CHIMERA: Combining Ring-LWE-based Fully Homomorphic Encryption Schemes

Mariya Georgieva [1,2]

[1] inpher.io

[2] EPFL ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

UNIVERSITÉ DE VERSAILLES ST-QUENTIN-EN-YVELINES
université PARIS-SACLAY

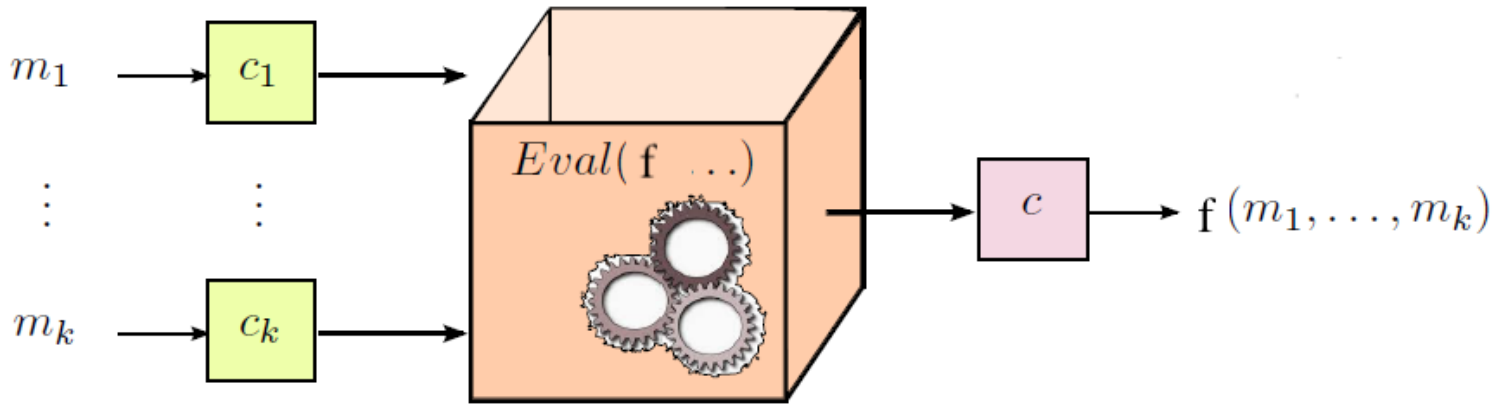
DPPH

Joint work with: C. Boura, N. Gama, D. Jetchev

Homomorphic encryption

- Given $(c_1, c_2, \ldots, c_k) = (E(m_1), E(m_2), \ldots, E(m_k))$

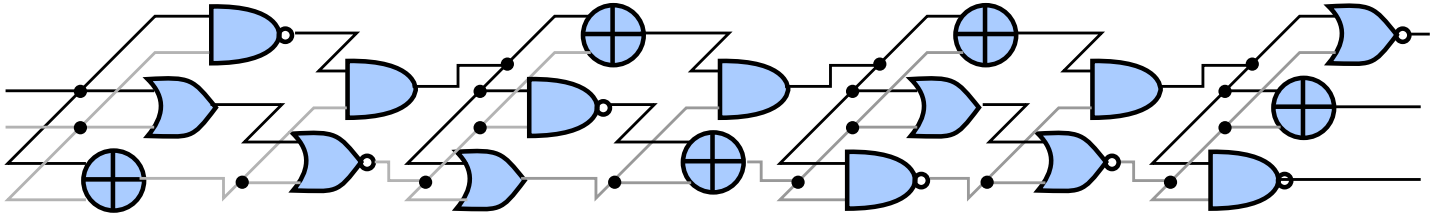
The homomorphic computation consists to compute $E(f(m_1, m_2, \ldots, m_k))$ without decryption.



A scheme that can homomorphically evaluate all function is said **Fully Homomorphic**

Model of computations

1 Binary, circuit computations



2 Integer arithmetic

decimal

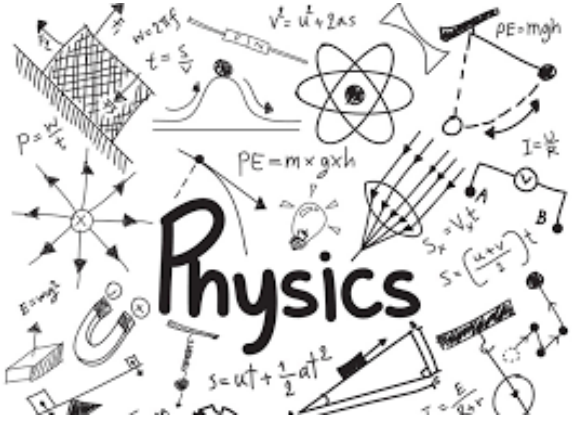
0 0 1 1 ← *carries*

4 5 6 7

3 6 6

4 9 3 3

3 Approximated (Fixed-point) computations



Plan

Integer/Real/Complex polynomials

inpher.io

- $R_{\mathsf{Z}} = \mathsf{Z}[X]/(X^N + 1)$: the ring of polynomials with integer coefficients module $X^N + 1$
- $R_{\mathsf{R}} = \mathsf{R}[X]/(X^N + 1)$: the ring of polynomials with real coefficients modulo $X^N + 1$
- $R_{\mathsf{C}} = \mathsf{C}[X]/(X^N + 1)$: the ring of polynomials with complex coefficients modulo $X^N + 1$

Examples (Real): $N = 2$

$$(1.2 + 2.3X) \cdot (3.2 + 4.1X) = 3.84 + 12.28X + 9.43X^2 = 12.28X - 5.59 \mod (X^2 + 1)$$

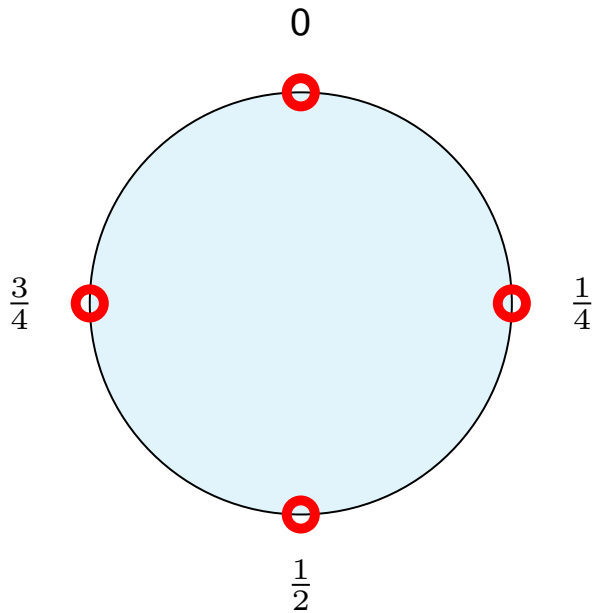
$(R_{\mathsf{Z}}, +, \times)$, $(R_{\mathsf{R}}, +, \times)$ and $(R_{\mathsf{C}}, +, \times)$ are well defined as Ring

- $(R_{\mathsf{Z}}, +)$, $(R_{\mathsf{R}}, +)$ and $(R_{\mathsf{C}}, +)$ are groups
- It is a Ring: $x \times y$ is defined!

Torus T and Torus polynomials T_R

$(T, +, \cdot)$ = R mod 1 is a Z-module ($\cdot : Z \times T \to T$ a valid external product)

- It is a group $x + y$ mod 1, and $-x$ mod 1
- It is a Z-module: $0 \cdot \frac{1}{2} = 0$ is defined!
- It is **not** a Ring: $0 \times \frac{1}{2}$ is **not** defined!



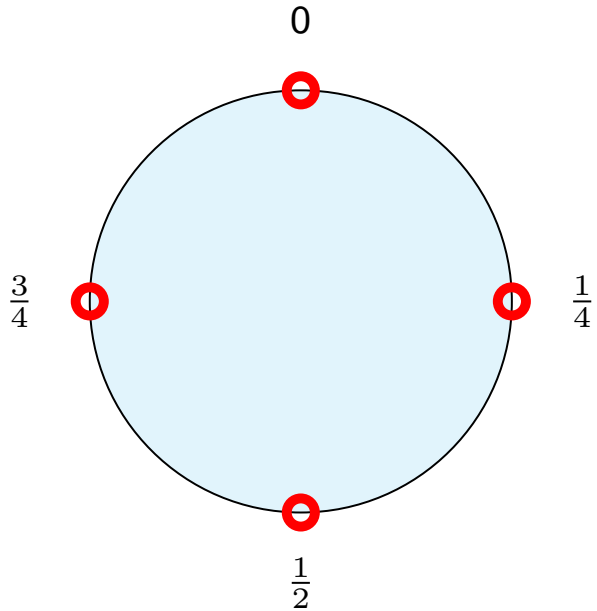
$(T_R, +, \cdot)$ is a R_Z-module

- Here, R_Z = Z[X] mod (X^N + 1)
- And T_R = R[X] mod (X^N + 1) mod 1

Torus T and Torus polynomials T_R

$(T, +, \cdot) = R \mod 1$ is a Z-module ($\cdot : Z \times T \to T$ a valid external product)

- It is a group $x + y \mod 1$, and $-x \mod 1$
- It is a Z-module: $0 \cdot \frac{1}{2} = 0$ is defined!
- It is **not** a Ring: $0 \times \frac{1}{2}$ is **not** defined!



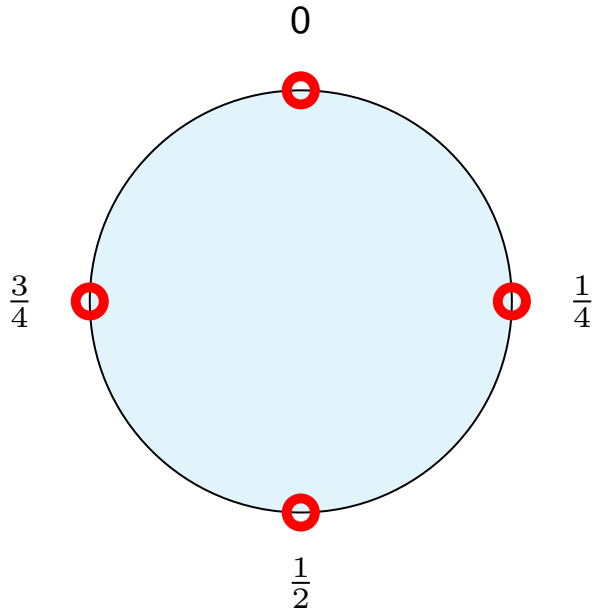
$(T_R, +, \cdot)$ is a R_Z-module

- Here, $R_Z = Z[X] \mod (X^N + 1)$
- And $T_R = R[X] \mod (X^N + 1) \mod 1$

Torus T and Torus polynomials T_R

$(T, +, \cdot)$ = R mod 1 is a Z-module ($\cdot : Z \times T \to T$ a valid external product)

- It is a group $x + y$ mod 1, and $-x$ mod 1
- It is a Z-module: $0 \cdot \frac{1}{2} = 0$ is defined!
- It is **not** a Ring: $0 \times \frac{1}{2}$ is **not** defined!



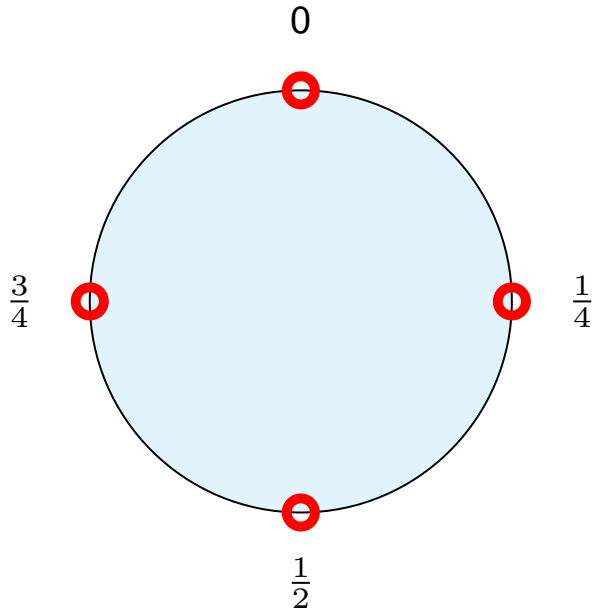
$(T_R, +, \cdot)$ is a R_Z-module

- Here, R_Z = Z[X] mod ($X^N + 1$)
- And T_R = R[X] mod ($X^N + 1$) mod 1

Torus T and Torus polynomials T_R

$(T, +, \cdot)$ = R mod 1 is a Z-module ($\cdot : Z \times T \to T$ a valid external product)

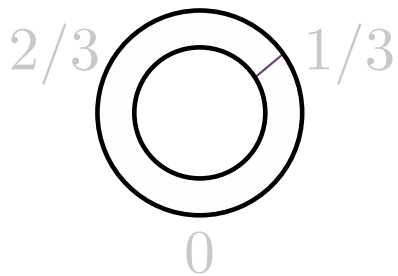
- It is a group $x + y$ mod 1, and $-x$ mod 1
- It is a Z-module: $0 \cdot \frac{1}{2} = 0$ is defined!
- It is **not** a Ring: $0 \times \frac{1}{2}$ is **not** defined!



