly}(n), q = \exp(n) \Rightarrow$ poly-time attack against NTRU

$\sigma = \mathrm{poly}(n), q = \mathrm{superpoly}(n) \Rightarrow$ subexp-time attack against NTRU

(correspondingly, possible threatening attacks vs. YASHE [LTV12, BLNN13], and multilinear maps [GGH13,15])

Subfield attacks [ABD16,CJL16]



if x is very short, then $\varphi(x)$ remains short

but dimension is halved so lattice reduction is easier!

If moreover q is large, any short vector will be a multiple of $\varphi(\mathrm{sk})$

!!

Then we lift it back to $\mathbb{Z}[\zeta_{2^k}]$

$\sigma = \mathrm{poly}(n), q = \exp(n) \Rightarrow$ poly-time attack against NTRU

$\sigma = \mathrm{poly}(n), q = \mathrm{superpoly}(n) \Rightarrow$ subexp-time attack against NTRU

(and correspondingly, attacks vs. YASHE [LTV12, BLNN13], and also multilinear maps [GGH13,15])

Subfield attacks [ABD16,CJL16] $<$ "LLL in the middle" [KF'17]

Principle 1: If q is large compared to σ, any short vector will be a multiple of sk

(because geometry and algebra)

Principle 2: Lattice reduction detects it earlier then expected

Subfield attacks [ABD16,CJL16] < "LLL in the middle" [KF'17]

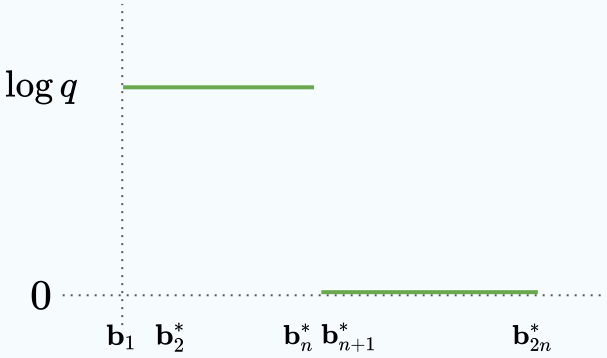
Principle 1: If q is large compared to σ, any short vector will be a multiple of sk

(because geometry and algebra)

Principle 2: Lattice reduction detects it earlier then expected

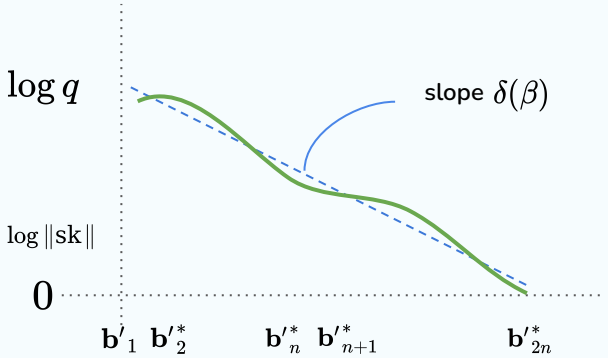
public basis

$q\mathbf{I}_n$	$\mathbf{0}$
$\mathbf{H}$	$\mathbf{I}_n$



block β lattice reduction

heuristic!!



Subfield attacks [ABD16,CJL16] < "LLL in the middle" [KF'17]

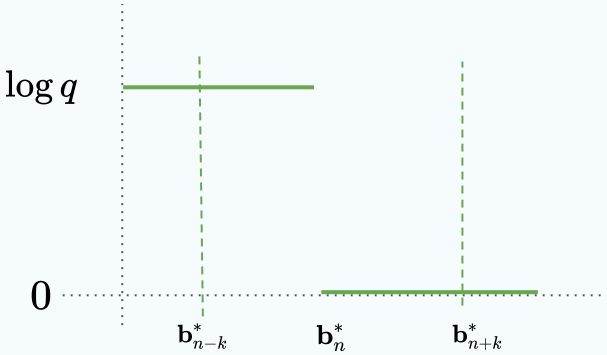
Principle 1: If q is large compared to σ, any short vector will be a multiple of sk

(because geometry and algebra)

Principle 2: Lattice reduction detects it earlier then expected

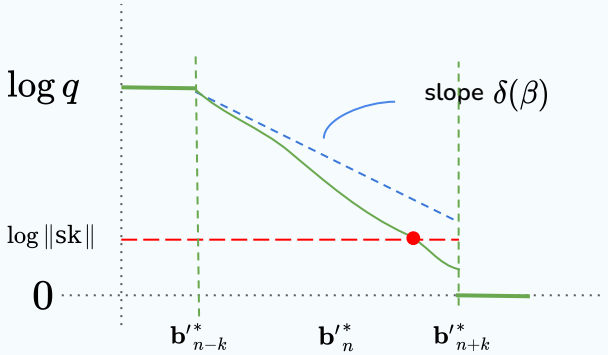
public basis

$q\mathbf{I}_n$	$\mathbf{0}$
$\mathbf{H}$	$\mathbf{I}_n$



block β lattice reduction

heuristic!!



