Fully Homomorphic Encryption

Francisco Vial-Prado

ASCrypto - LatinCrypt '19

IMFD Chile, Ecole Polytechnique, Université Paris-Saclay
Applied Cryptography @ ProtonMail

Overview

- Generic homomorphic encryption, *a priori* observations
- Gentry's blueprint
- Second and third generation schemes

The problem (Rivest, Adleman, Dertouzos, 1978)

On Data Banks And Privacy Homomorphisms - 1978

- *... a system working with encrypted data can at most store or retrieve data for the user; any more complicated operations seem to require that the data be decrypted before being operated on.*
- *... it appears likely that there exist [...] Privacy Homomorphisms.*

Privacy Homomorphisms

Find an encryption scheme S such that:

Let $y = S.\mathtt{Enc}_k(x)$. For any PPT function f mapping plaintexts to plaintexts, find y' publicly such that $S.\mathtt{Dec}_k(y') = f(x)$.

Example: If $S.\mathtt{plainspace}$ is a ring, provide functionalities $\mathtt{Add}, \mathtt{Mult}$ such that

$$\mathtt{Add}(\mathtt{Enc}(x), \mathtt{Enc}(y)) \text{ encrypts } x + y$$

$$\mathtt{Mult}(\mathtt{Enc}(x), \mathtt{Enc}(y)) \text{ encrypts } x \times y.$$

Disclaimer

Along with reasonable security properties!

Privacy Homomorphisms

Find an encryption scheme S such that:

Let $y = S.\mathtt{Enc}_k(x)$. For any PPT function f mapping plaintexts to plaintexts, find y' publicly such that $S.\mathtt{Dec}_k(y') = f(x)$.

Example: If $S.\mathtt{plainspace}$ is a ring, provide functionalities $\mathtt{Add}, \mathtt{Mult}$ such that

$$\mathtt{Add}(\mathtt{Enc}(x), \mathtt{Enc}(y)) \text{ encrypts } x + y$$

$$\mathtt{Mult}(\mathtt{Enc}(x), \mathtt{Enc}(y)) \text{ encrypts } x \times y.$$

Disclaimer

Along with reasonable security properties!

Privacy Homomorphisms

Find an encryption scheme S such that:

Let $y = S.\mathtt{Enc}_k(x)$. For any PPT function f mapping plaintexts to plaintexts, find y' publicly such that $S.\mathtt{Dec}_k(y') = f(x)$.

Example: If $S.\mathtt{plainspace}$ is a ring, provide functionalities $\mathtt{Add}, \mathtt{Mult}$ such that

$$\mathtt{Add}(\mathtt{Enc}(x), \mathtt{Enc}(y)) \text{ encrypts } x + y$$

$$\mathtt{Mult}(\mathtt{Enc}(x), \mathtt{Enc}(y)) \text{ encrypts } x \times y.$$

Disclaimer

Along with reasonable security properties!

A priori observations

HE is non determinist

1. Homomorphic encryption must be non-determinist

The attacker could solve ring equations

$$x = k \Leftrightarrow (x \neq 0) \wedge (x^2 = \underbrace{x + x + \cdots + x}_{k\text{times}})$$

1bis. Broccoli heuristics: If ciphertext spaces are distinguishable, they should be somewhat separable.

HE is non determinist

1. Homomorphic encryption must be non-determinist

The attacker could solve ring equations

$$x = k \Leftrightarrow (x \neq 0) \wedge (x^2 = \underbrace{x + x + \cdots + x}_{k \text{times}})$$

1bis. Broccoli heuristics: If ciphertext spaces are distinguishable, they should be somewhat separable.

HE runs in worst-case complexity for decision algorithms

2. `Logical conditions` translate to homomorphic comparison circuits.

