On Ideal Lattices and Learning With Errors Over Rings

Vadim Lyubashevsky[1] $\boxed{\text{Chris Peikert}}$[2] Oded Regev[1]

[1]Tel Aviv University

[2]Georgia Institute of Technology

Eurocrypt 2010

The 'Learning With Errors' Problem [Regev'05]

- Parameters: dimension n, prime modulus $q = \text{poly}(n)$.

The 'Learning With Errors' Problem [Regev'05]

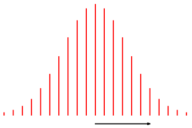
- Parameters: dimension n, prime modulus $q = \text{poly}(n)$.
- **Search:** find secret $\mathbf{s} \in \mathbb{Z}_q^n$ given many 'noisy inner products'

$$
\begin{aligned}
\mathbf{a}_1 \leftarrow \mathbb{Z}_q^n \quad &, \quad b_1 \approx \langle \mathbf{a}_1 \,,\, \mathbf{s} \rangle \bmod q \\
\mathbf{a}_2 \leftarrow \mathbb{Z}_q^n \quad &, \quad b_2 \approx \langle \mathbf{a}_2 \,,\, \mathbf{s} \rangle \bmod q \\
&\vdots
\end{aligned}
$$

The ‘Learning With Errors’ Problem [Regev’05]

- Parameters: dimension n, prime modulus $q = \text{poly}(n)$.
- **Search:** find <u>secret</u> $\mathbf{s} \in \mathbb{Z}_q^n$ given many ‘noisy inner products’

$$\begin{array}{lcl} \mathbf{a}_1 \leftarrow \mathbb{Z}_q^n & , & b_1 = \langle \mathbf{a}_1 \,,\, \mathbf{s} \rangle + e_1 \in \mathbb{Z}_q \\ \mathbf{a}_2 \leftarrow \mathbb{Z}_q^n & , & b_2 = \langle \mathbf{a}_2 \,,\, \mathbf{s} \rangle + e_2 \in \mathbb{Z}_q \\ & \vdots & \end{array}$$

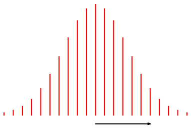


$\sqrt{n} \leq \text{error} \ll q$

The ‘Learning With Errors’ Problem [Regev’05]

- Parameters: dimension n, prime modulus $q = \text{poly}(n)$.
- **Search:** find <u>secret</u> $\mathbf{s} \in \mathbb{Z}_q^n$ given many ‘noisy inner products’

$$\begin{pmatrix} \vdots \\ \mathbf{A}^t \\ \vdots \end{pmatrix} , \quad \begin{pmatrix} \vdots \\ \mathbf{b} \\ \vdots \end{pmatrix} = \mathbf{A}^t\mathbf{s} + \mathbf{e}$$

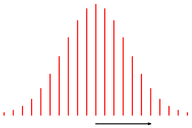


$\sqrt{n} \leq$ error $\ll q$

The 'Learning With Errors' Problem [Regev'05]

- Parameters: dimension n, prime modulus $q = \text{poly}(n)$.
- **Search**: find secret $\mathbf{s} \in \mathbb{Z}_q^n$ given many 'noisy inner products'

$$\begin{pmatrix} \vdots \\ \mathbf{A}^t \\ \vdots \end{pmatrix}, \quad \begin{pmatrix} \vdots \\ \mathbf{b} \\ \vdots \end{pmatrix} = \mathbf{A}^t\mathbf{s} + \mathbf{e}$$



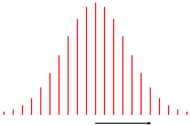
$\sqrt{n} \leq$ error $\ll q$

(After enough uniform $\mathbf{a}_i$'s, secret $\mathbf{s}$ is uniquely determined w/hp.)

The ‘Learning With Errors’ Problem [Regev’05]

- Parameters: dimension n, prime modulus $q = \text{poly}(n)$.
- **Search:** find secret $\mathbf{s} \in \mathbb{Z}_q^n$ given many ‘noisy inner products’

$$\begin{pmatrix} \vdots \\ \mathbf{A}^t \\ \vdots \end{pmatrix} , \quad \begin{pmatrix} \vdots \\ \mathbf{b} \\ \vdots \end{pmatrix} = \mathbf{A}^t\mathbf{s} + \mathbf{e}$$



$\sqrt{n} \leq$ error $\ll q$

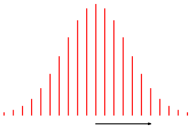
(After enough uniform $\mathbf{a}_i$’s, secret $\mathbf{s}$ is uniquely determined w/hp.)

- **Decision:** distinguish $(\mathbf{A}, \mathbf{b})$ from uniform $(\mathbf{A}, \mathbf{b})$

The 'Learning With Errors' Problem [Regev'05]

- Parameters: dimension n, prime modulus $q = \mathrm{poly}(n)$.
- **Search**: find <u>secret</u> $\mathbf{s} \in \mathbb{Z}_q^n$ given many 'noisy inner products'

$$\left(\begin{array}{c} \vdots \\ \mathbf{A}^t \\ \vdots \end{array} \right) , \quad \left(\begin{array}{c} \vdots \\ \mathbf{b} \\ \vdots \end{array} \right) = \mathbf{A}^t\mathbf{s} + \mathbf{e}$$



$\sqrt{n} \leq$ error $\ll q$

(After enough uniform $\mathbf{a}_i$'s, secret $\mathbf{s}$ is uniquely determined w/hp.)