Subfield attacks [ABD16,CJL16] $<$ "LLL in the middle" [KF'17]

Principle 1: If q is large compared to σ, any short vector will be a multiple of sk

(because geometry and algebra)

Principle 2: Lattice reduction detects it earlier then expected

Asymptotically: $\sigma = \mathrm{poly}(n)$ and $q = 2^{\widetilde{\Omega}(\sqrt{n})}$ give a polynomial time attack against NTRU.

Concretely: dimension roughly halved (at least), LLL gives a **practical attack** against several parameters of YASHE.

Subfield attacks [ABD16,CJL16] $<$ "LLL in the middle" [KF'17]

Principle 1: If q is large compared to σ, any short vector will be a multiple of sk (because geometry and algebra) ?

Principle 2: Lattice reduction detects it earlier then expected

Asymptotically: $\sigma = \mathrm{poly}(n)$ and $q = 2^{\widetilde{\Omega}(\sqrt{n})}$ give a polynomial time attack against NTRU.

Concretely: dimension roughly halved (at least), LLL gives a **practical attack** against several parameters of YASHE.

Subfield attacks [ABD16,CJL16] $<$ "LLL in the middle" [KF'17] $=$ MiNTRU, without the ring [L*20]

For $d > 1$, algebra just guarantees that $\mathcal{L}_{\mathrm{sk}}$ has large rank

Principle 3: One just needs a large rank and very dense sublattice to be "overstretched"

Principle 3bis: Too small "additional errors" do not help (aka. inhomogeneous version)

Subfield attacks [ABD16,CJL16] $<$ "LLL in the middle" [KF'17] $=$ MiNTRU, without the ring [L*20]

For $d > 1$, algebra just guarantees that $\mathcal{L}_{\mathrm{sk}}$ has large rank

Principle 3: One just needs a large rank and very dense sublattice to be "overstretched"

Principle 3bis: Too small "additional errors" do not help (aka. inhomogeneous version)

first block of a cipher : $\mathbf{S}^{-1}(\mathbf{I}_n - \mathbf{E}) = \mathbf{C} \mod q$

ternary?

Subfield attacks [ABD16,CJL16] $<$ "LLL in the middle" [KF'17] $=$ MiNTRU, without the ring [L*20]

For $d > 1$, algebra just guarantees that $\mathcal{L}_{\mathrm{sk}}$ has large rank

Principle 3: One just needs a large rank and very dense sublattice to be "overstretched"

Principle 3bis: Too small "additional errors" do not help (aka. inhomogeneous version)

rows span a lattice of rank k

$$\begin{pmatrix} \mathbf{S} & (\mathbf{I}_n - \mathbf{E}) \\ & q\mathbf{I}_n \end{pmatrix} \cdot \begin{pmatrix} \mathbf{C} \\ -\mathbf{I}_n \end{pmatrix} = \mathbf{0} \mod q$$

ternary?

Subfield attacks [ABD16,CJL16] $<$ "LLL in the middle" [KF'17] $=$ MiNTRU, without the ring [L*20]

For $d > 1$, algebra just guarantees that $\mathcal{L}_{\mathrm{sk}}$ has large rank

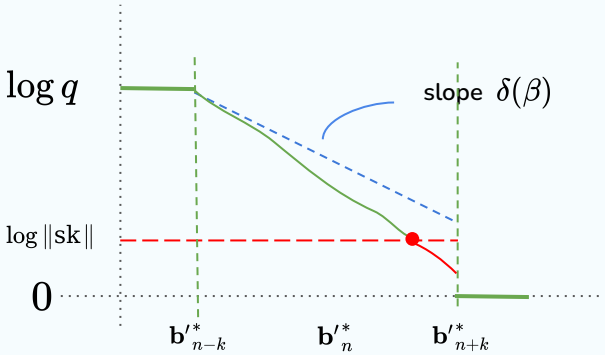
Principle 3: One just needs a large rank and very dense sublattice to be "overstretched"

Principle 3bis: Too small "additional errors" do not help (aka. inhomogeneous version)

Concretely: with $k \approx \frac{n}{4}$, (fplll) BKZ-20 in dimension $\approx \frac{n}{2}$ finds k short vectors in $\mathcal{L}_{\mathrm{sk}}$ in 15 hours. (this is the costly part of the attack).