$(T_R, +, \cdot)$ is a R_Z-module

- Here, R_Z = Z[X] mod ($X^N + 1$)
- And T_R = R[X] mod ($X^N + 1$) mod 1

LWE Encryption over the torus (T = R/Z = R mod 1)

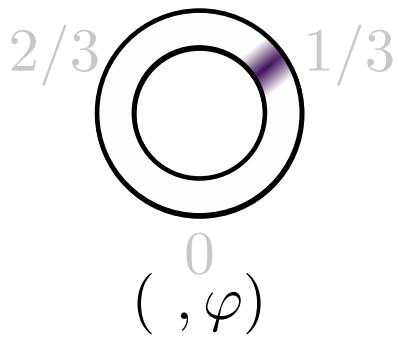


Example: $\mathcal{M} = \{0, 1/3, 2/3\}\ mod\ 1$

$\mu = 1/3\ mod\ 1 \in \mathcal{M}$

LWE Encryption over the torus (T = R/Z = R mod 1)

	message	ciphertext	key	lin. combin.	product
TLWE	T				



Example: $\mathcal{M} = \{0, 1/3, 2/3\}\ mod\ 1$

$\mu = 1/3\ mod\ 1 \in \mathcal{M}$

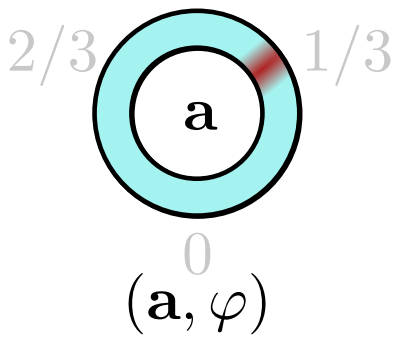
1. φ = μ + *Gaussian Error*
2. Random tag **a** T^n

LWE Encryption over the torus ($\mathbb{T} = \mathbb{R}/\mathbb{Z} = \mathbb{R} \mod 1$)

inpher.io

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$			

$$\textbf{secret key: } \mathbf{s} \in \{0,1\}^n$$



Example: $\mathcal{M} = \{0, 1/3, 2/3\}\ mod\ 1$

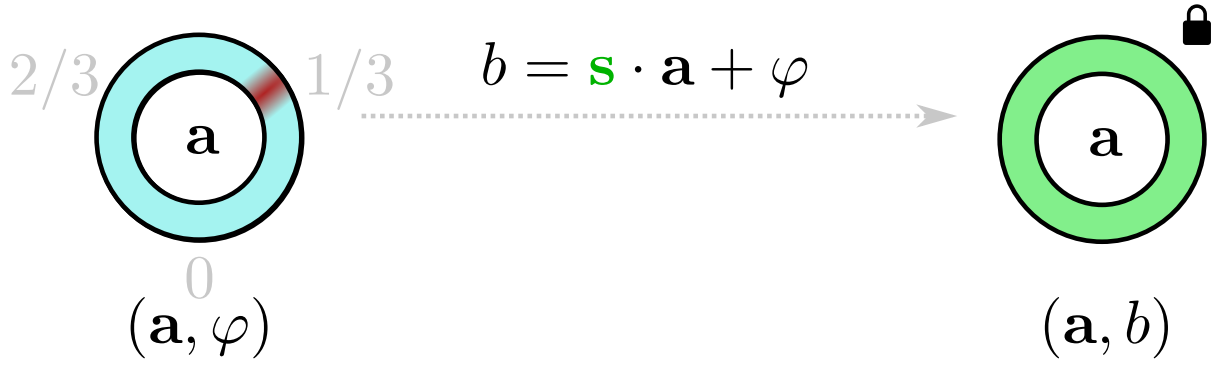
$\mu = 1/3\ mod\ 1 \in \mathcal{M}$

1. φ = μ + *Gaussian Error*
2. Random tag $\mathbf{a}$ $\quad \mathbb{T}^n$

LWE Encryption over the torus (T = R/Z = R mod 1)

	message	ciphertext	key	lin. combin.	product
TLWE	T	T^{n+1}			

$$\textbf{secret key: } \mathbf{s} \in \{0,1\}^n$$



Example: $\mathcal{M} = \{0, 1/3, 2/3\}\ mod\ 1$

$\mu = 1/3\ mod\ 1 \in \mathcal{M}$

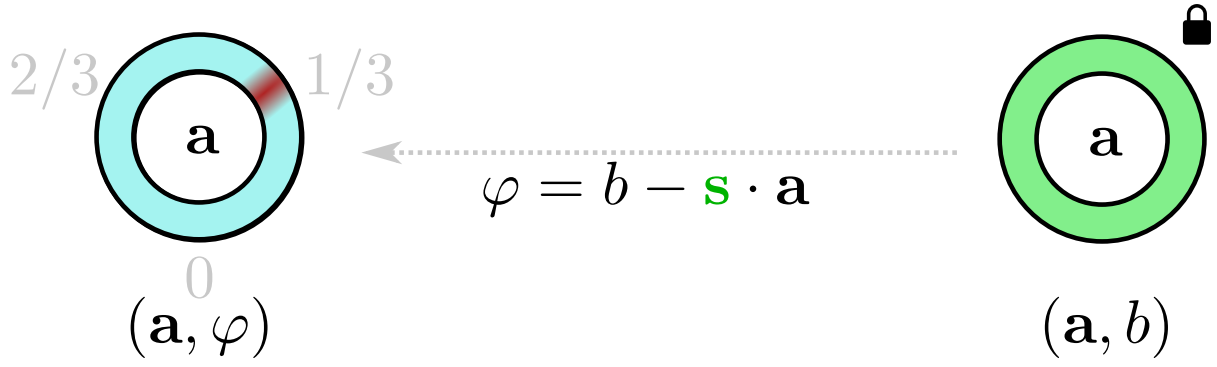
1. φ = μ + *Gaussian Error*
2. Random tag $\mathbf{a}$ T^n

LWE Encryption over the torus ($\mathbb{T} = \mathbb{R}/\mathbb{Z} = \mathbb{R} \mod 1$)

inpher.io

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$		

$$\textbf{secret key: } \mathbf{s} \in \{0,1\}^n$$



Example: $\mathcal{M} = \{0, 1/3, 2/3\}\ mod\ 1$

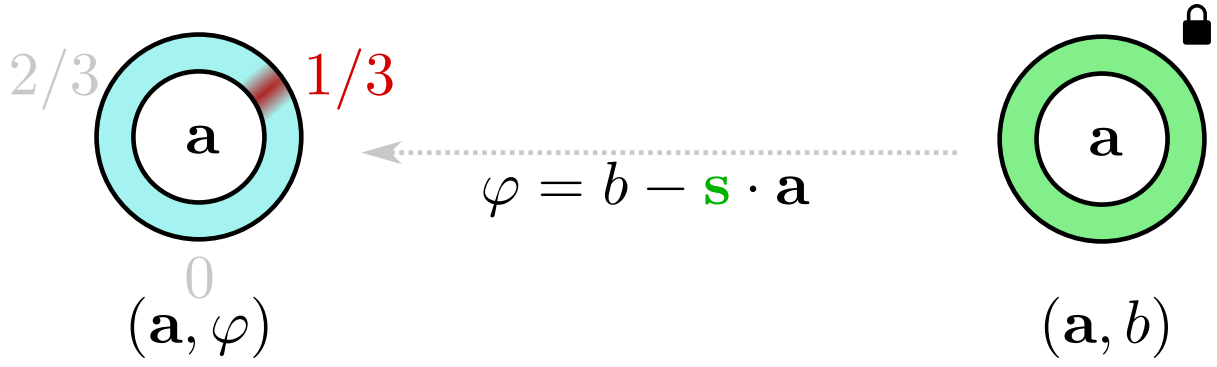
$\mu = 1/3\ mod\ 1 \in \mathcal{M}$

1. Unlock the representation $(\mathbf{a}, \varphi)$
2. Round φ to the nearest message μ M

LWE Encryption over the torus ($\mathbb{T} = \mathbb{R}/\mathbb{Z} = \mathbb{R} \mod 1$)

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$		

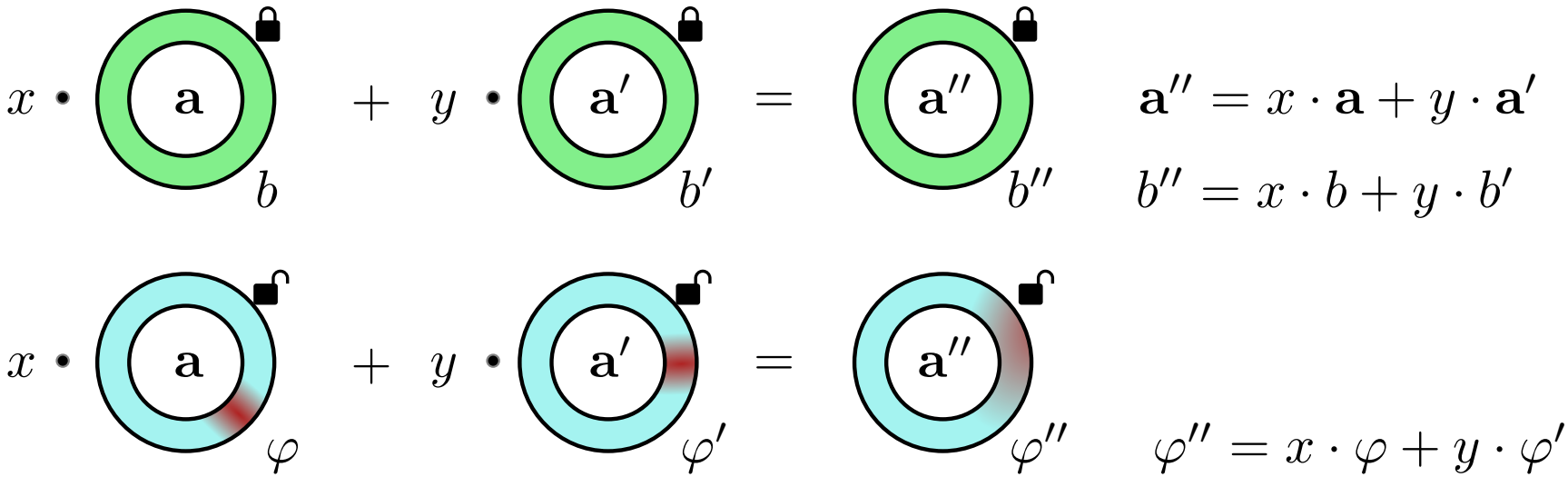
$$\textbf{secret key: } \mathbf{s} \in \{0,1\}^n$$



1. Unlock the representation $(\mathbf{a}, \varphi)$
2. Round φ to the nearest message μ $\quad M$

LWE Encryption over the torus

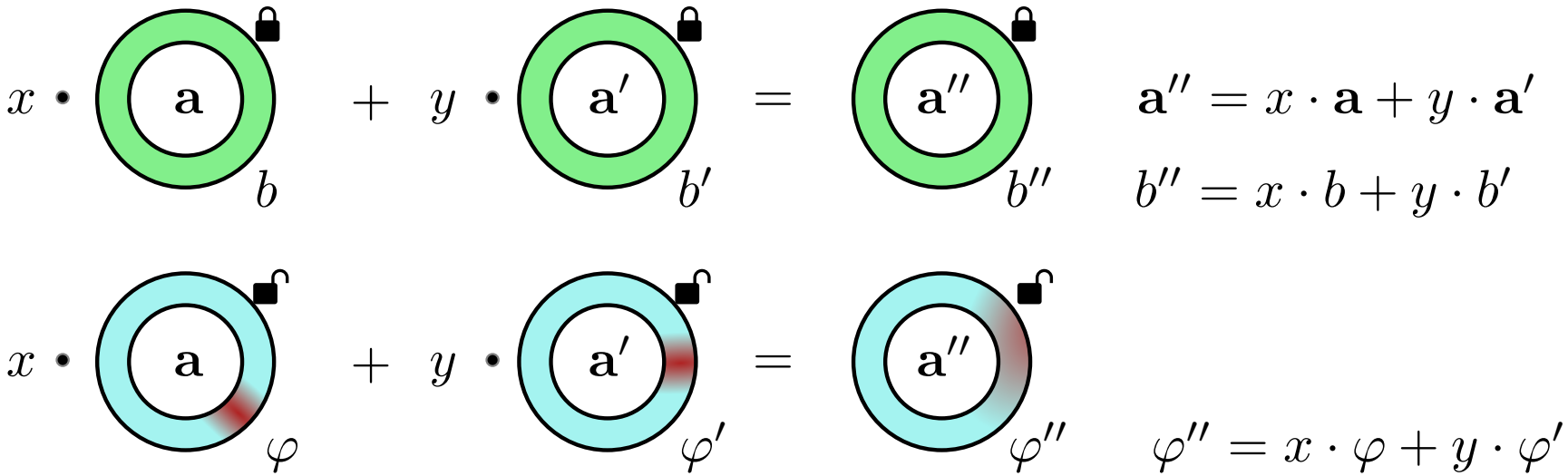
	message	ciphertext	key	lin. combin.	product
TLWE	T	T^{n+1}	B^n		
TRLWE	T_R	T_R^{k+1}	B^k		