- **Decision**: distinguish $(\mathbf{A} \;,\; \mathbf{b})$ from <u>uniform</u> $(\mathbf{A} \;,\; \mathbf{b})$

LWE is Hard (...maybe even for quantum!)

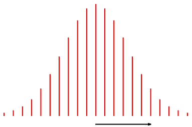
worst case lattice problems $\leq$ search-LWE $\leq$ decision-LWE $\leq$ crypto

(quantum [R'05]) [BFKL'93,R'05]

The 'Learning With Errors' Problem [Regev'05]

- Parameters: dimension n, prime modulus $q = \mathrm{poly}(n)$.
- **Search**: find <u>secret</u> $\mathbf{s} \in \mathbb{Z}_q^n$ given many 'noisy inner products'

$$\left(\begin{array}{c} \vdots \\ \mathbf{A}^t \\ \vdots \end{array} \right) \quad , \quad \left(\begin{array}{c} \vdots \\ \mathbf{b} \\ \vdots \end{array} \right) = \mathbf{A}^t\mathbf{s} + \mathbf{e}$$



$\sqrt{n} \leq$ error $\ll q$

(After enough uniform $\mathbf{a}_i$'s, secret $\mathbf{s}$ is uniquely determined w/hp.)

- **Decision**: distinguish $(\mathbf{A}\ ,\ \mathbf{b})$ from <u>uniform</u> $(\mathbf{A}\ ,\ \mathbf{b})$

LWE is Hard (...maybe even for quantum!)

worst case lattice problems $\leq$ search-LWE $\leq$ decision-LWE $\leq$ crypto

(quantum [R'05]) [BFKL'93,R'05]

- (Also some *classical* hardness for search-LWE [P'09])

LWE is Versatile

What kinds of crypto can we do with LWE?

LWE is Versatile

What kinds of crypto can we do with LWE?

- Public Key Encryption [R'05,PVW'08]

  CCA-Secure PKE (w/o RO) [PW'08,P'09]

LWE is Versatile

What kinds of crypto can we do with LWE?

- Public Key Encryption [R'05,PVW'08]

  CCA-Secure PKE (w/o RO) [PW'08,P'09]

  – – – – – – – – – – –

- Identity-Based Encryption (in RO model) [GPV'08]

  Hierarchical ID-Based Encryption (w/o RO) [CHKP'10,ABB'10]

LWE is Versatile

What kinds of crypto can we do with LWE?

- Public Key Encryption [R'05,PVW'08]

  CCA-Secure PKE (w/o RO) [PW'08,P'09]

  – – – – – – – – – – –

- Identity-Based Encryption (in RO model) [GPV'08]

  Hierarchical ID-Based Encryption (w/o RO) [CHKP'10,ABB'10]

  – – – – – – – – – – –

  UC Oblivious Transfer [PVW'08]

  Leakage Resilience [AGV'09,DGKPV'10,GKPV'10,ADNSWW'10,...]

  Circular/KDM-Secure Encryption [ACPS'09,BHHI'10]

  Quadratic-Formula Homomorphic Encryption [GHV'10]

  Bi-Deniable Encryption [OP'10]

  and more...

LWE is Efficient (. . . sort of)

$$\begin{pmatrix} — & \mathbf{a} & — \end{pmatrix} \begin{pmatrix} | \\ \mathbf{s} \\ | \end{pmatrix} + e \;=\; b \in \mathbb{Z}_q$$

- Getting one extra pseudorandom scalar requires an n-dim inner product

LWE is Efficient (. . . sort of)

$$\begin{pmatrix} — & \mathbf{a} & — \end{pmatrix} \begin{pmatrix} | \\ \mathbf{s} \\ | \end{pmatrix} + e = b \in \mathbb{Z}_q$$

- Getting one extra pseudorandom scalar requires an n-dim inner product
- Can amortize each $\mathbf{a}$ over many secrets $\mathbf{s}_i$, but still $\tilde{O}(n)$ work per scalar output.

LWE is Efficient (...sort of)

$$\left(\text{---}\ \mathbf{a}\ \text{---}\right) \begin{pmatrix} | \\ \mathbf{s} \\ | \end{pmatrix} + e = b \in \mathbb{Z}_q$$

- Getting one extra pseudorandom scalar requires an n-dim inner product
- Can amortize each $\mathbf{a}$ over many secrets $\mathbf{s}_i$, but still $\tilde{O}(n)$ work per scalar output.

- Public key crypto schemes have rather large keys:

$$pk = \underbrace{\begin{pmatrix} \vdots \\ \mathbf{A}^t \\ \vdots \end{pmatrix}}_{n}, \quad \left.\begin{pmatrix} \vdots \\ \mathbf{b} \\ \vdots \end{pmatrix}\right\} \Omega(n)$$

LWE is Efficient (…sort of)

$$(\text{— } \mathbf{a} \text{ —}) \begin{pmatrix} | \\ \mathbf{s} \\ | \end{pmatrix} + e = b \in \mathbb{Z}_q$$

- Getting one extra pseudorandom scalar requires an n-dim inner product
- Can amortize each $\mathbf{a}$ over many secrets $\mathbf{s}_i$, but still $\tilde{O}(n)$ work per scalar output.