Consider the equality circuit: Let $a, b \in \{0,1\}^\kappa$.

$$\mathrm{Eq}(a,b) = 1 \oplus \prod_{i=1}^{\kappa}(a_i \oplus b_i \oplus 1) = \begin{cases} 0 & \text{if} \quad a = b, \\ 1 & \text{if} \quad a \neq b. \end{cases}$$

Don't allow easy CCA's

3.– Decrypt **Verifiable Computations Only** If Possible (Homomorphic encryption schemes are known to be vulnerable to IND-CCA Key-Recovery attacks)

Connections with other cryptographic problems

- (implied by) Functional encryption
- (provides reduction of) Secure Multiparty Computation
- (compatible with) Identity/Attribute-Based Encryption
- (brick of?) Indistinguishability Obfuscation
- (first multi-hop?) Proxy Re-encryption

Connections with other cryptographic problems

- (implied by) Functional encryption
- (provides reduction of) Secure Multiparty Computation
- (compatible with) Identity/Attribute-Based Encryption
- (brick of?) Indistinguishability Obfuscation
- (first multi-hop?) Proxy Re-encryption

Connections with other cryptographic problems

- (implied by) Functional encryption
- (provides reduction of) Secure Multiparty Computation
- (compatible with) Identity/Attribute-Based Encryption
- (brick of?) Indistinguishability Obfuscation
- (first multi-hop?) Proxy Re-encryption

Connections with other cryptographic problems

- (implied by) Functional encryption
- (provides reduction of) Secure Multiparty Computation
- (compatible with) Identity/Attribute-Based Encryption
- (brick of?) Indistinguishability Obfuscation
- (first multi-hop?) Proxy Re-encryption

Connections with other cryptographic problems

- (implied by) Functional encryption
- (provides reduction of) Secure Multiparty Computation
- (compatible with) Identity/Attribute-Based Encryption
- (brick of?) Indistinguishability Obfuscation
- (first multi-hop?) Proxy Re-encryption

Gentry's solution

The Sophomore's Dream

Let R be some ring and I be an ideal of R. Let $m \in R/I$. Let $\mathtt{Enc}(m) := m + i$ where $i \in I$ is sampled randomly.

$\mathtt{Enc}(m_1) + \mathtt{Enc}(m_2) = m_1 + m_2 + i'$,
$\mathtt{Enc}(m_1) \times \mathtt{Enc}(m_2) = m_1 \times m_2 + i''$.

Good game; now look for

- Random efficient sampling from $\alpha + I$ for every $\alpha \in R/I$
- Secret decryption power: ideal annihilation procedure $\alpha + xI \mapsto \alpha$.
- Connection to hard problems.

Gentry's solution

The Sophomore's Dream

Let R be some ring and I be an ideal of R. Let $m \in R/I$. Let $\mathtt{Enc}(m) := m + i$ where $i \in I$ is sampled randomly.

$$\mathtt{Enc}(m_1) + \mathtt{Enc}(m_2) = m_1 + m_2 + i',$$
$$\mathtt{Enc}(m_1) \times \mathtt{Enc}(m_2) = m_1 \times m_2 + i''.$$

Good game; now look for

- Random efficient sampling from $\alpha + I$ for every $\alpha \in R/I$
- Secret decryption power: ideal annihilation procedure $\alpha + xI \mapsto \alpha$.
- Connection to hard problems.

Gentry's solution

The Sophomore's Dream

Let R be some ring and I be an ideal of R. Let $m \in R/I$. Let $\mathtt{Enc}(m) := m + i$ where $i \in I$ is sampled randomly.

$$\mathtt{Enc}(m_1) + \mathtt{Enc}(m_2) = m_1 + m_2 + i',$$
$$\mathtt{Enc}(m_1) \times \mathtt{Enc}(m_2) = m_1 \times m_2 + i''.$$

Good game; now look for

- Random efficient sampling from $\alpha + I$ for every $\alpha \in R/I$
- Secret decryption power: ideal annihilation procedure $\alpha + xI \mapsto \alpha$.
- Connection to hard problems.

Gentry's solution

The Sophomore's Dream

Let R be some ring and I be an ideal of R. Let $m \in R/I$. Let $\mathtt{Enc}(m) := m + i$ where $i \in I$ is sampled randomly.

$$\mathtt{Enc}(m_1) + \mathtt{Enc}(m_2) = m_1 + m_2 + i',$$
$$\mathtt{Enc}(m_1) \times \mathtt{Enc}(m_2) = m_1 \times m_2 + i''.$$

Good game; now look for

- Random efficient sampling from $\alpha + I$ for every $\alpha \in R/I$
- Secret decryption power: ideal annihilation procedure $\alpha + xI \mapsto \alpha$.
- Connection to hard problems.

