On the security of the m-RLWE problem

Carl Bootland, Wouter Castryck, Frederik Vercauteren

14th Algorithmic Number Theory Symposium

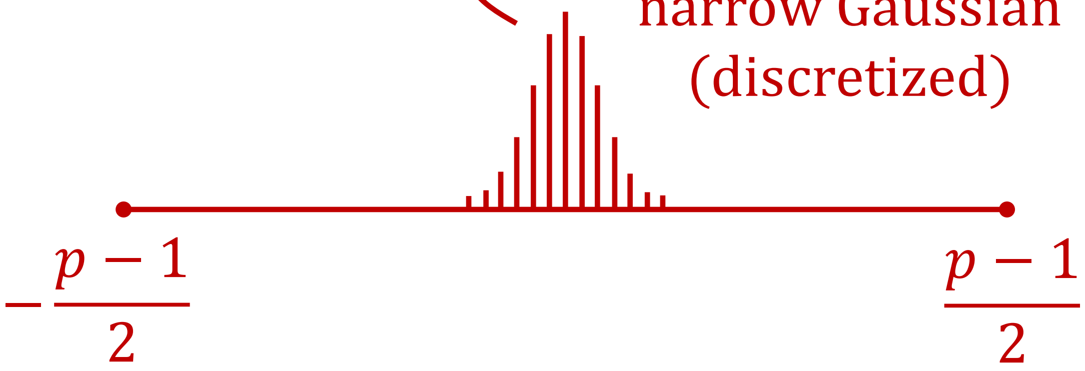
Auckland, New Zealand, June 29 – July 4, 2020

1. The **LWE problem** (Regev '05)

"learning with errors"

System of **approximate** linear equations over $\mathbf{F}_p$:

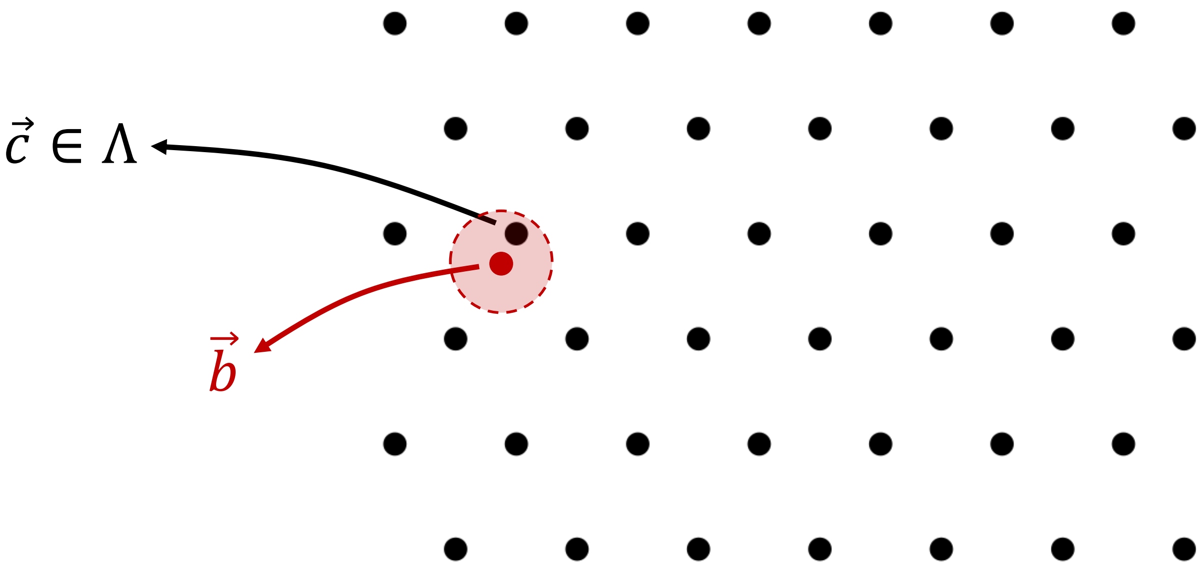
$$\begin{cases} a_{11}s_1 + a_{12}s_2 + \dots + a_{1n}s_n \approx c_1 + e_1 =: b_1 \\ \qquad\qquad\qquad \vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ a_{m1}s_1 + a_{m2}s_2 + \dots + a_{mn}s_n \approx c_m + e_m =: b_m \end{cases}$$



narrow Gaussian (discretized)