$\alpha = \mathrm{stdev}(\varphi)$ $\quad$ α' $\quad$ α'' $\quad$ $\alpha''^2 = x^2\alpha^2 + y^2\alpha'^2$

LWE Encryption over the torus

	message	ciphertext	key	lin. combin.	product
TLWE	T	T^{n+1}	B^n	4	8
TRLWE	T_R	T_R^{k+1}	B^k	4	8



$\alpha = \mathrm{stdev}(\varphi)$ $\quad$ α' $\quad$ α'' $\quad$ $\alpha''^2 = x^2\alpha^2 + y^2\alpha'^2$

	message	ciphertext	key	lin. combin.	product
TLWE	T	T^{n+1}	B^n	4	8
TRLWE	T_R	T_R^{k+1}	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k		

TR(GSW) ciphertexts of μ R_Z

$$\mathsf{TRGSW}(\mu) = \left(\begin{array}{c} \mathsf{TRLWE}_K(K \cdot \frac{\mu}{2}) \\ \mathsf{TRLWE}_K(K \cdot \frac{\mu}{4}) \\ \mathsf{TRLWE}_K(K \cdot \frac{\mu}{8}) \\ \hline \mathsf{TRLWE}_K(1 \cdot \frac{\mu}{2}) \\ \mathsf{TRLWE}_K(1 \cdot \frac{\mu}{4}) \\ \mathsf{TRLWE}_K(1 \cdot \frac{\mu}{8}) \end{array} \right)$$

1. **Internal Product (classical)**: : TRGSW × TRGSW $\longrightarrow$ TRGSW (Ring Structure)
2. **External product (Asiacrypt 2016)**: : TRGSW × TRLWE $\longrightarrow$ TRLWE (Module Structure)
   $(\mu_A, \mu_{\mathbf{b}}) \longrightarrow \mu_A \cdot \mu_{\mathbf{b}}$
   $(\epsilon_A, \epsilon_{\mathbf{b}}) \longrightarrow ||\mu_A||_1 \quad \epsilon_{\mathbf{b}} + O(\epsilon_A)$

If $||\mu_A||_1 = 1$ the noise propagation is linear!

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$	4	8
TRLWE	$\mathbb{T}_R$	$\mathbb{T}_R^{k+1}$	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k	4	4

TR(GSW) ciphertexts of μ R_Z

$$\text{TRGSW}(\mu) = \left(\begin{array}{c} \text{TRLWE}_K(K \cdot \frac{\mu}{2}) \\ \text{TRLWE}_K(K \cdot \frac{\mu}{4}) \\ \text{TRLWE}_K(K \cdot \frac{\mu}{8}) \\ \hline \text{TRLWE}_K(1 \cdot \frac{\mu}{2}) \\ \text{TRLWE}_K(1 \cdot \frac{\mu}{4}) \\ \text{TRLWE}_K(1 \cdot \frac{\mu}{8}) \end{array} \right)$$

1. **Internal Product (classical)**: : TRGSW × TRGSW $\longrightarrow$ TRGSW (Ring Structure)
2. **External product (Asiacrypt 2016)**: : TRGSW × TRLWE $\longrightarrow$ TRLWE (Module Structure)
   $(\mu_A, \mu_{\mathbf{b}}) \longrightarrow \mu_A \cdot \mu_{\mathbf{b}}$
   $(\epsilon_A, \epsilon_{\mathbf{b}}) \longrightarrow ||\mu_A||_1 \quad \epsilon_{\mathbf{b}} + O(\epsilon_A)$

If $||\mu_A||_1 = 1$ the noise propagation is linear!

	message	ciphertext	key	lin. combin.	product
TLWE	T	T^{n+1}	B^n	4	8
TRLWE	T_R	T_R^{k+1}	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k	4	4

TR(GSW) ciphertexts of μ R_Z

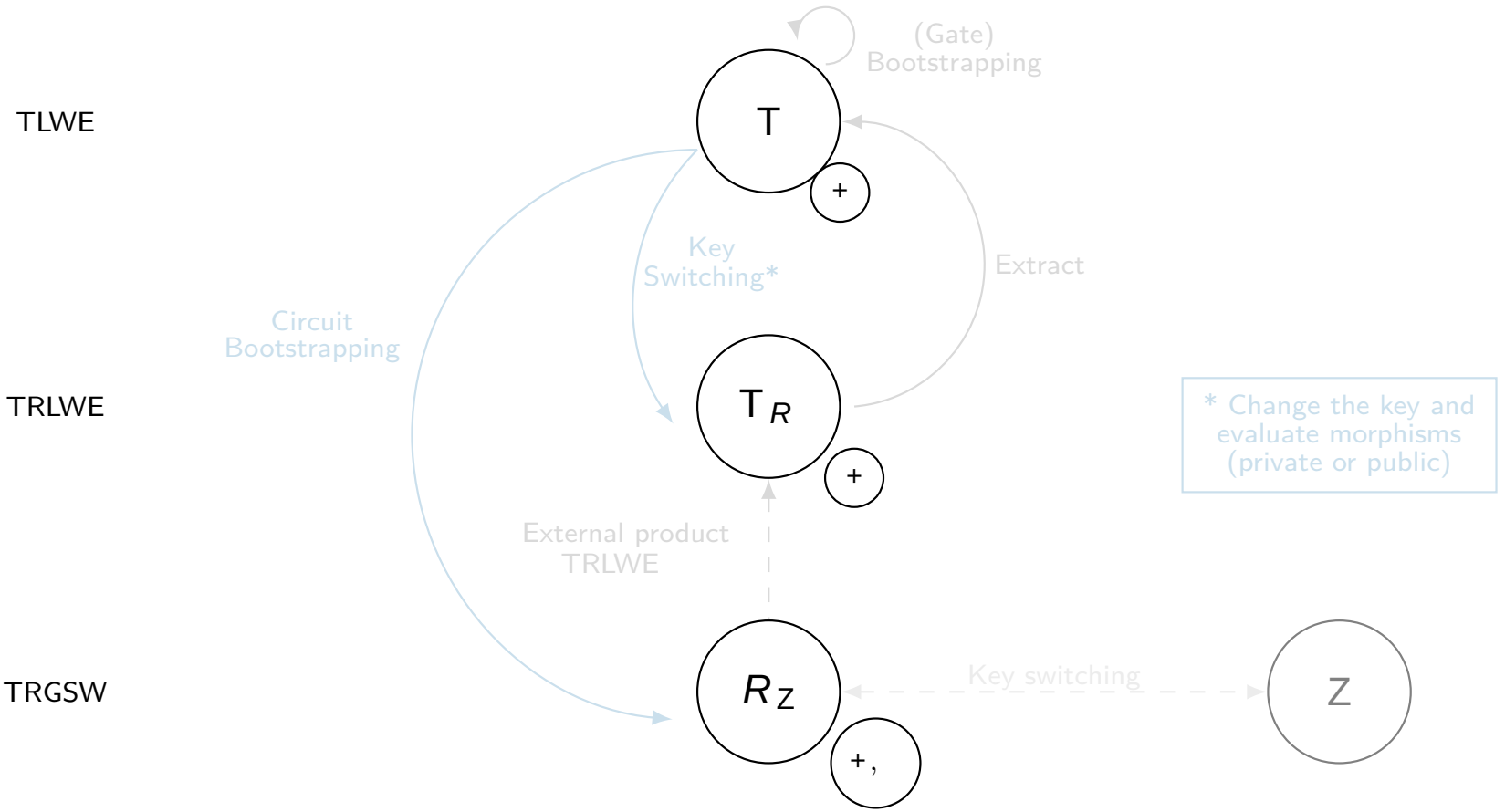
$$\mathsf{TRGSW}(\mu) = \left(\begin{array}{c} \mathsf{TRLWE}_K(K \cdot \frac{\mu}{2}) \\ \mathsf{TRLWE}_K(K \cdot \frac{\mu}{4}) \\ \mathsf{TRLWE}_K(K \cdot \frac{\mu}{8}) \\ \hline \mathsf{TRLWE}_K(1 \cdot \frac{\mu}{2}) \\ \mathsf{TRLWE}_K(1 \cdot \frac{\mu}{4}) \\ \mathsf{TRLWE}_K(1 \cdot \frac{\mu}{8}) \end{array}\right)$$

1. **Internal Product (classical)**: : TRGSW × TRGSW $\longrightarrow$ TRGSW (Ring Structure)
2. **External product (Asiacrypt 2016)**: : TRGSW × TRLWE $\longrightarrow$ TRLWE (Module Structure)
   $(\mu_A, \mu_{\mathbf{b}}) \longrightarrow \mu_A \cdot \mu_{\mathbf{b}}$
   $(\epsilon_A, \epsilon_{\mathbf{b}}) \longrightarrow ||\mu_A||_1 \quad \epsilon_{\mathbf{b}} + O(\epsilon_A)$

If $||\mu_A||_1 = 1$ the noise propagation is linear!

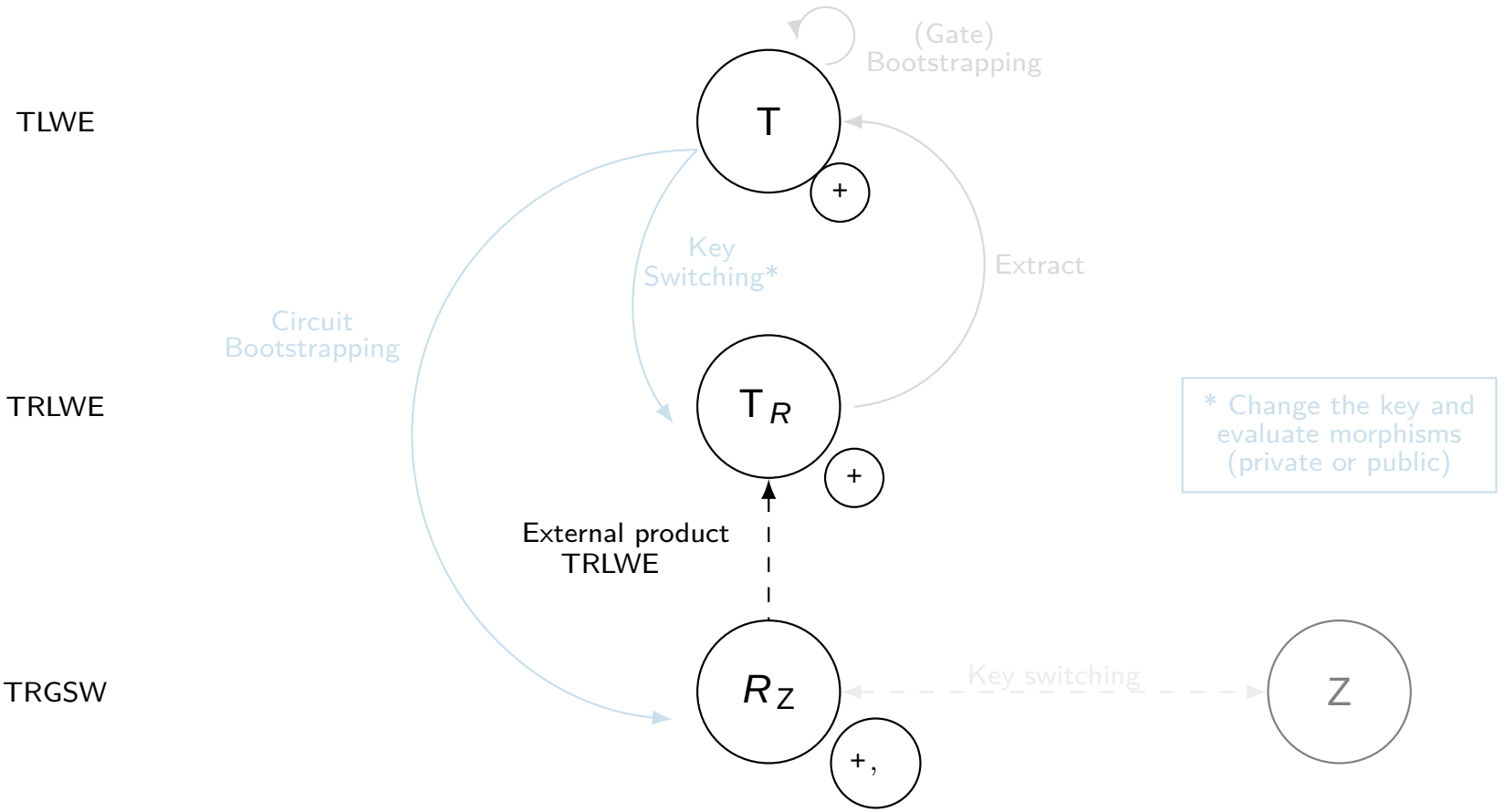
Homomorphic scheme

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$	4	8
TRLWE	$\mathbb{T}_R$	$\mathbb{T}_R^{k+1}$	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k	4	4