- Public key crypto schemes have rather large keys:

$$pk = \underbrace{\begin{pmatrix} \vdots \\ \mathbf{A}^t \\ \vdots \end{pmatrix}}_{n}, \quad \left.\begin{pmatrix} \vdots \\ \mathbf{b} \\ \vdots \end{pmatrix}\right\} \Omega(n)$$

- Can fix $\mathbf{A}$ for all users, but at best, still $\tilde{\Omega}(n^2)$ work to encrypt & decrypt an n-bit message

Wishful Thinking...

$$\begin{pmatrix} | \\ \mathbf{a} \\ | \end{pmatrix} \star \begin{pmatrix} | \\ \mathbf{s} \\ | \end{pmatrix} + \begin{pmatrix} | \\ \mathbf{e} \\ | \end{pmatrix} = \begin{pmatrix} | \\ \mathbf{b} \\ | \end{pmatrix} \in \mathbb{Z}_q^n$$

- Get n pseudorandom scalars from just one (cheap) product operation?

Wishful Thinking…

$$\begin{pmatrix} | \\ \mathbf{a} \\ | \end{pmatrix} \star \begin{pmatrix} | \\ \mathbf{s} \\ | \end{pmatrix} + \begin{pmatrix} | \\ \mathbf{e} \\ | \end{pmatrix} = \begin{pmatrix} | \\ \mathbf{b} \\ | \end{pmatrix} \in \mathbb{Z}_q^n$$

- Get n pseudorandom scalars from just one (cheap) product operation?

Question

- How to define the product '$\star$' so that distribution is pseudorandom?

Wishful Thinking...

$$\begin{pmatrix} | \\ \mathbf{a} \\ | \end{pmatrix} \star \begin{pmatrix} | \\ \mathbf{s} \\ | \end{pmatrix} + \begin{pmatrix} | \\ \mathbf{e} \\ | \end{pmatrix} = \begin{pmatrix} | \\ \mathbf{b} \\ | \end{pmatrix} \in \mathbb{Z}_q^n$$

- Get n pseudorandom scalars from just one (cheap) product operation?

Question

- How to define the product '$\star$' so that distribution is pseudorandom?
  - Careful: w/ small error, coordinate-wise multiplication is not secure!

Wishful Thinking...

$$\begin{pmatrix} | \\ \mathbf{a} \\ | \end{pmatrix} \star \begin{pmatrix} | \\ \mathbf{s} \\ | \end{pmatrix} + \begin{pmatrix} | \\ \mathbf{e} \\ | \end{pmatrix} = \begin{pmatrix} | \\ \mathbf{b} \\ | \end{pmatrix} \in \mathbb{Z}_q^n$$

- Get n pseudorandom scalars from just one (cheap) product operation?

Question

- How to define the product '$\star$' so that distribution is pseudorandom?
  - Careful: w/ small error, coordinate-wise multiplication is not secure!

Answer

- '$\star$' = Multiplication in a polynomial ring: e.g., $\mathbb{Z}_q[x]/(x^n + 1)$.

  Very fast and practical with FFT / NTT: $n \log n$ operations mod q.

Wishful Thinking...

$$\begin{pmatrix} | \\ \mathbf{a} \\ | \end{pmatrix} \star \begin{pmatrix} | \\ \mathbf{s} \\ | \end{pmatrix} + \begin{pmatrix} | \\ \mathbf{e} \\ | \end{pmatrix} = \begin{pmatrix} | \\ \mathbf{b} \\ | \end{pmatrix} \in \mathbb{Z}_q^n$$

- Get n pseudorandom scalars from just one (cheap) product operation?

Question

- How to define the product '$\star$' so that distribution is pseudorandom?
  - ★ Careful: w/ small error, coordinate-wise multiplication is not secure!

Answer

- '$\star$' = Multiplication in a polynomial ring: e.g., $\mathbb{Z}_q[x]/(x^n + 1)$.

  Very fast and practical with FFT / NTT: $n \log n$ operations mod q.

- Similar ring structures appear in heuristic NTRU scheme [HPS'98], in compact one-way / CR hash functions [Mic'02,PR'06,LM'06,...], and in fully homomorphic encryption [Gen'09].

Our Results

0. Definition: a suitable 'compact' version of LWE called Ring-LWE

Our Results

0. Definition: a suitable 'compact' version of LWE called Ring-LWE
1. Two main theorems:

worst case on *ideal* lattices $\leq$ search Ring-LWE $\leq$ decision Ring-LWE

(quantum, any ring of ints) (classical, any cyclotomic ring)

Our Results

0. Definition: a suitable ‘compact’ version of LWE called Ring-LWE
1. Two main theorems:

worst case on *ideal* lattices $\leq$ search Ring-LWE $\leq$ decision Ring-LWE

(quantum, any ring of ints) (classical, any cyclotomic ring)

- ⋆ Concurrently & using different techniques, [SSTX’09] proved a qualitatively weaker version of our first (quantum) reduction.

  (Specifically: hardness for bounded # of samples in a specific ring.)