Gentry's solution

The Sophomore's Dream

Let R be some ring and I be an ideal of R. Let $m \in R/I$. Let $\texttt{Enc}(m) := m + i$ where $i \in I$ is sampled randomly.

$$\texttt{Enc}(m_1) + \texttt{Enc}(m_2) = m_1 + m_2 + i',$$
$$\texttt{Enc}(m_1) \times \texttt{Enc}(m_2) = m_1 \times m_2 + i''.$$

Good game; now look for

- Random efficient sampling from $\alpha + I$ for every $\alpha \in R/I$
- Secret decryption power: ideal annihilation procedure $\alpha + xI \mapsto \alpha$.
- Connection to hard problems.

Ideals + Lattices = Ideal Lattices

Gentry's first FHE scheme

Specialized the latter construction using polynomial rings and two sets of ideal lattices.

Secret and public keys are parallelepipeds in $\mathbb{R}^n$, with large n, and plaintexts/ciphertexts are polynomials in $Z[X]/(X^n - 1)$.

Ideals + Lattices = Ideal Lattices

Gentry's first FHE scheme

Specialized the latter construction using polynomial rings and two sets of ideal lattices.

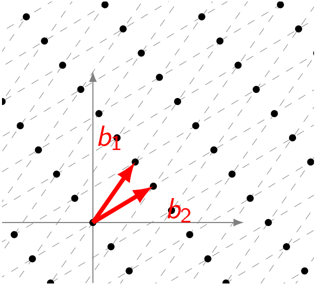
Secret and public keys are parallelepipeds in $\mathbb{R}^n$, with large n, and plaintexts/ciphertexts are polynomials in $Z[X]/(X^n - 1)$.

Disclaimer

What follows is an Unfair and Informal and Incomplete Description of Gentry's scheme

Lattices

More on lattices on yesterdays' talk:
Engineering lattice-based crypto – Peter Schwabe

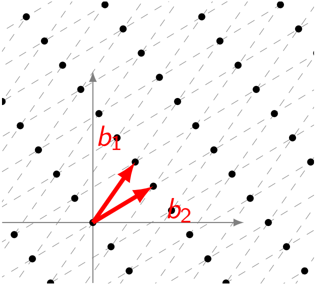


$\mathcal{L} = \mathbb{Z} \cdot \mathbf{b}_1 + \mathbb{Z} \cdot \mathbf{b}_2$

$\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2\}$ is called a basis of $\mathcal{L}$.

Lattices

More on lattices on yesterdays' talk:
Engineering lattice-based crypto – Peter Schwabe

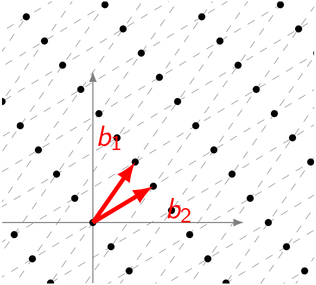


$$\mathcal{L} = \mathbb{Z} \cdot \mathbf{b}_1 + \mathbb{Z} \cdot \mathbf{b}_2$$

$\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2\}$ is called a basis of $\mathcal{L}$.

Lattices

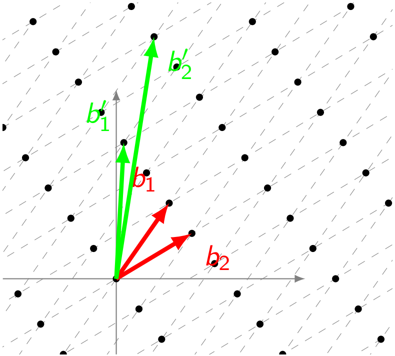
More on lattices on yesterdays' talk:
Engineering lattice-based crypto – Peter Schwabe



$$\mathcal{L} = \mathbb{Z} \cdot \mathbf{b}_1 + \mathbb{Z} \cdot \mathbf{b}_2$$

$\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2\}$ is called a basis of $\mathcal{L}$.