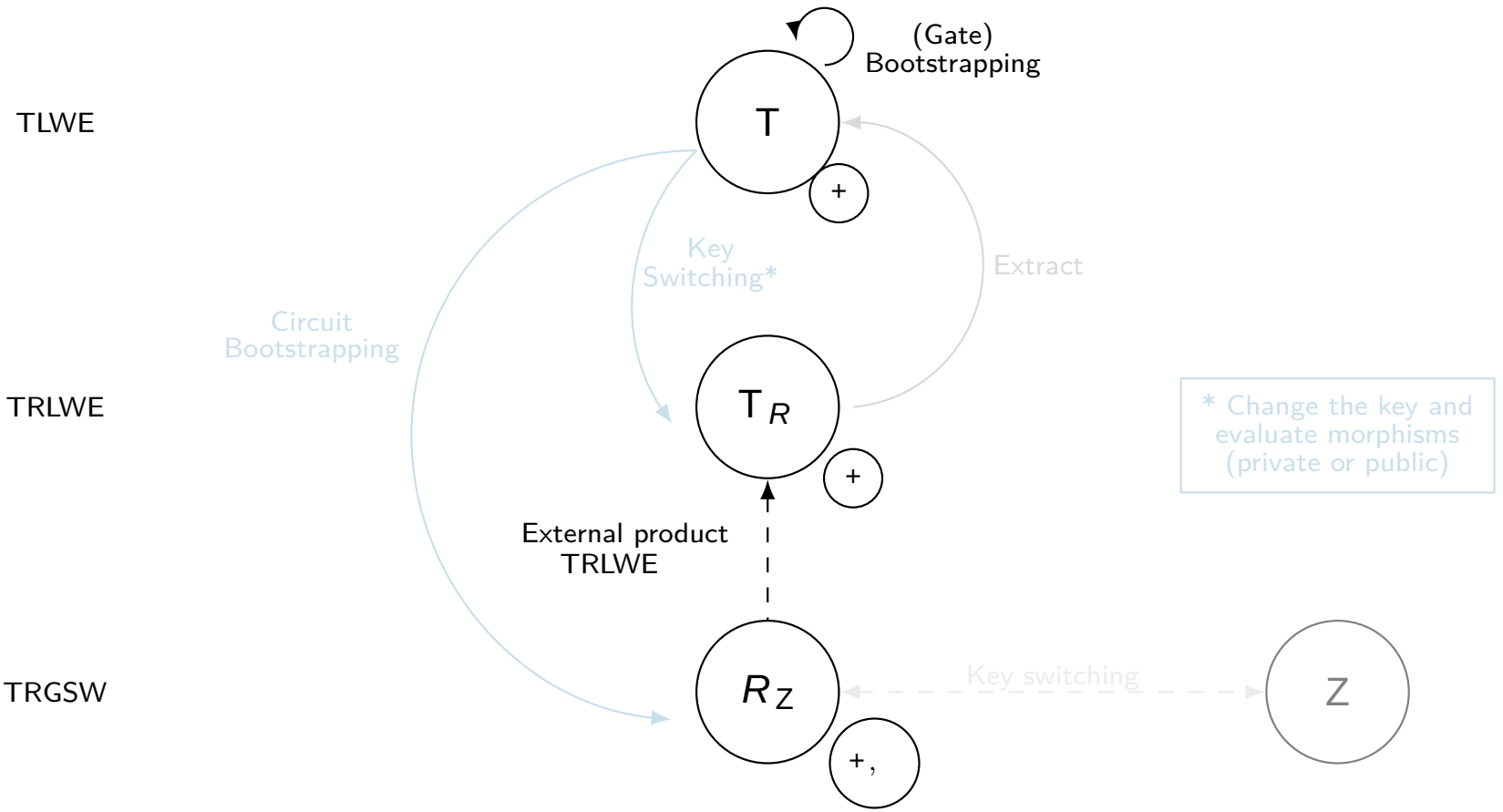
Homomorphic scheme

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$	4	8
TRLWE	$\mathbb{T}_R$	$\mathbb{T}_R^{k+1}$	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k	4	4



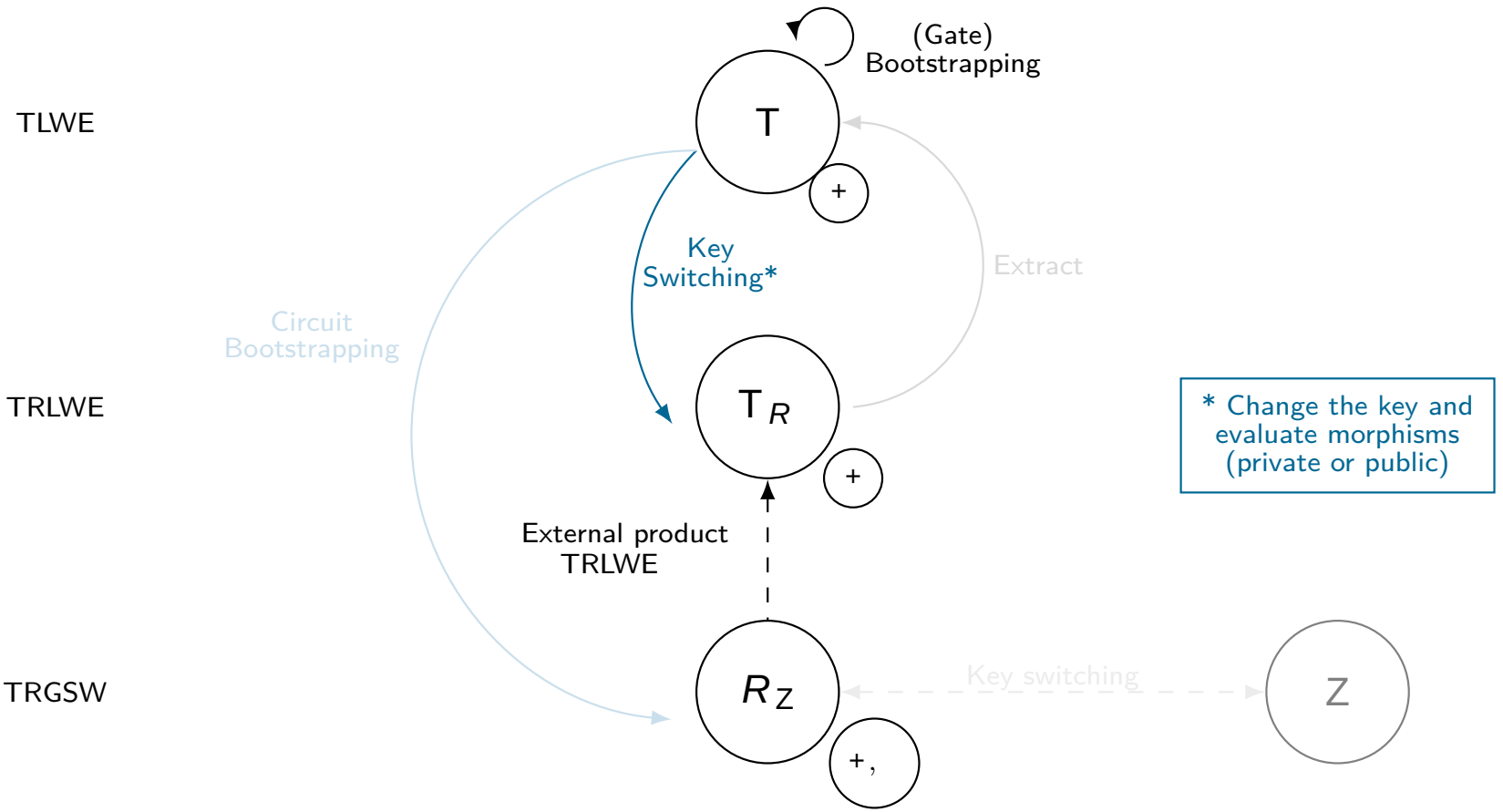
Homomorphic scheme

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$	4	8
TRLWE	$\mathbb{T}_R$	$\mathbb{T}_R^{k+1}$	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k	4	4



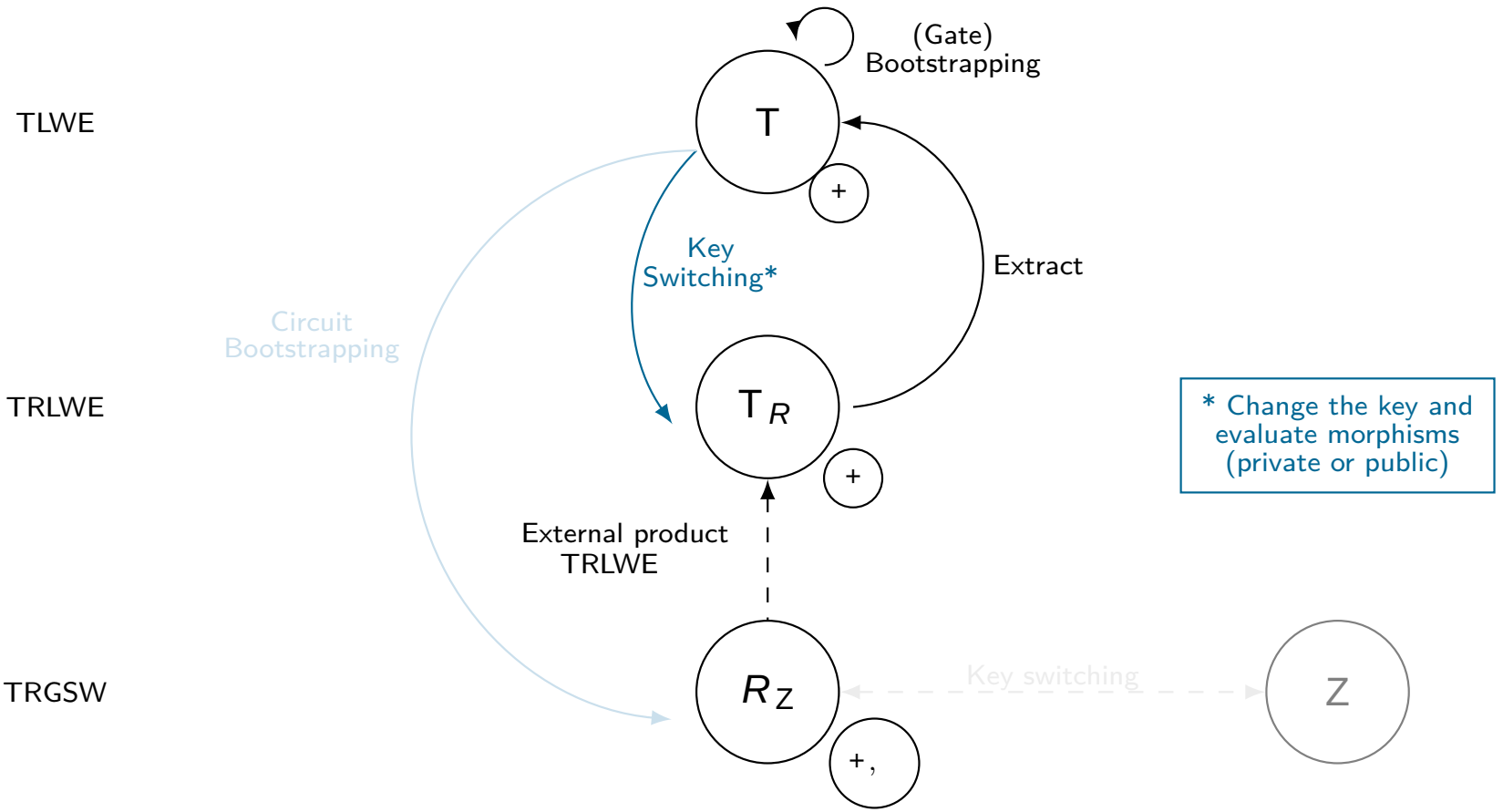
Homomorphic scheme

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$	4	8
TRLWE	$\mathbb{T}_R$	$\mathbb{T}_R^{k+1}$	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k	4	4