1. The **LWE problem** (Regev ‘05)

“learning with errors”



System of **approximate** linear equations over $\mathbf{F}_p$:

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \cdot \begin{pmatrix} s_1 \\ \vdots \\ s_n \end{pmatrix} \approx \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}$$

Lattice:

$$\Lambda = \left\{ \vec{y} \in \mathbf{Z}^m \;\middle|\; \begin{array}{l} A \cdot \vec{s} = \vec{y} \text{ has a} \\ \text{solution mod } p \end{array} \right\}$$

Goal: find $s_1, s_2, \dots, s_n$ (requires $m > n$ or extra assumptions on the s_i’s).

Notes:

- ~~Gaussian elimination?~~

  errors heap up and become indistuinguishable from uniform

- Can be viewed as instance of bounded distance decoding (BDD)

1. The **LWE problem** (Regev ‘05)

Good:

- ➢ Flexible and versatile

  ↪ key exchange, signatures, homomorphic encryption, …

- ➢ Random self-reducible

  ↪ average case as hard as worst case

- ➢ No known quantum attacks

  ↪ of the 26 second-round contenders to the NIST competition,
  9 schemes are based on some form of LWE

Bad:

- ➢ Quadratic key size

  ↪ need $\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}, \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}$ to hide $\begin{pmatrix} s_1 \\ \vdots \\ s_n \end{pmatrix}$

2. The **RLWE problem** (Lyubashevsky, Peikert, Regev ‘12)

“ring learning with errors”

Idea to reduce key size: use **structured** matrices, such as circulant matrices

$$\begin{pmatrix} \boxed{\begin{matrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{matrix}} & \begin{matrix} a_n \\ a_1 \\ \vdots \\ a_{n-1} \end{matrix} & \begin{matrix} \cdots \\ \cdots \\ \ddots \\ \cdots \end{matrix} & \begin{matrix} a_2 \\ a_3 \\ \vdots \\ a_1 \end{matrix} \end{pmatrix} \cdot \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{pmatrix} \approx \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

sufficient to store first column

matrix of multiplication by

$$a_1 + a_2x + \cdots + a_nx^{n-1}$$

in the ring $\mathbf{F}_p[x]/(x^n - 1)$

2. The **RLWE problem** (Lyubashevsky, Peikert, Regev ‘12)

↳ “ring learning with errors”

Idea to reduce key size: use **structured** matrices, such as circulant matrices

$$\underset{\substack{\shortparallel \\ a(x)}}{(a_1 + a_2x + \cdots + a_nx^{n-1})} \cdot \underset{\substack{\shortparallel \\ s(x)}}{(s_1 + s_2x + \cdots + s_nx^{n-1})} \approx \underset{\substack{\shortparallel \\ b(x) = c(x) + e(x)}}{b_1 + b_2x + \cdots + b_nx^{n-1}}$$

Security now depends on BDD in structured lattices.

↳ Can this structure be exploited?

2. The **RLWE problem** (Lyubashevsky, Peikert, Regev ‘12)

“ring learning with errors”

! Note: this does **not** apply to properly instantiated RLWE

Evaluation attack (Eisenträger, Hallgren, Lauter ‘14):

$$\frac{\mathbf{F}_p[x]}{(x^n - 1)} \to \mathbf{F}_p : f(x) \mapsto f(1)$$

most popular proper use:

$$\frac{\mathbf{F}_p[x]}{(x^n + 1)} \quad (\text{with } n = 2^k)$$

Therefore:

$$a(x) \cdot s(x) = b(x) - e(x)$$

$$e(x) = b(x) - a(x) \cdot s(x)$$

$$e(1) = b(1) - a(1) \cdot s(1)$$

$$\parallel$$

$$\sum_i e_i \longrightarrow \text{small}$$

leaks $s(1)$ when given enough samples

3. The **m-RLWE problem** (Pedrouzo-Ulloa, Troncoso-Pastoriza, Pérez-González ‘15)

↳ “multivariate ring learning with errors”

