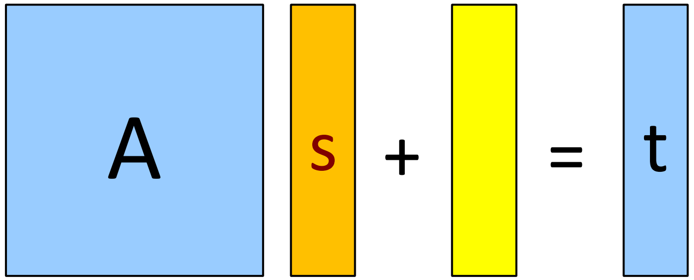
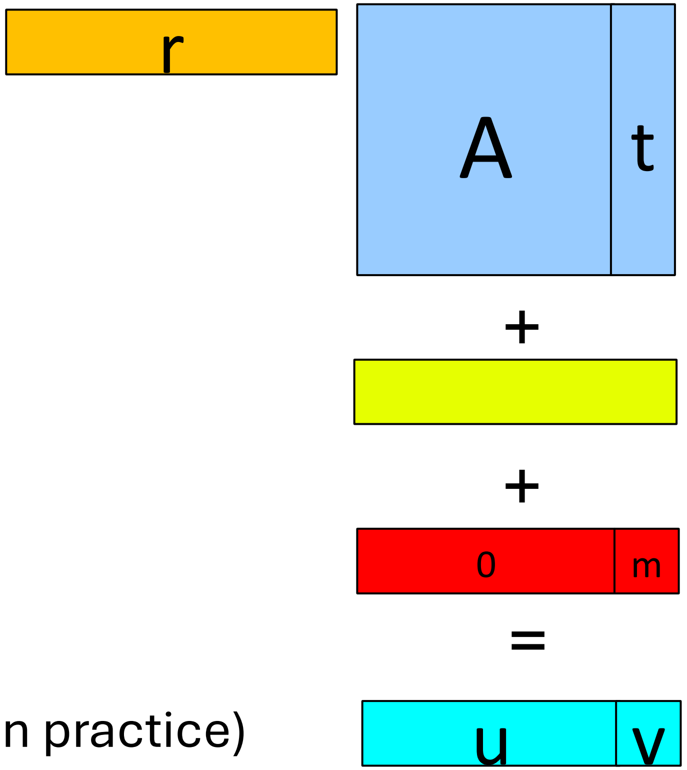
Lattice Cryptography
(2. Encryption Using Polynomial Rings)

Vadim Lyubashevsky

IBM Research Europe, Zurich

Encryption Scheme





Sources of inefficiency:

1. Multiplication As takes quadratic time (not so fast in practice)

2. <r,t> is an integer, and so we can only encrypt 1 bit at a time

Use Polynomials

f(x) is a polynomial $x^n + a_{n-1}x^{n-1} + \ldots + a_1x + a_0$

R = $Z_p[x]/(f(x))$ is a polynomial ring with

- Addition mod p
- Polynomial multiplication mod p and f(x)

Each element of R consists of n elements in Z_p

In R:

- small+small = small
- small*small = small (depending on f(x))

Example ring $Z_{17}[X]/(X^4+1)$

Elements are $z(X)=z_3X^3+z_2X^2+z_1X+z_0$

where z_i are integers mod 17

Addition is the usual coordinate-wise addition

Multiplication is the usual polynomial multiplication

followed by reduction modulo X^4+1

Multiplication and coefficient growth in **$Z[X]/(X^4+1)$**

Polynomial multiplication as Matrix-Vector multiplication

$a(X)=a_3X^3+a_2X^2+a_1X+a_0$
$b(x)=b_3X^3+b_2X^2+b_1X+b_0$

$a(X)*b(X) = b(X)*a_0 + (b(X) * X) * a_1 + (b(X)*X^2) * a_2 + (b(X)*X^3) * a_3$

$$\begin{bmatrix} | & | & | & | \\ b(X) & b(X)*X & b(X)*X^2 & b(X)*X^3 \\ | & | & | & | \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \xrightarrow{\text{when the ring is } Z[X]/(X^4+1)} \begin{bmatrix} b_0 & -b_3 & -b_2 & -b_1 \\ b_1 & b_0 & -b_3 & -b_2 \\ b_2 & b_1 & b_0 & -b_3 \\ b_3 & b_2 & b_1 & b_0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

because $b(X)*X^i$ has small coefficients, the coefficients grow “slowly” during multiplication

