Deniable Fully Homomorphic Encryption from LWE

Shweta Agrawal, Shafi Goldwasser, Saleet Mossel

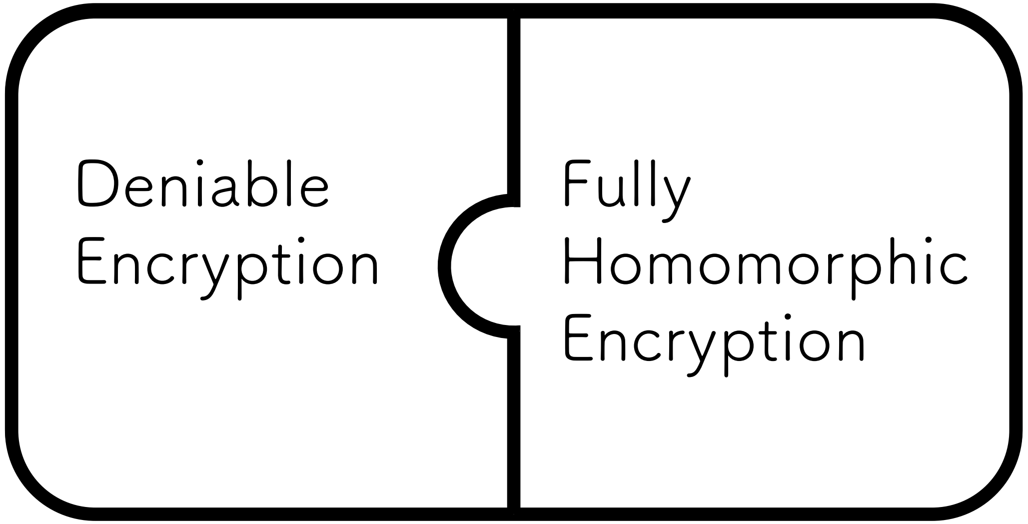
Crypto, 2021 (To appear)

Deniable Fully Homomorphic Encryption from LWE

Shweta Agrawal, Shafi Goldwasser, Saleet Mossel

Crypto, 2021 (To appear)

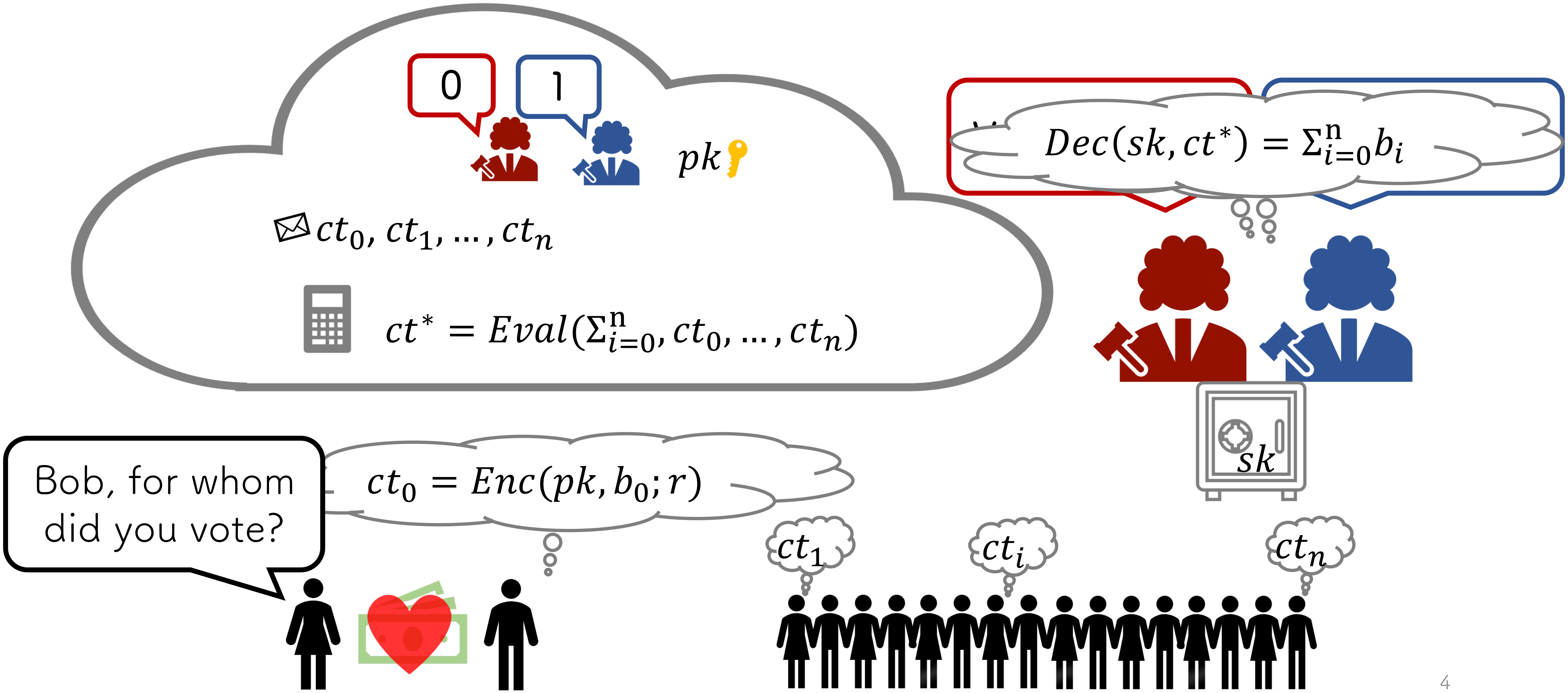
Most slides by Saleet Mossel



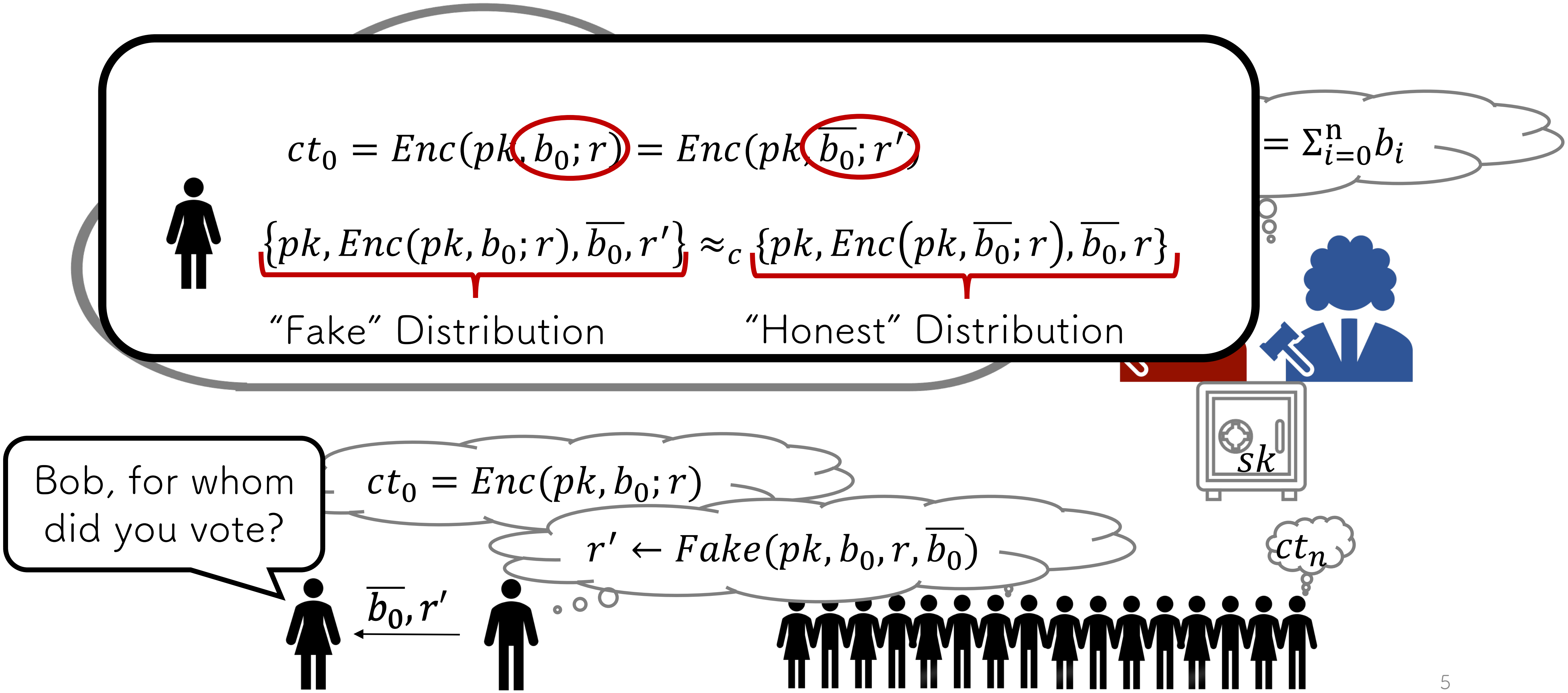
Deniable FHE

The notion of Deniable FHE

Deniable FHE



Deniable FHE



Elections require Deniability & FHE

- Benefit of Deniable Encryption in Elections:
  - Honest Participation

- Benefit of Fully Homomorphic Encryption in Elections:
  - Homomorphically compute the voting result

Any data-driven algorithm

Store Encrypted Data

Natural upgrade for all DE apps!

Deniable Encryption

- Introduced by **Canetti, Dwork, Naor and Ostrovsky 1997**
  - construction from trapdoor permutations, unique SVP
  - size of ct is the inverse of the detection probability

- Weak Deniable Encryption
  - can also lie about the encryption algorithm (Enc, Denc)
  - construction with compact ct and negligible deniability

- Lower bound (Efficiency vs. Deniability)
  - It **seems inherent** that the length of ct grows with the inverse of the detection probability in "separable" constructions.

- A significant step forward [SW14]
  - construction from iO and OWF
  - compact ct and negligible deniability

What does this mean given recent iO results?

Deniable Encryption

CDNO

- Based on TDP
- CT size inverse of detection prob

1997

SW

- Based on iO
- CT size indpt of detection prob

2014

In full model, nothing else known!

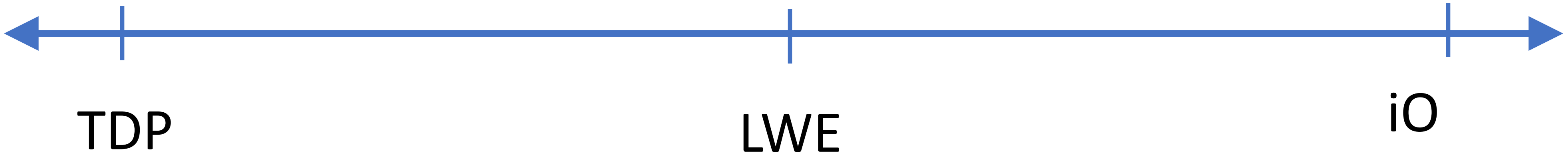
Our Results

- Notion of Deniable FHE (full and weak)
- Constructions based on Learning With Errors
- Compact ct : size does not depend on detection probability!
  - Our construction is separable (so not inherent)
  - Total encryption time grows with the inverse of the detection probability!
- Support large message space
  - All prior work encode large messages bit by bit
- Offline-Online Encryption
  - Online time independent of the detection probability

Our Results: Deniable Encryption

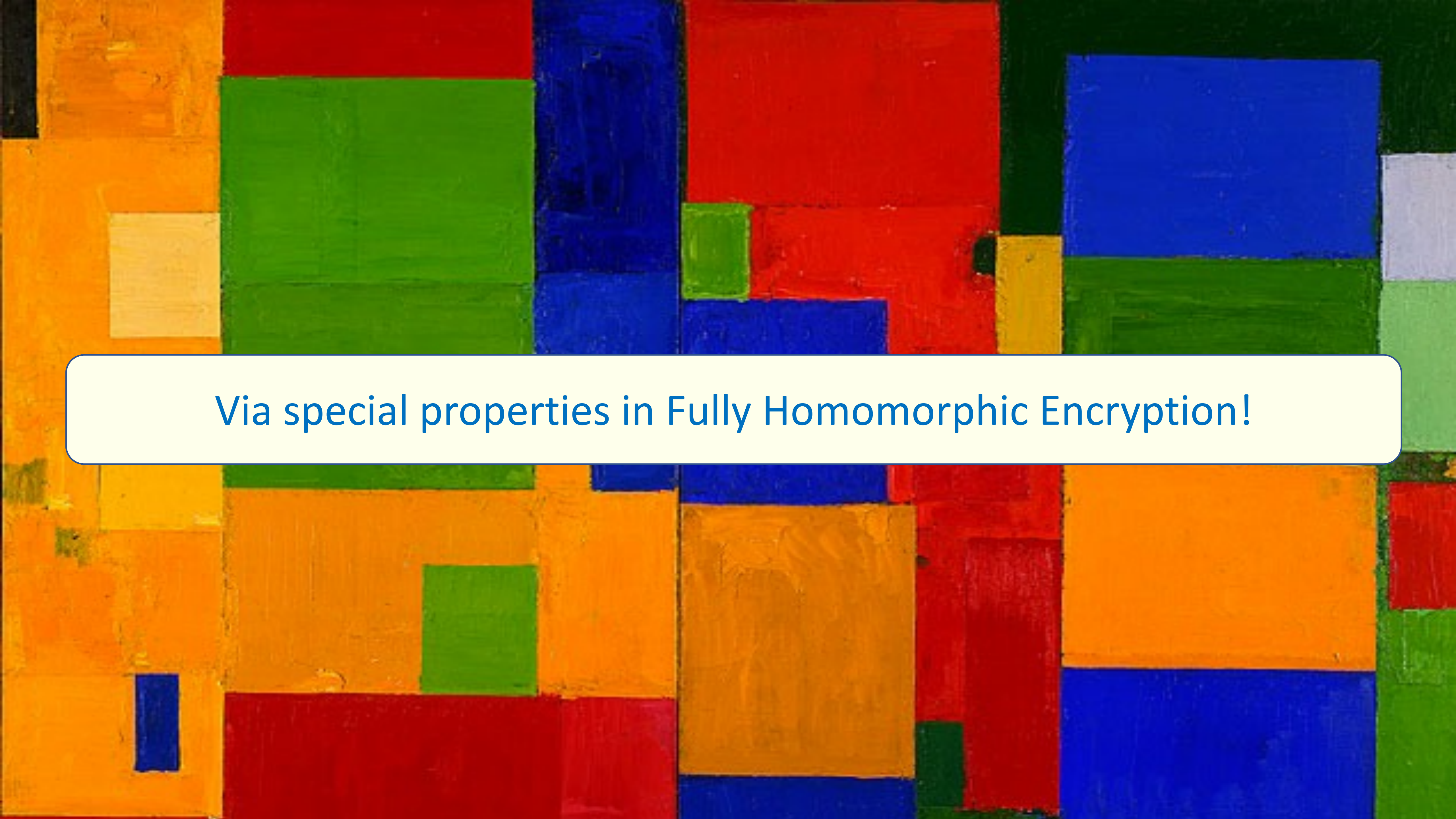
CDNO, 1997
CT size inverse of detection prob

SW, 2014
CT size indpt of detection prob



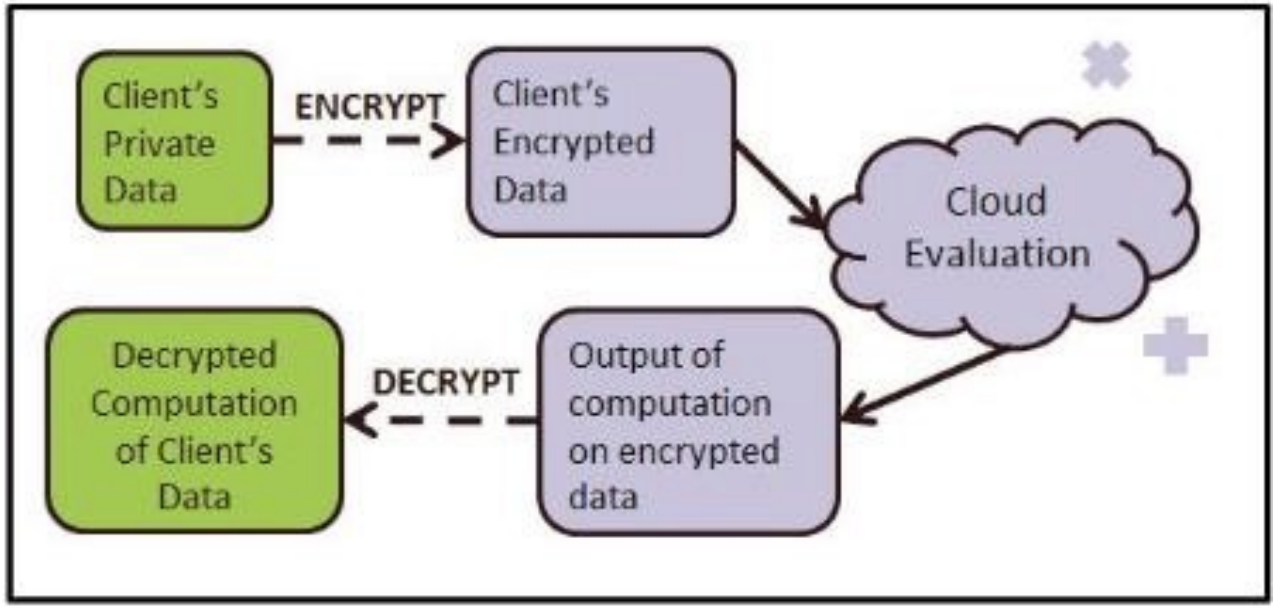
This Work

- CT size independent of detection prob
- (Offline) encryption time inverse of detection prob


Via special properties in Fully Homomorphic Encryption!

Fully Homomorphic Encryption

Can be built using LWE (BV11, BGV12, GSW13…)



Expressive Functionality: Supports arbitrary circuits

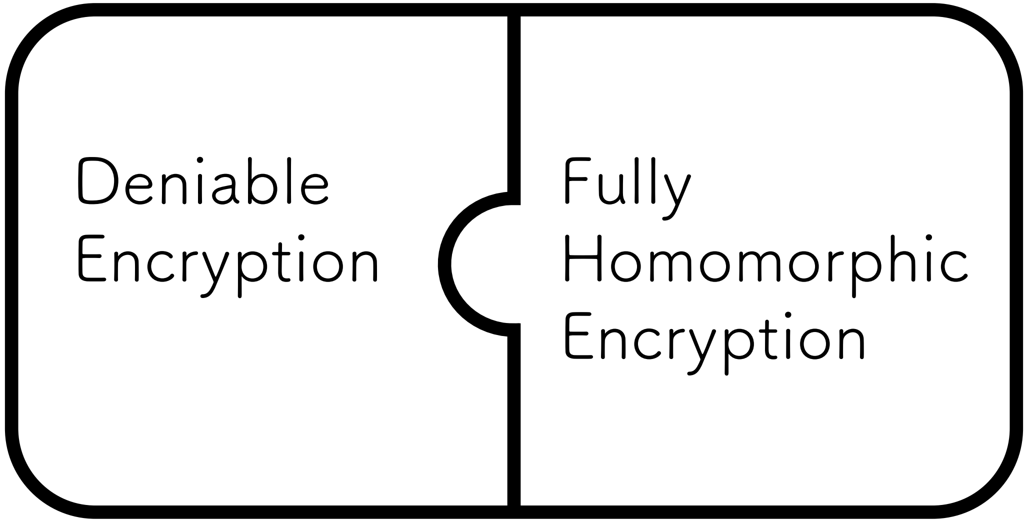
Compact ciphertext, independent of circuit size

Encryption and function evaluation commute!
Enc(f(x)) =* f(Enc(x))

* : roughly

Adding Deniability to the Mix

- A Deniable FHE scheme $(Gen, Enc, Eval, Dec, Fake)$
  - $(Gen, Enc, Eval, Dec)$ is an FHE scheme
  - $(Gen, Enc, Dec, Fake)$ is a Deniable Encryption scheme



Deniable FHE

A Deniable FHE scheme $(Gen, Enc, Eval, Dec, Fake)$ syntax

- $Gen \to (pk, sk)$
- $Enc(pk, m; r) = ct$
- $Dec(sk, ct) = b$
- $Eval(pk, f, ct_1, \dots, ct_k) = ct^*$
- $Fake(pk, b, r, \bar{b}) \to r'$

Deniable FHE

A Deniable FHE scheme $(Gen, Enc, Eval, Dec, Fake)$

1. Correctness
2. CPA-Security
3. Deniability
4. Compactness

Correctness versus Deniability

Correctness:

For every $\boldsymbol{f}$ and $\boldsymbol{m_1, \ldots, m_k}$:

$$\mathbf{Pr}\left[\boldsymbol{Dec}\left(\boldsymbol{sk}, \boldsymbol{Eval}(\boldsymbol{pk}, \boldsymbol{f}, \boldsymbol{ct_1}, \ldots, \boldsymbol{ct_k})\right) = \boldsymbol{f}(\boldsymbol{m_1}, \ldots, \boldsymbol{m_k})\right] = \mathbf{1} - \boldsymbol{negl}$$

where $\boldsymbol{ct_i} \leftarrow \boldsymbol{Enc}(\boldsymbol{pk}, \boldsymbol{m_i})$ and $(\boldsymbol{pk}, \boldsymbol{sk}) \leftarrow \boldsymbol{Gen}$

Cannot simultaneously satisfy <u>perfect</u> correctness and <u>deniability</u>

$\delta(\lambda)$- Deniability

We consider (inverse) polynomial deniability

For every bit $\boldsymbol{b}$, and PPT adversary $\boldsymbol{A}$

detection probability

$$\left|\mathbf{Pr}[A(pk, Enc(pk, b; r), b, r)] - \mathbf{Pr}\left[A\left(pk, Enc\left(pk, \bar{b}; r\right), b, r'\right)\right]\right| \leq \delta(\lambda)$$

"Honest" Distribution

"Fake" Distribution

where $(pk, sk) \leftarrow Gen$, $r \leftarrow \{0,1\}^{\ell'}$, and $r' \leftarrow Fake(pk, \bar{b}, r, b)$

Evaluation & Deniability Compactness

a) For every f and $m_1, \ldots, m_k$:

$$|Eval(pk, f, ct_1, \ldots, ct_k)| \leq poly$$

where $ct_i \leftarrow Enc(pk, m_i)$ and $(pk, sk) \leftarrow Gen$

Independent of k and the complexity of f

b) For every m:

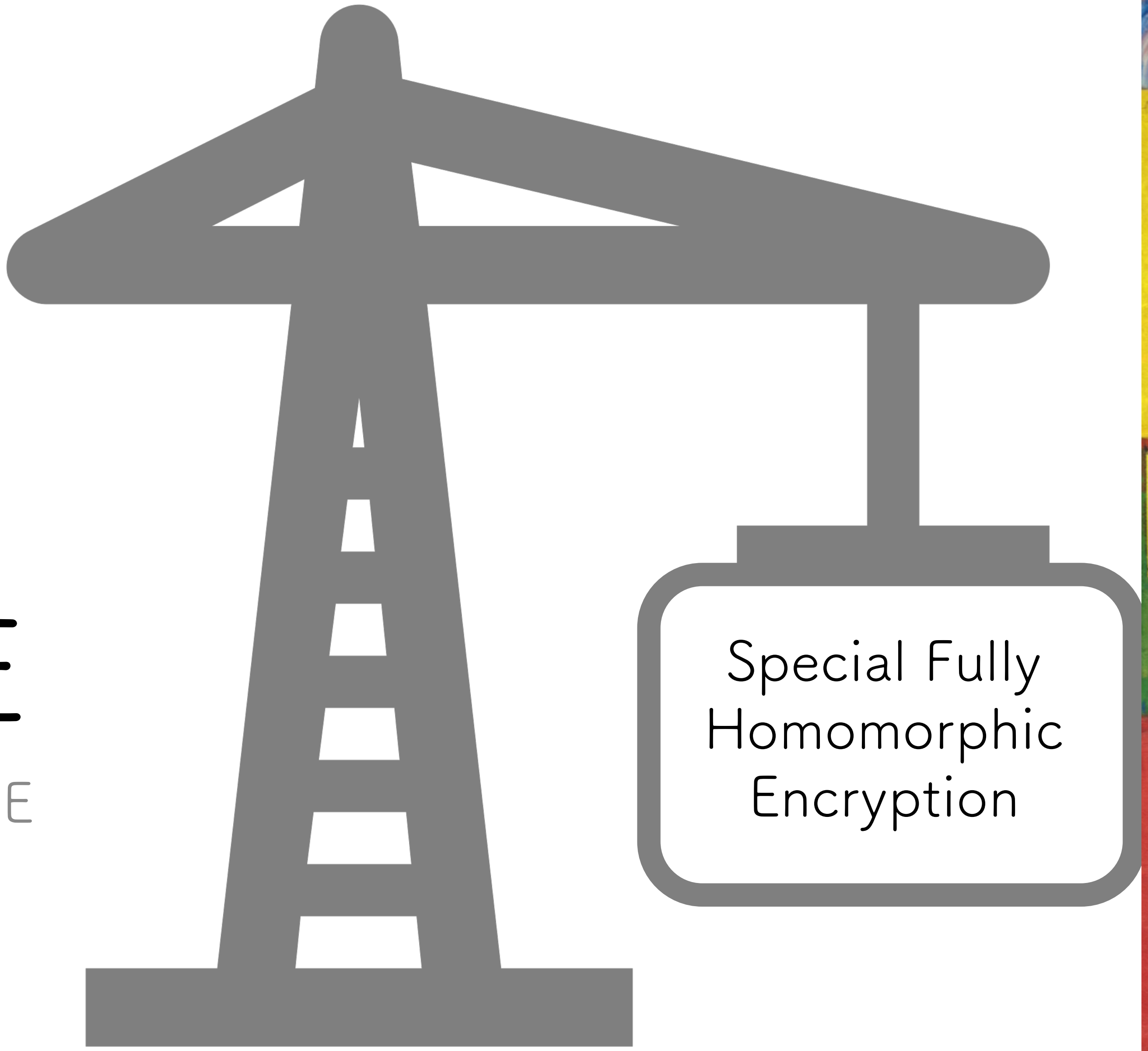
$$|Enc(pk, m)| \leq poly$$

where $(pk, sk) \leftarrow Gen$, regardless of the encryption running time

Independent of the detection probability

Deniable FHE

Our Construction of Deniable FHE



FHE from LWE: A Very Brief Recap

- All* known FHE schemes add noise in CT for security.
- Homomorphic evaluation of CTs (eval(f, $ct_1 \cdots ct_n$)) cause noise to grow
- Kills correctness after noise grows too much
- Limits number of homomorphic operations

How to keep going: Gentry's bootstrapping [Gen09]!

The Magic of Bootstrapping

- Assume that an FHE is powerful enough to support evaluation of its own decryption circuit Dec.

- By correctness of decryption, $\mathrm{Dec}(ct_x, sk) = x$

$$\mathrm{Dec}\left(\boxed{x}\ , sk \right) = x$$

- Define circuit $\mathrm{Dec}_{ct}(sk) = \mathrm{Dec}(sk, ct)$
- By correctness of homomorphic evaluation, $\mathrm{Eval}(F, ct_x) = ct(F(x))$

$$\mathrm{Eval}\left(\mathrm{Dec}_{ct}, \boxed{sk} \right) = \boxed{\mathrm{Dec}_{ct}(sk)} = \boxed{x}$$

The Magic of Bootstrapping

- Originally introduced to reduce noise in evaluated ciphertext

- Homomorphic evaluation of decryption
  - removes large old noise
  - adds small new noise (size small since decryption shallow)

This work: Oblivious Sampling of FHE ciphertexts!

The Magic of Bootstrapping

- Assume that decryption always outputs 0 or 1
  - even if input ct is not well formed

- Then, bootstrapping always outputs proper encryption of 0 or 1!

$$\text{Eval}\left(\text{Dec}_{ct}, \boxed{sk} \right) = \boxed{\text{Dec}_{ct}(sk)} = \boxed{x}$$

Even if input "ct" is a random element in ciphertext space!

The Magic of Bootstrapping

- Assume that decryption outputs 0 w.o.p for random input

- Then, bootstrapping outputs encryption of 0 w.o.p for random input

$$\text{Eval}\left(\text{Dec}_{\text{rand}}, \boxed{\text{sk}}\right) = \boxed{\text{Dec}_{\text{rand}}(\text{sk})} = \boxed{0}$$

Given enc(sk), run dec homomorphically on random to generate encryption of 0 w.o.p!

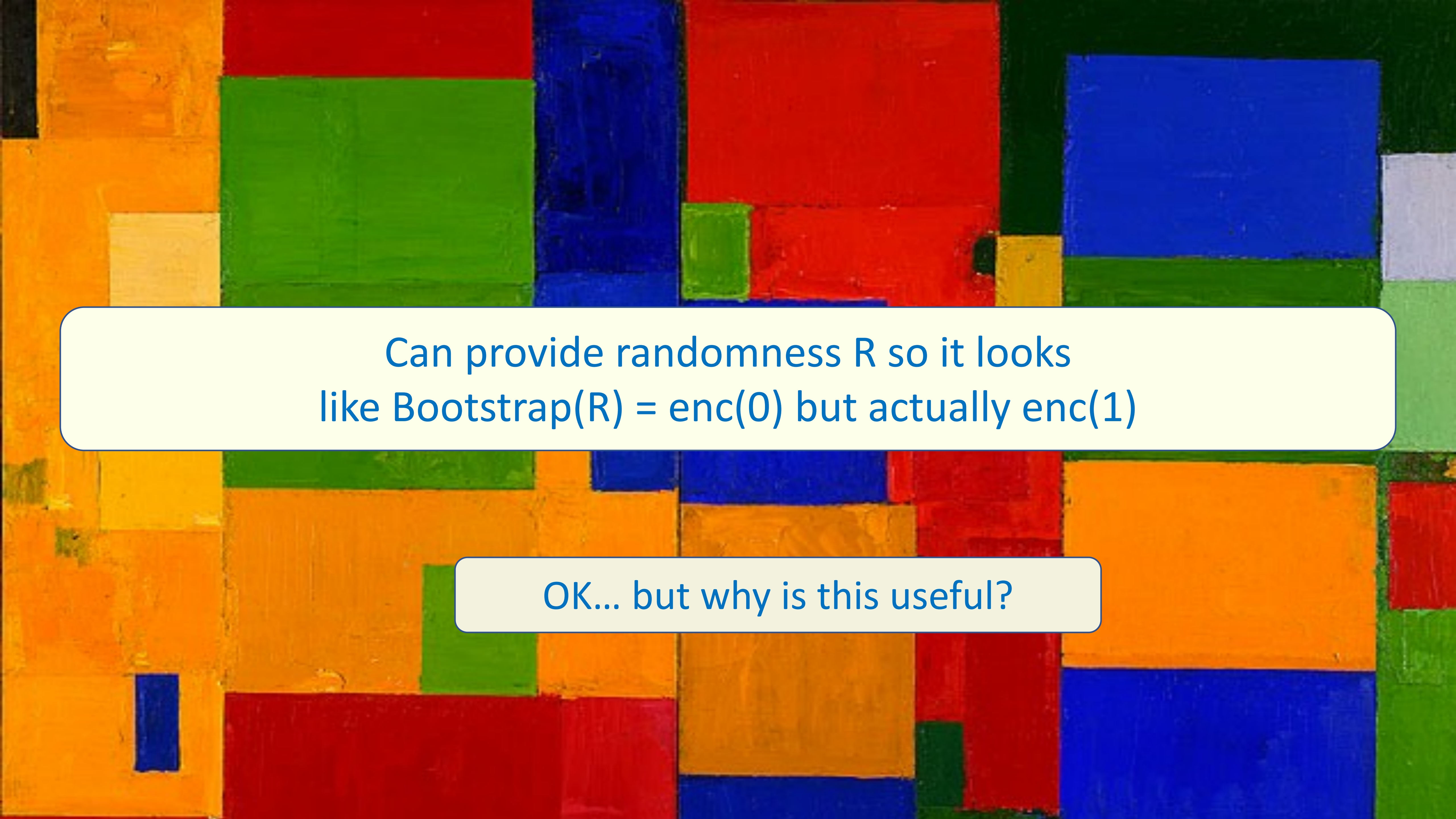
But, wait a minute…

- Given encryption of 1, decryption outputs 1 w.o.p
- Encryption of 1 is indistinguishable from random!

$$\mathrm{Eval}\left(\mathrm{Dec}_{ct1},\ \boxed{sk}\right) = \boxed{\mathrm{Dec}_{ct1}(sk)} = \boxed{1}$$

- Can pretend as if ct1 = enc(1) is a random string

Pretend bootstrapping outputs enc(0) but actually enc(1)!



Can provide randomness R so it looks like Bootstrap(R) = enc(0) but actually enc(1)

OK… but why is this useful?

Leveraging our trick (binary msg space)

- Let $B(x) = Eval(pk, Dec_x, ct_{sk})$ the bootstrapping procedure
  - recall $Dec_x(sk) = Dec(sk, x)$

- Denote homomorphic addition (mod 2) as

$$Eval(pk, +, ct_a, ct_b) = ct_a \oplus ct_b$$

$B(R_i)$
$= Enc(x_i)$

$$B(R_1) \oplus \cdots \oplus B(R_n) = \text{Enc}\,(\text{Parity}\,(x_1, \ldots, x_n))$$

Construction

Gen:

1. $(pk, sk) \leftarrow Gen$
2. $ct_{sk} \leftarrow Enc(pk, sk)$
3. Output $pk = (pk, ct_{sk}), sk = sk$

Construction

$rand = (x_1, \ldots, x_n, \{R_i\}_{x_i=0}, \{r_i\}_{x_i=1})$

$Enc(pk, b)$:

1. Sample $x_1, \ldots, x_n \leftarrow \{0,1\}$ s.t. $\sum_i x_i = b \ (mod\ 2)$
2. For $x_i = 0$, sample $R_i \leftarrow \mathcal{R}^{\ell}$
3. For $x_i = 1$, sample $r_i \leftarrow \{0,1\}^{\ell'}$ and set $R_i = Enc(pk, 1; r_i)$
4. Compute $ct = B(R_1) \oplus \cdots \oplus B(R_n)$
5. Output ct

$B(\mathcal{R}^{\ell})$ is a valid encryption of 0 w.h.p

Construction

$rand = (x_1, \ldots, x_n, \{R_i\}_{x_i=0}, \{r_i\}_{x_i=1})$

$Fake(pk, b, rand, \bar{b})$:

1. If $b = \bar{b}$, output $rand$
2. Sample $k \leftarrow [n]$ s.t. $x_k = 1$
3. Set $x'_k = 0$ and $R'_k = Enc(pk, 1; r_k)$
4. For $i \neq k$, set $R'_i = R_i$ and $r'_i = r_i$
5. Output $rand' = (x'_1, \ldots, x'_n, \{R'_i\}_{x'_i=0}, \{r'_i\}_{x'_i=1})$

Pseudorandom Ciphertext

By pretending one ciphertext enc(1) is random, parity flipped!

Statistical distance from honest dist is 1/poly(n)

Construction

$Eval(pk, f, ct_1, \ldots, ct_k)$:

1. Interpret ct_i as special FHE ciphertext ct_i
2. Output $Eval(pk, f, ct_1, \ldots, ct_k)$

$Dec(dsk, ct)$:

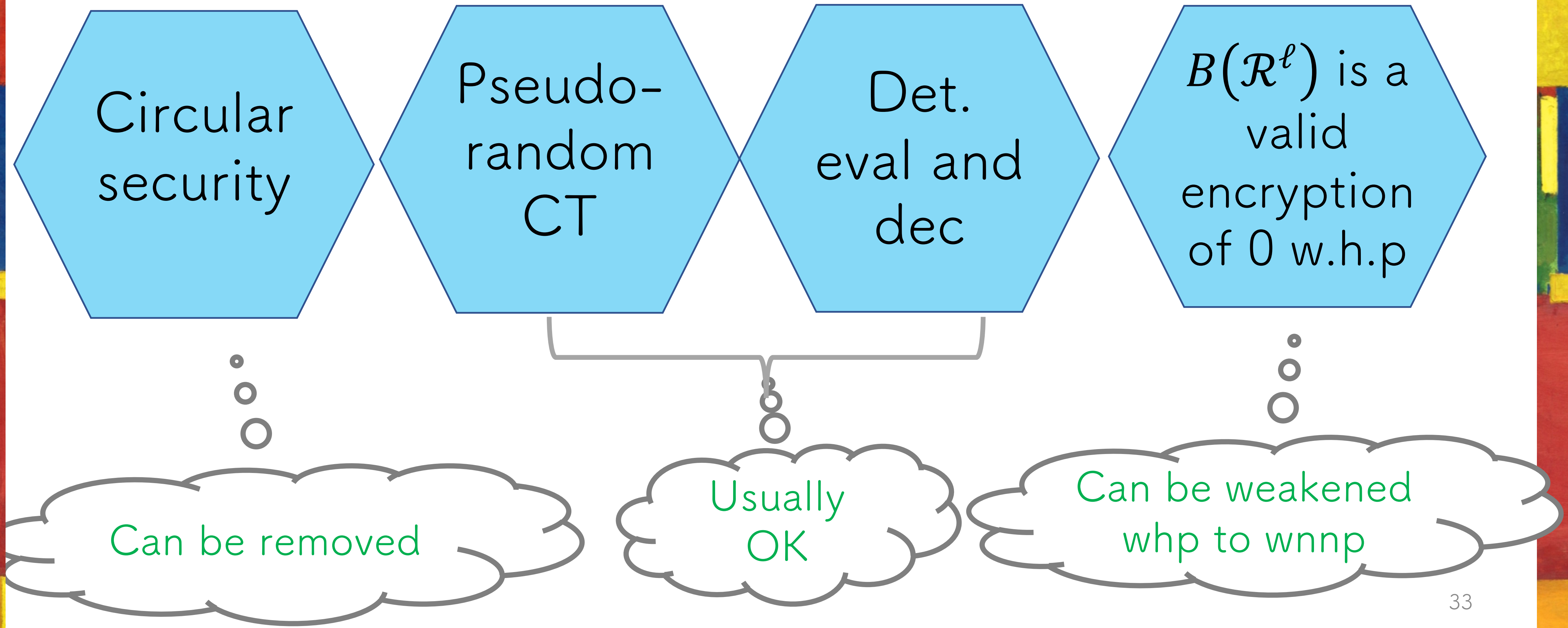
1. Interpret ct as special FHE ciphertext ct
2. Output $Dec(sk, ct)$

As before!

Special FHE

Definition and Instantiation

Special FHE


Circular security
Pseudo-random CT
Det. eval and dec
$B(\mathcal{R}^{\ell})$ is a valid encryption of 0 w.h.p
Can be removed
Usually OK
Can be weakened whp to wnnp

Weaker Special FHE

1. Pseudorandom Ciphertext
2. Deterministic evaluation and decry
3. Decryption always outputs a valid mess and

Almost always the case

$$\mathbf{Pr}[\boldsymbol{Dec}(\boldsymbol{sk}, \boldsymbol{R}) = \mathbf{0}] = \mathbf{1}/\boldsymbol{poly}$$

where $\boldsymbol{R} \leftarrow \mathcal{R}^{\ell}$ and $(\boldsymbol{pk}, \boldsymbol{sk}) \leftarrow \boldsymbol{Gen}$

[BGV14] FHE satisfies all properties!

Online-Offline Encryption

Bulk of the computation is independent of the message, and may be performed in an offline pre-processing phase.

$Enc(dpk, b)$:

1. Select $x_1, \ldots, x_n \leftarrow \{0,1\}$ s.t. $\sum_i x_i = b \ (mod\ 2)$
2. For $x_i = 0$, select $R_i \leftarrow \mathcal{R}^\ell$
3. For $x_i = 1$, select $r_i \leftarrow \{0,1\}^{\ell'}$ and set $R_i = Enc(pk, 1; r_i)$
4. Output $dct = B(R_1) \oplus \cdots \oplus B(R_n)$

n-1 computations of $B(R_i)$ can be done offline:
choose R_n depending on b and compute $B(R_n)$ online

Main Takeaway:
Evaluation compactness in FHE implies deniability compactness in DE!

Going Forward

- Compact CT ➔ compact encryption runtime?
  - Analogy to FE [LPST16,GKPVZ13]
- Technical barrier: unidirectional cheating
- Need: Invertible oblivious sampling with bias
  - SW construction may be viewed through this lens
- From LWE: can have oblivious sampling with bias (this work) or oblivious sampling with inversion but not both (so far).

Thank You

Images Credit: Hans Hoffman