Our Results

0. Definition: a suitable 'compact' version of LWE called Ring-LWE
1. Two main theorems:

worst case on *ideal* lattices $\underset{\text{(quantum, any ring of ints)}}{\leq}$ search Ring-LWE $\underset{\text{(classical, any cyclotomic ring)}}{\leq}$ decision Ring-LWE

- ⋆ Concurrently & using different techniques, [SSTX'09] proved a qualitatively weaker version of our first (quantum) reduction.

  (Specifically: hardness for bounded # of samples in a specific ring.)

- ⋆ Pseudorandomness is new, and important for crypto & efficiency.

  Proof requires very different techniques than for standard LWE.

Our Results

0. Definition: a suitable 'compact' version of LWE called Ring-LWE

1. Two main theorems:

worst case on *ideal* lattices $\leq$ search Ring-LWE $\leq$ decision Ring-LWE

(quantum, any ring of ints) (classical, any cyclotomic ring)

   ⋆ Concurrently & using different techniques, [SSTX'09] proved a qualitatively weaker version of our first (quantum) reduction.

   (Specifically: hardness for bounded # of samples in a specific ring.)

   ⋆ Pseudorandomness is new, and important for crypto & efficiency.

   Proof requires very different techniques than for standard LWE.

2. A 'cookbook' for porting LWE-based schemes to Ring-LWE, plus an entirely new & even more efficient PKE scheme.

LWE Over a Ring

- Example: ring $\boxed{R_q = \mathbb{Z}_q[x]/(x^n + 1)}$ for $n = 2^k$ and $q = 1 \bmod 2n$.

LWE Over a Ring

- Example: ring $\boxed{R_q = \mathbb{Z}_q[x]/(x^n + 1)}$ for $n = 2^k$ and $q = 1 \bmod 2n$.
  - Elements may be viewed as dim $< n$ polynomials with $\mathbb{Z}_q$ coeffs…
  - …or as vectors in $\mathbb{Z}_q^n$.

LWE Over a Ring

- Example: ring $\boxed{R_q = \mathbb{Z}_q[x]/(x^n + 1)}$ for $n = 2^k$ and $q = 1 \bmod 2n$.
  - Elements may be viewed as dim $< n$ polynomials with $\mathbb{Z}_q$ coeffs…
  - …or as vectors in $\mathbb{Z}_q^n$.

polynomial '$+$'	$\longleftrightarrow$	vector addition
polynomial '$\times$'	$\longleftrightarrow$	'anti-cyclic convolution'

LWE Over a Ring

- Example: ring $\boxed{R_q = \mathbb{Z}_q[x]/(x^n + 1)}$ for $n = 2^k$ and $q = 1 \bmod 2n$.
  - ⋆ Elements may be viewed as dim $< n$ polynomials with $\mathbb{Z}_q$ coeffs…
  - ⋆ …or as vectors in $\mathbb{Z}_q^n$.

$$\begin{array}{lcl} \text{polynomial '}+\text{'} & \longleftrightarrow & \text{vector addition} \\ \text{polynomial '}\times\text{'} & \longleftrightarrow & \text{'anti-cyclic convolution'} \end{array}$$

- **Search**: find the secret $\mathbf{s} \in R_q$, given:

$$\begin{array}{lcl} \mathbf{a}_1 \leftarrow R_q & , & \mathbf{b}_1 = \mathbf{a}_1 \times \mathbf{s} + \mathbf{e}_1 \in R_q \\ \mathbf{a}_2 \leftarrow R_q & , & \mathbf{b}_2 = \mathbf{a}_2 \times \mathbf{s} + \mathbf{e}_2 \in R_q \\ & \vdots & \end{array}$$

Error vectors

LWE Over a Ring

- Example: ring $\boxed{R_q = \mathbb{Z}_q[x]/(x^n + 1)}$ for $n = 2^k$ and $q = 1 \bmod 2n$.
  - Elements may be viewed as dim $< n$ polynomials with $\mathbb{Z}_q$ coeffs…
  - …or as vectors in $\mathbb{Z}_q^n$.

$$\begin{array}{lcl} \text{polynomial `}+\text{'} & \longleftrightarrow & \text{vector addition} \\ \text{polynomial `}\times\text{'} & \longleftrightarrow & \text{`anti-cyclic convolution'} \end{array}$$

- **Search**: find the secret $\mathbf{s} \in R_q$, given:

$$\begin{array}{lcl} \mathbf{a}_1 \leftarrow R_q & , & \mathbf{b}_1 = \mathbf{a}_1 \times \mathbf{s} + \mathbf{e}_1 \in R_q \\ \mathbf{a}_2 \leftarrow R_q & , & \mathbf{b}_2 = \mathbf{a}_2 \times \mathbf{s} + \mathbf{e}_2 \in R_q \\ & \vdots & \end{array}$$

Error vectors

- **Decision**: distinguish $(\mathbf{a}_i \,,\, \mathbf{b}_i)$ from uniform $(\mathbf{a}_i \,,\, \mathbf{b}_i)$.