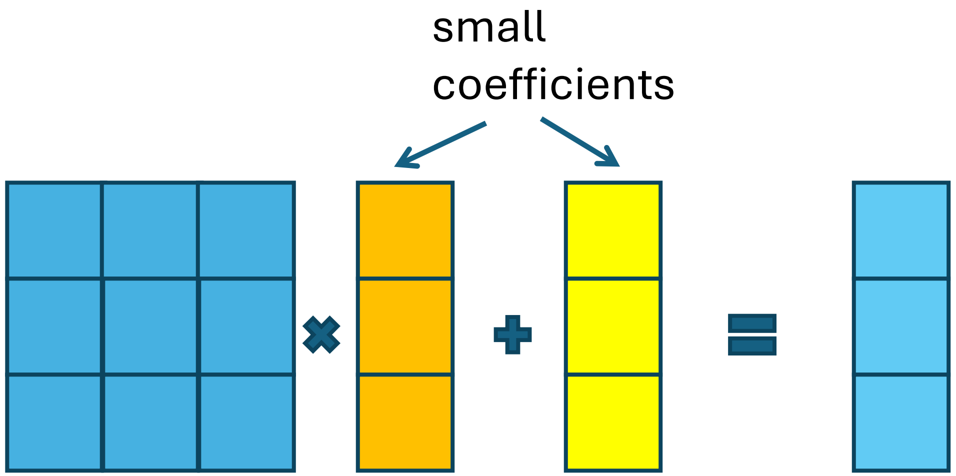
if the ring were $Z[X]/(X^n + 2X^{n-1} + 1)$, then $b(X)*X^{n-1}$ can have coefficients 2^n times larger than $b(X)$

Operations

Basic Computational Domain:

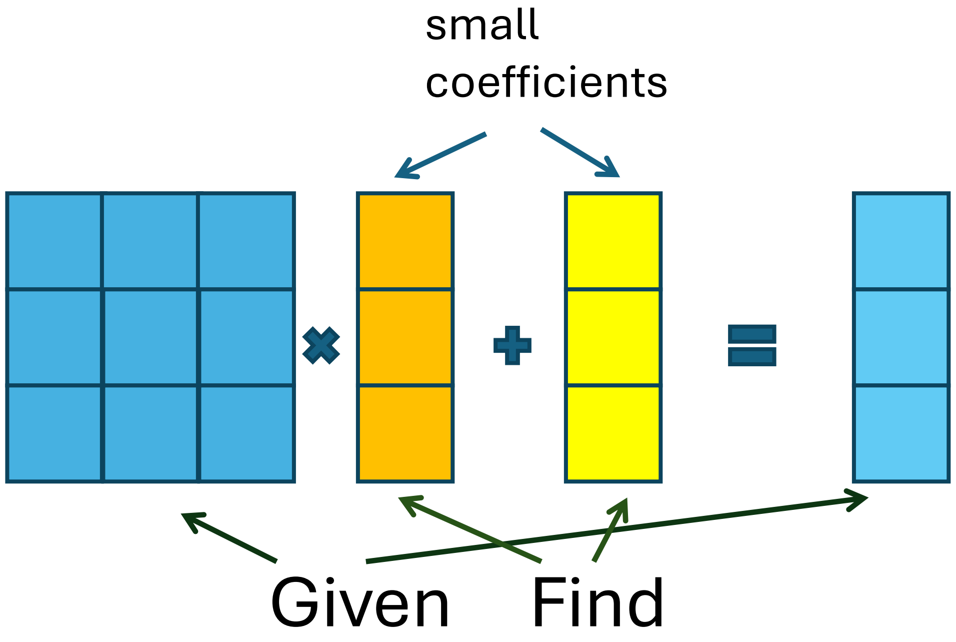
Polynomial ring $Z_p[X]/(X^n+1)$

Operations used in the schemes: $+$ and $\times$ in the ring:

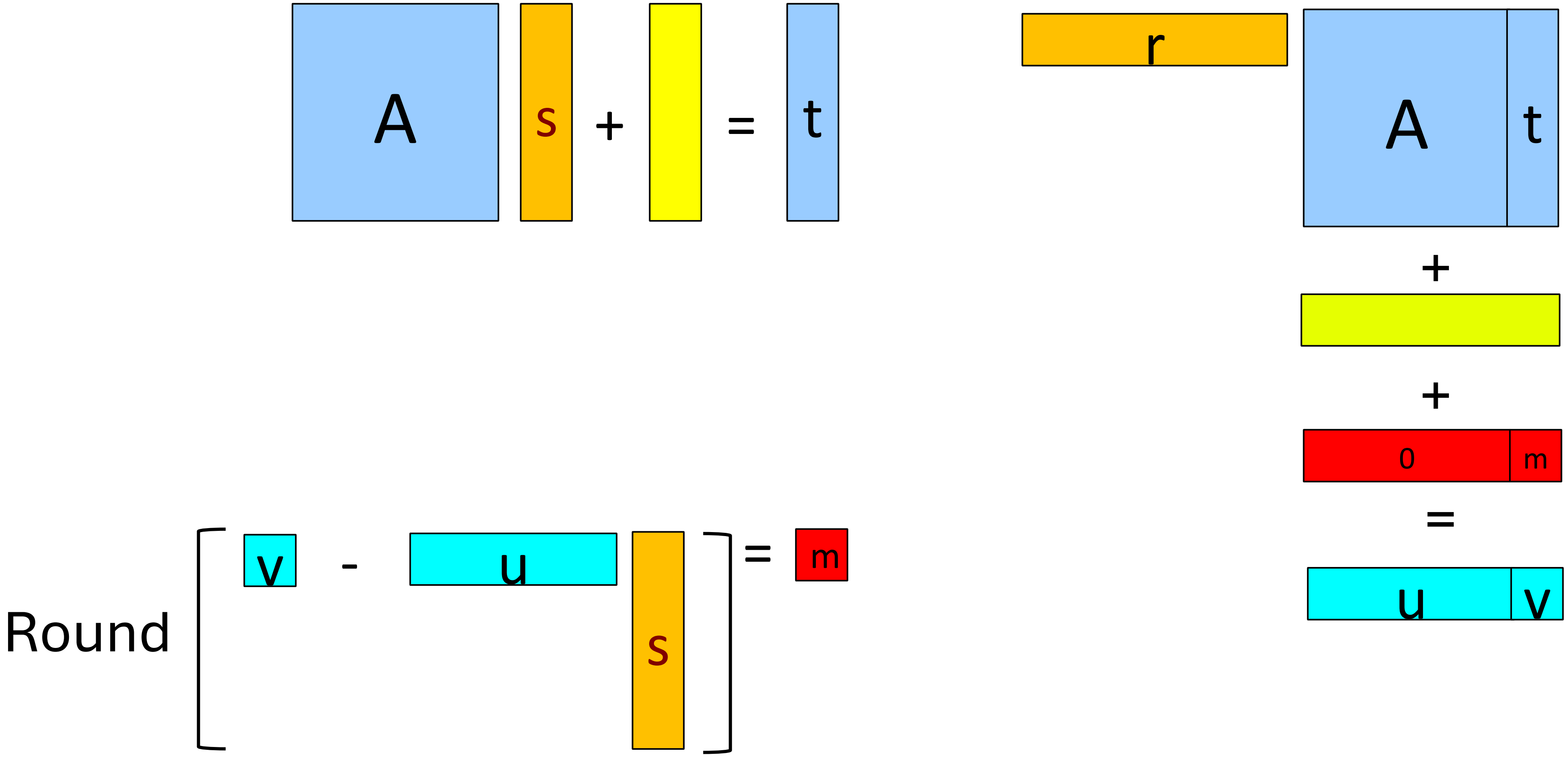


Hard problem LWE over Rings

Basic Hard Problem:

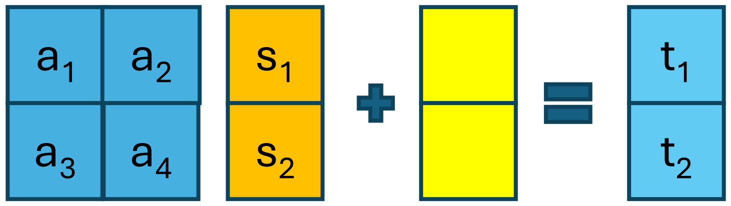


Encryption Scheme

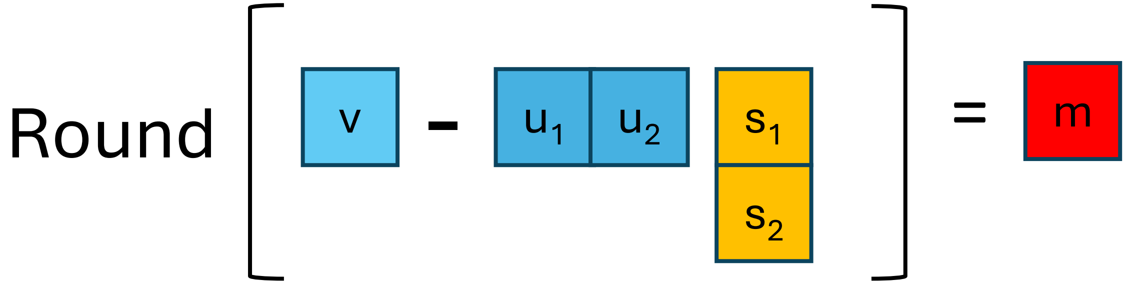


Encryption scheme over $Z_q[X]/(X^n+1)$

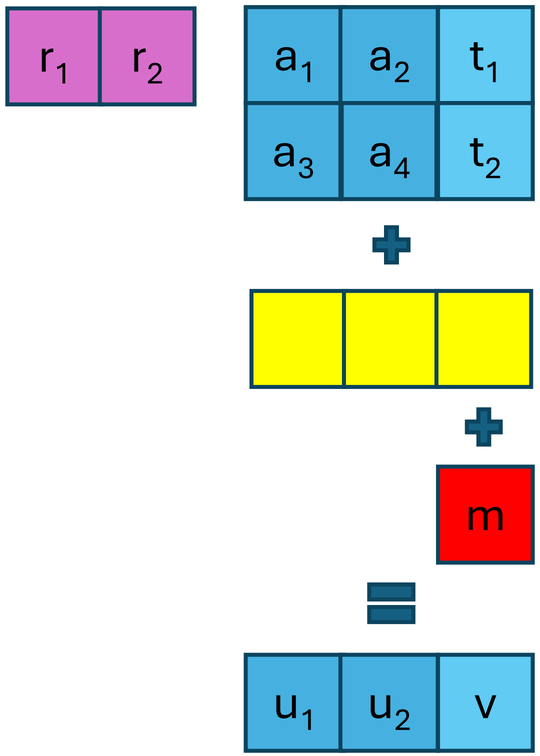
<u>Key Generation</u>



<u>Decryption</u>



<u>Encryption</u>



Operations in $R=Z_p[X]/(X^n+1)$

- Additions are easy and cheap

- To get cheap multiplications, we pick p = 1 (mod 2n)

  Can then write $X^n+1 = (X-r_1)(X-r_2) \cdot \cdot \cdot (X-r_n) \bmod p$

- Polynomials in R can be represented in two ways

  Coefficient representation

  $$f = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1}$$

  Chinese Remainder Theorem (CRT) representation

  $$\hat{f} = (f(r_1), f(r_2), \ldots, f(r_n))$$

Multiplication using NTT (Number Theoretic Transform)

Want to multiply $f * g$, which are in coefficient representation

1. $f \rightarrow \hat{f} = (f(r_1), f(r_2), \dots, f(r_n))$
2. $g \rightarrow \hat{g} = (g(r_1), g(r_2), \dots, g(r_n))$
3. $\widehat{f * g} = \left(f(r_1) * g(r_1), \dots, f(r_n) * g(r_n)\right)$
4. $\widehat{f * g} \rightarrow f * g$

Example in $R=Z_{17}[x]/(X^4+1)$

$$x^4 + 1 = (x-2)(x+2)(x-8)(x+8)\ mod\ 17$$

We will show how to efficiently convert between the coefficient and CRT representations

This is the main part of polynomial multiplication

$$f \to \hat{f}$$

$$f \ mod \ (x^4 + 1)$$

$$f_{11} = f \ mod \ (x^2 - 4) \qquad f_{12} = f \ mod \ (x^2 + 4)$$

$$f_{21} = f_{11} \ mod \ (x - 2) \qquad f_{22} = f_{11} \ mod \ (x + 2) \qquad f_{23} = f_{12} \ mod \ (x - 8) \qquad f_{24} = f_{12} \ mod \ (x + 8)$$

$$f(2) \qquad f(-2) \qquad f(8) \qquad f(-8)$$

$$f \rightarrow \hat{f}$$

$$f \ mod \ (x^4 + 1)$$

$$\mathbf{1 - 4x + 3x^2 - 5x\hat{}3}$$

$$f_{11} = f \ mod \ (x^2 - 4)$$

$$\mathbf{1 - 4x + 3 * 4 - 5 * 4x}$$

$$f_{12} = f \ mod \ (x^2 + 4)$$

$$\mathbf{1 - 4x - 3 * 4 + 5 * 4x}$$

$f_{21} = f_{11} \ mod \ (x - 2)$ | $f_{22} = f_{11} \ mod \ (x + 2)$ | $f_{23} = f_{12} \ mod \ (x - 8)$ | $f_{24} = f_{12} \ mod \ (x + 8)$