Lattices

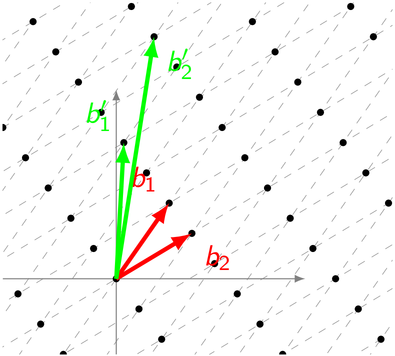


$\mathbf{B} = U \cdot \mathbf{B}'$ for $U \in \mathrm{GL}_n(\mathbb{Z})$.

In particular, for any base,

$$\det(\mathcal{L}) := \sqrt{\det(\mathbf{B} \cdot \mathbf{B}^t)}.$$

Lattices

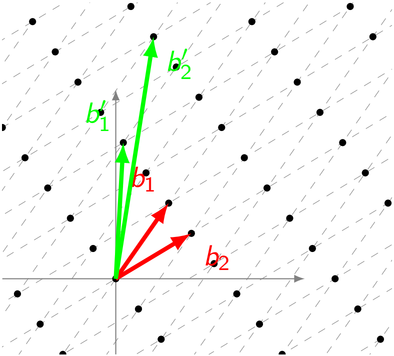


$\mathbf{B} = U \cdot \mathbf{B}'$ for $U \in \mathrm{GL}_n(\mathbb{Z})$.

In particular, for any base,

$$\det(\mathcal{L}) := \sqrt{\det(\mathbf{B} \cdot \mathbf{B}^t)}.$$

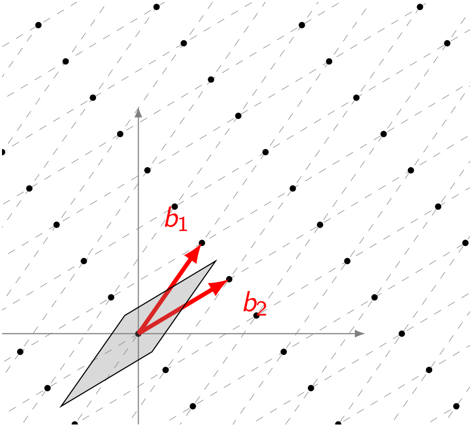
Lattices



$\mathbf{B} = U \cdot \mathbf{B}'$ for $U \in \mathrm{GL}_n(\mathbb{Z})$.

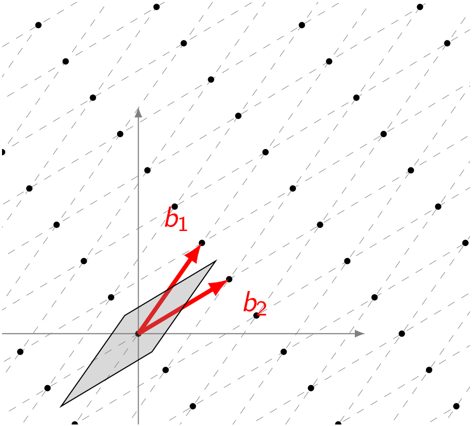
In particular, for any base,

$$\det(\mathcal{L}) := \sqrt{\det(\mathbf{B} \cdot \mathbf{B}^t)}.$$



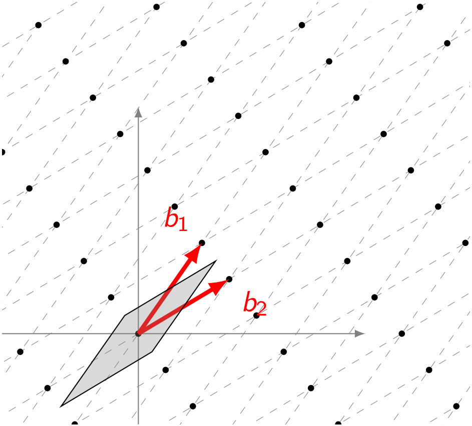
$$\mathcal{P}(\mathbf{B}) := \left[-\frac{1}{2}, \frac{1}{2}\right) \cdot \mathbf{b}_1 + \left[-\frac{1}{2}, \frac{1}{2}\right) \cdot \mathbf{b}_2$$