A New Kind of LWE Cryptosystem

- Secret key: 'short' $\mathbf{s} \in R_q$. Public key: $(\mathbf{a}\ ,\ \mathbf{b} = \mathbf{a} \times \mathbf{s} + \mathbf{e})$

A New Kind of LWE Cryptosystem

- Secret key: 'short' $\mathbf{s} \in R_q$. Public key: $(\mathbf{a}\ ,\ \mathbf{b} = \mathbf{a} \times \mathbf{s} + \mathbf{e})$
- Encrypt $m \in \{0,1\}^n$: choose 'short' $\mathbf{t} \in R_q$. Output ciphertext

$$
\begin{aligned}
(\mathbf{c}_1, \mathbf{c}_2) &= (\mathbf{a} \times \mathbf{t} + \mathbf{e}_1\ ,\ \mathbf{b} \times \mathbf{t} + \mathbf{e}_2 \quad + m \cdot [\tfrac{q}{2}]) \\
&\approx (\mathbf{a} \times \mathbf{t}\ ,\ \mathbf{a} \times \mathbf{s} \times \mathbf{t} \quad + m \cdot [\tfrac{q}{2}])
\end{aligned}
$$

A New Kind of LWE Cryptosystem

- Secret key: 'short' $\mathbf{s} \in R_q$. Public key: $(\mathbf{a}\ ,\ \mathbf{b} = \mathbf{a} \times \mathbf{s} + \mathbf{e})$
- Encrypt $m \in \{0,1\}^n$: choose 'short' $\mathbf{t} \in R_q$. Output ciphertext

$$\begin{aligned}(\mathbf{c}_1, \mathbf{c}_2) &= (\mathbf{a} \times \mathbf{t} + \mathbf{e}_1\ ,\ \mathbf{b} \times \mathbf{t} + \mathbf{e}_2 \quad + m \cdot [\tfrac{q}{2}]) \\ &\approx (\mathbf{a} \times \mathbf{t}\ ,\ \mathbf{a} \times \mathbf{s} \times \mathbf{t} \quad + m \cdot [\tfrac{q}{2}])\end{aligned}$$

- Decrypt: recover m from $\mathbf{c}_2 - \mathbf{c}_1 \times \mathbf{s}$.

A New Kind of LWE Cryptosystem

- Secret key: 'short' $\mathbf{s} \in R_q$. Public key: $(\mathbf{a}\ ,\ \mathbf{b} = \mathbf{a} \times \mathbf{s} + \mathbf{e})$
- Encrypt $m \in \{0,1\}^n$: choose 'short' $\mathbf{t} \in R_q$. Output ciphertext

$$\begin{aligned}(\mathbf{c}_1, \mathbf{c}_2) &= (\mathbf{a} \times \mathbf{t} + \mathbf{e}_1\ ,\ \mathbf{b} \times \mathbf{t} + \mathbf{e}_2 \quad + m \cdot [\tfrac{q}{2}]) \\ &\approx (\mathbf{a} \times \mathbf{t}\ ,\ \mathbf{a} \times \mathbf{s} \times \mathbf{t} \quad + m \cdot [\tfrac{q}{2}])\end{aligned}$$

- Decrypt: recover m from $\mathbf{c}_2 - \mathbf{c}_1 \times \mathbf{s}$.

- Works just like subset-sum encryption [LPS'10] and . . . ElGamal ?!?

A New Kind of LWE Cryptosystem

- Secret key: 'short' $\mathbf{s} \in R_q$. Public key: $(\mathbf{a}\ ,\ \mathbf{b} = \mathbf{a} \times \mathbf{s} + \mathbf{e})$
- Encrypt $m \in \{0,1\}^n$: choose 'short' $\mathbf{t} \in R_q$. Output ciphertext

$$
\begin{aligned}
(\mathbf{c}_1, \mathbf{c}_2) &= (\mathbf{a} \times \mathbf{t} + \mathbf{e}_1\ ,\ \mathbf{b} \times \mathbf{t} + \mathbf{e}_2 \quad + m \cdot [\tfrac{q}{2}]) \\
&\approx (\mathbf{a} \times \mathbf{t}\ ,\ \mathbf{a} \times \mathbf{s} \times \mathbf{t} \quad + m \cdot [\tfrac{q}{2}])
\end{aligned}
$$

- Decrypt: recover m from $\mathbf{c}_2 - \mathbf{c}_1 \times \mathbf{s}$.

- Works just like subset-sum encryption [LPS'10] and ... ElGamal ?!?
  But only $\tilde{O}(n)$ key size, Enc, Dec for n-bit message.

A New Kind of LWE Cryptosystem

- Secret key: 'short' $\mathbf{s} \in R_q$. Public key: $(\mathbf{a}\ ,\ \mathbf{b} = \mathbf{a} \times \mathbf{s} + \mathbf{e})$
- Encrypt $m \in \{0,1\}^n$: choose 'short' $\mathbf{t} \in R_q$. Output ciphertext

$$\begin{aligned}(\mathbf{c}_1, \mathbf{c}_2) &= (\mathbf{a} \times \mathbf{t} + \mathbf{e}_1\ ,\ \mathbf{b} \times \mathbf{t} + \mathbf{e}_2 \quad + m \cdot [\tfrac{q}{2}]) \\ &\approx (\mathbf{a} \times \mathbf{t}\ ,\ \mathbf{a} \times \mathbf{s} \times \mathbf{t} \quad + m \cdot [\tfrac{q}{2}])\end{aligned}$$

- Decrypt: recover m from $\mathbf{c}_2 - \mathbf{c}_1 \times \mathbf{s}$.