Idea: use quotients of **multivariate** polynomial rings, such as

$$\frac{\mathbf{F}_p[x,y]}{(x^{n_1}+1,y^{n_2}+1)} \qquad (n_1 = 2^{k_1}, n_2 = 2^{k_2}).$$

Samples now look like:

$$\underset{\substack{\shortparallel \\ a(x,y)}}{\left(\sum_{i=0}^{n_1-1}\sum_{j=0}^{n_2-1} a_{ij}x^i y^j\right)} \cdot \underset{\substack{\shortparallel \\ s(x,y)}}{\left(\sum_{i=0}^{n_1-1}\sum_{j=0}^{n_2-1} s_{ij}x^i y^j\right)} \approx \underset{\substack{\shortparallel \\ b(x,y) = c(x,y) + e(x,y)}}{\sum_{i=0}^{n_1-1}\sum_{j=0}^{n_2-1} b_{ij}x^i y^j}$$

Motivation: matrix/tensor arithmetic in homomorphic encryption (e.g., signal processing).

3. The **m-RLWE problem** (Pedrouzo-Ulloa, Troncoso-Pastoriza, Pérez-González '15)

→ "multivariate ring learning with errors"

This paper: m-RLWE **falls prey to evaluation attack** (also observed by Cheon, Kim, Yhee '18):

$$\frac{\mathbf{F}_p[x,y]}{(x^{n_1}+1, y^{n_2}+1)} = \frac{\frac{\mathbf{F}_p[x]}{(x^{n_1}+1)}[y]}{(y^{n_2}+1)}$$

← admits x^{n_1/n_2} as a root (assume $n_2|n_1$)

3. The **m-RLWE problem** (Pedrouzo-Ulloa, Troncoso-Pastoriza, Pérez-González ‘15)

“multivariate ring learning with errors”

This paper: m-RLWE **falls prey to evaluation attack** (also observed by Cheon, Kim, Yhee ‘18):

$$\frac{\mathbf{F}_p[x,y]}{(x^{n_1}+1, y^{n_2}+1)} \to \frac{\mathbf{F}_p[x]}{(x^{n_1}+1)} : f(x,y) \mapsto f(x, x^{n_1/n_2})$$

As before:

$$\begin{aligned} a(x,y)\cdot s(x,y) &= b(x,y) - e(x,y) \\ e(x,y) &= b(x,y) - a(x,y)\cdot s(x,y) \\ e(x,x^{n_1/n_2}) &= b(x,x^{n_1/n_2}) - a(x,x^{n_1/n_2})\cdot s(x,x^{n_1/n_2}) \end{aligned}$$

solving RLWE in dim n_1 leaks $s(x,x^{n_1/n_2})$

$$e(x,x^{n_1/n_2}) = \sum_{i=0}^{n_1-1} \underbrace{\left(\sum_{j=0}^{n_2-1} \pm e_{r(i,j),j} \right)}_{\text{small}} x^i$$

3. The **m-RLWE problem** (Pedrouzo-Ulloa, Troncoso-Pastoriza, Pérez-González ‘15)

→ “<u>m</u>ultivariate <u>r</u>ing <u>l</u>earning <u>w</u>ith <u>e</u>rrors”

It does not stop there:

$$\frac{\dfrac{\mathbf{F}_p[x]}{(x^{n_1}+1)}[y]}{(y^{n_2}+1)}$$

→ factors **completely** as

$$\left(y - x^{n_1/n_2}\right)\left(y - x^{3n_1/n_2}\right)\cdots\left(y - x^{(2n_2-1)n_1/n_2}\right)$$

Thus: solving n_2 instances of RLWE in dim n_1 leaks

$$\left.\begin{array}{c} s(x, x^{n_1/n_2}) \\ s(x, x^{3n_1/n_2}) \\ \vdots \\ s(x, x^{(2n_2-1)n_1/n_2}) \end{array}\right] \xrightarrow{\text{easy linear algebra}} s(x,y).$$

3. The **m-RLWE problem** (Pedrouzo-Ulloa, Troncoso-Pastoriza, Pérez-González ‘15)

“multivariate ring learning with errors”

It does not stop there:

$$\frac{\frac{\mathbf{F}_p[x]}{(x^{n_1}+1)}[y]}{(y^{n_2}+1)}$$

factors **completely** as

$$\left(y - x^{n_1/n_2}\right)\left(y - x^{3n_1/n_2}\right)\cdots\left(y - x^{(2n_2-1)n_1/n_2}\right)$$

Concrete example:

- $n_1 = n_2 = 128, p = 2^{42} + 15, \sigma = 1$ were expected to reach security level > 2500,
- actual security level ≈ 32, easy to run full break in practice.