$f(2)$ | $f(-2)$ | $f(8)$ | $f(-8)$

$$f \rightarrow \hat{f}$$

$$\boxed{f \ mod \ (x^4 + 1)}$$

$\mathbf{1 - 4x + 3x^2 - 5x}$**^3**

$$\boxed{f_{11} = f \ mod \ (x^2 - 4)} \qquad \boxed{f_{12} = f \ mod \ (x^2 + 4)}$$

$\mathbf{-4 - 7x}$ $\qquad$ $\mathbf{6 - x}$

$$\boxed{f_{21} = f_{11} \ mod \ (x - 2)} \quad \boxed{f_{22} = f_{11} \ mod \ (x + 2)} \quad \boxed{f_{23} = f_{12} \ mod \ (x - 8)} \quad \boxed{f_{24} = f_{12} \ mod \ (x + 8)}$$

$\mathbf{-1}$ $\qquad$ $\mathbf{-7}$ $\qquad$ $\mathbf{-2}$ $\qquad$ $\mathbf{-3}$

$$\boxed{f(2)} \quad \boxed{f(-2)} \quad \boxed{f(8)} \quad \boxed{f(-8)}$$

$$\hat{f} \to f$$

$f \; mod \; (x^4 + 1)$

$f_{11} = f \; mod \; (x^2 - 4)$

$f_{12} = f \; mod \; (x^2 + 4)$

$\boldsymbol{a + bx}$

$\boldsymbol{a + 2b}$ **= -1**

$\boldsymbol{a - 2b}$ **= -7**

$f_{21} = f_{11} \; mod \; (x - 2)$

$f_{22} = f_{11} \; mod \; (x + 2)$

$f_{23} = f_{12} \; mod \; (x - 8)$

$f_{24} = f_{12} \; mod \; (x + 8)$

$\boldsymbol{-1}$ $\boldsymbol{-7}$ $\boldsymbol{-2}$ $\boldsymbol{-3}$

$f(2)$ $f(-2)$ $f(8)$ $f(-8)$

$$\hat{f} \to f$$

$f \; mod \; (x^4 + 1)$

$1 - 4x + 3x^2 - 5x$^3

$f_{11} = f \; mod \; (x^2 - 4)$

$-4 - 7x$

$f_{12} = f \; mod \; (x^2 + 4)$

$6 - x$

$f_{21} = f_{11} \; mod \; (x - 2)$

-1

$f(2)$

$f_{22} = f_{11} \; mod \; (x + 2)$

-7

$f(-2)$

$f_{23} = f_{12} \; mod \; (x - 8)$

-2

$f(8)$

$f_{24} = f_{12} \; mod \; (x + 8)$

-3

$f(-8)$

Time complexity

$$X^n - r \equiv (X^{n/2} - \sqrt{r})(X^{n/2} + \sqrt{r})$$

$$a = \sum_{i=0}^{n-1} a_i X^i,$$

$$a \mod X^{n/2} - \sqrt{r} = \sum_{i=0}^{n/2-1} b_i X^i,$$

$$a \mod X^{n/2} + \sqrt{r} = \sum_{i=0}^{n/2-1} c_i X^i,$$

Computing the NTT

$$b_i = a_i + \sqrt{r} \cdot a_{i+n/2}$$

$$c_i = a_i - \sqrt{r} \cdot a_{i+n/2}$$

n additions and n/2 multiplications

Computing the inverse NTT

$$2 \cdot a_i = b_i + c_i$$

$$2 \cdot a_{i+n/2} = (\sqrt{r})^{-1} \cdot (b_i - c_i)$$

n additions and n/2 multiplications

Computation time for $R=Z_p[x]/(x^n+1)$

$f \rightarrow \hat{f}$ and $\hat{f} \rightarrow f$

log n levels

(n/2 multiplications) and (n additions) modulo p per level

Problem session

Implement NTT and NTT^{-1} over $Z_{257}[X]/(X^{32} + 1)$

- Create a tree (given to you) of roots and inverse roots that will be used for all NTT and NTT^{-1} computations
- All the operations are to be done “in place” -- do not create new vectors
  - At level 0, your vector consists of a polynomial with 32 coefficients
  - At level 1, your vector consists of 2 polynomials with 16 coefficients
  - …
  - At the bottom level, your vector consists of 32 polynomials with 1 coefficient

Supplementary reading: Section 4 of
https://github.com/VadimLyubash/LatticeTutorial