- Works just like subset-sum encryption [LPS'10] and … ElGamal ?!?
  But only $\tilde{O}(n)$ key size, Enc, Dec for n-bit message.

Proof of CPA Security

1. Public key $(\mathbf{a}, \mathbf{b}) \approx_c (\mathbf{a}, \mathbf{b})$ by decision Ring-LWE
   (even for 'short' $\mathbf{s}$ [ACPS'09])
2. Ciphertext $(\mathbf{c}_1, \mathbf{c}_2) \approx_c (\mathbf{c}_1, \mathbf{c}_2)$, again by decision Ring-LWE

Hardness of Search Ring-LWE

Theorem 1

For any large enough q, solving search Ring-LWE is as hard as quantumly solving poly(n)-approx SVP in any (worst-case) ideal lattice from the ring.

Hardness of Search Ring-LWE

Theorem 1

For any large enough q, solving search Ring-LWE
is as hard as quantumly solving poly(n)-approx SVP
in any (worst-case) ideal lattice from the ring.

- Proof follows the outline of [Regev'05] for LWE & arbitrary lattices.

  Quantum component used as 'black-box;' only classical part needs adaptation to the ring setting.

Hardness of Search Ring-LWE

Theorem 1

For any large enough q, solving search Ring-LWE is as hard as quantumly solving poly(n)-approx SVP in any (worst-case) ideal lattice from the ring.

- Proof follows the outline of [Regev'05] for LWE & arbitrary lattices.

  Quantum component used as 'black-box;' only classical part needs adaptation to the ring setting.

- New reduction technique for 'clearing the ideal' ($\mathcal{I}/q\mathcal{I} \mapsto R/qR$), in an 'algebra-preserving' way.

  Uses Chinese remainder theorem and theory of duality for ideals.

A Word on Ideal Lattices

- Recall example ring $R = \mathbb{Z}[x]/(x^n + 1)$ for $n = 2^k$.
- An ideal $\mathcal{I} \subseteq R$ is closed under $+$ and $-$, and under $\times$ with R.

A Word on Ideal Lattices

- Recall example ring $R = \mathbb{Z}[x]/(x^n + 1)$ for $n = 2^k$.
- An ideal $\mathcal{I} \subseteq R$ is closed under $+$ and $-$, and under $\times$ with R.

To get ideal lattices, embed R and its ideals into $\mathbb{R}^n$. How?

A Word on Ideal Lattices

- Recall example ring $R = \mathbb{Z}[x]/(x^n + 1)$ for $n = 2^k$.
- An ideal $\mathcal{I} \subseteq R$ is closed under $+$ and $-$, and under $\times$ with R.

To get ideal lattices, embed R and its ideals into $\mathbb{R}^n$. How?

- [HPS'98,M'02,PR'06,LM'06,G'09,...]: 'coefficient embedding'

$$a(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1} \quad \mapsto \quad (a_0, \ldots, a_{n-1}) \in \mathbb{Z}^n$$

A Word on Ideal Lattices

- Recall example ring $R = \mathbb{Z}[x]/(x^n + 1)$ for $n = 2^k$.
- An ideal $\mathcal{I} \subseteq R$ is closed under $+$ and $-$, and under $\times$ with R.

To get ideal lattices, embed R and its ideals into $\mathbb{R}^n$. How?

- [HPS'98,M'02,PR'06,LM'06,G'09,...]: 'coefficient embedding'

$$a(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1} \quad \mapsto \quad (a_0, \ldots, a_{n-1}) \in \mathbb{Z}^n$$

$+$ is coordinate-wise, but analyzing $\times$ is cumbersome (esp. for rv's)

A Word on Ideal Lattices

- Recall example ring $R = \mathbb{Z}[x]/(x^n + 1)$ for $n = 2^k$.
- An ideal $\mathcal{I} \subseteq R$ is closed under $+$ and $-$, and under $\times$ with R.

To get ideal lattices, embed R and its ideals into $\mathbb{C}^n$. How?

- [HPS'98,M'02,PR'06,LM'06,G'09,...]: 'coefficient embedding'

$$a(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1} \quad \mapsto \quad (a_0, \ldots, a_{n-1}) \in \mathbb{Z}^n$$

  $+$ is coordinate-wise, but analyzing $\times$ is cumbersome (esp. for rv's)

- [Minkowski'18??,...]: 'canonical embedding.' Let $\omega = \exp(\pi i/n)$:

$$a(x) \quad \mapsto \quad (a(\omega^1)\,, a(\omega^3)\,, \ldots, a(\omega^{2n-1})) \in \mathbb{C}^n$$

A Word on Ideal Lattices

- Recall example ring $R = \mathbb{Z}[x]/(x^n + 1)$ for $n = 2^k$.
- An ideal $\mathcal{I} \subseteq R$ is closed under $+$ and $-$, and under $\times$ with R.

To get ideal lattices, embed R and its ideals into $\mathbb{C}^n$. How?

- [HPS'98,M'02,PR'06,LM'06,G'09,...]: 'coefficient embedding'

$$a(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1} \quad \mapsto \quad (a_0, \ldots, a_{n-1}) \in \mathbb{Z}^n$$

  $+$ is coordinate-wise, but analyzing $\times$ is cumbersome (esp. for rv's)

- [Minkowski'18??,...]: 'canonical embedding.' Let $\omega = \exp(\pi i/n)$:

$$a(x) \quad \mapsto \quad (a(\omega^1), a(\omega^3), \ldots, a(\omega^{2n-1})) \in \mathbb{C}^n$$

  Both $+$ and $\times$ are coordinate-wise! Nice geometric behavior.