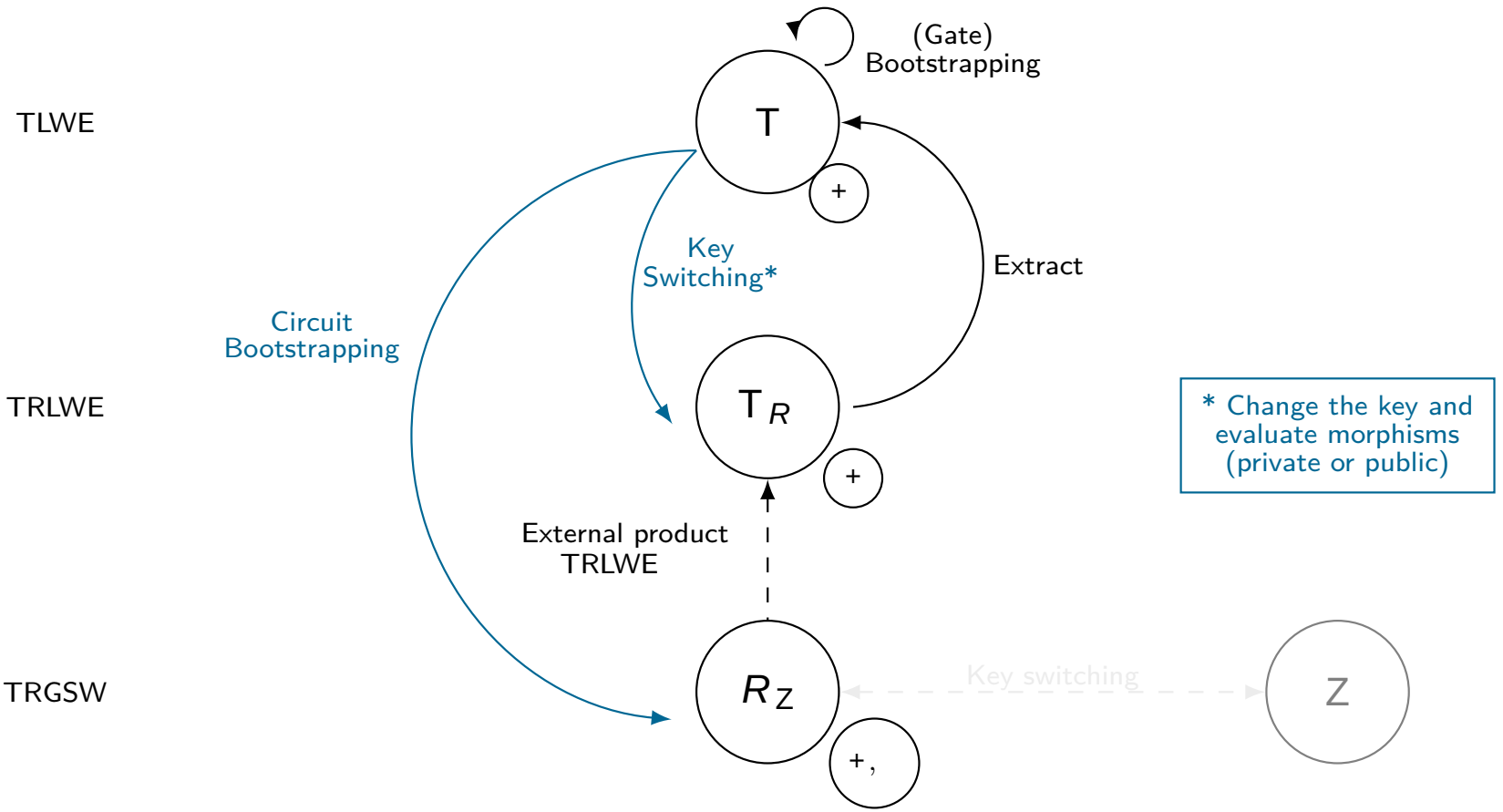
Homomorphic scheme

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$	4	8
TRLWE	$\mathbb{T}_R$	$\mathbb{T}_R^{k+1}$	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k	4	4



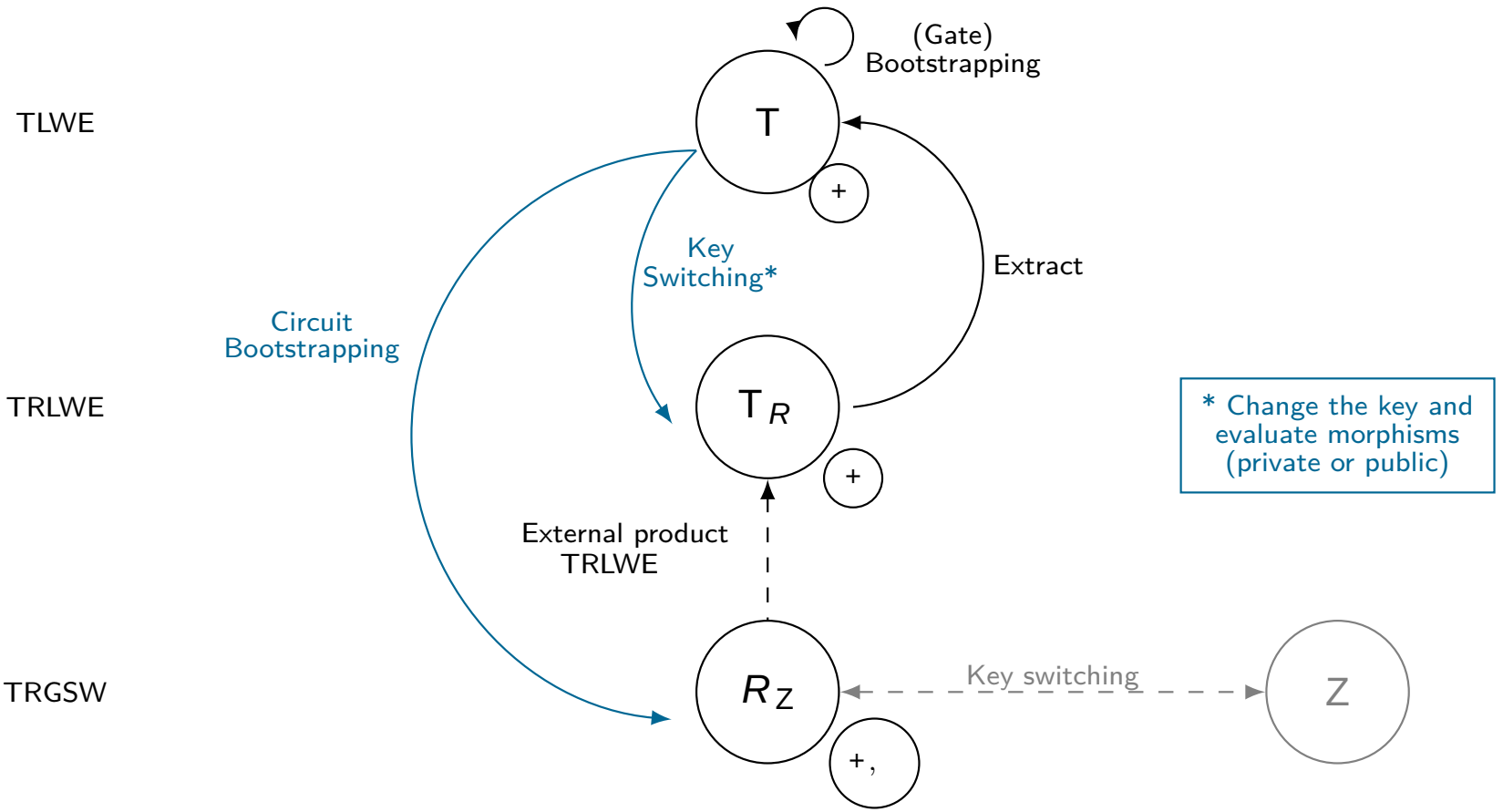
Homomorphic scheme

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$	4	8
TRLWE	$\mathbb{T}_R$	$\mathbb{T}_R^{k+1}$	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k	4	4



Homomorphic scheme

	message	ciphertext	key	lin. combin.	product
TLWE	$\mathbb{T}$	$\mathbb{T}^{n+1}$	$\mathbb{B}^n$	4	8
TRLWE	$\mathbb{T}_R$	$\mathbb{T}_R^{k+1}$	B^k	4	8
TRGSW	R_Z	ℓ-vector of TRLWE	B^k	4	4



Plan

How choose the homomorphic scheme?

inpher.io

Strengths of HE libraries

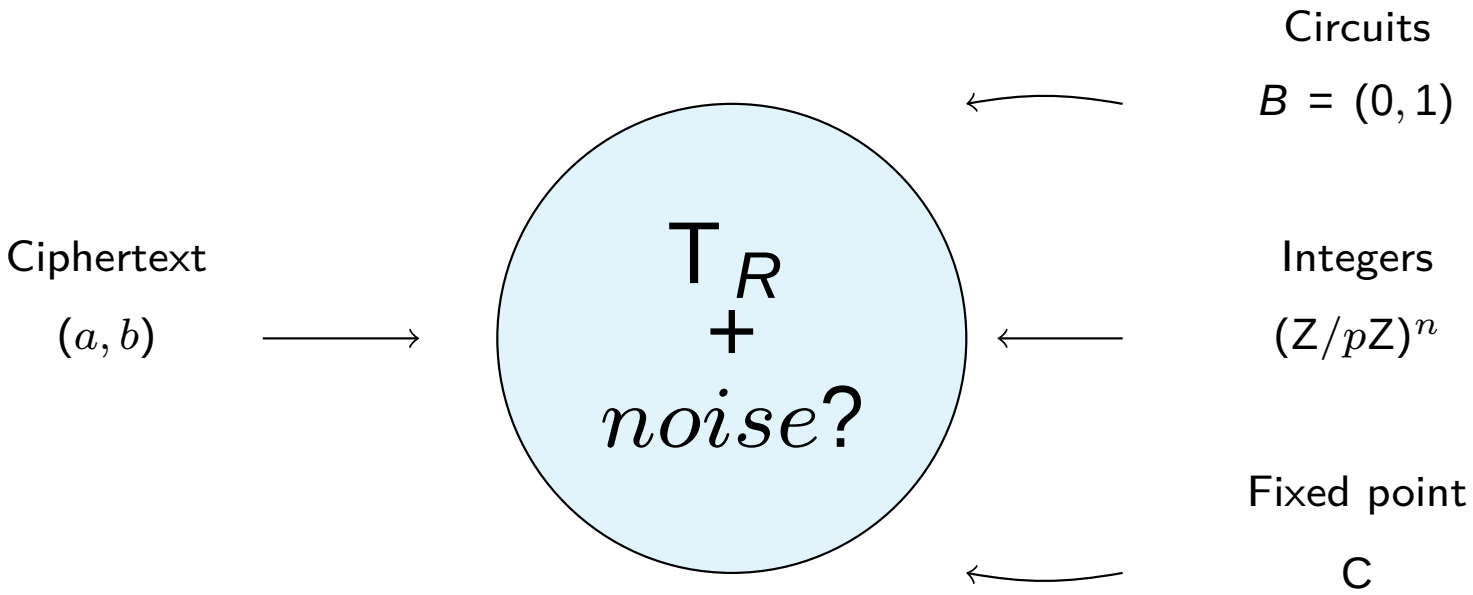
- BGV/Helib: SIMD finite field arithmetic
- B/FV, Seal: SIMD vector mod p
- HEAAN: SIMD fixed point arithmetic
- TFHE: single evaluation, boolean logic, comparison, threshold, complex circuits
- etc...

How to get all the benefits without the limitations?

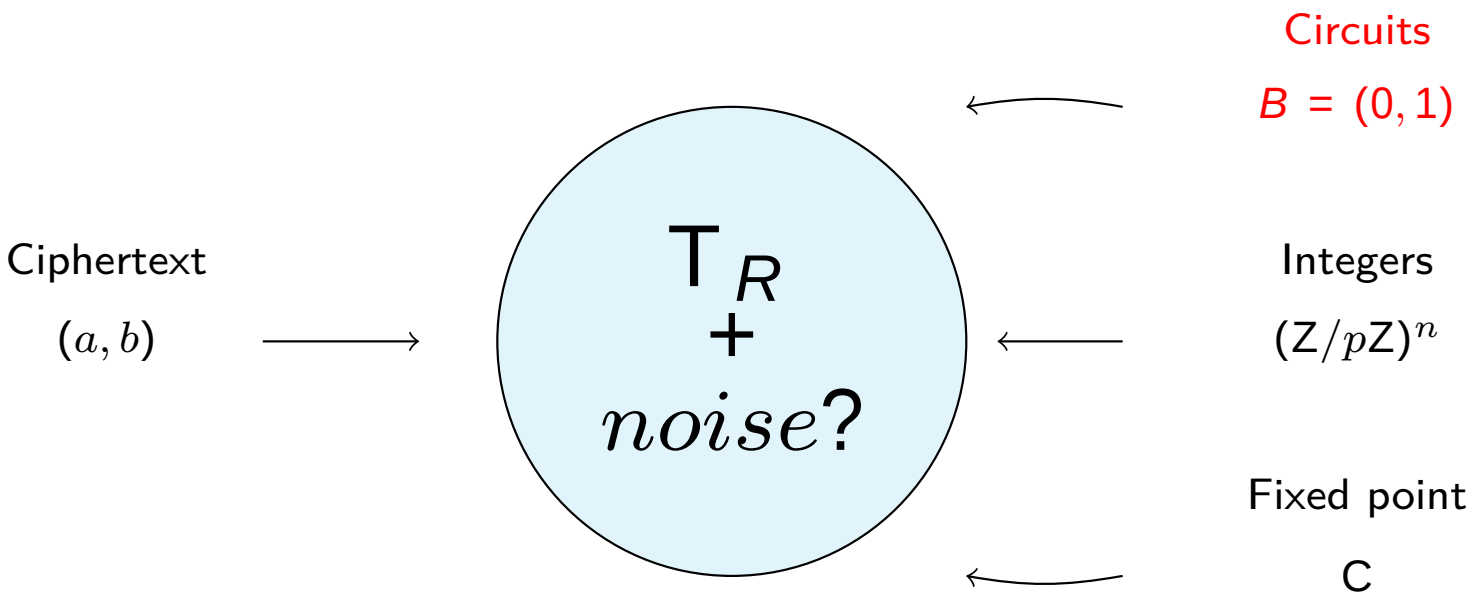
Solution: Chimera

- Unified plaintext space over the Torus
- Switch between ciphertext representations
- Implement bridges between TFHE, B/FV and HEAAN

How we can represent all plaintexts over the T_R?