(larger parameters are even more overstretched and broken by BKZ-25 and LLL [EFK21])

Subfield attacks [ABD16,CJL16] < "LLL in the middle" [KF'17] = MiNTRU, without the ring [L*20] < NTRU Fatigue [DvW21]



Lattice reduction may not find a multiple of sk, but as soon as it starts to find vectors in $\mathcal{L}_{\mathrm{sk}}$, inserting them and restarting "cascades" into recovering $\mathcal{L}_{\mathrm{sk}}$

call this the "DSD event"

Asymptotically: for $\beta \approx \frac{n \log \sigma}{(\log q)^2}$, BKZ-$\beta$ over $\mathcal{L}(\mathbf{b_1}, \ldots \mathbf{b}_{n+\frac{1}{2}\beta})$ triggers DSD at position $n - \frac{1}{2}\beta$

For ternary secret keys, this gives a "overstretched point" at $q = n^{2.484+o(1)}$

Concretely: for matrix-NTRU, the fatigue point is at $q \approx 0.004 \cdot n^{2.484}$ (experimental prediction)

Summing-up & factoring

Principle 1: $\frac{q}{\sigma}$ very large implies a very large gap between an NTRU lattice's minima

Principle 2: In most cryptographic constructions, this implies the existence of a very dense and large rank sublattice.

Principle 3: Lattice reduction overperforms in such context (provable under heuristics) and detects "quickly" the dense sublattice, without running in the full dimension.

Conclusion: Be careful when you set parameters and do not overstretch NTRU's fatigue!

What’s next with NTRU?

Still fatigued?

[DvW21] analyzed “Vintage” NTRU and Matrix-NTRU.
Experiments in dim 128 used to predict average behaviour of NTRU lattices.

- “*Variance of instances’ hardness bigger for vintage NTRU*” -> Investigate
- Same predictions for (power-of-two) cyclotomic rings?
- Are module variants “in-between”?

Interested? [DvW21] explains very well what to do :)

Pseudo-randomness of NTRU-lattices

Random q-ary lattice means $\mathbf{H}$ is uniformly random in R/qR

Let $\mathrm{NTRU}_{q,n,m} : \mathbf{F} \leftarrow \chi_f^{n\times n}, \mathbf{G} \leftarrow \chi_g^{n\times m}$, outputs $\mathbf{F}^{-1}\mathbf{G} \bmod q$

What we know so far:

- σ small means $\mathrm{NTRU}_{q,n,m}$ cannot be pseudo-random
- [SS'13] $\mathrm{NTRU}_{q,1,1}$ is pseudo-random for χ Gaussian with $\sigma = O(d) \cdot \sqrt{q}$

Open problems:

- ▷ Give a subexp distinguisher that would not be a search-solver?
- ▷ What about matrix variants?

Pseudo-randomness of NTRU-lattices

Random q-ary lattice means $\mathbf{H}$ is uniformly random in R/qR

Let $\mathrm{NTRU}_{q,n,m} : \mathbf{F} \leftarrow \chi_f^{n\times n}, \mathbf{G} \leftarrow \chi_g^{n\times m}$, outputs $\mathbf{F}^{-1}\mathbf{G} \bmod q$

What we know so far:

- σ small means $\mathrm{NTRU}_{q,n,m}$ cannot be pseudo-random
- [SS'13] $\mathrm{NTRU}_{q,1,1}$ is pseudo-random for χ Gaussian with $\sigma = O(d) \cdot \sqrt{q}$
- [CPS*X20] $\mathrm{NTRU}_{q,n,m}$ is pseudo-random for χ Gaussian with $\sigma = O(d) \cdot q^{\frac{m}{n+m}}$, <u>if q has large prime factors</u>

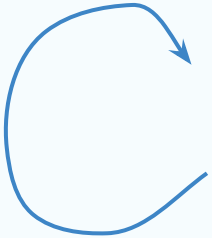
Open problems:

- ▷ Give a subexp distinguisher that would not be a search-solver?
- ▷ What about matrix variants?
- ▷ Can we extend to any q ?