A Word on Ideal Lattices

- Recall example ring $R = \mathbb{Z}[x]/(x^n + 1)$ for $n = 2^k$.
- An ideal $\mathcal{I} \subseteq R$ is closed under $+$ and $-$, and under $\times$ with R.

To get ideal lattices, embed R and its ideals into $\mathbb{Z}_q^n$. How?

- [HPS'98,M'02,PR'06,LM'06,G'09,...]: 'coefficient embedding'

$$a(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1} \quad \mapsto \quad (a_0, \ldots, a_{n-1}) \in \mathbb{Z}^n$$

  $+$ is coordinate-wise, but analyzing $\times$ is cumbersome (esp. for rv's)

- [Minkowski'18??,...]: 'canonical embedding.' Let $\omega = \exp(\pi i/n)$:

$$a(x) \quad \mapsto \quad (a(\omega^1), a(\omega^3), \ldots, a(\omega^{2n-1})) \in \mathbb{C}^n$$

  Both $+$ and $\times$ are coordinate-wise! Nice geometric behavior.

- Modulo any prime $q = 1 \bmod 2n$, $(x^n + 1)$ has n roots $\omega^{2i-1} \in \mathbb{Z}_q$.
  For Ring-LWE schemes, this gives an embedding into $\mathbb{Z}_q^n$.

Pseudorandomness of Ring-LWE

Theorem 2

Solving decision Ring-LWE in $R_q = \mathbb{Z}_q[x]/(x^n + 1)$

(for any poly(n)-bounded prime $q = 1 \bmod 2n$)

is as hard as solving search Ring-LWE.

Pseudorandomness of Ring-LWE

Theorem 2

Solving decision Ring-LWE in $R_q = \mathbb{Z}_q[x]/(x^n + 1)$

(for any poly(n)-bounded prime $q = 1 \bmod 2n$)

is as hard as solving search Ring-LWE.

Proof Outline

Given: $\mathcal{O}$ distinguishes samples $(\mathbf{a}, \mathbf{b} \approx \mathbf{a} \times \mathbf{s})$ from uniform $(\mathbf{a}, \mathbf{b})$.

Goal: Find $\mathbf{s} \in R_q$. Equivalent to finding $\mathbf{s}(\omega^{2j-1}) \in \mathbb{Z}_q$ for $j = 1, \ldots, n$.

Pseudorandomness of Ring-LWE

Theorem 2

Solving decision Ring-LWE in $R_q = \mathbb{Z}_q[x]/(x^n + 1)$

(for any poly(n)-bounded prime $q = 1 \bmod 2n$)

is as hard as solving search Ring-LWE.

Proof Outline

Given: $\mathcal{O}$ distinguishes samples $(\mathbf{a}, \mathbf{b} \approx \mathbf{a} \times \mathbf{s})$ from uniform $(\mathbf{a}, \mathbf{b})$.

Goal: Find $\mathbf{s} \in R_q$. Equivalent to finding $\mathbf{s}(\omega^{2j-1}) \in \mathbb{Z}_q$ for $j = 1, \ldots, n$.

1. Hybrid argument: randomize $\mathbf{b}(\omega^1) \in \mathbb{Z}_q$, then $(\mathbf{b}(\omega^1), \mathbf{b}(\omega^3)), \ldots$
   Then $\mathcal{O}$ must distinguish relative to some ω^{2i-1}.

Pseudorandomness of Ring-LWE

Theorem 2

Solving decision Ring-LWE in $R_q = \mathbb{Z}_q[x]/(x^n + 1)$

(for any poly(n)-bounded prime $q = 1 \bmod 2n$)

is as hard as solving search Ring-LWE.

Proof Outline

Given: $\mathcal{O}$ distinguishes samples $(\mathbf{a}, \mathbf{b} \approx \mathbf{a} \times \mathbf{s})$ from uniform $(\mathbf{a}, \mathbf{b})$.

Goal: Find $\mathbf{s} \in R_q$. Equivalent to finding $\mathbf{s}(\omega^{2j-1}) \in \mathbb{Z}_q$ for $j = 1, \ldots, n$.

1. Hybrid argument: randomize $\mathbf{b}(\omega^1) \in \mathbb{Z}_q$, then $(\mathbf{b}(\omega^1), \mathbf{b}(\omega^3)), \ldots$ Then $\mathcal{O}$ must distinguish relative to some ω^{2i-1}.
2. Using $\mathcal{O}$, guess-and-check to find $\mathbf{s}(\omega^{2i-1}) \in \mathbb{Z}_q$ (a la [BFKL'93,R'05]).

Pseudorandomness of Ring-LWE

Theorem 2

Solving decision Ring-LWE in $R_q = \mathbb{Z}_q[x]/(x^n + 1)$

(for any poly(n)-bounded prime $q = 1 \bmod 2n$)

is as hard as solving search Ring-LWE.

Proof Outline

Given: $\mathcal{O}$ distinguishes samples $(\mathbf{a}, \mathbf{b} \approx \mathbf{a} \times \mathbf{s})$ from uniform $(\mathbf{a}, \mathbf{b})$.