3. The **m-RLWE problem** (Pedrouzo-Ulloa, Troncoso-Pastoriza, Pérez-González ‘15)

↳ “multivariate ring learning with errors”

High-level viewpoint: hardness of LWE in (the reduction mod p of)

tensor product $\underbrace{\mathbf{Z}[\zeta_{2n_1}] \otimes_{\mathbf{Z}} \mathbf{Z}[\zeta_{2n_2}]}_{\dim = n_1 n_2}$ reduces to that in **compositum** $\underbrace{\mathbf{Z}[\zeta_{2n_1}, \zeta_{2n_2}] = \mathbf{Z}[\zeta_{2n_1}]}_{\dim = n_1}$

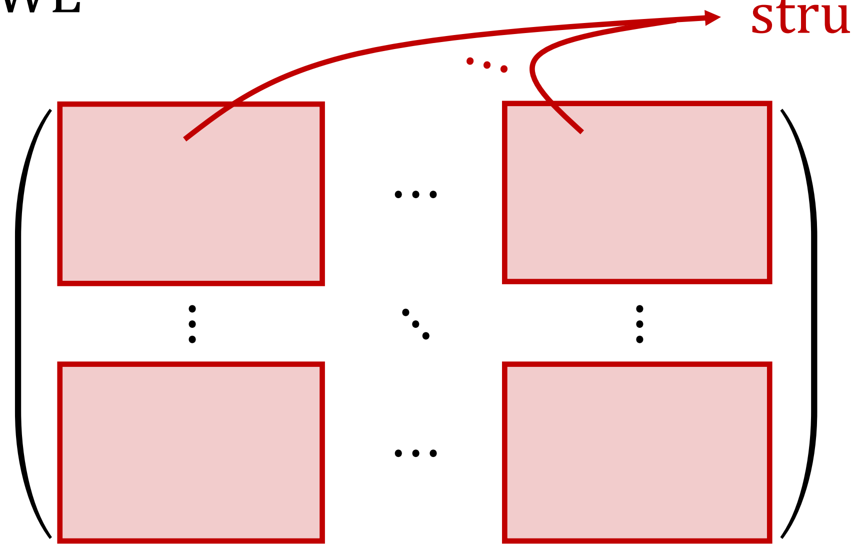
Generalizes:

- from 2-RLWE to m-RLWE, i.e., to arbitrary number of factors,
- to arbitrary products with Galois compositum and good error growth.

4. The **MLWE problem** (Langlois, Stehlé ‘15)

“module learning with errors”

Interpolation LWE ⟶ RLWE



structured blocks:

matrices of multiplication by elements of $\mathbf{F}_p[x]/(x^n + 1)$

⚠ DANGER

4. The **MLWE problem** (Langlois, Stehlé '15)

“module learning with errors”

Interpolation LWE ⟶ RLWE

what if, in a similar attempt to save space,
we endow this module with a ring structure …

$$\begin{pmatrix} a_{11}(x) & \cdots & a_{1m}(x) \\ \vdots & \ddots & \vdots \\ a_{m1}(x) & \cdots & a_{mm}(x) \end{pmatrix}$$

… like

$$\frac{\frac{\mathbf{F}_p[x]}{(x^n+1)}[y]}{(y^m+1)} \ ?$$

☣

… and choose this matrix to be a
matrix of multiplication?

linear transformation of the rank m **module**

$$\frac{\mathbf{F}_p[x]}{(x^n+1)} \oplus \cdots \oplus \frac{\mathbf{F}_p[x]}{(x^n+1)}$$

Note: warning does not apply to all parameters
(e.g., Kyber-768 uses $n = 256$ and $m = 3$).