$$\mathrm{Vol}(\mathcal{P}) = \det(\mathcal{L})$$



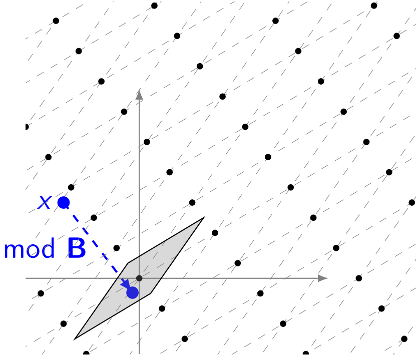
$$\mathcal{P}(\mathbf{B}) := \left[-\frac{1}{2}, \frac{1}{2}\right) \cdot \mathbf{b}_1 + \left[-\frac{1}{2}, \frac{1}{2}\right) \cdot \mathbf{b}_2$$

$$\mathrm{Vol}(\mathcal{P}) = \det(\mathcal{L})$$



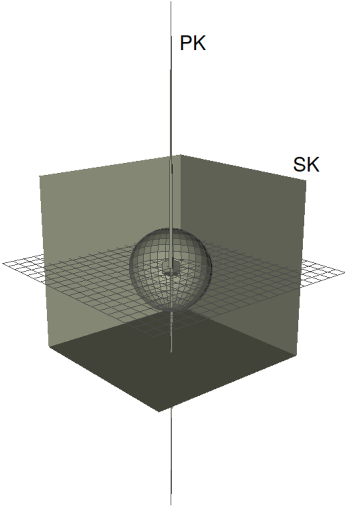
$$\mathcal{P}(\mathbf{B}) := \left[-\frac{1}{2}, \frac{1}{2}\right) \cdot \mathbf{b}_1 + \left[-\frac{1}{2}, \frac{1}{2}\right) \cdot \mathbf{b}_2$$

$$\mathrm{Vol}(\mathcal{P}) = \det(\mathcal{L})$$



$$\forall x \in \mathbb{R}^n \quad x \bmod \mathbf{B} := x - \mathbf{B}\lfloor \mathbf{B}^{-1} \cdot x \rceil$$

Gentry's scheme



A message $m = (1, 0, 0, 0, 1, 1)$ is encrypted by

$$c = m \bmod \mathbf{B}_{\mathrm{pk}}.$$

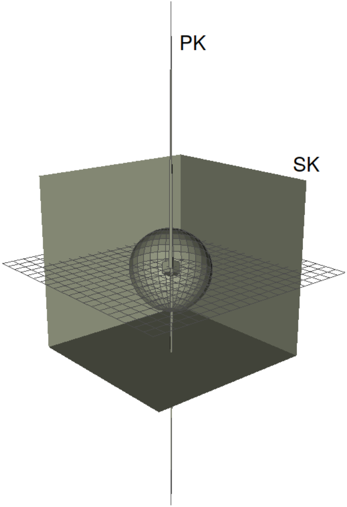
Then,

$$c = (1, 3, 0, -2, 0, -521159786514568)$$

is decrypted by

$$m = c \bmod \mathbf{B}_{\mathrm{sk}}.$$

Gentry's scheme



A message $m = (1, 0, 0, 0, 1, 1)$ is encrypted by

$$c = m \bmod \mathbf{B}_{\mathtt{pk}}.$$

Then,

$$c = (1, 3, 0, -2, 0, -521159786514568)$$

is decrypted by

$$m = c \bmod \mathbf{B}_{\mathtt{sk}}.$$

Gentry's scheme

Concretely:

Let $p \in \mathbb{Z}[X]/(X^n - 1)$. Then

$$\mathbf{B}_{\mathtt{sk}} = \{p(x), xp(x), x^2 p(x), \ldots, x^{n-1} p(x)\}$$

In order to decrypt a ciphertext $c = (c_0, \ldots, c_{n-1})$,

$$\begin{aligned} c \bmod \mathbf{B}_{\mathtt{sk}} &= c - \mathbf{B}_{\mathtt{sk}} \cdot \lfloor \mathbf{B}_{\mathtt{sk}}^{-1} \cdot c \rceil && (\text{in } \mathbb{Z}^n) \\ &= c(x) - p(x) \cdot \lfloor p(x)^{-1} \cdot c(x) \rceil && (\text{in } \tfrac{\mathbb{Z}[X]}{X^n - 1}). \end{aligned}$$