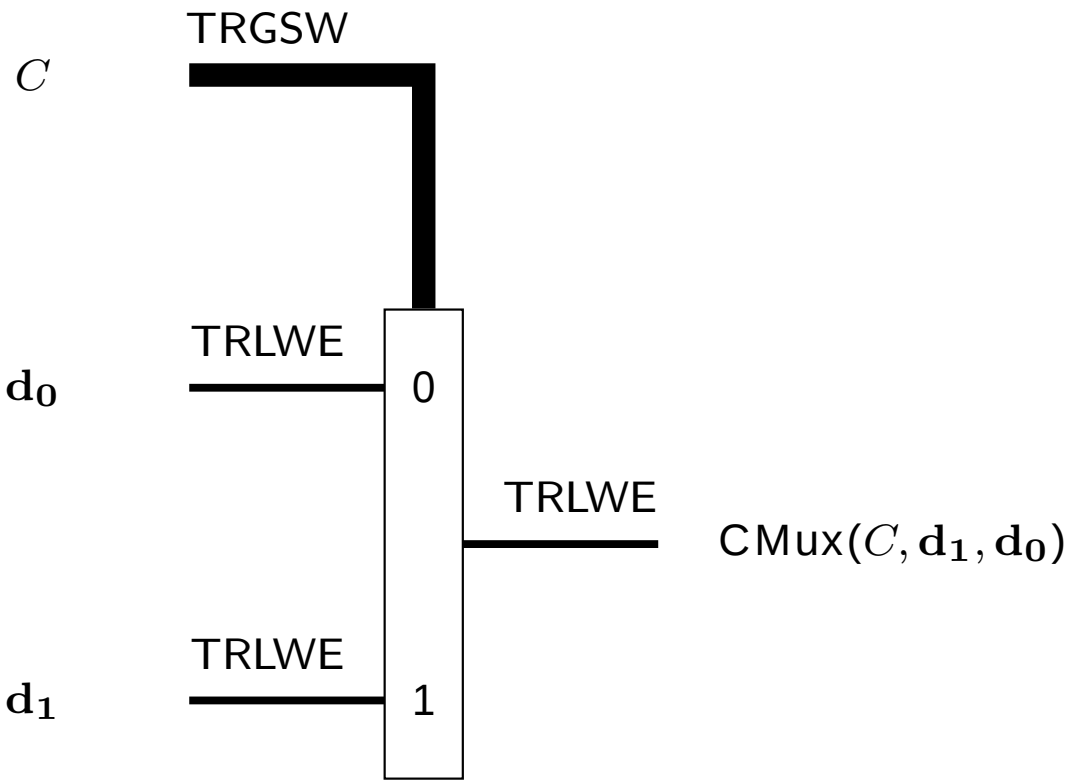


Circuit


Circuits
$B = (0, 1)$
Ciphertext
(a, b)
T_R
+
$noise?$
Integers
$(\mathbb{Z}/p\mathbb{Z})^n$
Fixed point
C

Circuit: CMux

$\mathsf{CMux}(C, d_1, d_0) = C \quad (d_1 - d_0) + d_0$



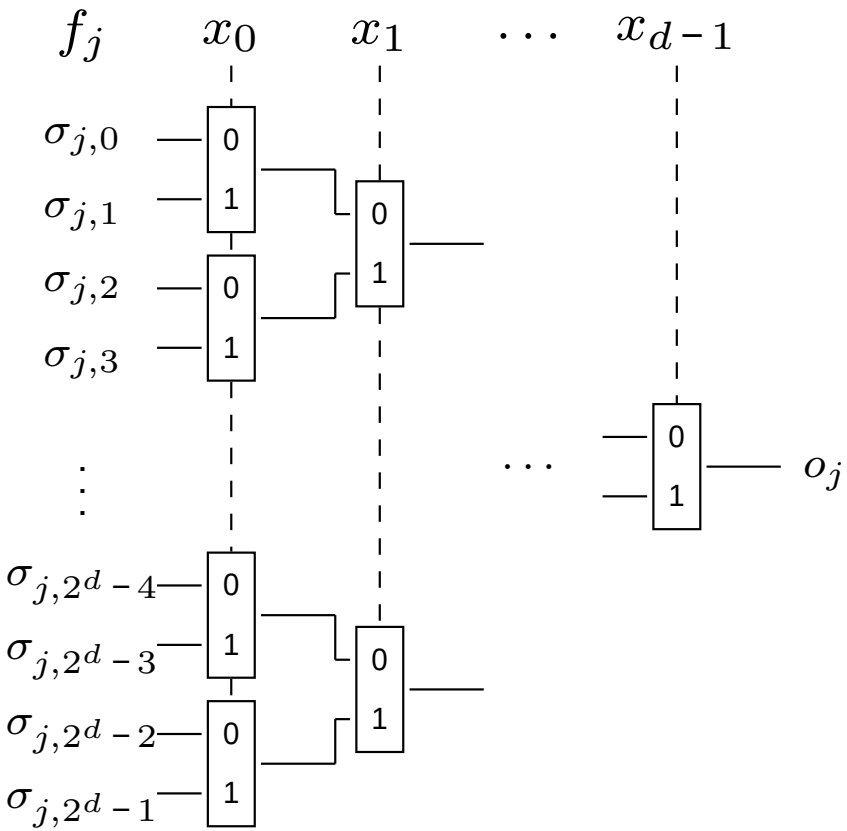
LUT evaluation

LookUp Tables (LUT) to evaluate arbitrary functions:

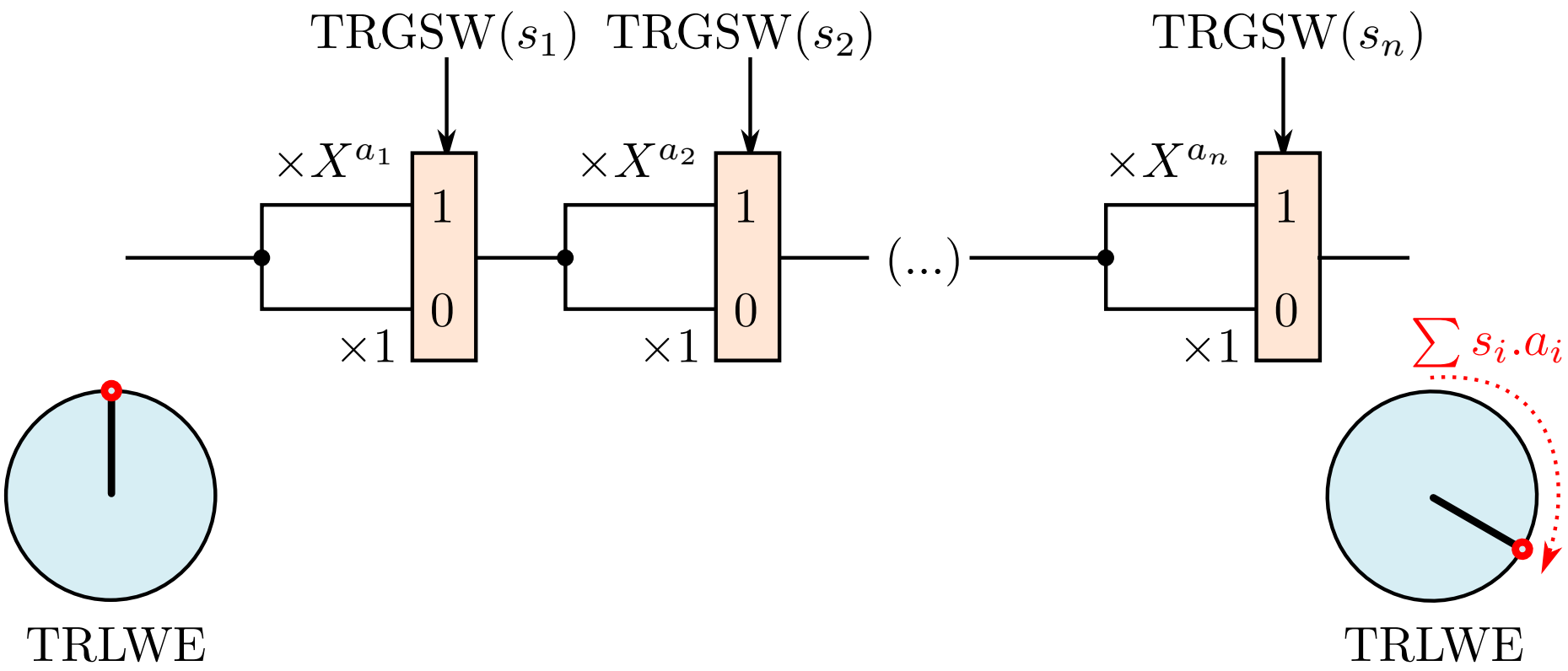
$$f : \mathsf{B}^d \longrightarrow \mathsf{T}^s$$

$$x = (x_0, \ldots, x_{d-1}) \longrightarrow f(x) = (f_0(x), \ldots, f_{s-1}(x))$$

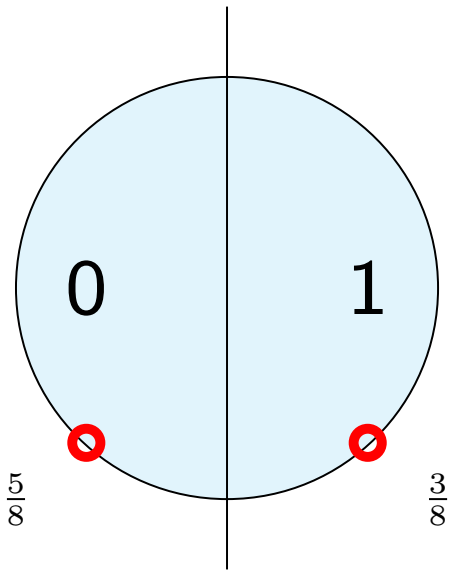
x_0	$\cdots$	x_{d-1}	f_0	$\cdots$	f_{s-1}
0	$\cdots$	0	$\sigma_{0,0}$	$\cdots$	$\sigma_{s-1,0}$
1	$\cdots$	0	$\sigma_{0,1}$	$\cdots$	$\sigma_{s-1,1}$
0	$\cdots$	0	$\sigma_{0,2}$	$\cdots$	$\sigma_{s-1,2}$
1	$\cdots$	0	$\sigma_{0,3}$	$\cdots$	$\sigma_{s-1,3}$
$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
0	$\cdots$	1	$\sigma_{0,2^d-4}$	$\cdots$	$\sigma_{s-1,2^d-4}$
1	$\cdots$	1	$\sigma_{0,2^d-3}$	$\cdots$	$\sigma_{s-1,2^d-3}$
0	$\cdots$	1	$\sigma_{0,2^d-2}$	$\cdots$	$\sigma_{s-1,2^d-2}$
1	$\cdots$	1	$\sigma_{0,2^d-1}$	$\cdots$	$\sigma_{s-1,2^d-1}$