  New technique(s)/tool(s)?

Pseudo-randomness bis & NTRU variants

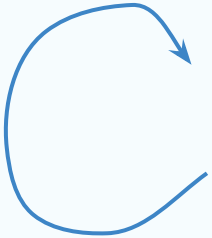
self-reducibility?
reduction between
variants ?



$$\mathrm{NTRU}_{q,n,m} : \mathbf{F} \leftarrow \chi_f^{n\times n}, \mathbf{G} \leftarrow \chi_g^{n\times m},$$
$$\text{outputs } \mathbf{F}^{-1}\mathbf{G} \bmod q$$

Pseudo-randomness bis & NTRU variants

self-reducibility?
reduction between
variants ?



$$\mathrm{NTRU}_{q,n,m} : \mathbf{F} \leftarrow \chi_f^{n\times n}, \mathbf{G} \leftarrow \chi_g^{n\times m},$$
$$\text{outputs } \mathbf{F}^{-1}\mathbf{G} \bmod q$$

“Inhomogeneous” NTRU

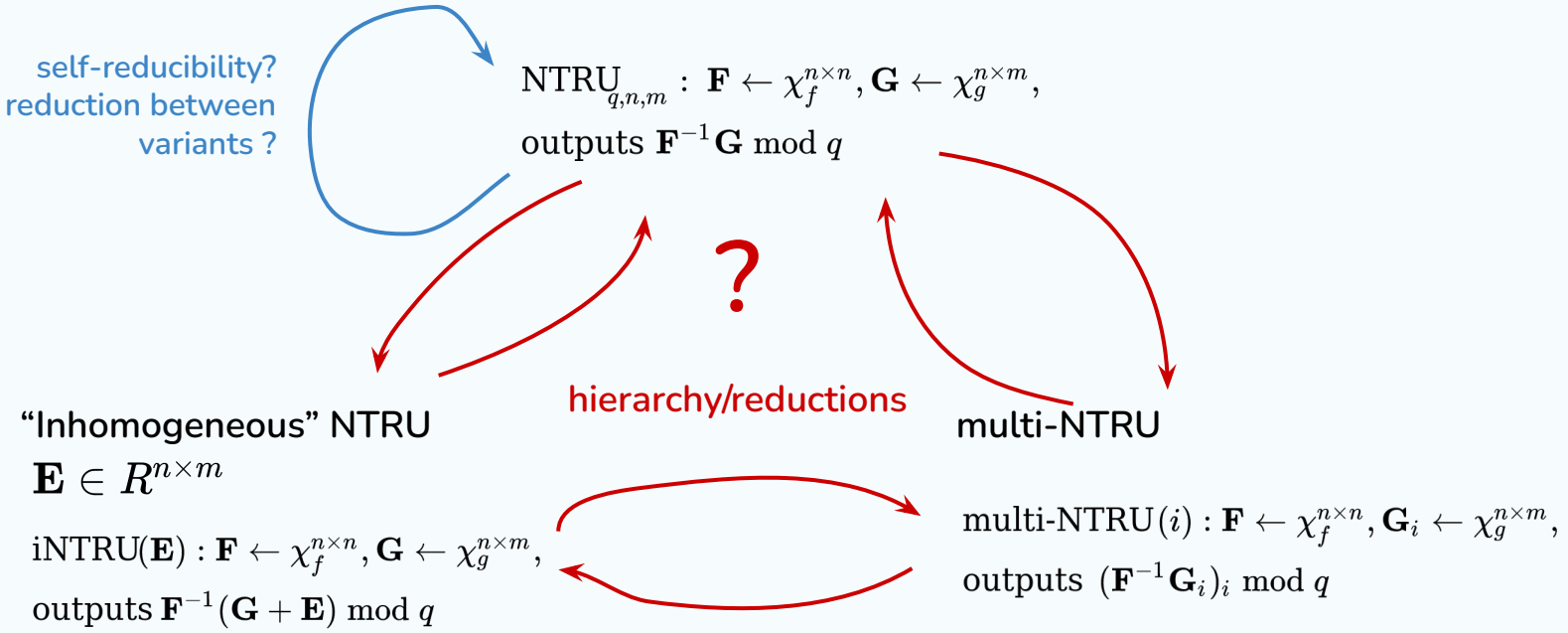
$$\mathbf{E} \in R^{n\times m}$$

$$\mathrm{iNTRU}(\mathbf{E}) : \mathbf{F} \leftarrow \chi_f^{n\times n}, \mathbf{G} \leftarrow \chi_g^{n\times m},$$
$$\text{outputs } \mathbf{F}^{-1}(\mathbf{G}+\mathbf{E}) \bmod q$$