Gentry's scheme

Concretely:

Let $p \in \mathbb{Z}[X]/(X^n - 1)$. Then

$$\mathbf{B}_{\mathtt{sk}} = \{p(x), xp(x), x^2 p(x), \ldots, x^{n-1} p(x)\}$$

In order to decrypt a ciphertext $c = (c_0, \ldots, c_{n-1})$,

$$\begin{aligned} c \bmod \mathbf{B}_{\mathtt{sk}} &= c - \mathbf{B}_{\mathtt{sk}} \cdot \lfloor \mathbf{B}_{\mathtt{sk}}^{-1} \cdot c \rceil & \text{(in } \mathbb{Z}^n\text{)} \\ &= c(x) - p(x) \cdot \lfloor p(x)^{-1} \cdot c(x) \rceil & \text{(in } \tfrac{\mathbb{Z}[X]}{X^n - 1}\text{)}. \end{aligned}$$

Gentry's scheme

Concretely:

Let $p \in \mathbb{Z}[X]/(X^n - 1)$. Then

$$\mathbf{B}_{\mathtt{sk}} = \{p(x), xp(x), x^2 p(x), \dots, x^{n-1} p(x)\}$$

In order to decrypt a ciphertext $c = (c_0, \dots, c_{n-1})$,

$$\begin{aligned} c \bmod \mathbf{B}_{\mathtt{sk}} &= c - \mathbf{B}_{\mathtt{sk}} \cdot \lfloor \mathbf{B}_{\mathtt{sk}}^{-1} \cdot c \rceil && (\text{in } \mathbb{Z}^n) \\ &= c(x) - p(x) \cdot \lfloor p(x)^{-1} \cdot c(x) \rceil && (\text{in } \tfrac{\mathbb{Z}[X]}{X^n - 1}). \end{aligned}$$

Gentry's scheme

Homomorphic operations? Ring structure transport from $R = \mathbb{Z}[X]/(P(X))$, to $\mathbb{Z}^n$ via the coefficients homomorphism.

The noise problem and Gentrys' Glovebox

Encryption $m + xl$ is subject to the 'size' of x. After a threshold, decryption breaks.

Bootstrapping operation: *Homomorphically decrypt*

The noise problem and Gentrys' Glovebox

Encryption $m + xI$ is subject to the 'size' of x. After a threshold, decryption breaks.

Bootstrapping operation: *Homomorphically decrypt*

Second and third gen schemes

Second and third generation schemes

Same blueprint

- Provide `Add`, `Mult` operations, bootstrap to reduce noise, repeat

Improved efficiency and security

- RLWE, NTRU-based, Approximate Eigenvectors
- Better noise growth, key sizes, ciphertext compression, ciphertext packing, SIMD style
- Efficient bootstrapping

New flavors, properties, and already practical for applications.

Second and third generation schemes

Same blueprint

- Provide Add, Mult operations, bootstrap to reduce noise, repeat

Improved efficiency and security

- RLWE, NTRU-based, Approximate Eigenvectors
- Better noise growth, key sizes, ciphertext compression, ciphertext packing, SIMD style
- Efficient bootstrapping

New flavors, properties, and already practical for applications.

Second and third generation schemes

Same blueprint

- Provide `Add`, `Mult` operations, bootstrap to reduce noise, repeat

Improved efficiency and security

- RLWE, NTRU-based, Approximate Eigenvectors
- Better noise growth, key sizes, ciphertext compression, ciphertext packing, SIMD style
- Efficient bootstrapping

New flavors, properties, and already practical for applications.

Second and third generation schemes

Same blueprint

- Provide `Add`, `Mult` operations, bootstrap to reduce noise, repeat

Improved efficiency and security

- RLWE, NTRU-based, Approximate Eigenvectors
- Better noise growth, key sizes, ciphertext compression, ciphertext packing, SIMD style
- Efficient bootstrapping

New flavors, properties, and already practical for applications.

Second and third generation schemes

Same blueprint

- Provide `Add`, `Mult` operations, bootstrap to reduce noise, repeat

Improved efficiency and security

- RLWE, NTRU-based, Approximate Eigenvectors
- Better noise growth, key sizes, ciphertext compression, ciphertext packing, SIMD style
- Efficient bootstrapping

New flavors, properties, and already practical for applications.

Second and third generation schemes

Same blueprint

- Provide `Add`, `Mult` operations, bootstrap to reduce noise, repeat

Improved efficiency and security

- RLWE, NTRU-based, Approximate Eigenvectors
- Better noise growth, key sizes, ciphertext compression, ciphertext packing, SIMD style
- Efficient bootstrapping

New flavors, properties, and already practical for applications.

Learning With Errors

Regev's folklore example: Recover an integer vector $s = (s_1, s_2, s_3, s_4) \in \mathbb{Z}_{17}^4$ satisfying

$$\begin{cases} 14s_1 + 15s_2 + 5s_3 + 2s_4 & \approx & 8 \bmod 17, \\ 13s_1 + 14s_2 + 14s_3 + 6s_4 & \approx & 16 \bmod 17, \\ 6s_1 + 10s_2 + 13s_3 + 1s_4 & \approx & 3 \bmod 17, \\ 10s_1 + 4s_2 + 12s_3 + 16s_4 & \approx & 12 \bmod 17, \\ 9s_1 + 5s_2 + 9s_3 + 6s_4 & \approx & 9 \bmod 17, \\ 3s_1 + 6s_2 + 4s_3 + 5s_4 & \approx & 16 \bmod 17, \end{cases}$$

where "$\approx$" means that the equation is correct up to an error of ± 1.

BGV (2011) FHE scheme

Ring Learning With Errors

Let χ be an error distribution over $R = \mathbb{F}_q[X]/(P_n(X))$.Let $s_i(x) \leftarrow \chi$ and for $i = 0, 1, 2, \ldots,$ $a_i(x) \xleftarrow{\$} R$, $s_i \leftarrow \chi$. Finally, let $b_i := a_i \cdot s + e_i$.

Search-RLWE

Guess s given a list of pairs $(a_i, b_i) = (a_i, a_i \cdot s + e_i)$.

Decision-RLWE

Given a list of pairs $(a_i(x), b_i(x))$, decide whether the b_i's were sampled randomly, or constructed as above.

BFV (2012) FHE scheme - with new techniques

$\rightarrow$ *See LatinCrypt'19 - Compact and simple RLWE based key encapsulation mechanism*

Ring Learning With Errors

Let χ be an error distribution over $R = \mathbb{F}_q[X]/(P_n(X))$.Let $s_i(x) \leftarrow \chi$ and for $i = 0, 1, 2, \ldots,$ $a_i(x) \xleftarrow{\$} R$, $s_i \leftarrow \chi$. Finally, let $b_i := a_i \cdot s + e_i$.

Search-RLWE

Guess s given a list of pairs $(a_i, b_i) = (a_i, a_i \cdot s + e_i)$.

Decision-RLWE

Given a list of pairs $(a_i(x), b_i(x))$, decide whether the b_i's were sampled randomly, or constructed as above.

BFV (2012) FHE scheme - with new techniques

$\rightarrow$ *See LatinCrypt'19 - Compact and simple RLWE based key encapsulation mechanism*

Ring Learning With Errors

Let χ be an error distribution over $R = \mathbb{F}_q[X]/(P_n(X))$.Let $s_i(x) \leftarrow \chi$ and for $i = 0, 1, 2, \dots,$ $a_i(x) \xleftarrow{\$} R$, $s_i \leftarrow \chi$. Finally, let $b_i := a_i \cdot s + e_i$.

Search-RLWE

Guess s given a list of pairs $(a_i, b_i) = (a_i, a_i \cdot s + e_i)$.

Decision-RLWE

Given a list of pairs $(a_i(x), b_i(x))$, decide whether the b_i's were sampled randomly, or constructed as above.

BFV (2012) FHE scheme - with new techniques

$\rightarrow$ *See LatinCrypt'19 - Compact and simple RLWE based key encapsulation mechanism*

Ring Learning With Errors

Let χ be an error distribution over $R = \mathbb{F}_q[X]/(P_n(X))$.Let $s_i(x) \leftarrow \chi$ and for $i = 0, 1, 2, \ldots,$ $a_i(x) \overset{\$}{\leftarrow} R$, $s_i \leftarrow \chi$. Finally, let $b_i := a_i \cdot s + e_i$.

Search-RLWE

Guess s given a list of pairs $(a_i, b_i) = (a_i, a_i \cdot s + e_i)$.

Decision-RLWE

Given a list of pairs $(a_i(x), b_i(x))$, decide whether the b_i's were sampled randomly, or constructed as above.

BFV (2012) FHE scheme - with new techniques

$\rightarrow$ *See LatinCrypt'19 - Compact and simple RLWE based key encapsulation mechanism*

Ring Learning With Errors

Let χ be an error distribution over $R = \mathbb{F}_q[X]/(P_n(X))$.Let $s_i(x) \leftarrow \chi$ and for $i = 0, 1, 2, \ldots,$ $a_i(x) \xleftarrow{\$} R$, $s_i \leftarrow \chi$. Finally, let $b_i := a_i \cdot s + e_i$.

Search-RLWE

Guess s given a list of pairs $(a_i, b_i) = (a_i, a_i \cdot s + e_i)$.

Decision-RLWE

Given a list of pairs $(a_i(x), b_i(x))$, decide whether the b_i's were sampled randomly, or constructed as above.

BFV (2012) FHE scheme - with new techniques

$\rightarrow$ *See LatinCrypt'19 - Compact and simple RLWE based key encapsulation mechanism*

NTRU-based

N-th truncated: Security problems related to Gaussian distributions and inversions in polynomial rings. Exposed strong connections with MPC (LTV12 scheme)

Subfield lattice attacks on overstretched NTRU assumptions - ABD 2016.

→ *Same ideas behind the new Mersenne cryptosystem (AJPS17), see LatinCrypt'19, Quantum LLL with an Application to Mersenne Number Cryptosystems*

NTRU-based

N-th truncated: Security problems related to Gaussian distributions and inversions in polynomial rings. Exposed strong connections with MPC (LTV12 scheme)

Subfield lattice attacks on overstretched NTRU assumptions - ABD 2016.

→ Same ideas behind the new Mersenne cryptosystem (AJPS17), see LatinCrypt'19, Quantum LLL with an Application to Mersenne Number Cryptosystems

NTRU-based

N-th truncated: Security problems related to Gaussian distributions and inversions in polynomial rings. Exposed strong connections with MPC (LTV12 scheme)

Subfield lattice attacks on overstretched NTRU assumptions - ABD 2016.

$\rightarrow$ *Same ideas behind the new Mersenne cryptosystem (AJPS17), see LatinCrypt'19, Quantum LLL with an Application to Mersenne Number Cryptosystems*

Third Generation

GSW and Approximate Eigenvectors

$$C \cdot \mathbf{v} = m.\mathbf{v} + \mathbf{e} \bmod q$$

- Asymmetric nose growth
- Bootstrapping after each gate - the homomorphic brick
- Ring variant and inspired optimizations: TorusFHE (`https://tfhe.github.io/tfhe/`)

Third Generation

GSW and Approximate Eigenvectors

$$C \cdot \mathbf{v} = m.\mathbf{v} + \mathbf{e} \bmod q$$

- Asymmetric nose growth
- Bootstrapping after each gate - the homomorphic brick
- Ring variant and inspired optimizations: TorusFHE (https://tfhe.github.io/tfhe/)

Third Generation

GSW and Approximate Eigenvectors

$$C \cdot \mathbf{v} = m.\mathbf{v} + \mathbf{e} \bmod q$$

- Asymmetric nose growth
- Bootstrapping after each gate - the homomorphic brick
- Ring variant and inspired optimizations: TorusFHE (`https://tfhe.github.io/tfhe/`)

Conclusion

Thank you!

Conclusion

Thank you!