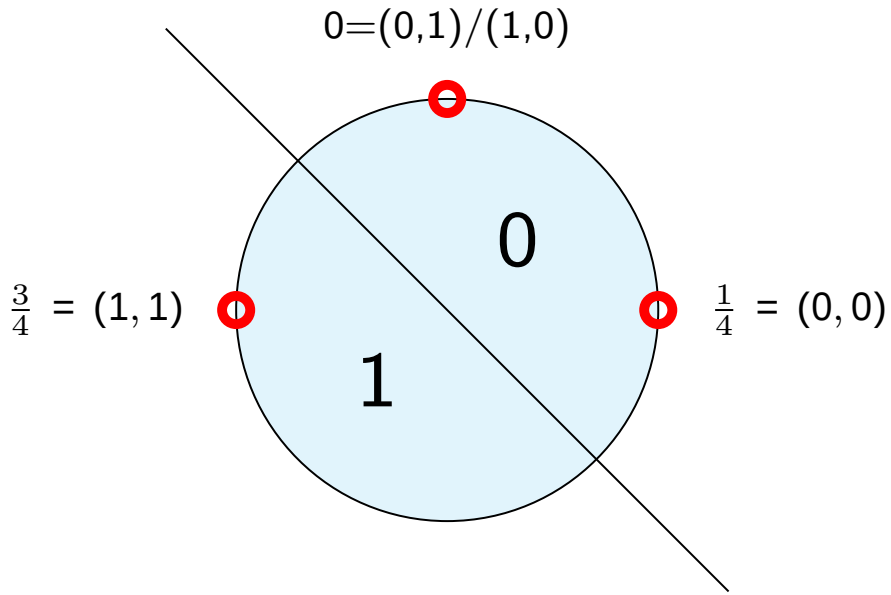


Blindrotate



Exemple AND



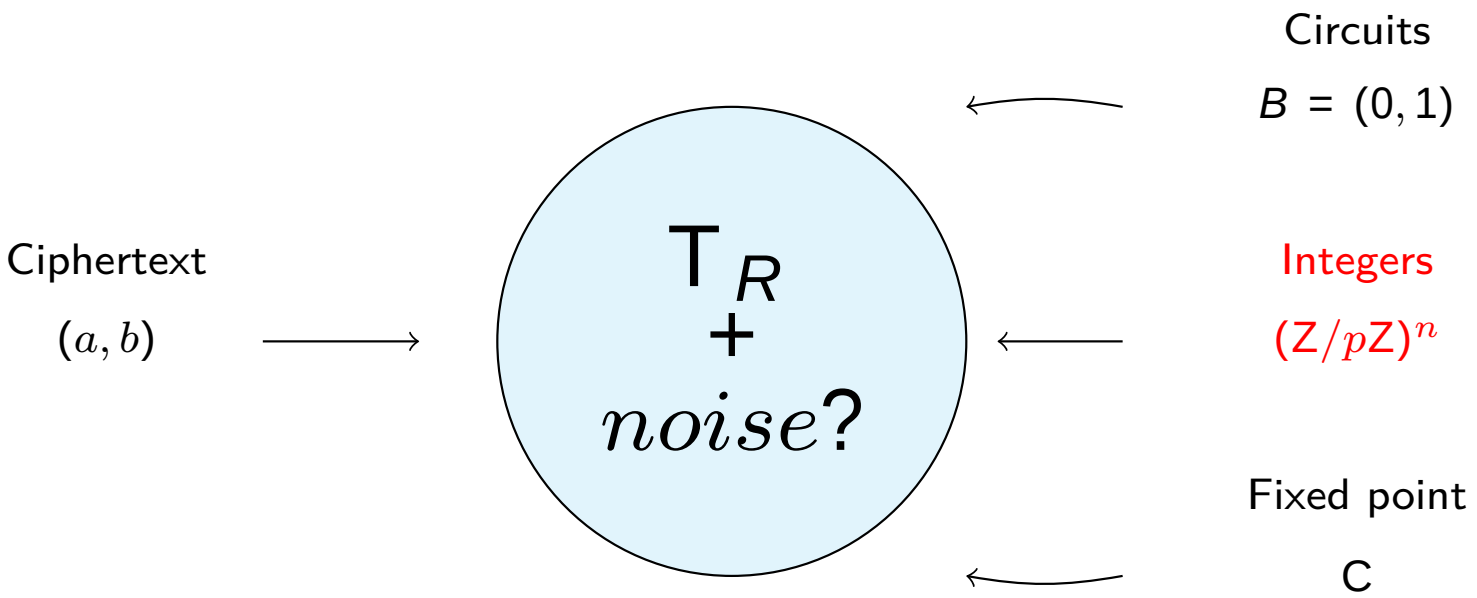


AND

Sum + BlindRotate

NAND, OR, NOT ...

Integers


Circuits
$B = (0, 1)$
Ciphertext
(a, b)
T_R
$+$
$noise?$
Integers
$(\mathbb{Z}/p\mathbb{Z})^n$
Fixed point
C

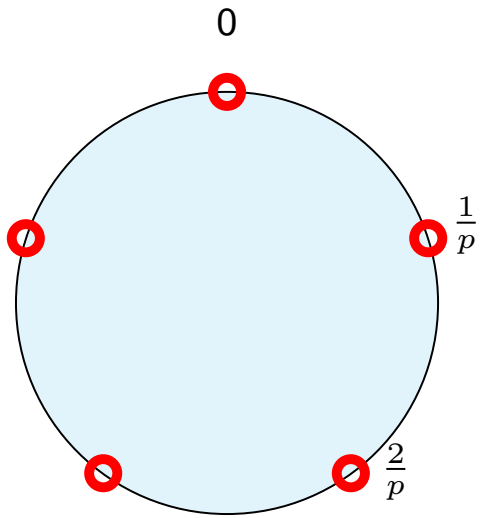
BFV scheme (encoding)

$R_Z \mod p$: the ring of polynomials with integer $\mod p$ coefficients module $X^N + 1$

If $X^N + 1$ has N roots mod p, Z/pZ^N is isomorphic to $R_Z \mod p$

$$(Z/pZ)^N \quad R_Z \mod p \quad \frac{1}{p} R_Z \mod 1$$

The plaintext space M is composed by exact multiples of $\frac{1}{p}$.



Plaintext addition $(\mu_1(X), \mu_2(X))$

$\mu_1(X) + \mu_2(X) := \mu_1(X) + \mu_2(X) \mod 1.$

Plaintext product (Montgomery) $(\mu_1(X), \mu_2(X))$

$\mu_1(X) \ {}_p \ \mu_2(X) := p \cdot \mu_1(X) \cdot \mu_2(X) \mod 1.$

Problem of lift

inpher.io

Examples: $p = 3$, $\mu_1 = \frac{1}{3}$ and $\mu_2 = \frac{2}{3}$

- Exact product: $3(I_1 + \frac{1}{3})(I_2 + \frac{2}{3}) = I + \frac{2}{3} = +\frac{2}{3} \mod 1$, for all I_1, I_2 integers
- Product with noise and small element: 3 5.33333 10.66665 = 170.6662
- Product with noise and big element: 3 12345678.33333 7654321.66665 = −.839...

- We need a small representative of the plaintext to keep the result correct.
- We should lift the ciphertext to small representative in R[X] (all coefficients in $[-1/2, 1/2)$).
- $\frac{1}{p}$ $noise$

Homomorphic operations

inpher.io

Homomorphic addition $c_1 = (a_1, b_1)$, $c_2 = (a_2, b_2)$

$(a, b) = (a_1 + a_2, b_1 + b_2)$

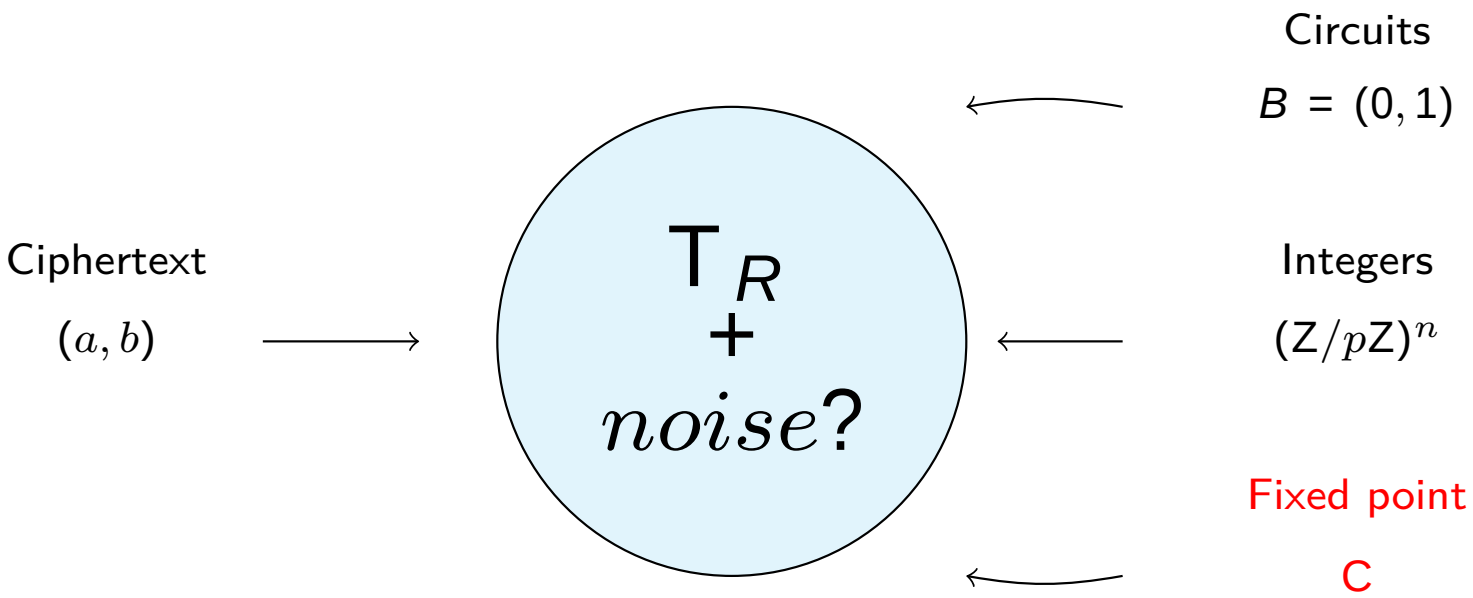
Homomorphic product $c_1 = (a_1, b_1), c_2 = (a_2, b_2)$

$$p(b_1 - s.a_1)(b_2 - s.a_2) = \underbrace{(p.b_1.b_2)}_{C_0} - s.\underbrace{(p.a_1.b_2 + p.a_2.b_1)}_{C_1} + s^2.\underbrace{(p.a_1.a_2)}_{C_2}$$
$$= (b - s.a)$$

Relinearize the term $(p.a_1.a_2)s^2$ using the external product:

$$c_1 \quad {}_p\, c_2 = (C_1, C_0) - TRGSW(s) \quad (C_2, 0)$$

Fixed point


Circuits
$B = (0, 1)$
Ciphertext
(a, b)
T_R
$+$
$noise?$
Integers
$(\mathsf{Z}/p\mathsf{Z})^n$
Fixed point
C

There are two models: Fixed points and Floating point

inpher.io

Floating point (float, double in C):

- $x = m.2^{\tau}$, with $m \quad 2^{-\rho}.\mathbb{Z}$ and $\frac{1}{2} \leq |m| < 1$
- $\tau = log_2(x)$ data dependent and not public (not FHE-friendly)
- The exponent is always in sync with the data
  ex: $(1.23 \cdot 10^{-4}) \quad (7.24 \cdot 10^{-4}) = (8.90 \cdot 10^{-8})$

Fixed point:

- $x = m.2^{\tau}$, with $m \quad 2^{-\rho}.\mathbb{Z}$ and $0 \leq |m| < 1$,
- τ is public, thus FHE-friendly
- Risk of overflow (τ too small)
- Risk of underflow (τ too large)
  ex: $(0.000123 \cdot 10^{0}) \quad (0.000724 \cdot 10^{0}) = (0.000000 \cdot 10^{0})$

Addition is much tricker than you think!

- Given (m_1, τ_1), (m_2, τ_2), and τ.
- How do you compute $m.2^{\tau} = m_1.2^{\tau_1} + m_2.2^{\tau_2}$ with ρ bits of precision?
- Addition requires right shift and roundings, which are non-linear!

There are two models: Fixed points and Floating point

inpher.io

Floating point (float, double in C):

- $x = m.2^{\tau}$, with $m \quad 2^{-\rho}.\mathbb{Z}$ and $\frac{1}{2} \leq |m| < 1$
- $\tau = log_2(x)$ data dependent and not public (not FHE-friendly)
- The exponent is always in sync with the data
  ex: $(1.23 \cdot 10^{-4}) \quad (7.24 \cdot 10^{-4}) = (8.90 \cdot 10^{-8})$

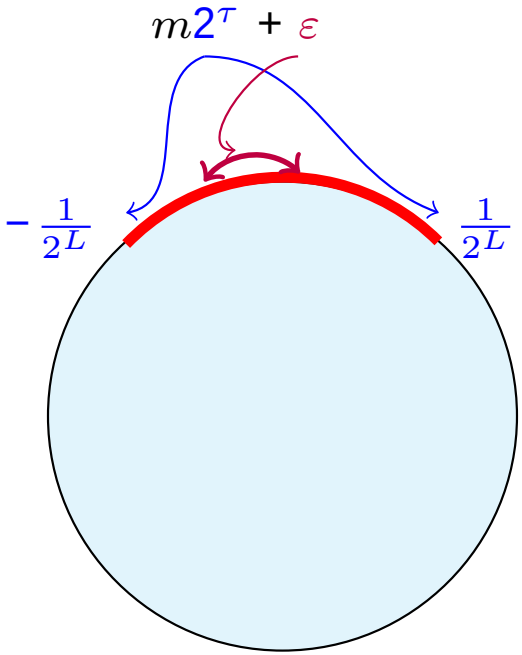
Fixed point:

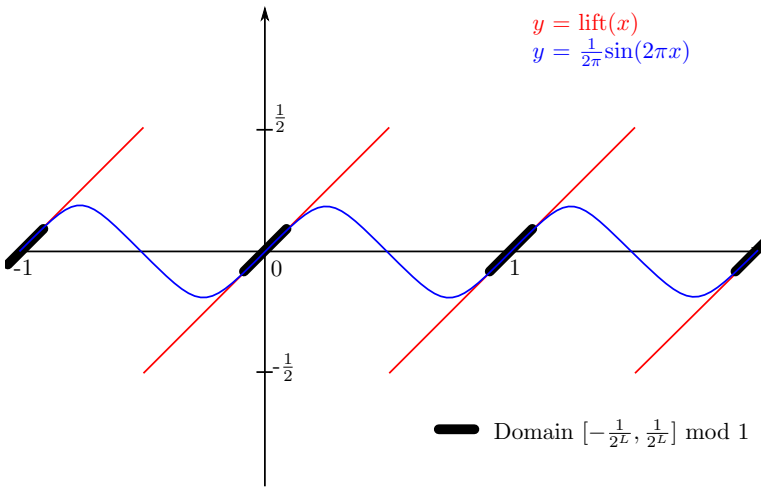
- $x = m.2^{\tau}$, with $m \quad 2^{-\rho}.\mathbb{Z}$ and $0 \leq |m| < 1$,
- τ is public, thus FHE-friendly
- Risk of overflow (τ too small)
- Risk of underflow (τ too large)
  ex: $(0.000123 \cdot 10^{0}) \quad (0.000724 \cdot 10^{0}) = (0.000000 \cdot 10^{0})$

Addition is much tricker than you think!

- Given (m_1, τ_1), (m_2, τ_2), and τ.
- How do you compute $m.2^{\tau} = m_1.2^{\tau_1} + m_2.2^{\tau_2}$ with ρ bits of precision?
- Addition requires right shift and roundings, which are non-linear!

HEAAN





Continuous approach

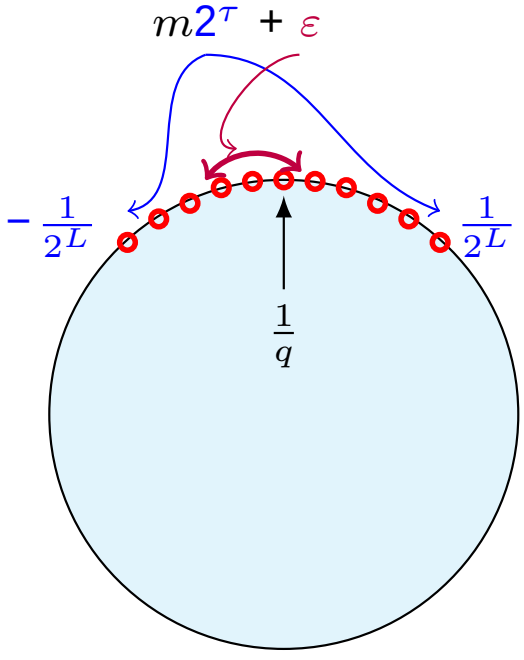
- $x \times y = \text{Lift}(x) \quad \text{Lift}(y) \mod 1$.

4 This approach can preserve (or reduce) the interval $[-\frac{1}{2^L}, \frac{1}{2^L}]$

4 Lift is a periodic function: approx by sinus (or other Fourier serie) wherever it matters...

8 ...but sinus can only be approx by a polynomial, which recursively requires a product.

Fixed point: HEAAN



Discrete approach

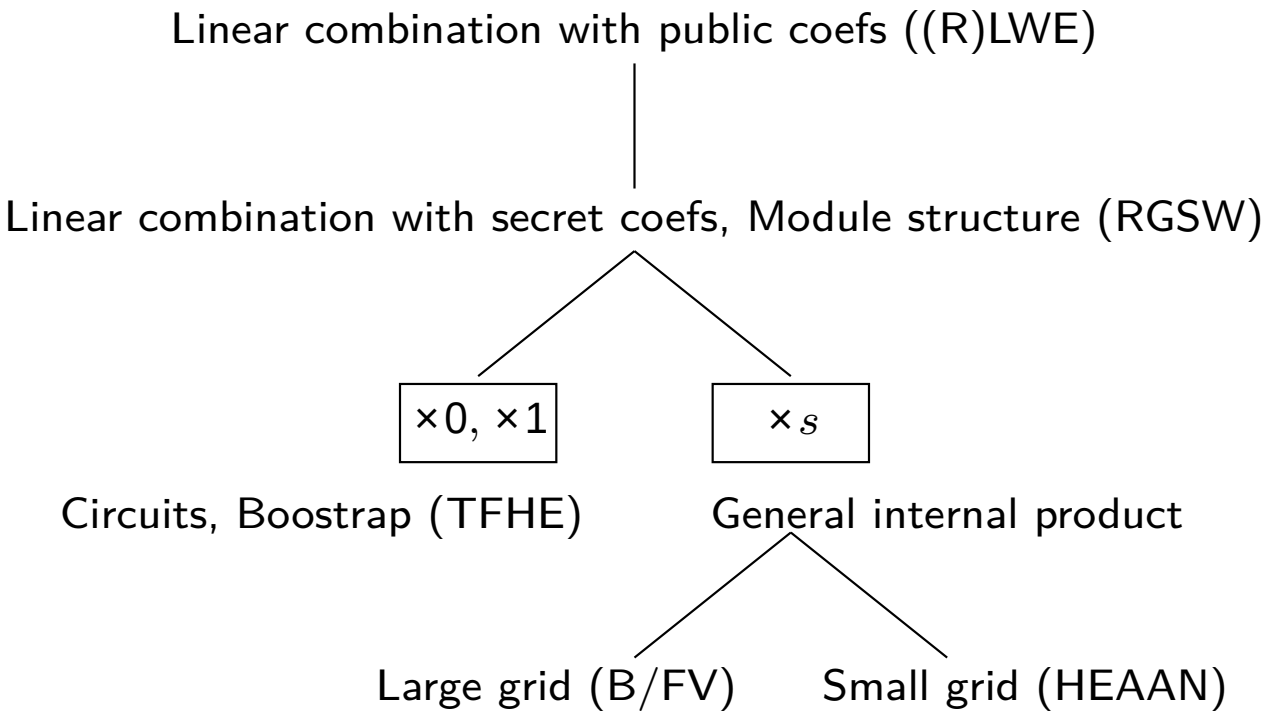
- round a, b (and thus μ) on exact multiples of $\frac{1}{q}$ where $q \approx 2^{L+\rho}$.

4 Brings us in the ring $\frac{1}{q}R_Z \mod 1$ (avoids lifting)

4 Exact Montgomery product $q(b_1 - sa_1)(b_2 - sa_2)$

8 Blows up the interval $[-\frac{1}{2^L}, \frac{1}{2^L}] \to [-\frac{1}{2^{L-\rho}}, \frac{1}{2^{L-\rho}}]$...
...works a leveled number of times.

Homomorphic operations hierarchy



Coefficient and Slot packing

inpher.io

Coefficient packing

$$\mathbf{m} = \sum_{i=0}^{N-1} m_i \cdot X^i \qquad \mathbf{m} = (m_0, m_1, \ldots, m_{N-1})$$

m_0	m_1	m_2	$\cdots$	m_{N-2}	m_{N-1}

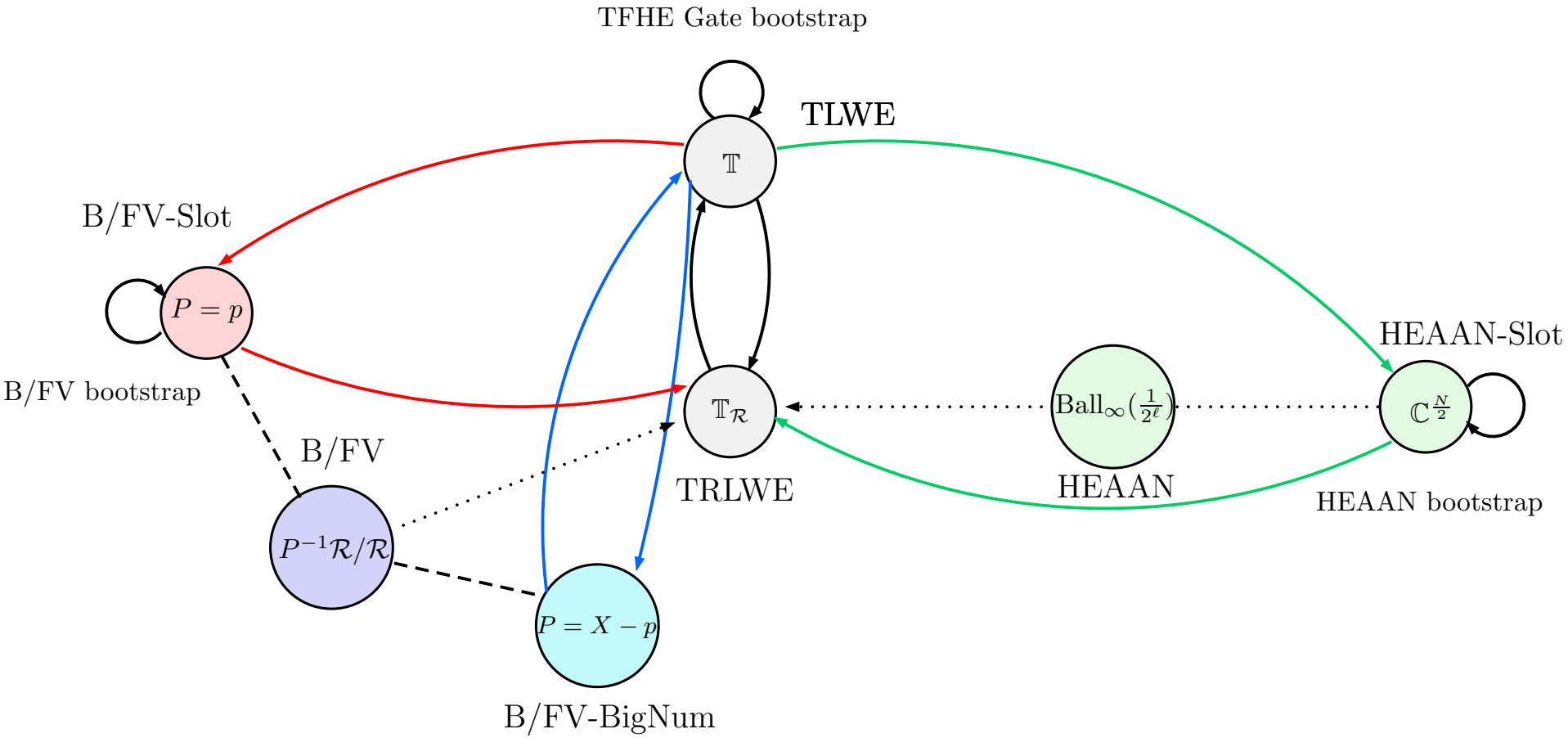
Slot packing

$$X^N + 1 = \prod_{i=0}^{N-1} (X - \omega_i) \qquad \mathbf{m} = (\mathbf{m}(\omega_0), \mathbf{m}(\omega_1), \ldots, \mathbf{m}(\omega_{N-1}))$$

$\mathbf{m}(\omega_0)$	$\mathbf{m}(\omega_1)$	$\mathbf{m}(\omega_2)$	$\cdots$	$\mathbf{m}(\omega_{N-2})$	$\mathbf{m}(\omega_{N-1})$

There exists morphism to switch between the coefficient and slot representation!
(Vandermonde, DFT,...)

Conclusion


TFHE Gate bootstrap
TLWE
$\mathbb{T}$
B/FV-Slot
$P = p$
B/FV bootstrap
B/FV
$P^{-1}\mathcal{R}/\mathcal{R}$
$\mathbb{T}_{\mathcal{R}}$
TRLWE
$\mathrm{Ball}_\infty(\frac{1}{2^\ell})$
HEAAN
HEAAN-Slot
$\mathbb{C}^{\frac{N}{2}}$
HEAAN bootstrap
$P = X - p$
B/FV-BigNum

Questions?

inpher.io