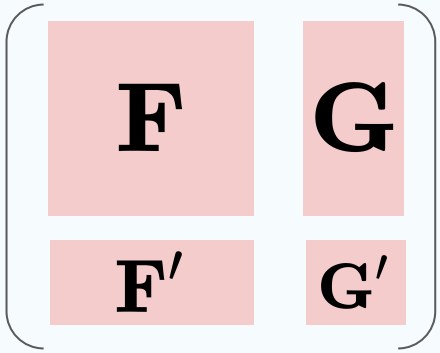
multi-NTRU

$$\text{multi-NTRU}(i) : \mathbf{F} \leftarrow \chi_f^{n\times n}, \mathbf{G}_i \leftarrow \chi_g^{n\times m},$$
$$\text{outputs } (\mathbf{F}^{-1}\mathbf{G}_i)_i \bmod q$$

Pseudo-randomness bis & NTRU variants



Trapdoors and the unimodularity problem

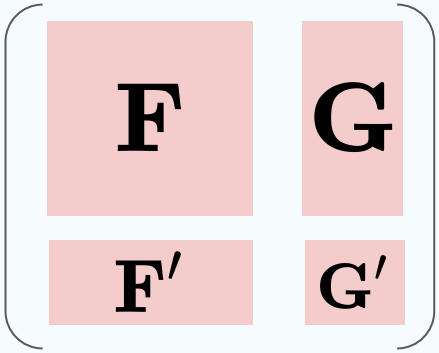


Completing (F,G) into a full basis of an NTRU lattice can be done in poly time

For crypto, we also need **(F',G') short** and **the GSO to be balanced**

And it should be **practically fast** to find one such (Lattice, basis)

Trapdoors and the unimodularity problem



Completing (F,G) into a full basis of an NTRU lattice can be done in poly time

For crypto, we also need **(F',G') short** and **the GSO to be balanced**

And it should be **practically fast** to find one such (Lattice, basis)

Case $n = m = 1$:

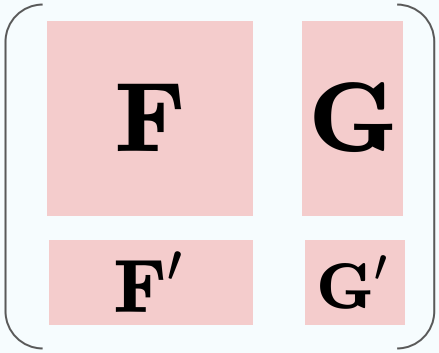
[HPSS00]

Power-of-two cyclo:
[DLP'14]+[EFGRTT*Y22]

+[PP19] (fast with towers)

Other rings?

Trapdoors and the unimodularity problem



Completing (F,G) into a full basis of an NTRU lattice can be done in poly time

For crypto, we also need **(F',G') short** and **the GSO to be balanced**

And it should be **practically fast** to find one such (Lattice, basis)

Case $n = m = 1$:

[HPSS00]

Power-of-two cyclo:
[DLP'14]+[EFGRTT*Y22]

+[PP19] (fast with towers)

Other rings?

Case $n \geq m = 1$:

[CPS*X20]+[CKKS20]

Not well-studied, and only power-of-two cyclo.

Case $n \geq m > 1$:

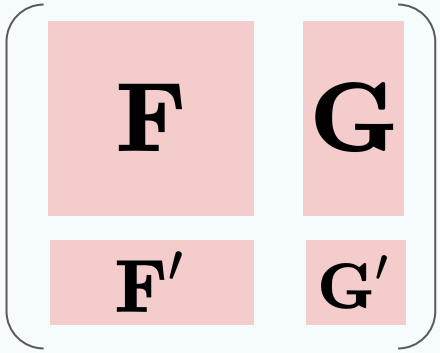
Nothing yet.

Seems like a difficult problem.

"ad-hoc" approaches?

reduction from/to module-SVP?

Trapdoors and the unimodularity problem



Completing (F,G) into a full basis of an NTRU lattice can be done in poly time

For crypto, we also need **(F',G') short** and **the GSO to be balanced**

And it should be **practically fast** to find one such (Lattice, basis)

Thank you! Questions?