Goal: Find $\mathbf{s} \in R_q$. Equivalent to finding $\mathbf{s}(\omega^{2j-1}) \in \mathbb{Z}_q$ for $j = 1, \ldots, n$.

1. Hybrid argument: randomize $\mathbf{b}(\omega^1) \in \mathbb{Z}_q$, then $(\mathbf{b}(\omega^1), \mathbf{b}(\omega^3)), \ldots$
   Then $\mathcal{O}$ must distinguish relative to some ω^{2i-1}.
2. Using $\mathcal{O}$, guess-and-check to find $\mathbf{s}(\omega^{2i-1}) \in \mathbb{Z}_q$ (a la [BFKL'93,R'05]).
3. How to find other $\mathbf{s}(\omega^{2j-1})$? Couldn't $\mathcal{O}$ be useless on other roots?

Pseudorandomness of Ring-LWE

Theorem 2

Solving decision Ring-LWE in $R_q = \mathbb{Z}_q[x]/(x^n + 1)$

(for any poly(n)-bounded prime $q = 1 \bmod 2n$)

is as hard as solving search Ring-LWE.

Proof Outline

Given: $\mathcal{O}$ distinguishes samples $(\mathbf{a}, \mathbf{b} \approx \mathbf{a} \times \mathbf{s})$ from uniform $(\mathbf{a}, \mathbf{b})$.

Goal: Find $\mathbf{s} \in R_q$. Equivalent to finding $\mathbf{s}(\omega^{2j-1}) \in \mathbb{Z}_q$ for $j = 1, \ldots, n$.

1. Hybrid argument: randomize $\mathbf{b}(\omega^1) \in \mathbb{Z}_q$, then $(\mathbf{b}(\omega^1), \mathbf{b}(\omega^3)), \ldots$
   Then $\mathcal{O}$ must distinguish relative to some ω^{2i-1}.
2. Using $\mathcal{O}$, guess-and-check to find $\mathbf{s}(\omega^{2i-1}) \in \mathbb{Z}_q$ (a la [BFKL'93,R'05]).
3. How to find other $\mathbf{s}(\omega^{2j-1})$? Couldn't $\mathcal{O}$ be useless on other roots?
   Map $\omega \mapsto \omega^k$ permutes roots of $x^n + 1$. Can send each to ω^{2i-1}.

Pseudorandomness of Ring-LWE

Theorem 2

Solving decision Ring-LWE in $R_q = \mathbb{Z}_q[x]/(x^n + 1)$

(for any poly(n)-bounded prime $q = 1 \bmod 2n$)

is as hard as solving search Ring-LWE.

Proof Outline

Given: $\mathcal{O}$ distinguishes samples $(\mathbf{a}, \mathbf{b} \approx \mathbf{a} \times \mathbf{s})$ from uniform $(\mathbf{a}, \mathbf{b})$.

Goal: Find $\mathbf{s} \in R_q$. Equivalent to finding $\mathbf{s}(\omega^{2j-1}) \in \mathbb{Z}_q$ for $j = 1, \ldots, n$.

1. Hybrid argument: randomize $\mathbf{b}(\omega^1) \in \mathbb{Z}_q$, then $(\mathbf{b}(\omega^1), \mathbf{b}(\omega^3)), \ldots$ Then $\mathcal{O}$ must distinguish relative to some ω^{2i-1}.
2. Using $\mathcal{O}$, guess-and-check to find $\mathbf{s}(\omega^{2i-1}) \in \mathbb{Z}_q$ (a la [BFKL'93,R'05]).
3. How to find other $\mathbf{s}(\omega^{2j-1})$? Couldn't $\mathcal{O}$ be useless on other roots? Map $\omega \mapsto \omega^k$ permutes roots of $x^n + 1$. Can send each to ω^{2i-1}.

(Math jargon: use the automorphism (Galois) group of the cyclotomic number field.)

Summary and Conclusions

- In any cyclotomic ring, Ring-LWE is pseudorandom if ideal lattice problems are (quantumly) hard in the worst case.

Summary and Conclusions

- In any cyclotomic ring, Ring-LWE is pseudorandom if ideal lattice problems are (quantumly) hard in the worst case.
- Ring-LWE allows for much more compact and efficient encryption schemes than standard LWE.

  E.g., PKE in $\tilde{O}(1)$ work per message bit.

Summary and Conclusions

- In any cyclotomic ring, Ring-LWE is pseudorandom if ideal lattice problems are (quantumly) hard in the worst case.
- Ring-LWE allows for much more compact and efficient encryption schemes than standard LWE.

  E.g., PKE in $\tilde{O}(1)$ work per message bit.
- Main open direction: develop new kinds of constructions unlike those based on standard LWE (e.g., fully homomorphic PKE?)

Summary and Conclusions

- In any cyclotomic ring, Ring-LWE is pseudorandom if ideal lattice problems are (quantumly) hard in the worst case.
- Ring-LWE allows for much more compact and efficient encryption schemes than standard LWE.

  E.g., PKE in $\tilde{O}(1)$ work per message bit.
- Main open direction: develop new kinds of constructions unlike those based on standard LWE (e.g., fully homomorphic PKE?)
- Questions? More details? Find me here: