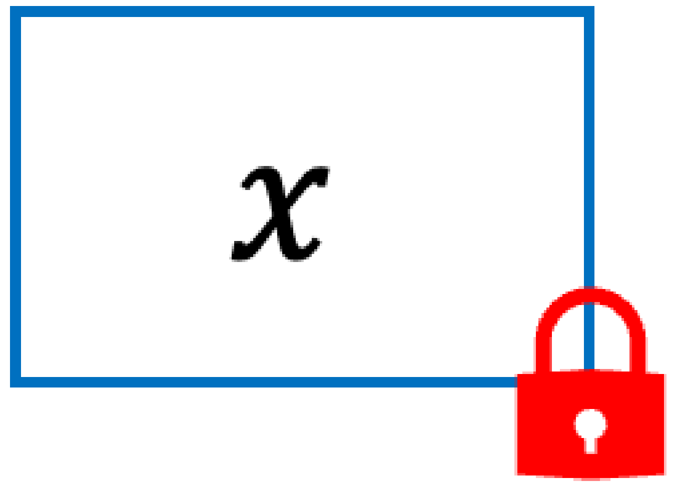
Sometimes-Decryptable Homomorphic Encryption from Sub-exponential DDH

Abhishek Jain
NTT and Johns Hopkins University

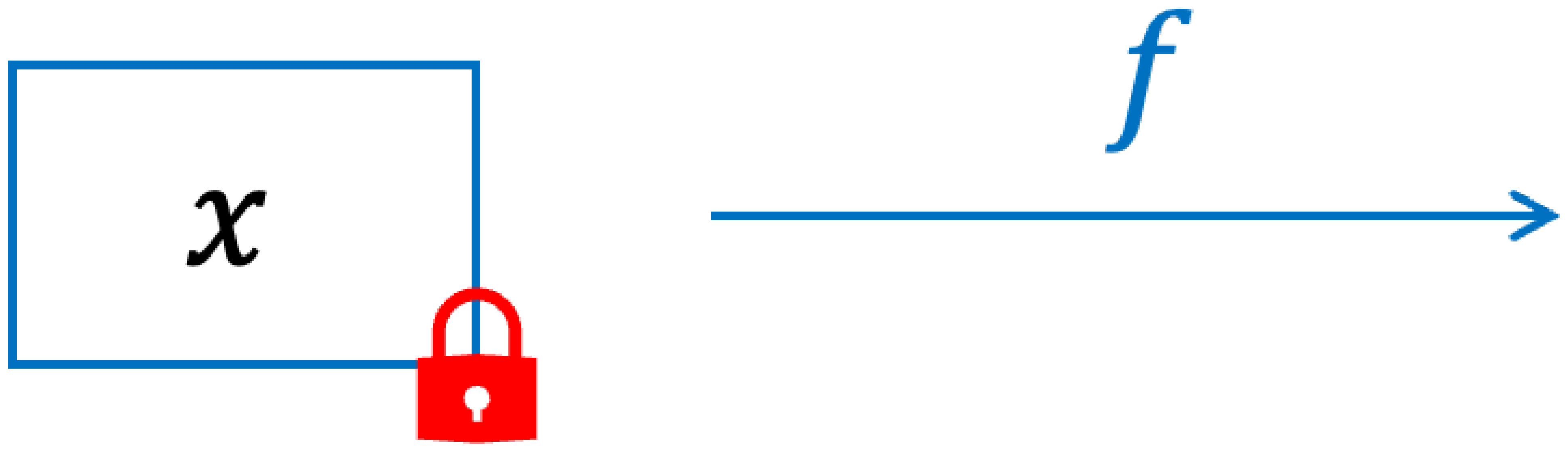
Zhengzhong Jin
Northeastern University

Homomorphic Encryption 🔒 (HE)

Homomorphic Encryption 🔒 (HE)



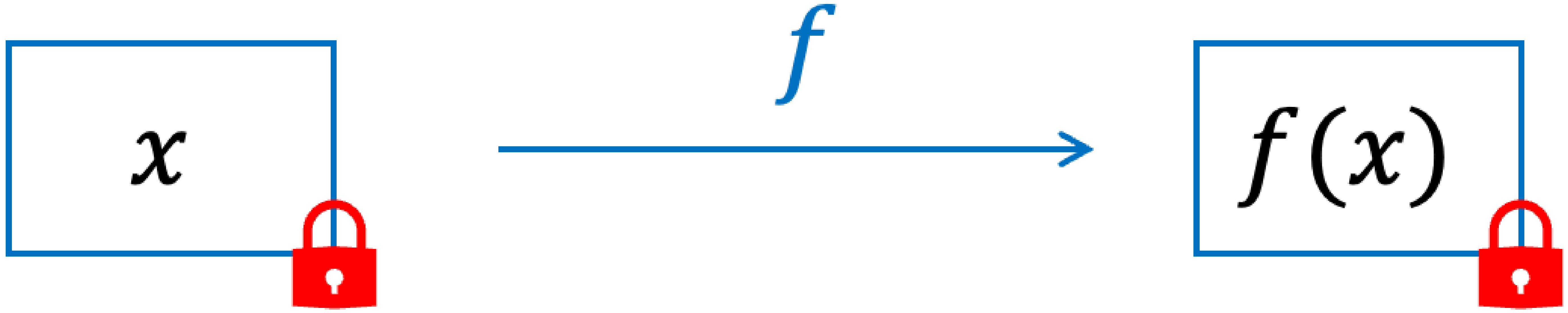
Homomorphic Encryption 🔒 (HE)



Homomorphic Encryption 🔒 (HE)

$\boxed{x}$ 🔒 $\xrightarrow{f}$ $\boxed{f(x)}$ 🔒

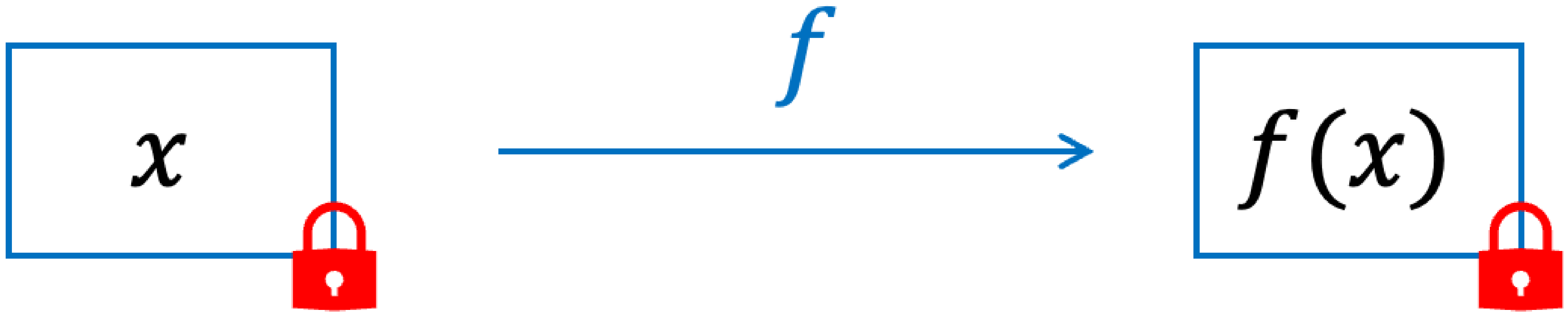
Homomorphic Encryption 🔒 (HE)



$$\boxed{x} \xrightarrow{f} \boxed{f(x)}$$

Many Applications: computing over encrypted data

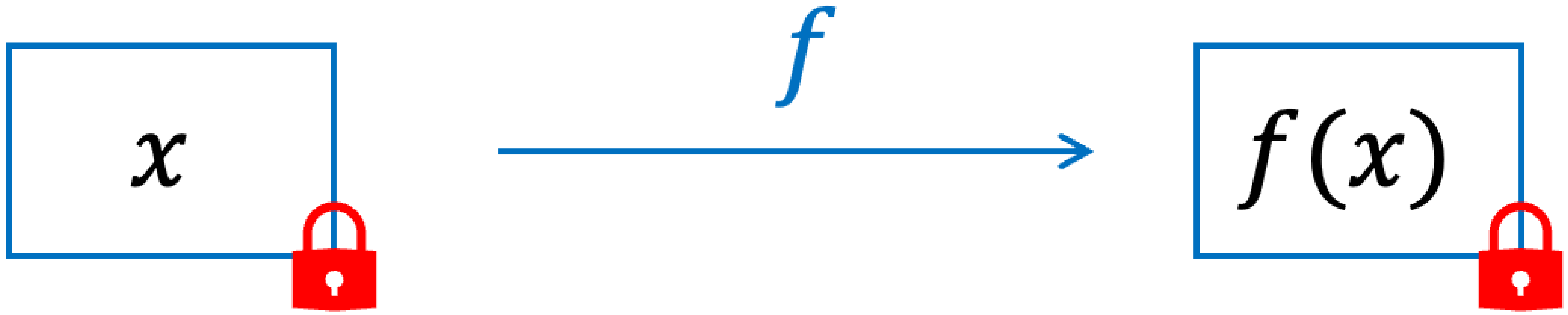
Homomorphic Encryption 🔒 (HE)



Many Applications: computing over encrypted data

Prior Work

Homomorphic Encryption 🔒 (HE)



Many Applications: computing over encrypted data

Prior Work

- Fully homomorphism:

Homomorphic Encryption 🔒 (HE)

$$\boxed{x}\ 🔒 \quad \xrightarrow{\quad f \quad} \quad \boxed{f(x)}\ 🔒$$

Many Applications: computing over encrypted data

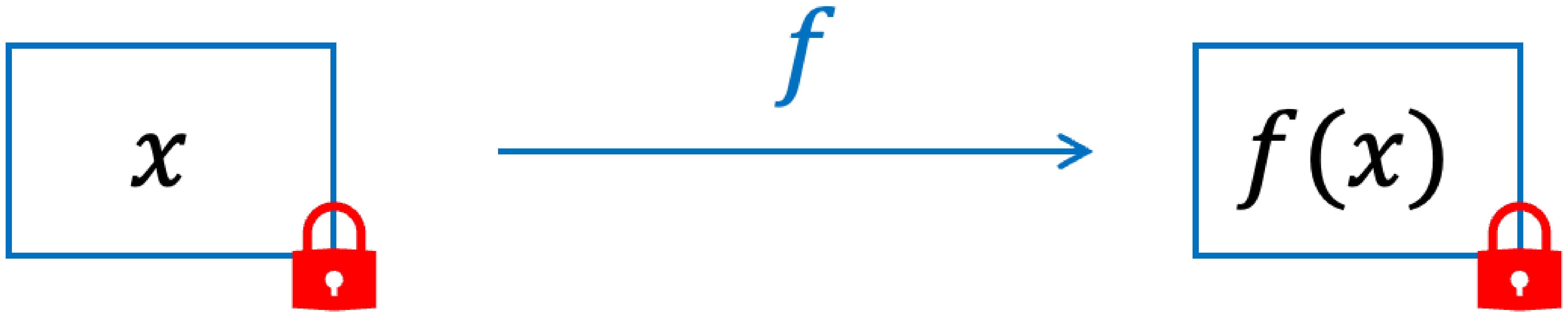
Prior Work

- Fully homomorphism:

  Lattice [Gentry’09, Dijk-Gentry-Halevi-Vaikuntanathan’10, Brakerski-Vaikuntanathan’11, Brakerski-Gentry-Vaikuntanathan’12, Gentry-Sahai-Waters’13],

  iO [Canetti-Lin-Tessaro-Vaikuntanathan’15, Jain-Lin-Sahai’21, Jain-Lin-Sahai’22, Ragavan-Vafa-Vaikuntanathan’24]

Homomorphic Encryption 🔒 (HE)



Many Applications: computing over encrypted data

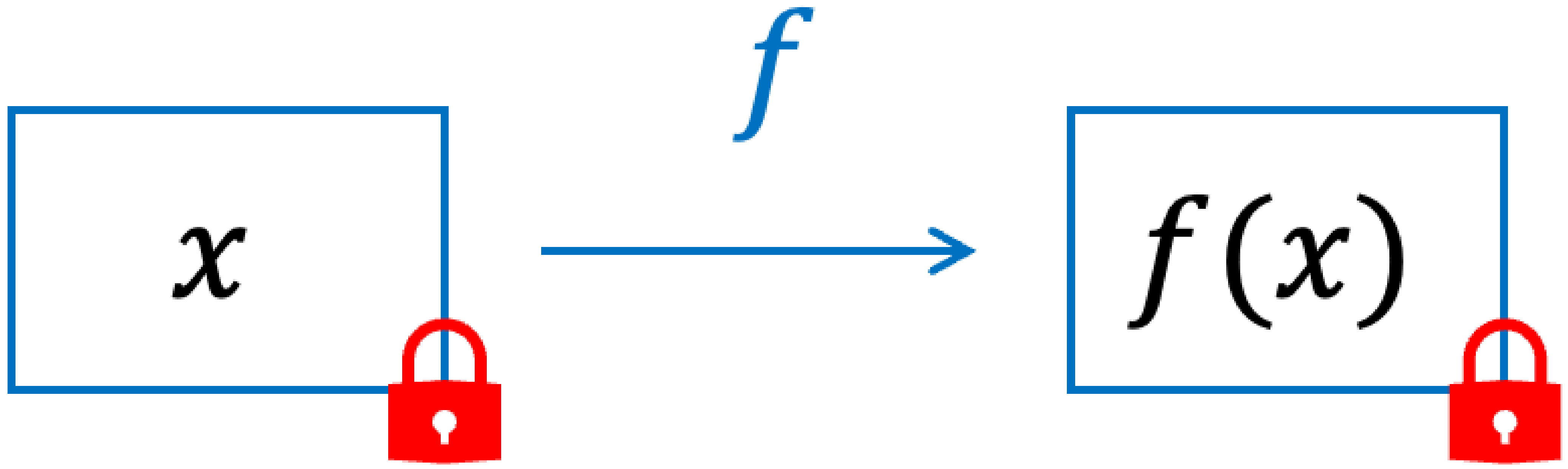
Prior Work

- Fully homomorphism:
  Lattice [Gentry'09, Dijk-Gentry-Halevi-Vaikuntanathan'10, Brakerski-Vaikuntanathan'11, Brakerski-Gentry-Vaikuntanathan'12, Gentry-Sahai-Waters'13],
  iO [Canetti-Lin-Tessaro-Vaikuntanathan'15, Jain-Lin-Sahai'21, Jain-Lin-Sahai'22, Ragavan-Vafa-Vaikuntanathan'24]
- 2-DNF: bilinear maps [Boneh-Goh-Nissim'05]

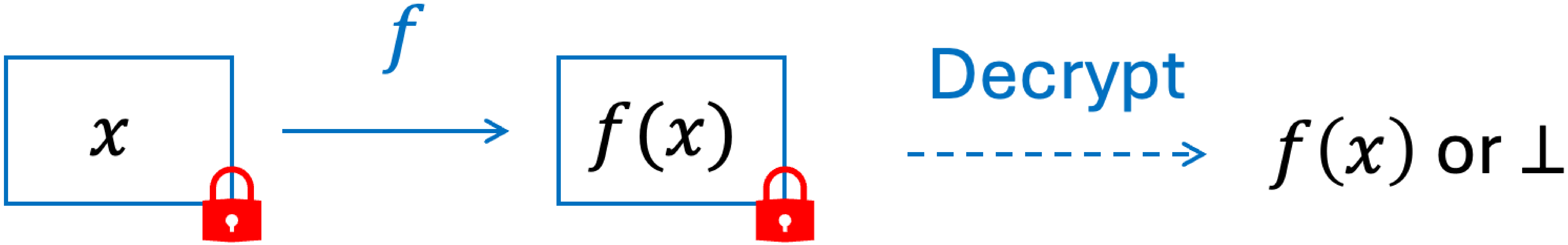
Can we build HE from group-based assumptions, for a larger class of functionality?

This Work: Sometimes-Decryptable HE

This Work: Sometimes-Decryptable HE



This Work: Sometimes-Decryptable HE



x

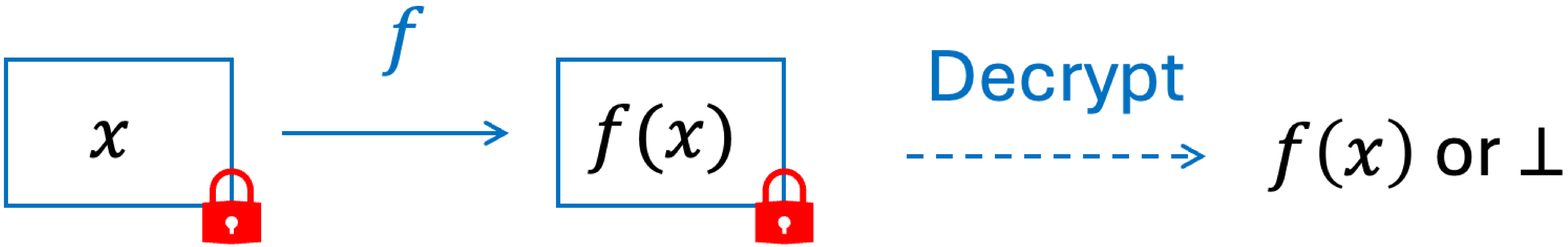
f

$f(x)$

Decrypt

$f(x)$ or $\perp$

This Work: Sometimes-Decryptable HE

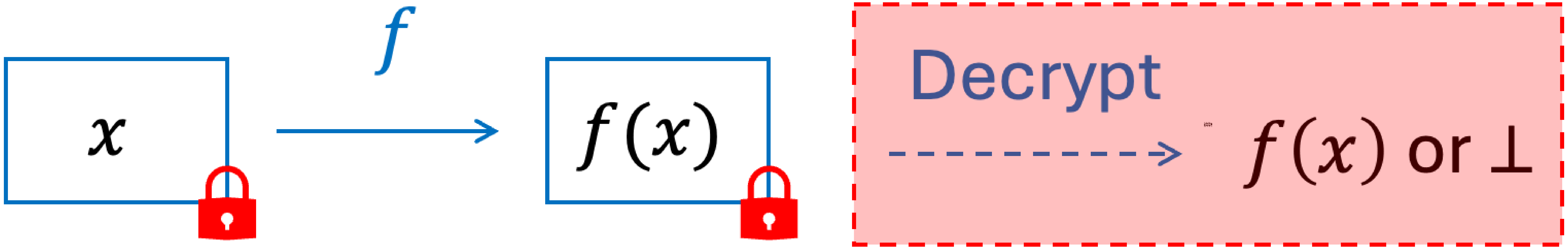


$x \xrightarrow{f} f(x) \overset{\text{Decrypt}}{\dashrightarrow} f(x)$ or $\perp$

Sometimes-Decryptable: $\Pr[\text{Decrypt correct}] > 2^{-\lambda^c}$, where $c \in (0,1)$

Formal Definition: later

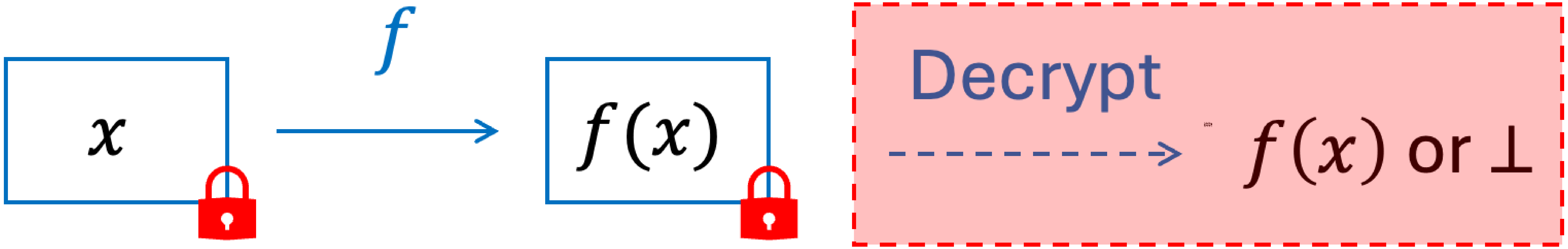
This Work: Sometimes-Decryptable HE



Sometimes-Decryptable: $\Pr[\text{Decrypt correct}] > 2^{-\lambda^{c}}$, where $c \in (0,1)$

Formal Definition: later

This Work: Sometimes-Decryptable HE

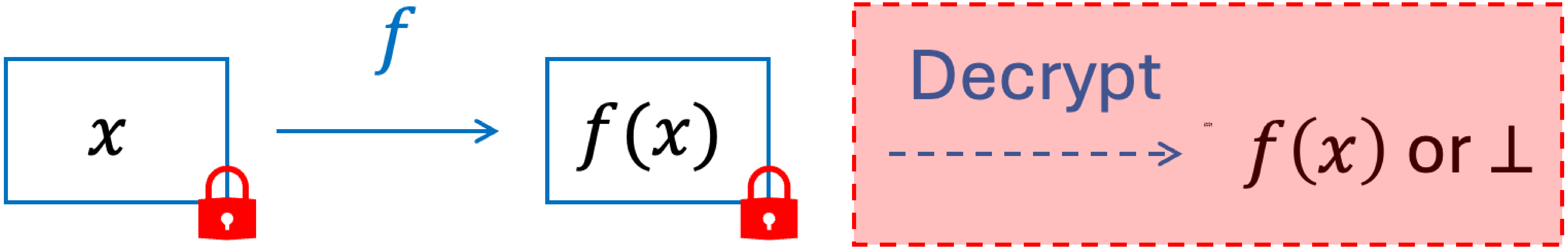


Sometimes-Decryptable: $\Pr[\text{Decrypt correct}] > 2^{-\lambda^c}$, where $c \in (0,1)$

Formal Definition: later

Useful When: **Decryption is only needed in security proof**
e.g. proof systems

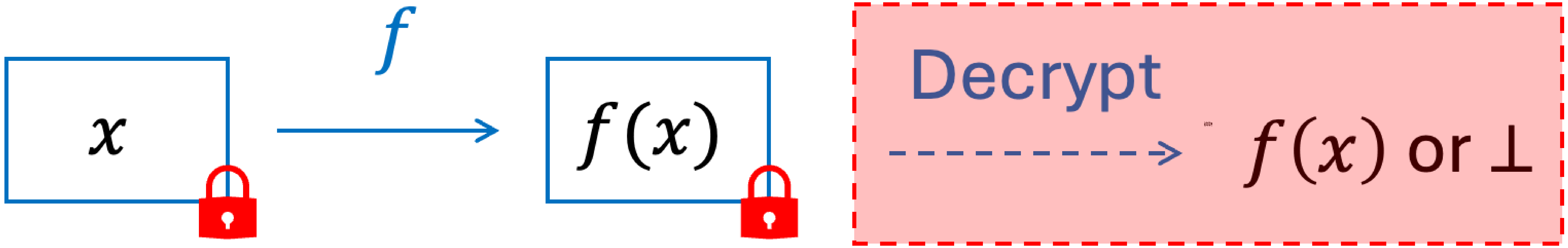
This Work: Sometimes-Decryptable HE



Sometimes-Decryptable: $\Pr[\text{Decrypt correct}] > 2^{-\lambda^c}$, where $c \in (0,1)$

Formal Definition: later

This Work: Sometimes-Decryptable HE



x $\xrightarrow{f}$ $f(x)$

Decrypt $\dashrightarrow$ $f(x)$ or $\perp$

Sometimes-Decryptable: $\Pr[\text{Decrypt correct}] > 2^{-\lambda^c}$, where $c \in (0,1)$

Formal Definition: later

Decryption/Extraction only in the Security Proof:

- Sometimes extractable commitment → *statistical Zaps* [Kalai-Khurana-Sahai'18]
- Somewhere extractable commitment [Hubacek-Wichs'15], predicate-extractable commitment [Brakerski-Brodsky-Kalai-Lombardi-Paneth'23] → *SNARGs*
- Correlation intractable hash [Canetti-Chen-Holmgren-Lombardi-Rothblum-Rothblum-Wichs, Peikert-Shiehian'19] → *NIZKs/SNARGs*

Our Result

Assuming sub-exponential hardness of Decisional Diffie-Hellman (DDH), there exists a sometimes-decryptable homomorphic encryption for TC^0.

Our Result

Assuming sub-exponential hardness of Decisional Diffie-Hellman (DDH), there exists a sometimes-decryptable homomorphic encryption for TC^0.

(TC^0: constant-depth threshold circuits)

Application: Succinct Non-interactive ARGuments (SNARGs)

CRS: Common Reference String

P

V

Application: Succinct Non-interactive ARGuments (SNARGs)

CRS: Common Reference String

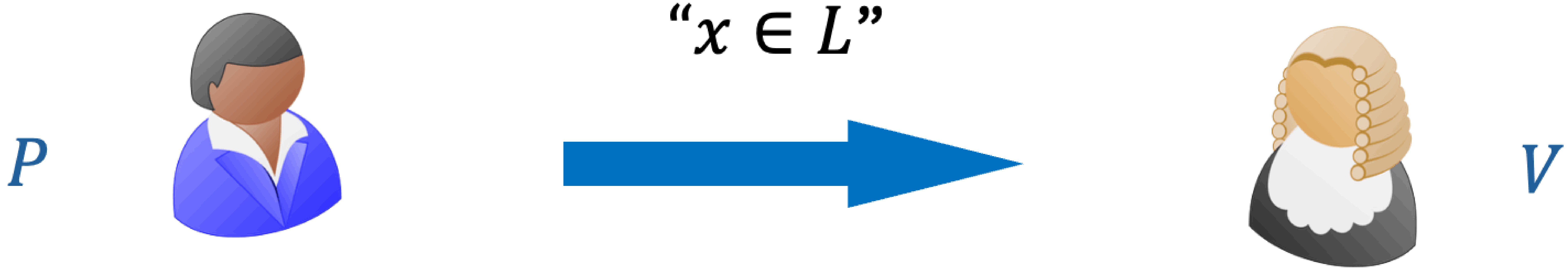
“$x \in L$”

P

V

Application: Succinct Non-interactive ARGuments (SNARGs)

CRS: Common Reference String



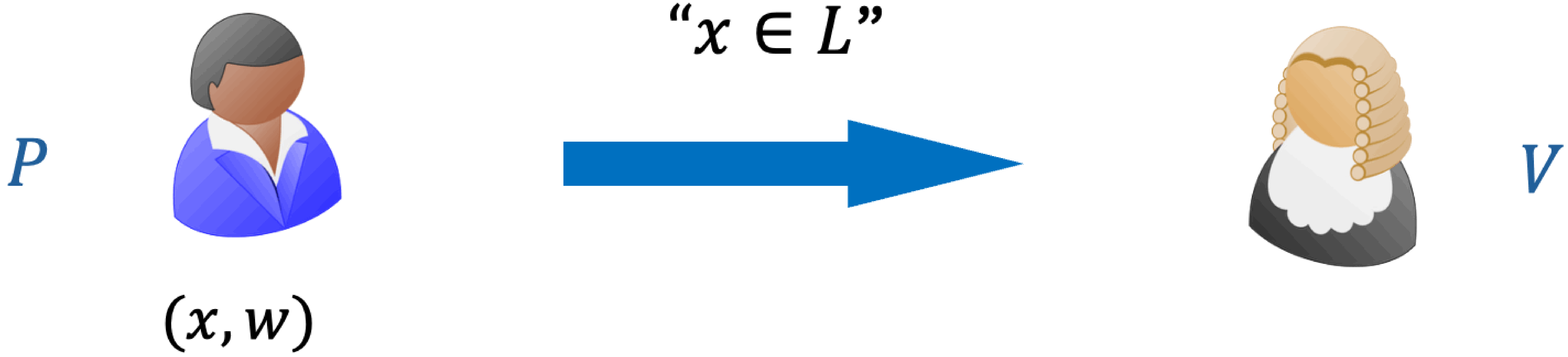
P

“$x \in L$”

V

Application: Succinct Non-interactive ARGuments (SNARGs)

CRS: Common Reference String



P

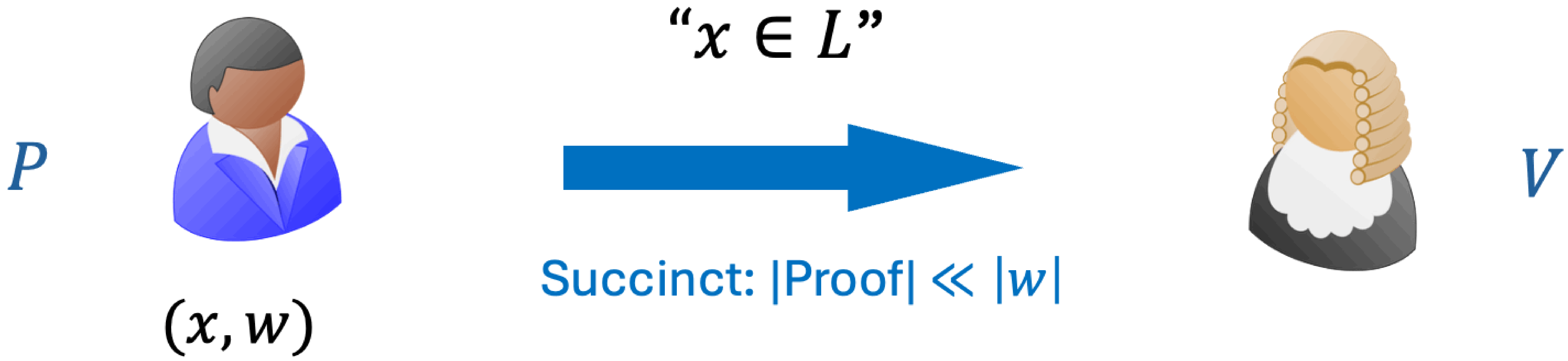
(x, w)

“$x \in L$”

V

Application: Succinct Non-interactive ARGuments (SNARGs)

CRS: Common Reference String



P

"$x \in L$"

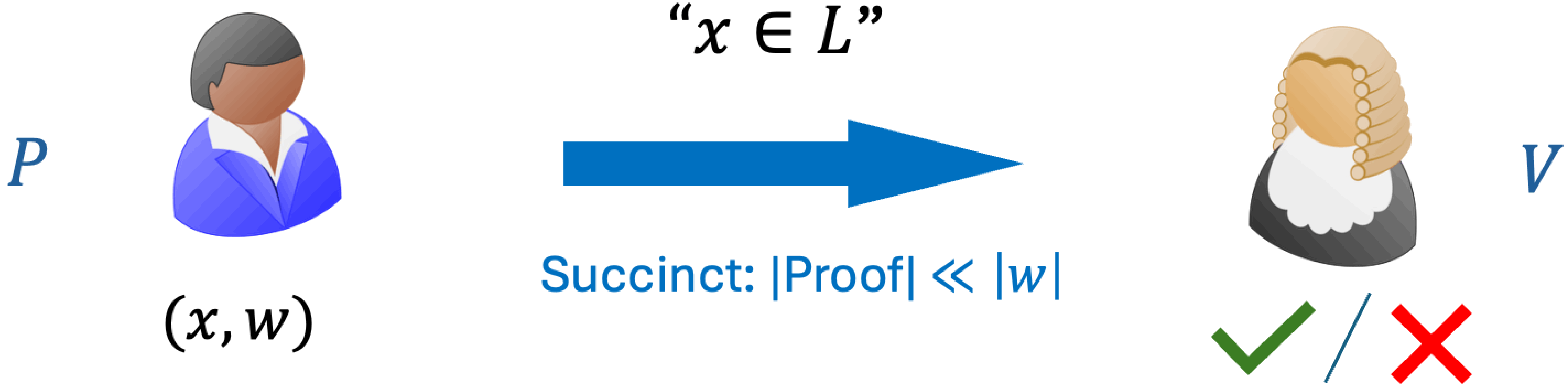
V

(x, w)

Succinct: |Proof| ≪ $|w|$

Application: Succinct Non-interactive ARGuments (SNARGs)

CRS: Common Reference String



P

(x, w)

“$x \in L$”

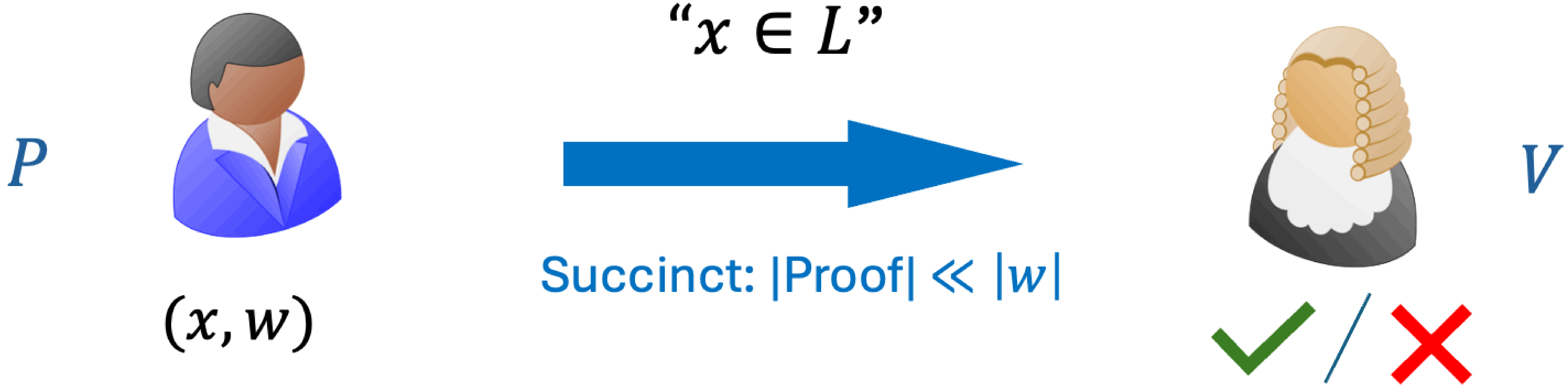
Succinct: |Proof| $\ll$ $|w|$

V

✓ / ✗

Application: Succinct Non-interactive ARGuments (SNARGs)

CRS: Common Reference String



“$x \in L$”

P

(x, w)

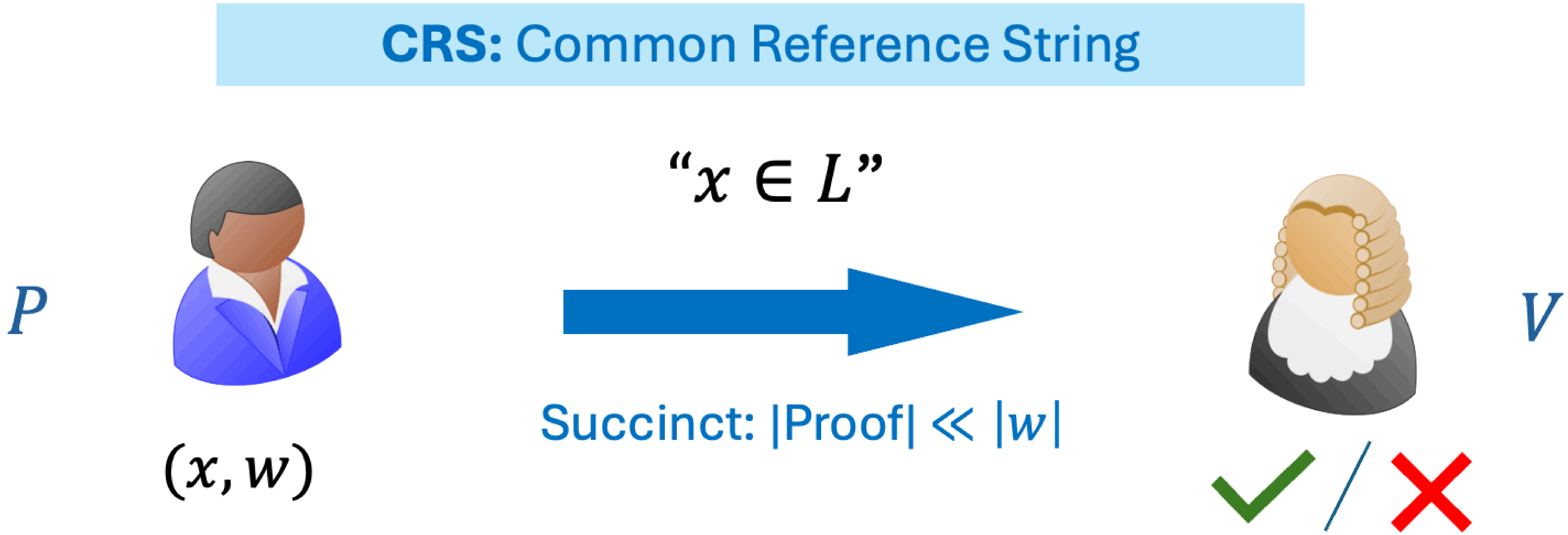
Succinct: |Proof| ≪ $|w|$

V

✓ / ✗

- **Completeness:** $\forall x \in L$, the honestly generated proof is accepted.

Application: Succinct Non-interactive ARGuments (SNARGs)



CRS: Common Reference String

“$x \in L$”

P

(x, w)

Succinct: |Proof| $\ll$ $|w|$

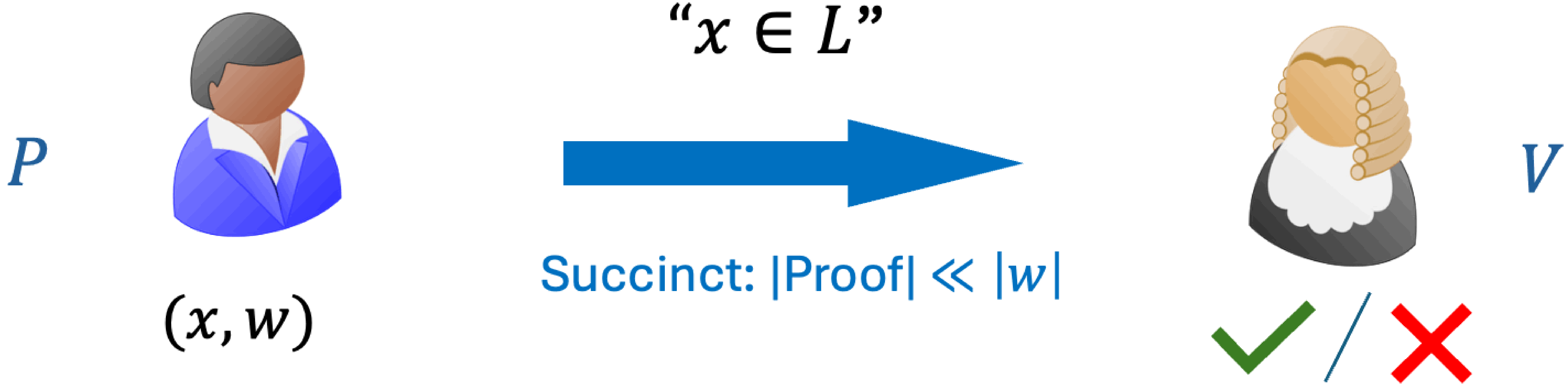
V

✓ / ✗

- **Completeness:** $\forall x \in L$, the honestly generated proof is accepted.
- **Soundness:** for any $x \notin L$, and any PPT. adversary, the cheating proof should be rejected.

Application: Succinct Non-interactive ARGuments (SNARGs)

CRS: Common Reference String



P

(x, w)

"$x \in L$"

Succinct: |Proof| $\ll |w|$

V

✓ / ✗

- **Completeness:** $\forall x \in L$, the honestly generated proof is accepted.
- **Soundness:** for any $x \notin L$, and any PPT. adversary, the cheating proof should be rejected.

Many applications: delegation of computation, blockchain and cryptocurrency, etc.

Application (1): SNARGs

Assuming sub-exponential hardness of DDH, there exists a SNARG for any NP language that has a poly-size TC^0 Frege Logic proof of non-membership.

Application (1): SNARGs

Assuming sub-exponential hardness of DDH, there exists a SNARG for any NP language that has a poly-size TC^0 Frege Logic proof of non-membership.

(a subclass of NP ∩ coNP)

Application (1): SNARGs

Assuming sub-exponential hardness of DDH, there exists a SNARG for any NP language that has a poly-size TC^0 Frege Logic proof of non-membership.

(a subclass of NP ∩ coNP)

Prior work on SNARGs via Logic Proofs of Non-membership:
[Jain-J'22] from iO, [J-Kalai-Lombardi-Vaikuntanathan'24] from LWE

Application (1): SNARGs

Assuming sub-exponential hardness of DDH, there exists a SNARG for any NP language that has a poly-size TC^0 Frege Logic proof of non-membership.

(a subclass of NP ∩ coNP)

Prior work on SNARGs via Logic Proofs of Non-membership:
[Jain-J'22] from iO, [J-Kalai-Lombardi-Vaikuntanathan'24] from LWE

Example: DDH Language

$$\{(g, h, g^s, h^s) | s \in \mathbb{Z}, g, h \in \mathbb{G}\}$$

Implication: Monotone-Policy Batch Arguments

Implication: Monotone-Policy Batch Arguments

CRS

P

V

$x_1 \ldots x_k, w_1 \ldots w_k$

Implication: Monotone-Policy Batch Arguments

CRS

“$f\left(1_{x_1 \in L}, \dots, 1_{x_k \in L}\right) = 1$”, f: a monotone circuit

P

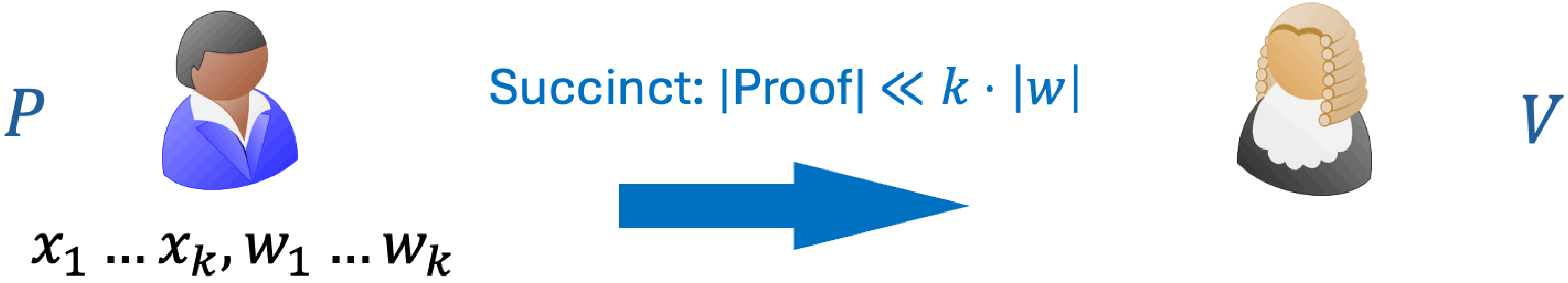
V

$x_1 \dots x_k, w_1 \dots w_k$

Implication: Monotone-Policy Batch Arguments

CRS

"$f\left(1_{x_1 \in L}, \ldots, 1_{x_k \in L}\right) = 1$", f: a monotone circuit



P

Succinct: |Proof| $\ll k \cdot |w|$

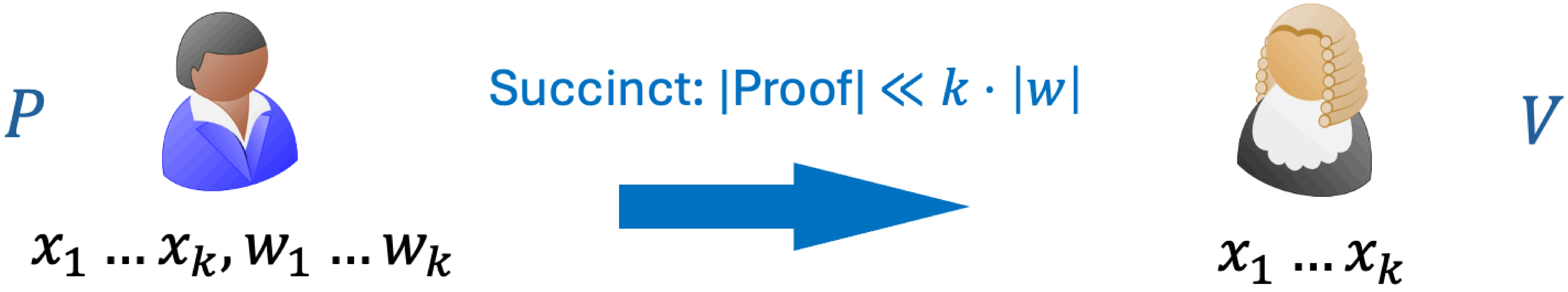
V

$x_1 \ldots x_k, w_1 \ldots w_k$

Implication: Monotone-Policy Batch Arguments

CRS

"$f(1_{x_1 \in L}, \ldots, 1_{x_k \in L}) = 1$", f: a monotone circuit



P

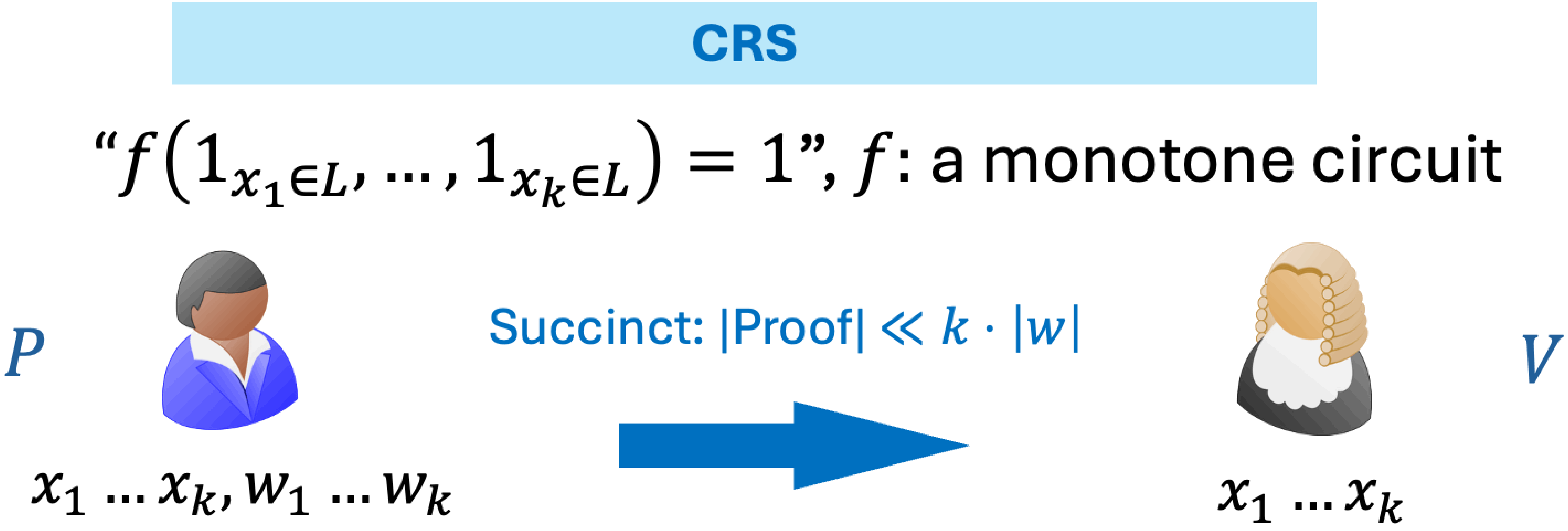
$x_1 \ldots x_k, w_1 \ldots w_k$

Succinct: |Proof| $\ll k \cdot |w|$

V

$x_1 \ldots x_k$

Implication: Monotone-Policy Batch Arguments



CRS

“$f\left(1_{x_1 \in L}, \ldots, 1_{x_k \in L}\right) = 1$”, f: a monotone circuit

P

$x_1 \ldots x_k, w_1 \ldots w_k$

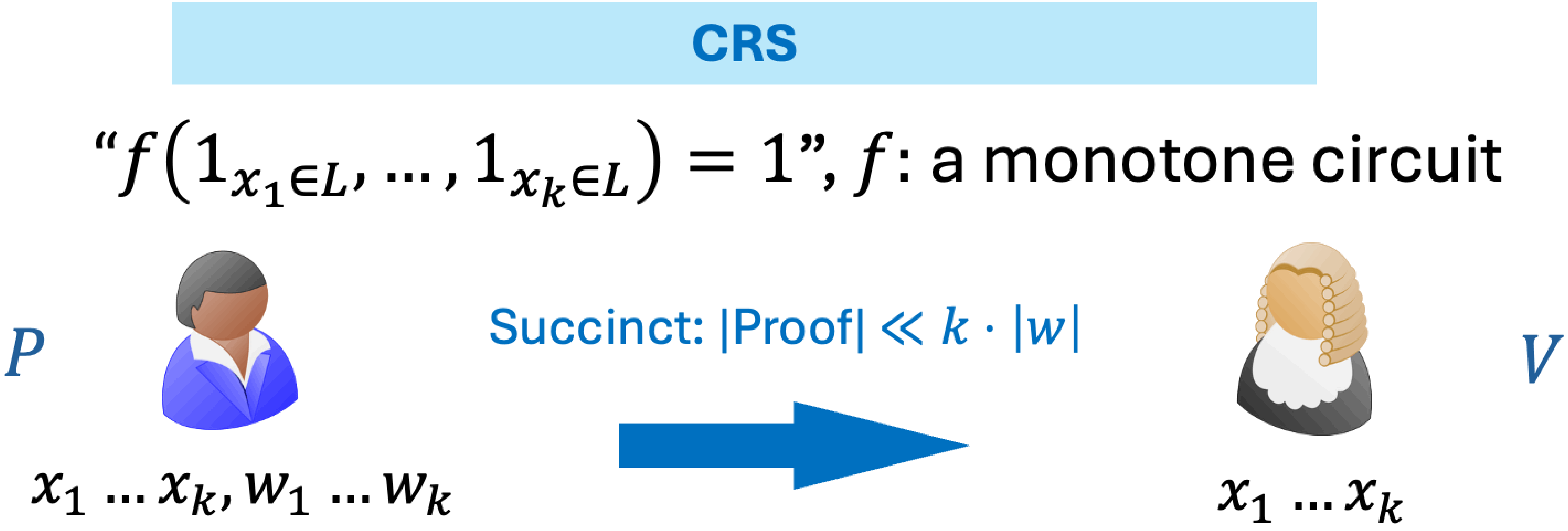
Succinct: |Proof| $\ll k \cdot |w|$

V

$x_1 \ldots x_k$

- Prior work: [Brakerski-Brodsky-Kalai-Paneth’23] Monotone Policy BARGs from LWE

Implication: Monotone-Policy Batch Arguments



CRS

"$f\left(1_{x_1 \in L}, \ldots, 1_{x_k \in L}\right) = 1$", f: a monotone circuit

P

$x_1 \ldots x_k, w_1 \ldots w_k$

Succinct: |Proof| $\ll k \cdot |w|$

V

$x_1 \ldots x_k$

- Prior work: [Brakerski-Brodsky-Kalai-Paneth'23] Monotone Policy BARGs from LWE
- Concurrent: [Nassar-Waters-Wu'24] from sub-exp DDH (different approach), or poly-hard k-Lin in pairing groups

Application (2): Monotone-Policy BARGs

Assuming sub-exponential hardness of DDH, there exists a monotone-policy BARGs for all polynomial-size monotone circuits.

More in the paper: Predicate-Extractable hash and
Correlation-Intractable hash from sub-exp DDH.

Rest of the Talk

Rest of the Talk

Formal Definition of Sometimes-Decryptable HE

Rest of the Talk

Formal Definition of Sometimes-Decryptable HE

Construction of Sometimes-Decryptable HE

Rest of the Talk

Formal Definition of Sometimes-Decryptable HE

Construction of Sometimes-Decryptable HE

Defining Sometimes-Decryptable HE (s-HE)

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

$$\mathrm{Gen}\left(1^{\lambda}\right) \rightarrow (pk, sk, \mathrm{Pred})$$

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

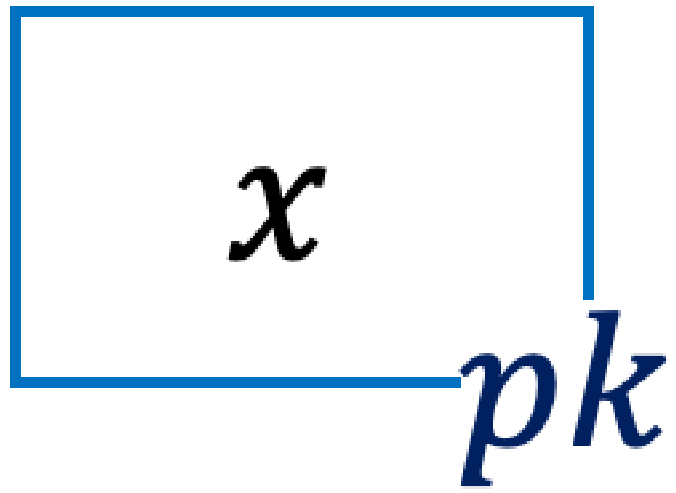
$$\mathrm{Gen}(1^\lambda) \rightarrow (pk, sk, \mathrm{Pred})$$

Only **privately** computable

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

$\mathrm{Gen}(1^\lambda) \to (pk, sk, \mathrm{Pred})$ Only **privately** computable



Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

$$\mathrm{Gen}(1^{\lambda}) \rightarrow (pk, sk, \mathrm{Pred})$$

Only **privately** computable

$$\boxed{x}_{pk} \xrightarrow{f} \mathrm{CT}$$

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

$\mathrm{Gen}(1^{\lambda}) \to (pk, sk, \mathrm{Pred})$ Only **privately** computable

$$\boxed{x}_{pk} \xrightarrow{f} \mathrm{CT}$$

If $\mathrm{Pred}(\mathrm{CT}) = 1$, then $\mathrm{Dec}(\mathrm{CT}) = f(x)$.

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

$$\mathrm{Gen}(1^{\lambda}) \rightarrow (pk, sk, \mathrm{Pred})$$ Only **privately** computable

$$\boxed{x}_{pk} \xrightarrow{f} \mathrm{CT}$$

If $\mathrm{Pred}(\mathrm{CT}) = 1$, then $\mathrm{Dec}(\mathrm{CT}) = f(x)$.

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

$\mathrm{Gen}(1^\lambda) \rightarrow (pk, sk, \mathrm{Pred})$ Only **privately** computable

$$\boxed{x}_{pk} \xrightarrow{f} \mathrm{CT}$$

If $\mathrm{Pred}(\mathrm{CT}) = 1$, then $\mathrm{Dec}(\mathrm{CT}) = f(x)$.

Sometimes Decryptable (attempt)

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

$\mathrm{Gen}(1^\lambda) \to (pk, sk, \mathrm{Pred})$ Only **privately** computable

$$\boxed{x}_{pk} \xrightarrow{f} \mathrm{CT}$$

If $\mathrm{Pred}(\mathrm{CT}) = 1$, then $\mathrm{Dec}(\mathrm{CT}) = f(x)$.

Sometimes Decryptable (attempt)

(for malicious CT)

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

$$\mathrm{Gen}(1^\lambda) \to (pk, sk, \mathrm{Pred})$$

Only **privately** computable

$$\boxed{x}_{pk} \xrightarrow{f} \mathrm{CT}$$

If $\mathrm{Pred}(\mathrm{CT}) = 1$, then $\mathrm{Dec}(\mathrm{CT}) = f(x)$.

<u>Sometimes Decryptable (attempt)</u>

(for malicious CT)

PPT.

Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

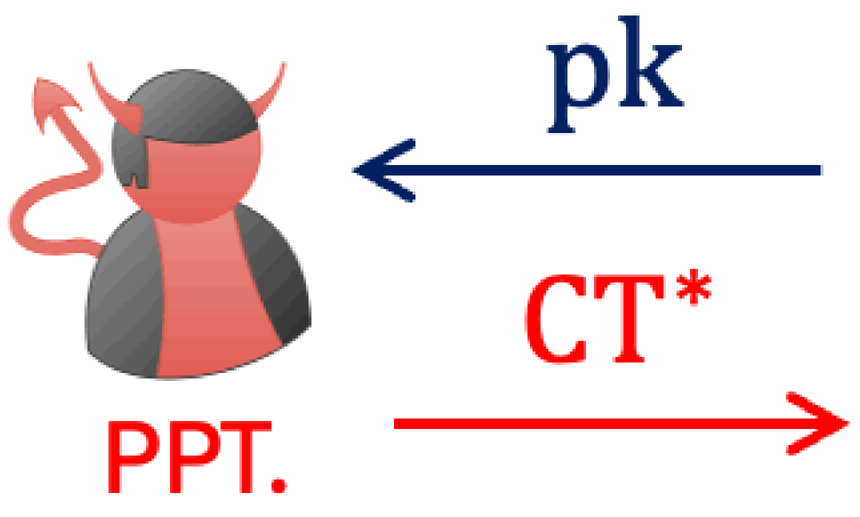
$\mathrm{Gen}(1^\lambda) \to (pk, sk, \mathrm{Pred})$ Only **privately** computable

$\boxed{x}_{pk} \xrightarrow{f} \mathrm{CT}$

If $\mathrm{Pred}(\mathrm{CT}) = 1$, then
$\mathrm{Dec}(\mathrm{CT}) = f(x)$.

Sometimes Decryptable (attempt)

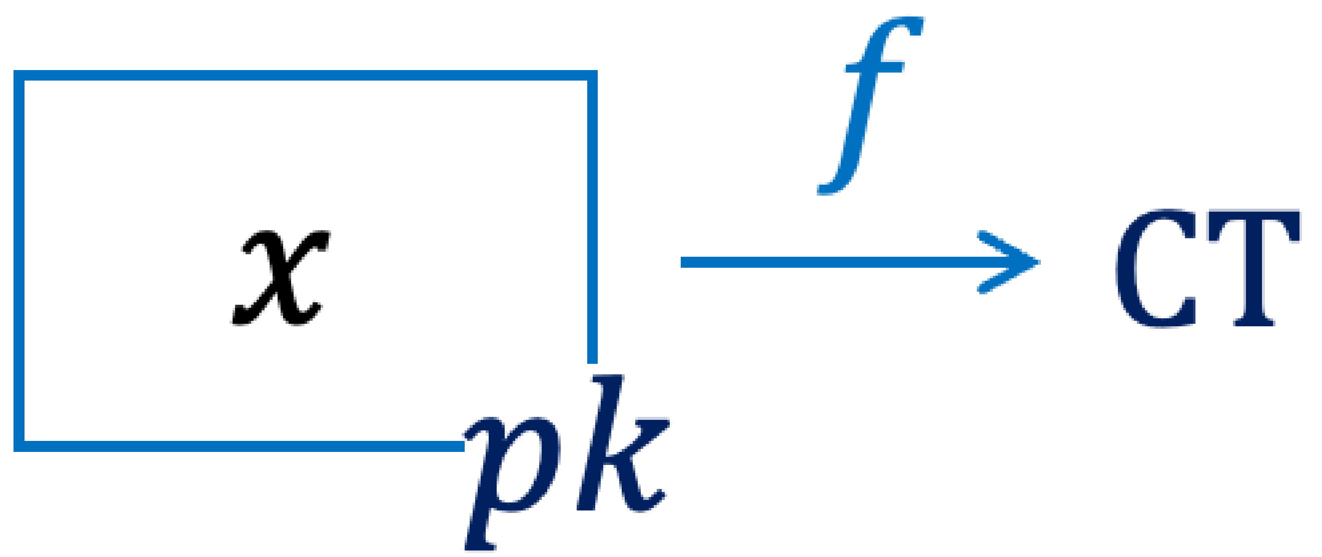
(for malicious CT)



Defining Sometimes-Decryptable HE (s-HE)

We will define the properties of s-HE step by step.

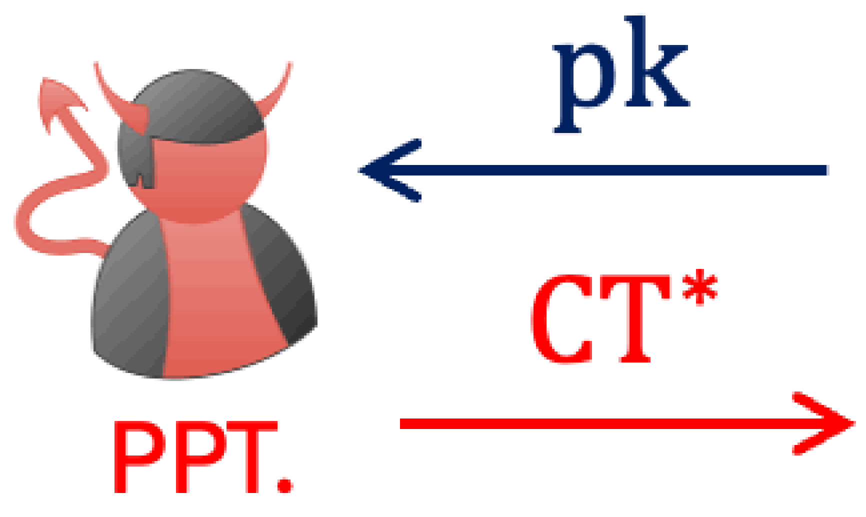
$\mathrm{Gen}(1^\lambda) \to (pk, sk, \mathrm{Pred})$ Only **privately** computable



If $\mathrm{Pred}(\mathrm{CT}) = 1$, then
$\mathrm{Dec}(\mathrm{CT}) = f(x)$.

<u>Sometimes Decryptable (attempt)</u>

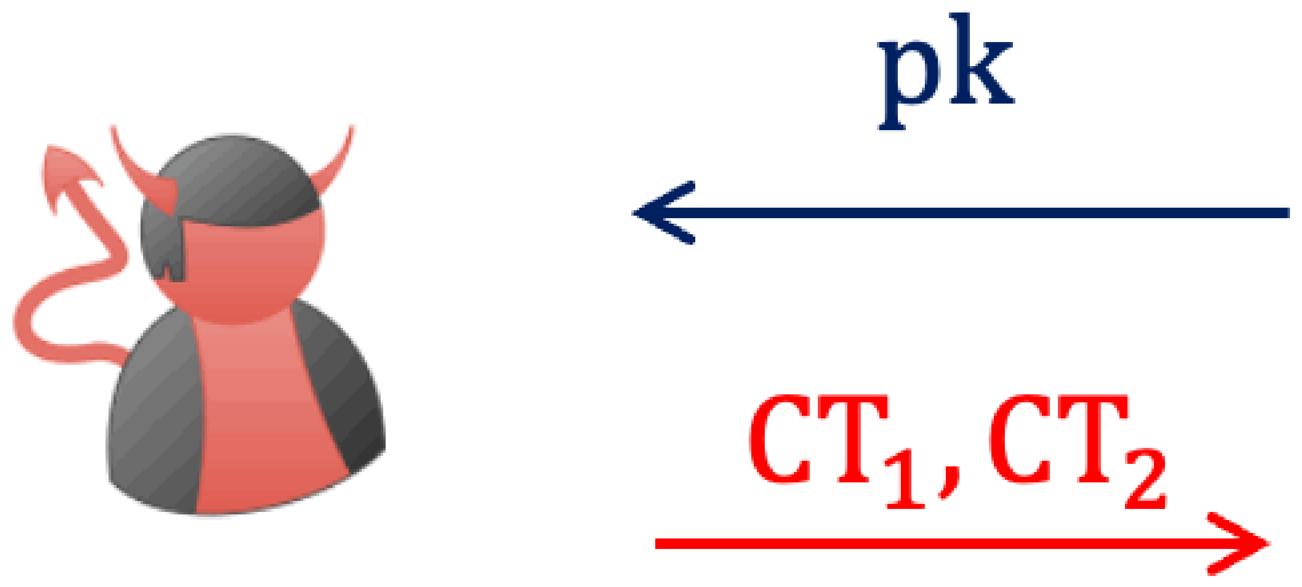
(for malicious CT)



$\Pr[\mathrm{Pred}(\mathrm{CT}^*) = 1] > 2^{-\lambda^c}$

Issue: Probability Can’t Compose

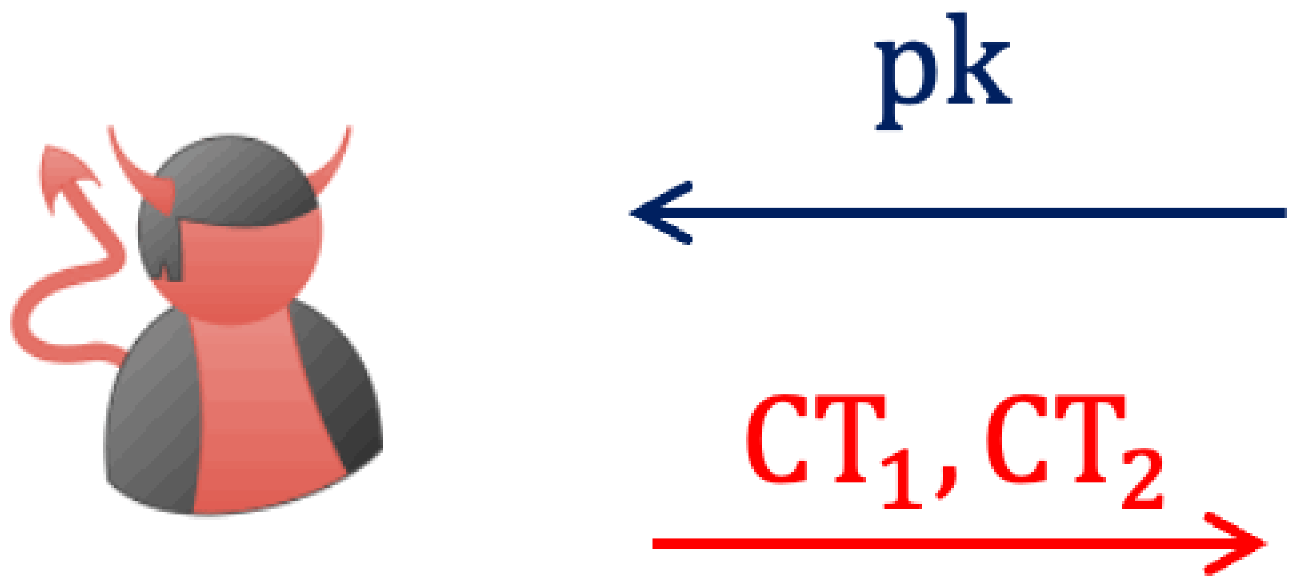
Issue: Probability Can't Compose



pk

$\mathrm{CT}_1, \mathrm{CT}_2$

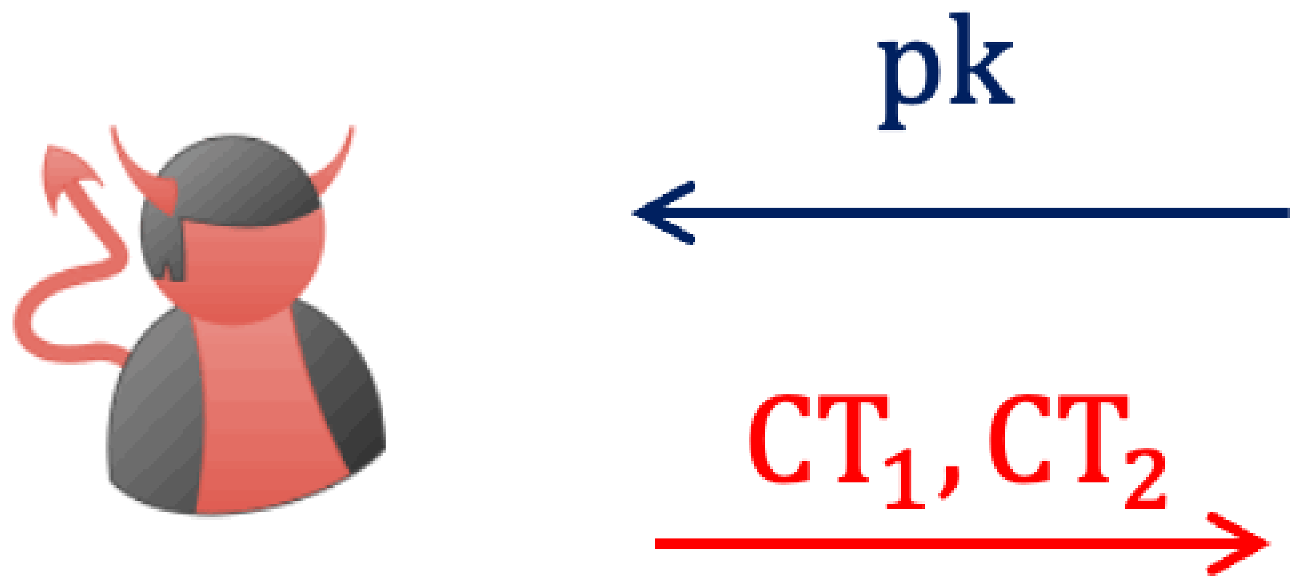
Issue: Probability Can’t Compose



$$\Pr[\mathrm{Pred}(\mathrm{CT}_1) = 1] > 2^{-\lambda^c} := \mu$$

$$\Pr[\mathrm{Pred}(\mathrm{CT}_2) = 1] > \mu$$

Issue: Probability Can’t Compose



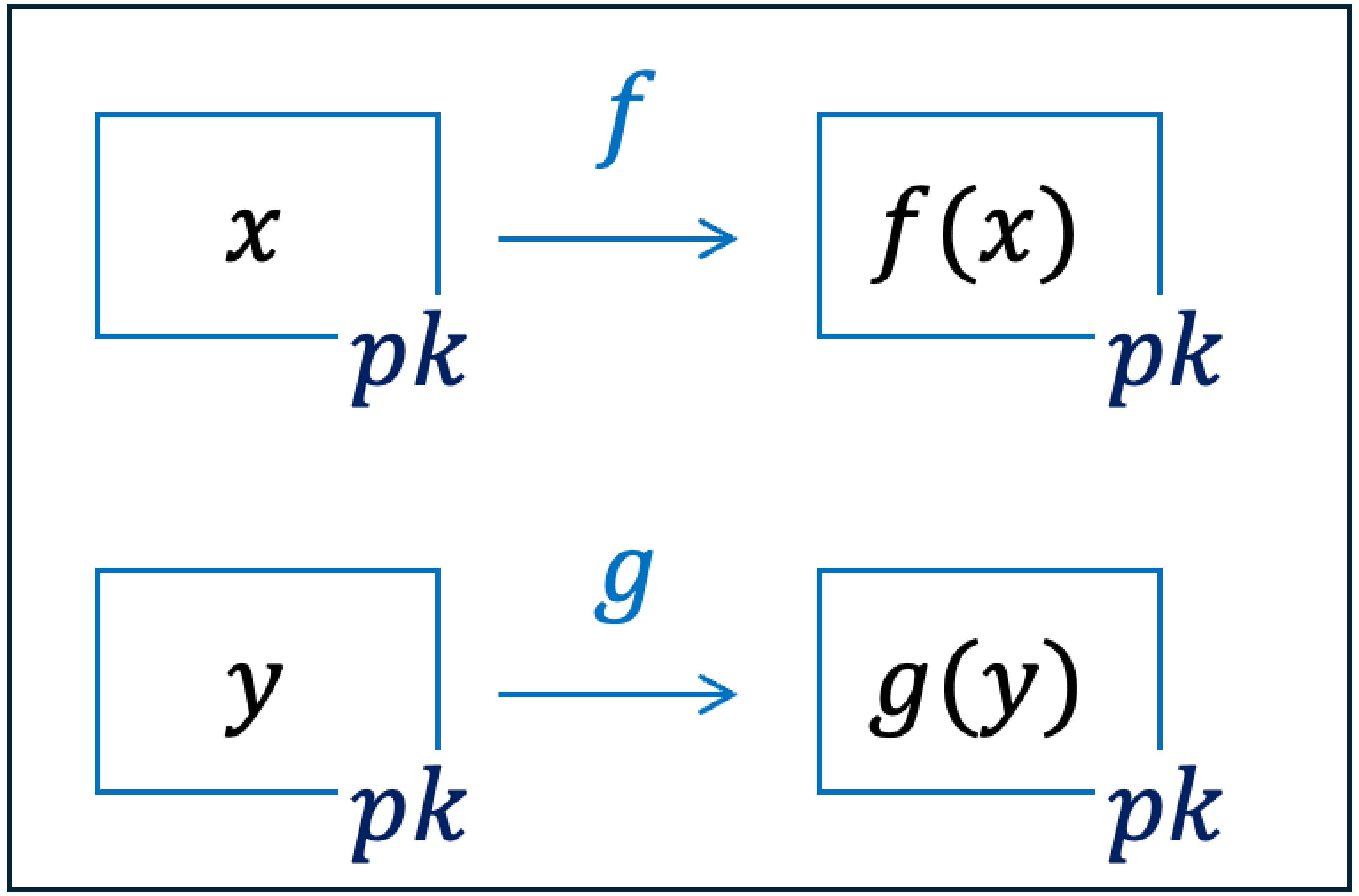
$\Pr[\mathrm{Pred}(\mathrm{CT}_1) = 1] > 2^{-\lambda^c} := \mu$

$\Pr[\mathrm{Pred}(\mathrm{CT}_2) = 1] > \mu$

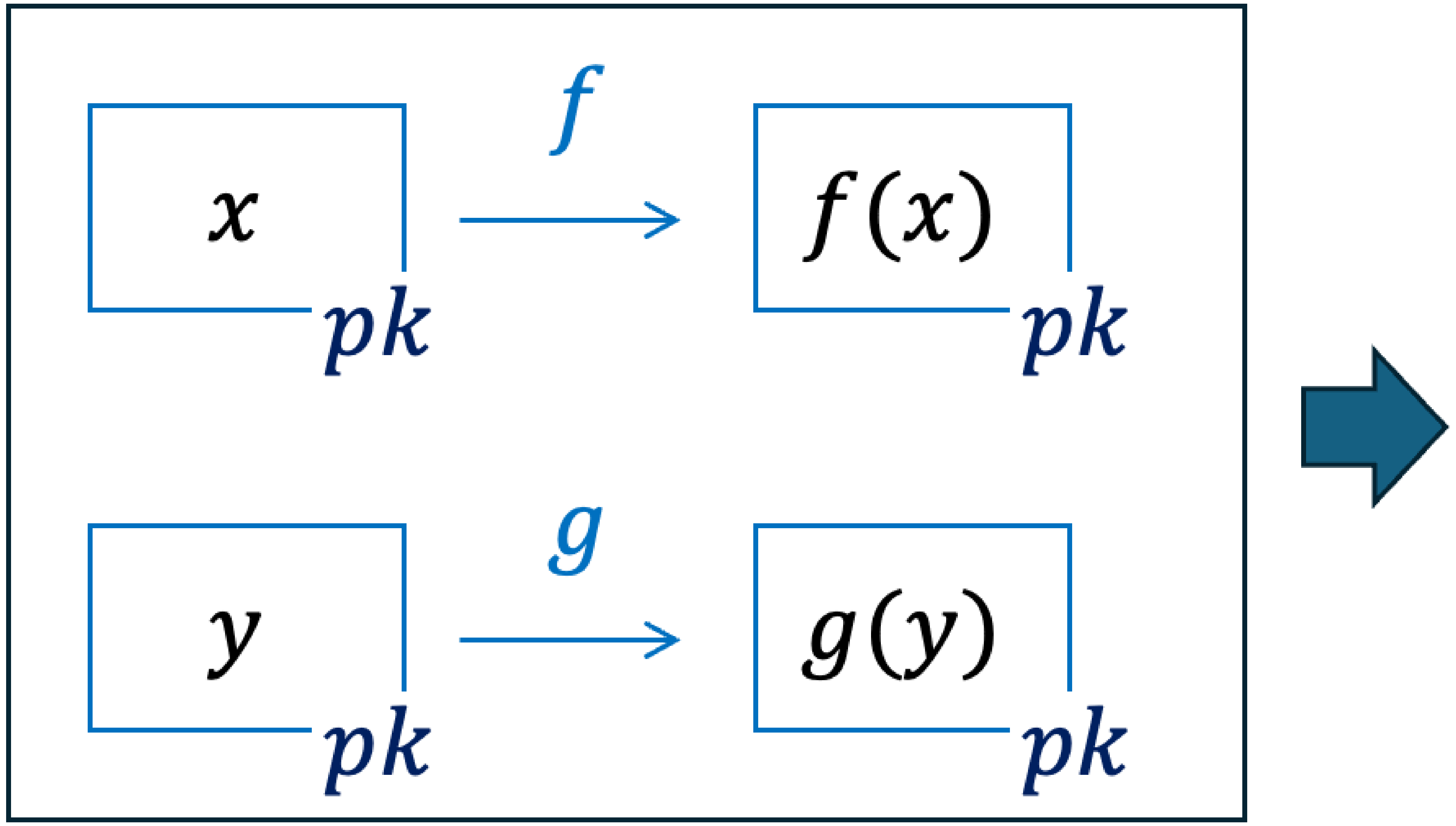
We can’t conclude $\Pr[\mathrm{Pred}(\mathrm{CT}_1) \wedge \mathrm{Pred}(\mathrm{CT}_2)] \geq \mu^2$

Avoid Composition via Multi-bit Evaluation

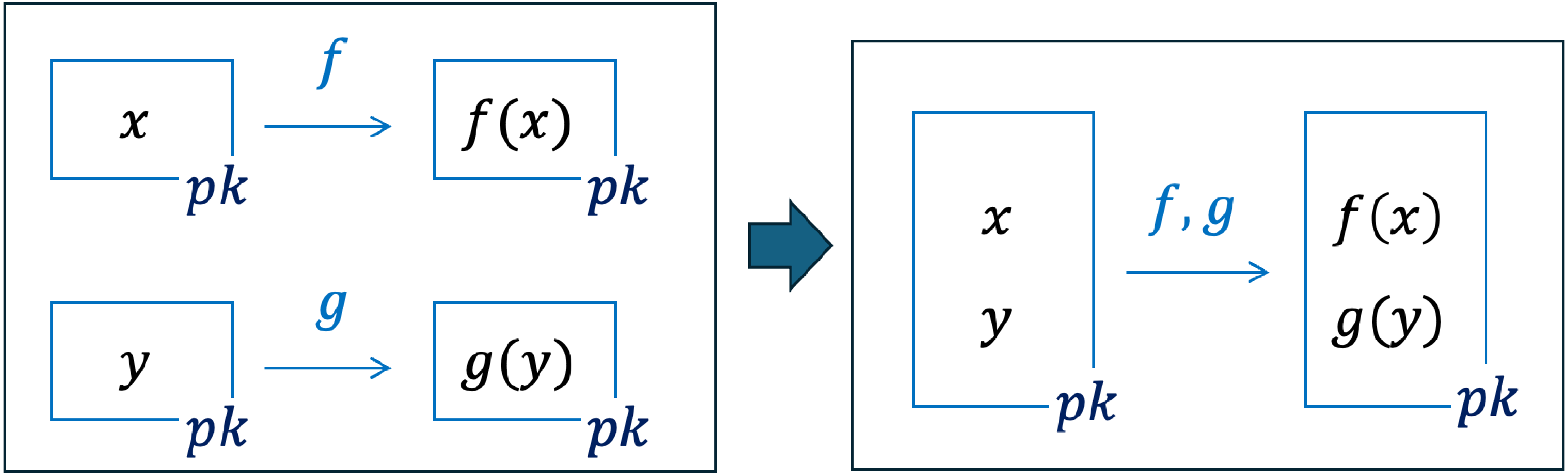
Avoid Composition via Multi-bit Evaluation



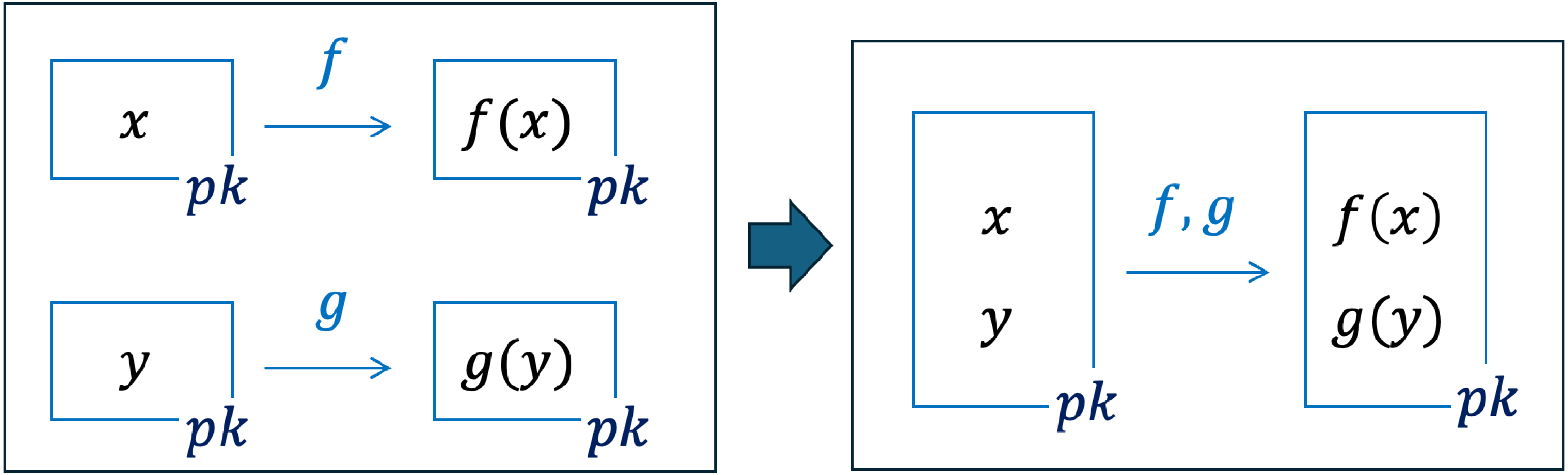
Avoid Composition via Multi-bit Evaluation



Avoid Composition via Multi-bit Evaluation



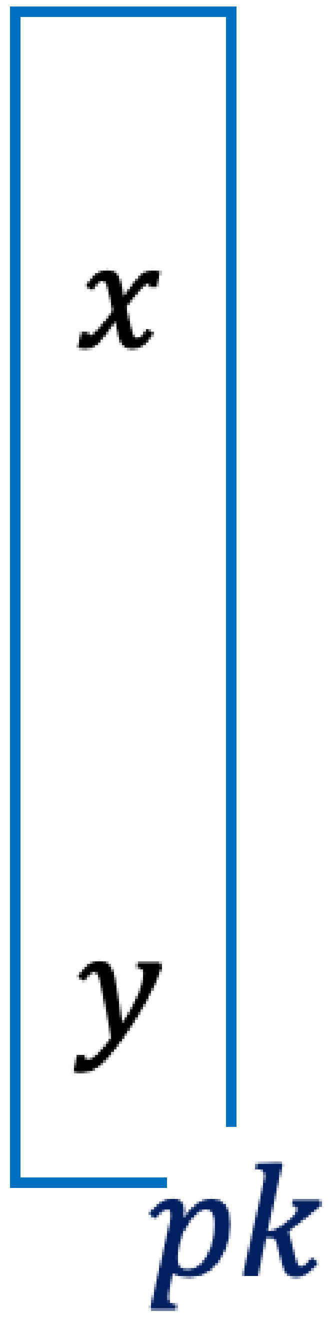
Avoid Composition via Multi-bit Evaluation



New Issue: we lost ‘**gate-by-gate**’ structure in HE evaluation——
Can’t talk about ‘intermediate ciphertext’ for a gate in f, g.

Homomorphic Eval Provides Intermediate CT

Homomorphic Eval Provides Intermediate CT

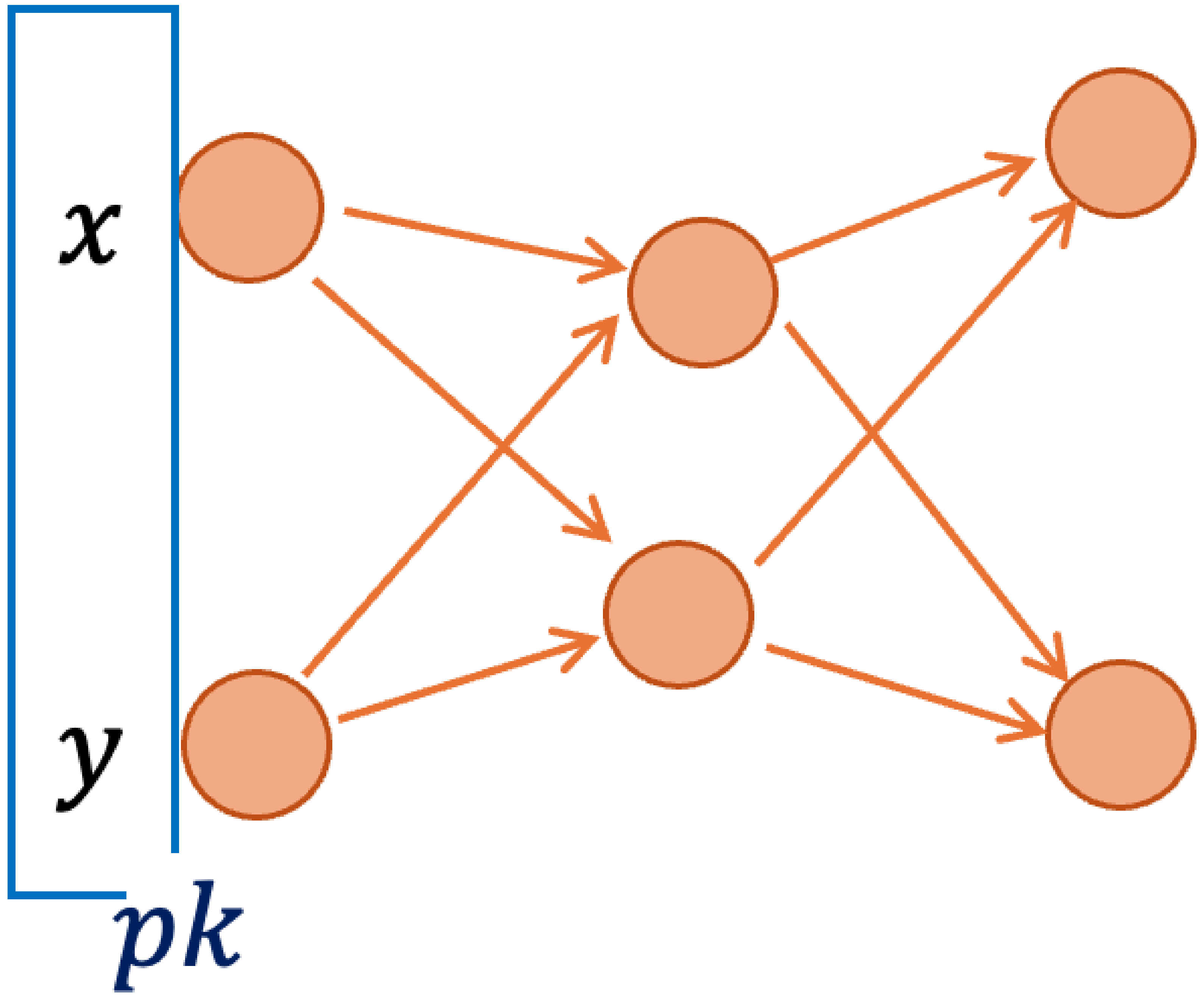


x

y

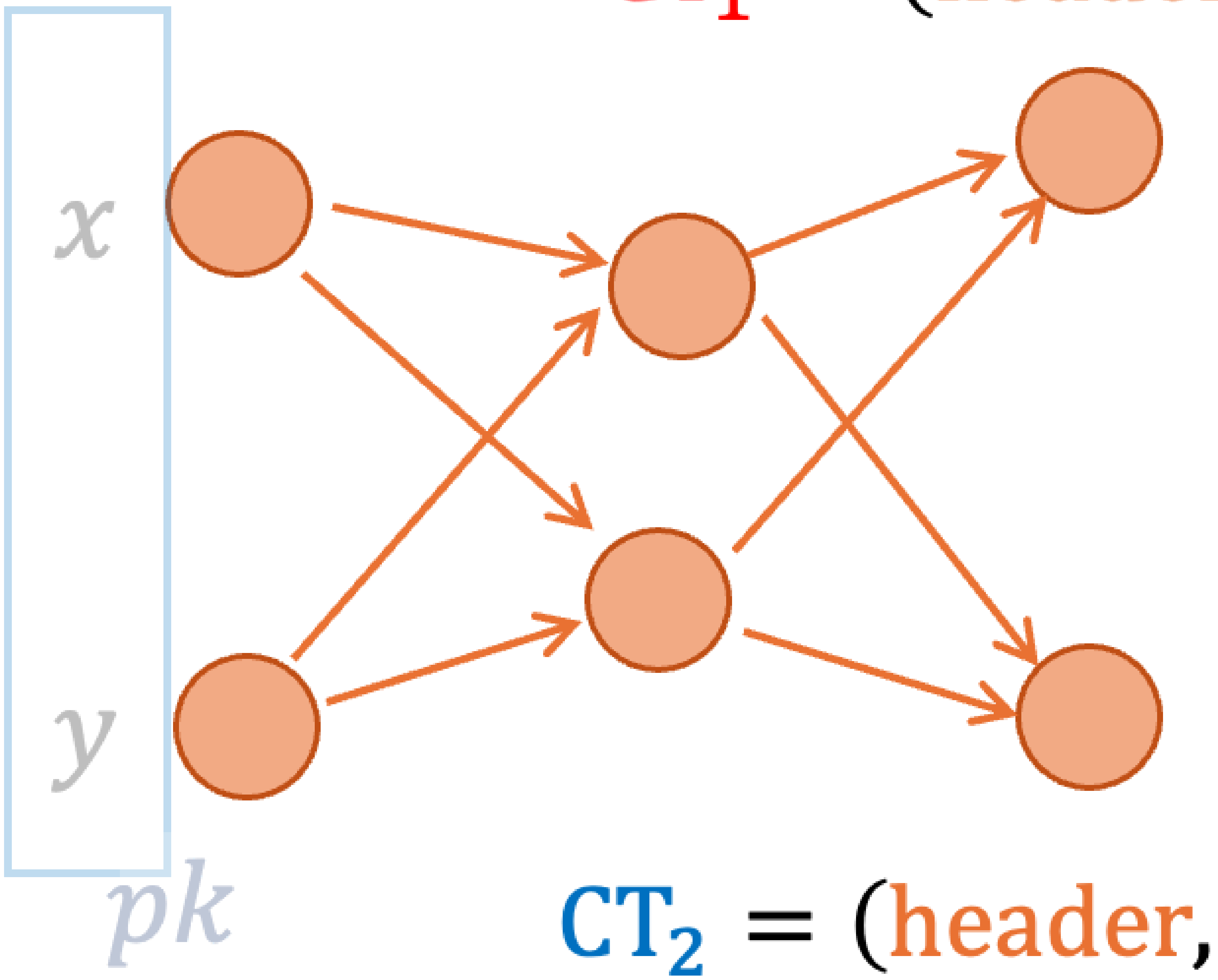
pk

Homomorphic Eval Provides Intermediate CT



Homomorphic Eval Provides Intermediate CT

$\mathrm{CT}_1 = (\mathrm{header}, \mathrm{payload}_1)$

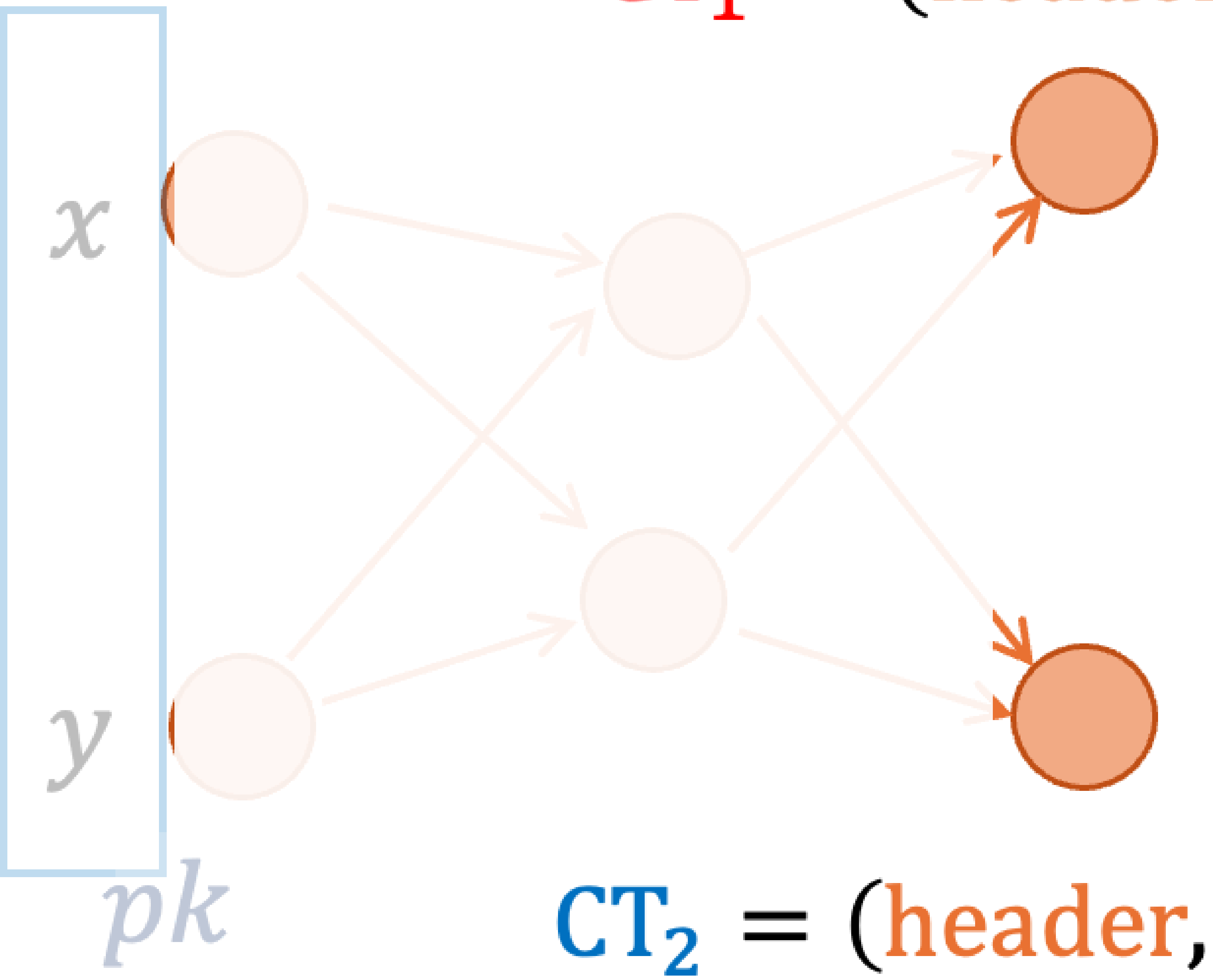


Eval also outputs **intermediate CT** for each gate

$\mathrm{CT}_2 = (\mathrm{header}, \mathrm{payload}_2)$

Homomorphic Eval Provides Intermediate CT

$\mathrm{CT}_1 = (\mathrm{header}, \mathrm{payload}_1)$



$\mathrm{CT}_2 = (\mathrm{header}, \mathrm{payload}_2)$

Eval also outputs **intermediate CT** for each gate

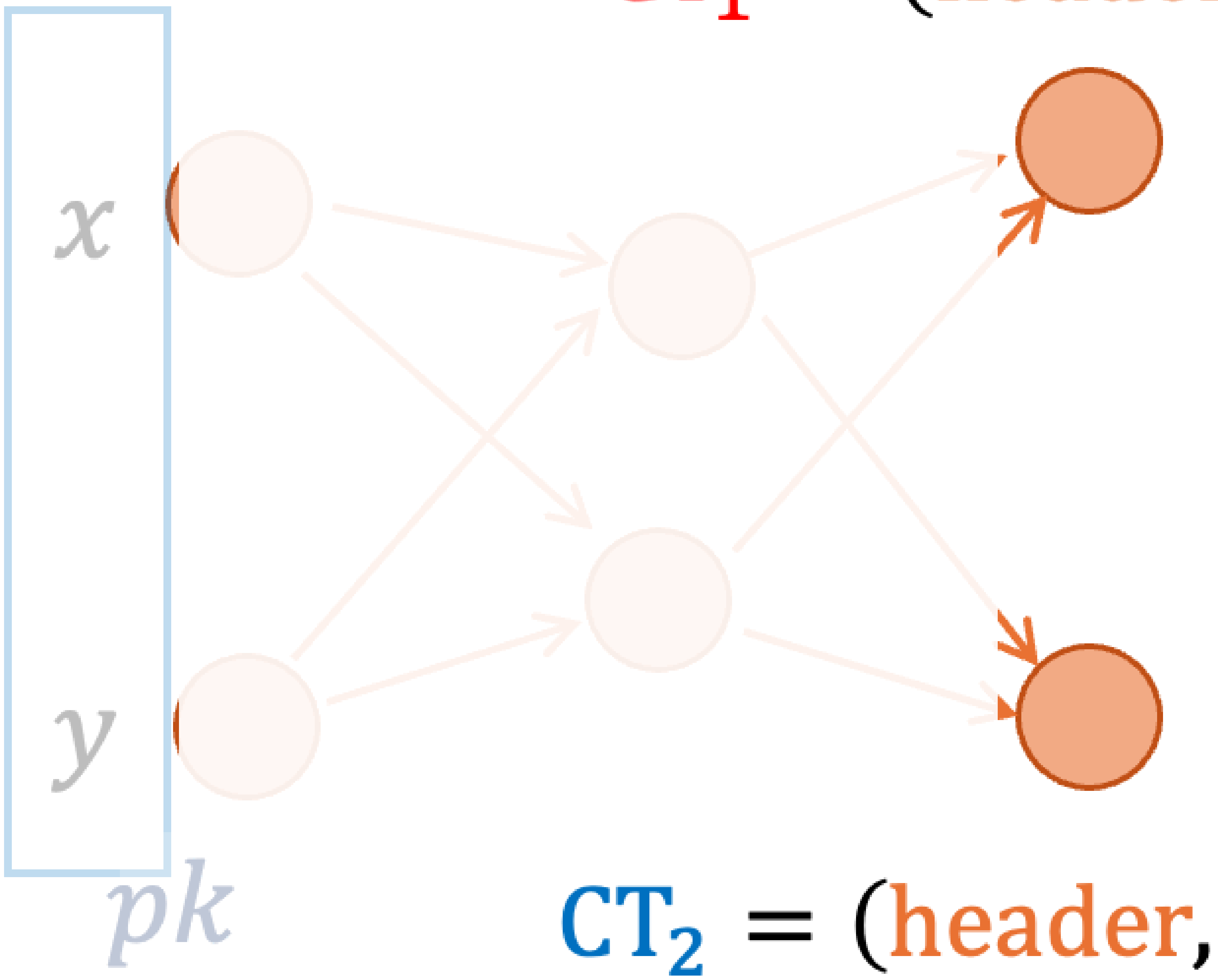
Header-Payload Structure

CT = (header, payload)

headers are the same for all gates

Homomorphic Eval Provides Intermediate CT

$$\mathrm{CT}_1 = (\mathrm{header}, \mathrm{payload}_1)$$



$$\mathrm{CT}_2 = (\mathrm{header}, \mathrm{payload}_2)$$

Eval also outputs **intermediate CT** for each gate

Header-Payload Structure

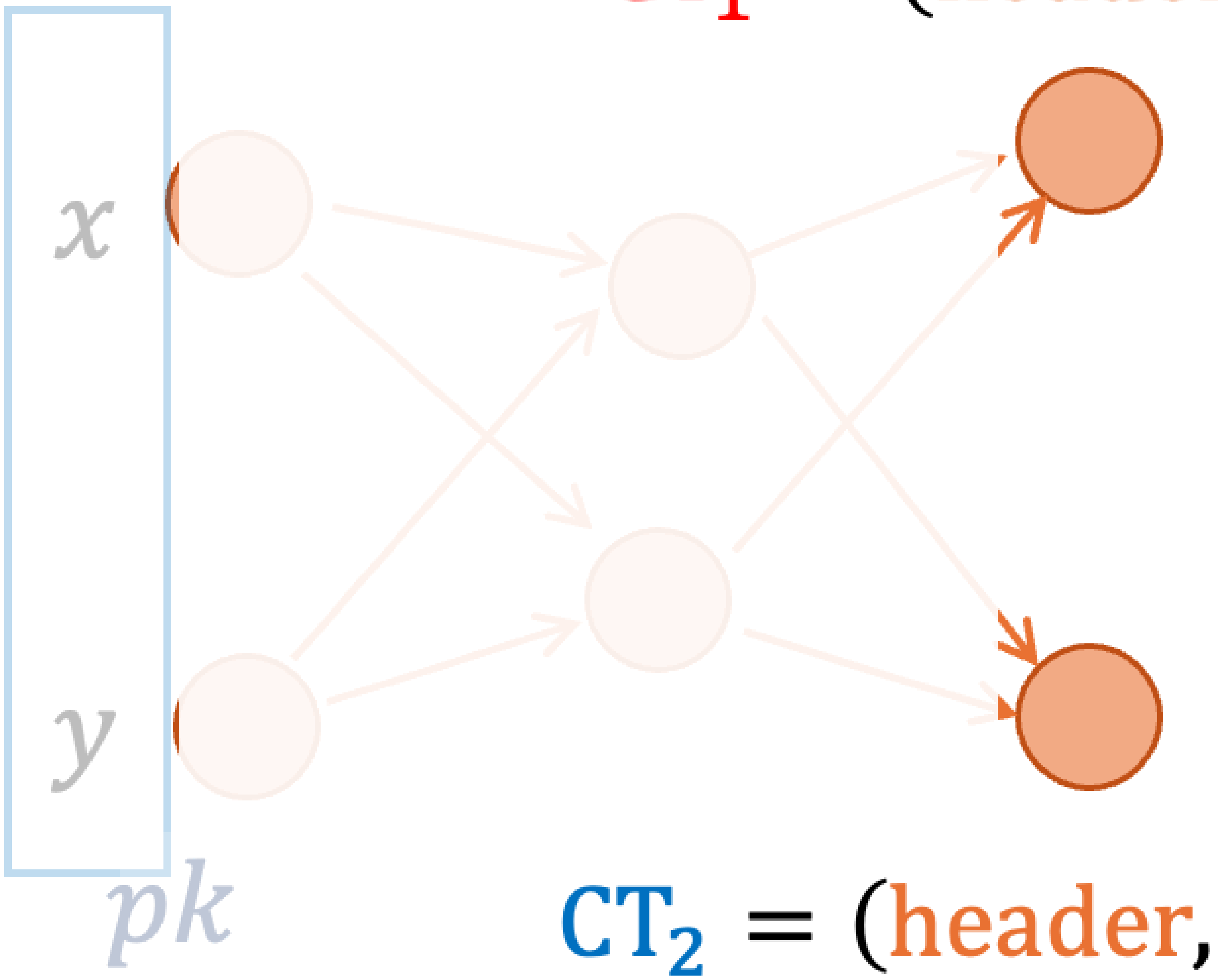
CT = (header, payload)

headers are the same for all gates

(Implicit in many FHE constructions)

Homomorphic Eval Provides Intermediate CT

$\mathrm{CT}_1 = (\mathrm{header}, \mathrm{payload}_1)$



$\mathrm{CT}_2 = (\mathrm{header}, \mathrm{payload}_2)$

Eval also outputs **intermediate CT** for each gate

Header-Payload Structure

CT = (header, payload)

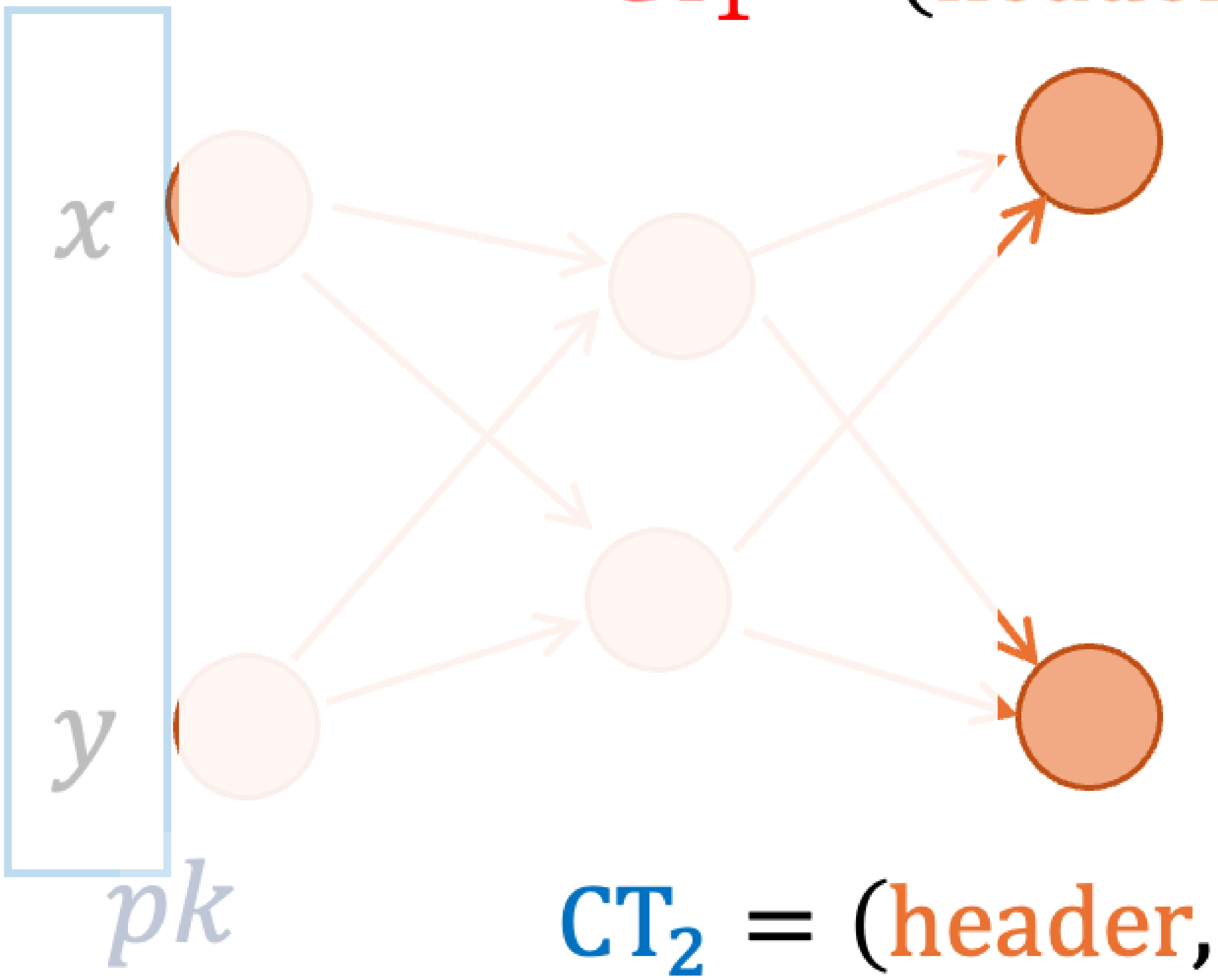
headers are the same for all gates

(Implicit in many FHE constructions)

Pred now only depends on header: Pred(header) = 1 => Dec correct.

Homomorphic Eval Provides Intermediate CT

$\mathrm{CT}_1 = (\mathrm{header}, \mathrm{payload}_1)$



$\mathrm{CT}_2 = (\mathrm{header}, \mathrm{payload}_2)$

Eval also outputs **intermediate CT** for each gate

Header-Payload Structure

CT = (header, payload)

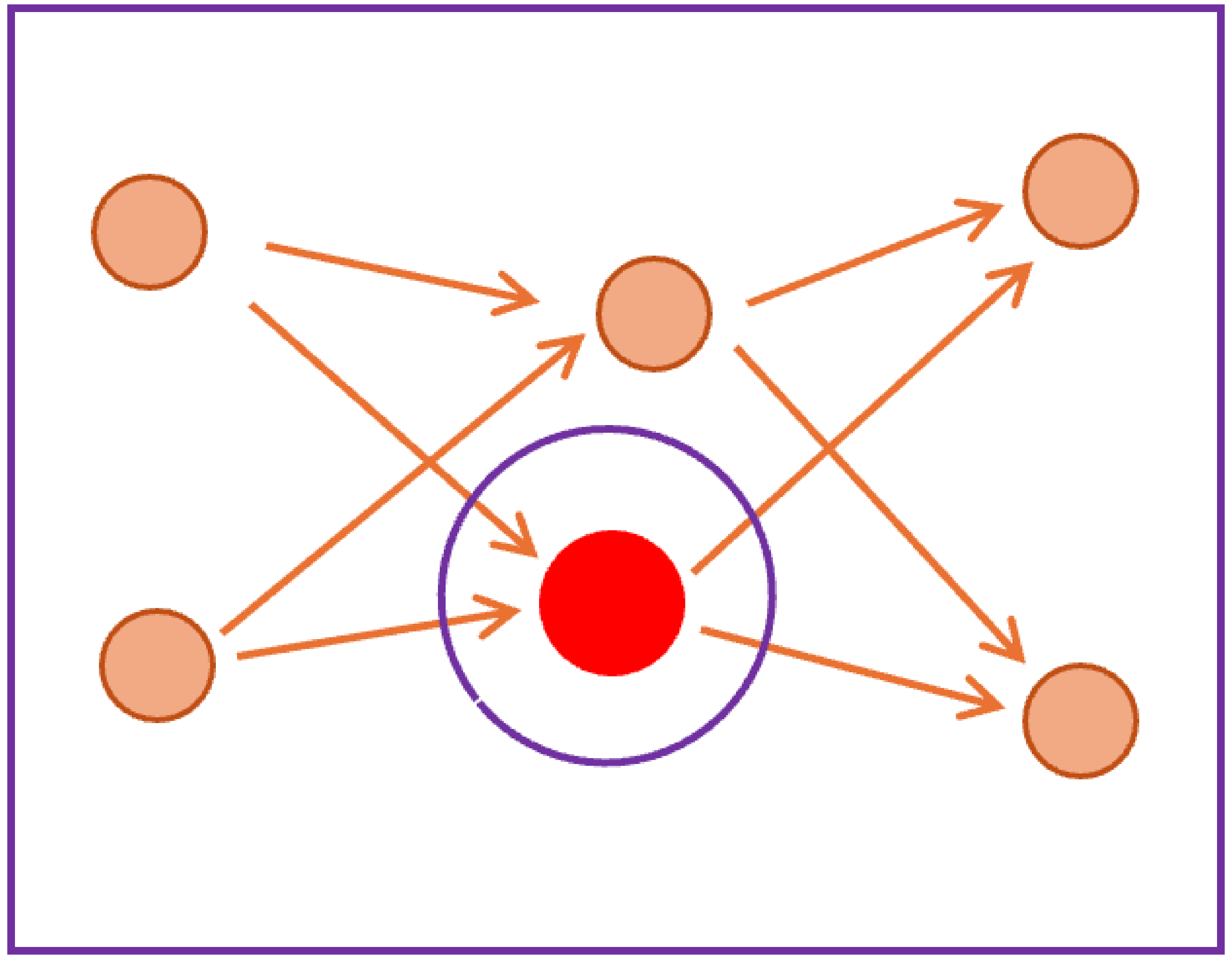
headers are the same for all gates

(Implicit in many FHE constructions)

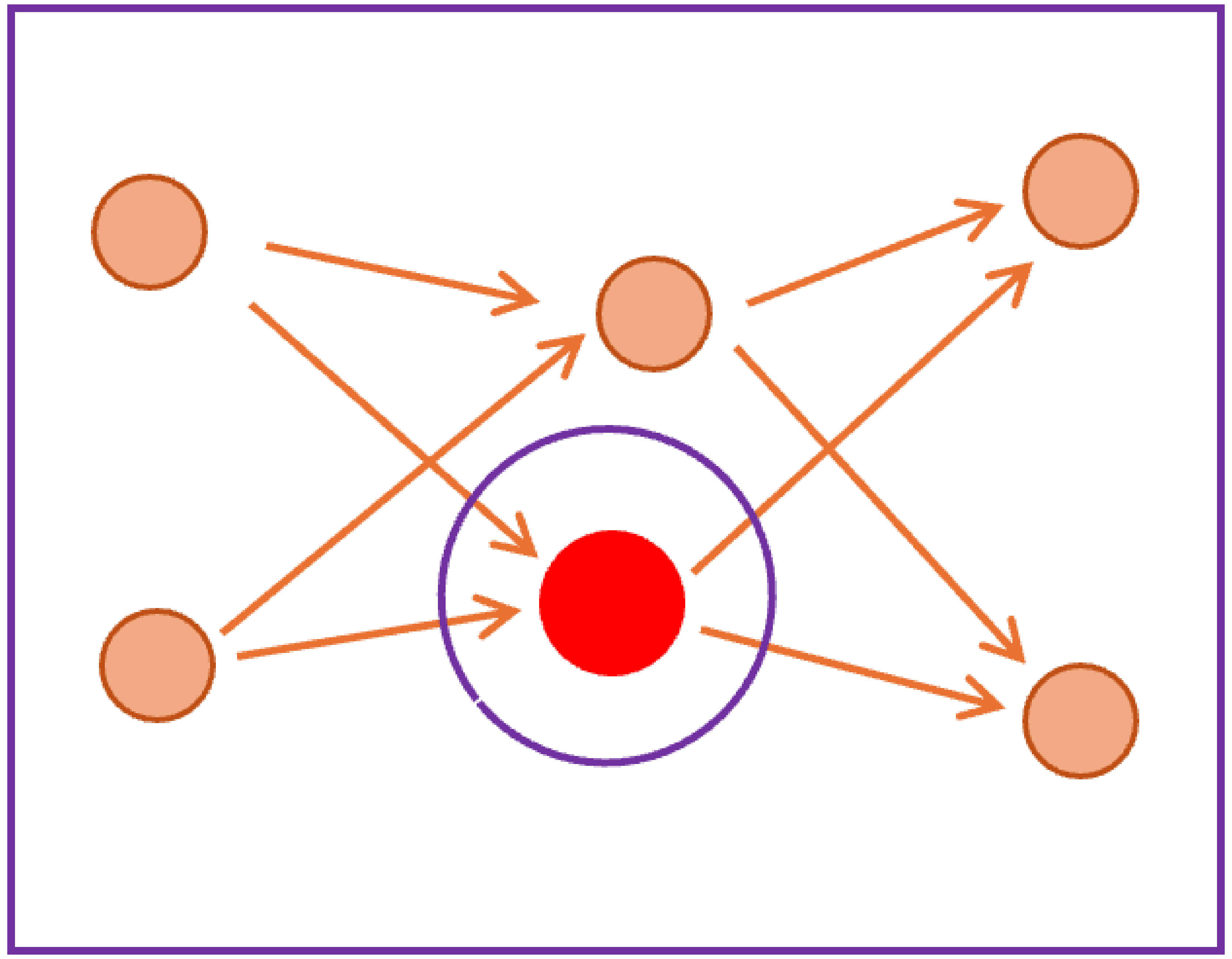
Pred now only depends on header: Pred(header) = 1 => Dec correct.

How to locally certify the correctness of intermediate ciphertexts?

SNARG for Local Correctness



SNARG for Local Correctness



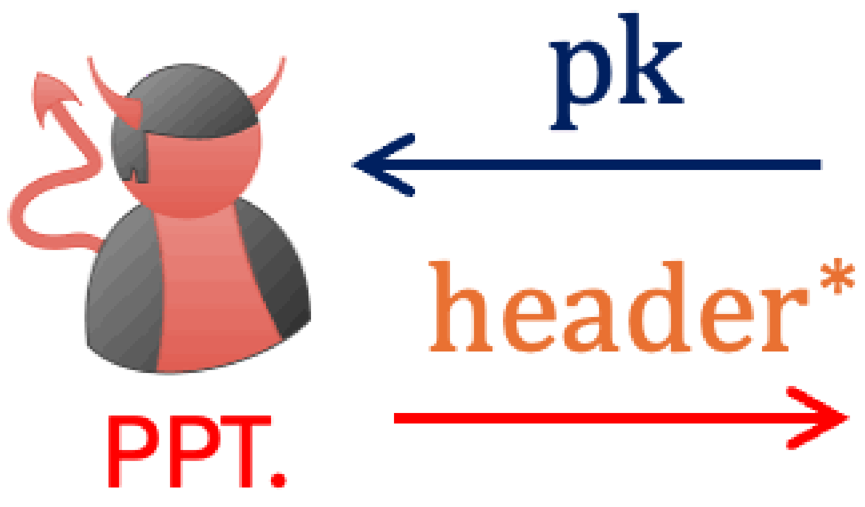
We generate a SNARG proof to certify the correctness for each intermediate ciphertext.

Summary of Definition for s-HE

$\mathrm{Gen}(1^{\lambda}) \to (pk, sk, \mathrm{Pred})$ Pred : privately computable

Homomorphic Evaluation

- Header-payload structure: CT = (header, payload)
- If $\mathrm{Pred}(\mathrm{header}) = 1$, then decryption is correct.
- Sometimes Decryptable for Malicious CT:



$$\Pr[\mathrm{Pred}(\mathrm{header}^{*}) = 1] > 2^{-\lambda^{c}}$$

- SNARGs for local correctness of intermediate CT

Rest of the Talk

Formal Definition of Sometimes-Decryptable HE

Construction of Sometimes-Decryptable HE

Rest of the Talk

Formal Definition of Sometimes-Decryptable HE

Construction of Sometimes-Decryptable HE

Starting Point: HE for Linear Functions

Starting Point: HE for Linear Functions

(A variant of ElGamal)

Starting Point: HE for Linear Functions

(A variant of ElGamal)

- KeyGen: pk = $(g, g^{\boxed{s}})$ $sk = \boxed{s}$

Starting Point: HE for Linear Functions

(A variant of ElGamal)

- KeyGen: pk = $(g, g^{\boxed{\quad s \quad}})$ $\quad sk = \overbrace{\boxed{\quad s \quad}}^{\text{Output length for linear functions}}$

Starting Point: HE for Linear Functions

(A variant of ElGamal)

- KeyGen: pk = $(g, g^{\boxed{\quad s \quad}})$ $\quad sk = \overbrace{\boxed{\quad s \quad}}^{\text{Output length for linear functions}}$
- Enc(pk, $\boldsymbol{m} \in \{0,1\}^n$): CT = $\left(g^{\boxed{r}}, g^{\boxed{r}\,\boxed{\quad s \quad} + \boxed{\begin{matrix} \boldsymbol{m} & & \\ & \cdots & \\ & & \boldsymbol{m} \end{matrix}}}\right)$

Starting Point: HE for Linear Functions

(A variant of ElGamal)

- KeyGen: pk = $(g, g^{\boxed{\quad s \quad}})$ $\quad sk = \overbrace{\boxed{\quad s \quad}}^{\text{Output length for linear functions}}$

- Enc(pk, $\boldsymbol{m} \in \{0,1\}^n$): CT = $\left(g^{\boxed{r}}, g^{\boxed{r}\,\boxed{\quad s \quad} + \boxed{\begin{matrix} \boldsymbol{m} & & \\ & \cdots & \\ & & \boldsymbol{m} \end{matrix}}}\right)$

Eval(pk, CT, $\boldsymbol{f}$)

Starting Point: HE for Linear Functions

(A variant of ElGamal)

- KeyGen: pk = $(g, g^{\boxed{\quad s \quad}})$ $\qquad sk = \overbrace{\boxed{\quad s \quad}}^{\text{Output length for linear functions}}$

- Enc(pk, $\boldsymbol{m} \in \{0,1\}^n$): CT = $\left(g^{\boxed{r}}, g^{\boxed{r}\,\boxed{\quad s \quad} + \boxed{\begin{matrix} \boldsymbol{m} & & \\ & \cdots & \\ & & \boldsymbol{m} \end{matrix}}}\right)$

Eval(pk, CT, $\boldsymbol{f}$)

Represent $\boldsymbol{f}: \{0,1\}^n \rightarrow \{0,1\}^\ell$ as $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_\ell \in \{0,1\}^n$

Starting Point: HE for Linear Functions

(A variant of ElGamal)

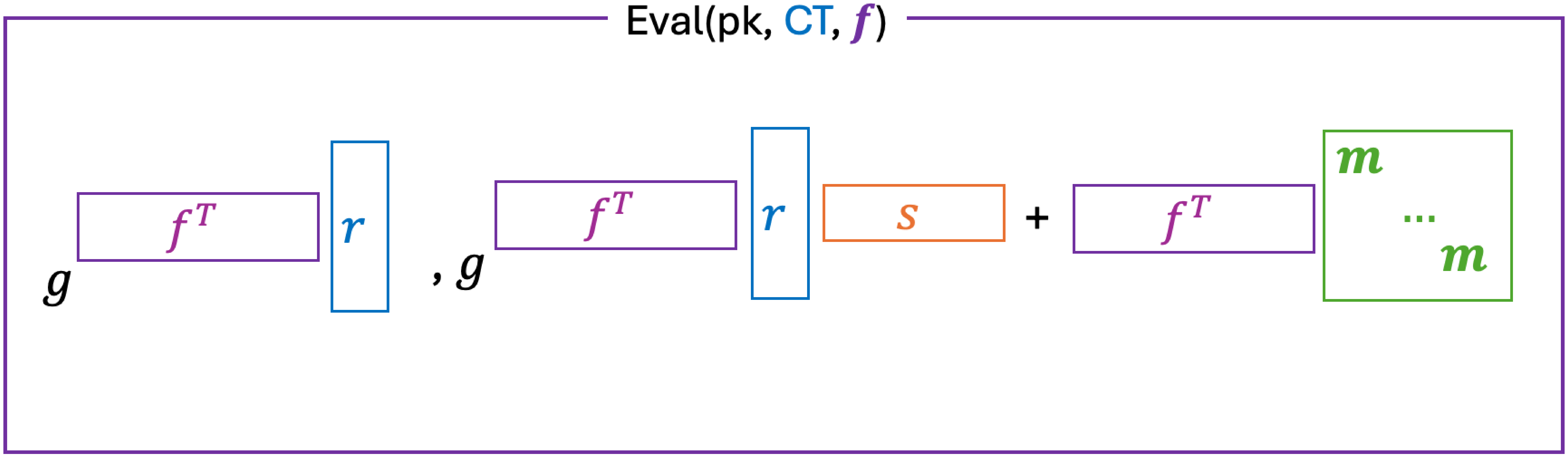
- KeyGen: pk = $(g, g^{\boxed{\quad s \quad}})$ $\quad sk = \overbrace{\boxed{\quad s \quad}}^{\text{Output length for linear functions}}$
- Enc(pk, $\boldsymbol{m} \in \{0,1\}^n$): CT = $\left(g^{\boxed{r}}, g^{\boxed{r}\boxed{\quad s \quad} + \boxed{\begin{matrix} \boldsymbol{m} & & \\ & \ldots & \\ & & \boldsymbol{m} \end{matrix}}}\right)$

Eval(pk, CT, $\boldsymbol{f}$)

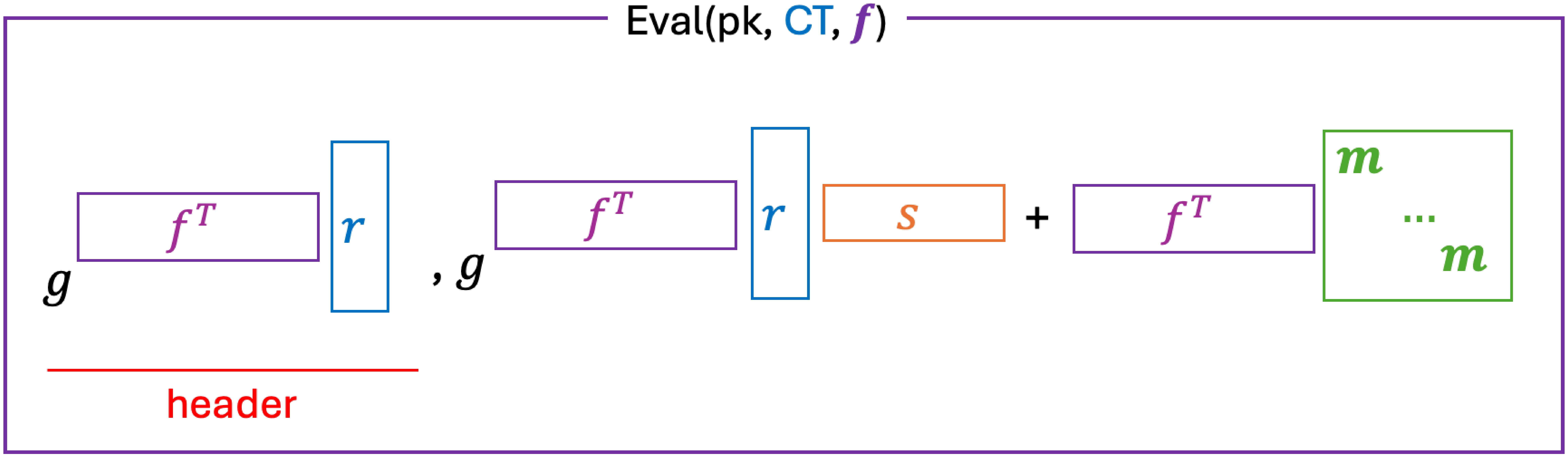
Represent $\boldsymbol{f}$: $\{0,1\}^n \rightarrow \{0,1\}^\ell$ as $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_\ell \in \{0,1\}^n$

$$g^{\boxed{f_1^T, \ldots, f_\ell^T}\boxed{r}}, g^{\boxed{f_1^T, \ldots, f_\ell^T}\boxed{r}\boxed{\quad s \quad} + \boxed{f_1^T, \ldots, f_\ell^T}\boxed{\begin{matrix} \boldsymbol{m} & & \\ & \ldots & \\ & & \boldsymbol{m} \end{matrix}}}$$

HE for Linear Functions: Decryption



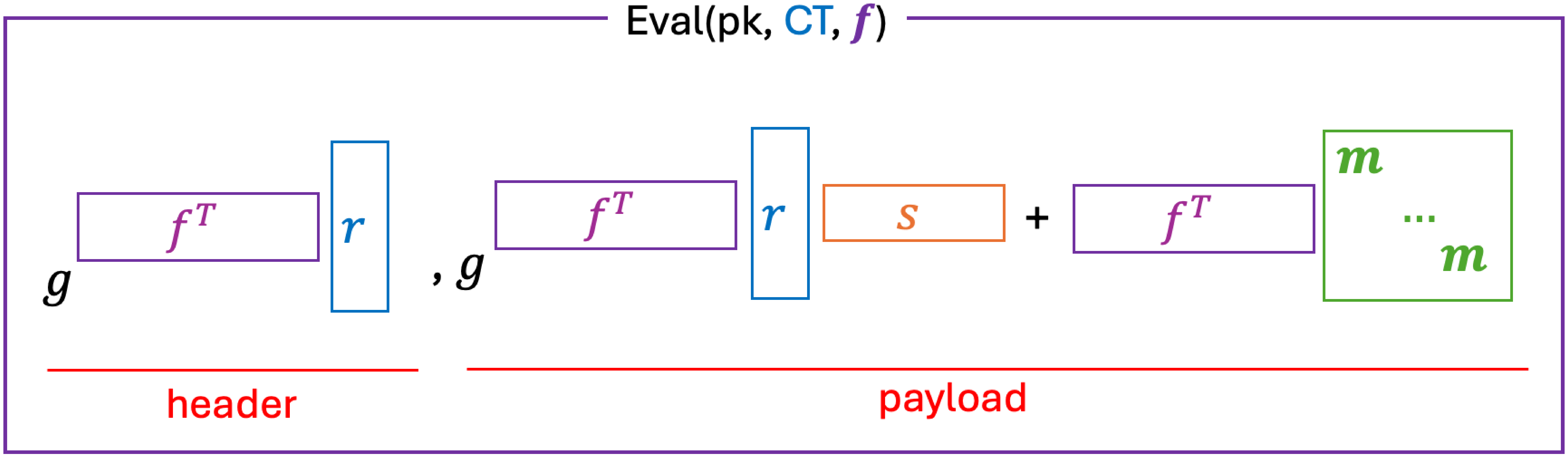
HE for Linear Functions: Decryption



Eval(pk, CT, f)

header

HE for Linear Functions: Decryption



HE for Linear Functions: Decryption

Eval(pk, CT, f)

$$\underbrace{g^{\boxed{f^T}\boxed{r}}}_{\text{header}},\ \underbrace{g^{\boxed{f^T}\boxed{r}\boxed{s} + \boxed{f^T}\boxed{\begin{matrix} m & & \\ & \cdots & \\ & & m \end{matrix}}}}_{\text{payload}}$$

- Decryption: divide payload by header$^{\boxed{s}}$, get $g^{\boxed{f_1 \cdot m, \ldots, f_\ell \cdot m}}$

SNARGs for Local Correctness

$$\mathsf{CT} = \left(g^{\boxed{r}},\ g^{\boxed{r}\,\boxed{\ s\ } + \boxed{\begin{matrix} m & & \\ & \dots & \\ & & m \end{matrix}}}\right)$$

Eval(pk, CT, f)

$$g^{\boxed{f_1^T, \dots, f_\ell^T}\,\boxed{r}},\ g^{\boxed{f_1^T, \dots, f_\ell^T}\,\boxed{r}\,\boxed{\ s\ } + \boxed{f_1^T, \dots, f_\ell^T}\,\boxed{\begin{matrix} m & & \\ & \dots & \\ & & m \end{matrix}}}$$

SNARGs for Local Correctness

$$\mathrm{CT} = \left(g^{\boxed{r}}, g^{\boxed{r}\,\boxed{\;s\;} + \boxed{\begin{matrix} m & & \\ & \dots & \\ & & m \end{matrix}}}\right)$$

Eval(pk, CT, f)

$$g^{\boxed{f_1^T, \dots, f_\ell^T}\,\boxed{r}}, g^{\boxed{f_1^T, \dots, f_\ell^T}\,\boxed{r}\,\boxed{\;s\;} + \boxed{f_1^T, \dots, f_\ell^T}\,\boxed{\begin{matrix} m & & \\ & \dots & \\ & & m \end{matrix}}}$$

$$\boxed{f_1^T, \dots, f_\ell^T} = \boxed{\boldsymbol{f_1^T \dots 0 \dots f_\ell^T}} + \boxed{\boldsymbol{0 \dots f_i^T \dots 0}}$$

SNARGs for Local Correctness

SNARGs for Local Correctness

i-th Output of Eval$(f,\cdot)$

SNARGs for Local Correctness

i-th Output of Eval$(f,\cdot)$

$$g^{\boxed{\boldsymbol{f}_1^T \dots \boldsymbol{0} \dots \boldsymbol{f}_\ell^T}\boxed{r}}$$

$$\cdot\, g^{\boxed{\boldsymbol{0} \dots \boldsymbol{f}_i^T \dots \boldsymbol{0}}\boxed{r}}$$

SNARGs for Local Correctness

i-th Output of Eval$(f, \cdot)$

$$g^{\boxed{f_1^T \ldots 0 \ldots f_\ell^T}\boxed{r}}, \; g^{\boxed{f_1^T \ldots 0 \ldots f_\ell^T}\boxed{r}\, s_i + \boxed{f_1^T \ldots 0 \ldots f_\ell^T}\boxed{\begin{matrix}0\\ \ldots\\ m\\ \ldots\\ 0\end{matrix}}}$$

$$\cdot\, g^{\boxed{0 \ldots f_i^T \ldots 0}\boxed{r}} \qquad \cdot\, g^{\boxed{0 \ldots f_i^T \ldots 0}\boxed{r}\, s_i + f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval(f,·)

$$g^{\boxed{f_1^T \ldots 0 \ldots f_\ell^T}\boxed{r}} \quad , g^{\boxed{f_1^T \ldots 0 \ldots f_\ell^T}\boxed{r}\, s_i \;+\; \boxed{f_1^T \ldots 0 \ldots f_\ell^T}\boxed{\begin{matrix}0\\ \ldots\\ m\\ \ldots\\ 0\end{matrix}}}$$

$$\cdot\, g^{\boxed{0 \ldots f_i^T \ldots 0}\boxed{r}} \quad \cdot\, g^{\boxed{0 \ldots f_i^T \ldots 0}\boxed{r}\, s_i \;+\; f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval(f,·)

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}} \;,\; g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i \;+\; \boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{\begin{matrix}0\\ \dots\\ m\\ \dots\\ 0\end{matrix}}}$$

$$\cdot\; g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \qquad \cdot\; g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i \;+\; f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval$(f, \cdot)$

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}}, \; g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i \;+\; \boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{\begin{matrix}0\\ \dots\\ m\\ \dots\\ 0\end{matrix}}}$$

$$\cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \qquad \cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i \;+\; f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval$(f,\cdot)$

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}}, \; g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i + \boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{\begin{matrix}0 \\ \dots \\ m \\ \dots \\ 0\end{matrix}}}$$

$$\cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \quad \cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i + f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval$(f, \cdot)$

$$g^{\boxed{f_1^T \ldots 0 \ldots f_\ell^T}\boxed{r}} \;,\; g^{\boxed{f_1^T \ldots 0 \ldots f_\ell^T}\boxed{r}\, s_i \;+\; \boxed{f_1^T \ldots 0 \ldots f_\ell^T}\boxed{\begin{matrix}0\\ \ldots\\ m\\ \ldots\\ 0\end{matrix}}}$$

$$\cdot\, g^{\boxed{0 \ldots f_i^T \ldots 0}\boxed{r}} \qquad \cdot\, g^{\boxed{0 \ldots f_i^T \ldots 0}\boxed{r}\, s_i \;+\; f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval(f,·)

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}} \;,\; g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i \;+\; \boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{\begin{matrix} 0 \\ \dots \\ m \\ \dots \\ 0 \end{matrix}}}$$

$$\cdot\; g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \qquad \cdot\; g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i \;+\; f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval$(f,\cdot)$

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}}, g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i} \quad + \boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{\begin{matrix}0\\ \dots\\ m\\ \dots\\ 0\end{matrix}}$$

$$\cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \quad \cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i \,+\, f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval(f,·)

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}} \quad , g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i} \qquad + \boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{\begin{matrix}0\\ \dots\\ m\\ \dots\\ 0\end{matrix}}$$

$$\cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \qquad \cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i \;+\; f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval$(f,\cdot)$

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}} ,\; g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i \qquad + \boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{\begin{matrix}0\\ \dots\\ m\\ \dots\\ 0\end{matrix}}}$$

$$\cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \qquad \cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i \;+\; f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval$(f,\cdot)$

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}} \;,\; g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i \; + \boxed{f_1^T \dots 0 \dots f_\ell^T}\left|\begin{matrix}0\\ \dots\\ m\\ \dots\\ 0\end{matrix}\right.}$$

$$\cdot\; g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \qquad \cdot\; g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i \;+\; f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval$(f, \cdot)$

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}} \quad , \; g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i}$$

$$\cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \quad \cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i \,+\, f_i^T \cdot m}$$

SNARGs for Local Correctness

i-th Output of Eval$(f,\cdot)$

$$g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}}, \; g^{\boxed{f_1^T \dots 0 \dots f_\ell^T}\boxed{r}\, s_i}$$

$$\cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}} \qquad \cdot\, g^{\boxed{0 \dots f_i^T \dots 0}\boxed{r}\, s_i + f_i^T \cdot m}$$

SNARGs for Local Correctness

$$g^{\boxed{\boldsymbol{f}_1^T \dots \mathbf{0} \dots \boldsymbol{f}_\ell^T}\boxed{\boldsymbol{r}}} \quad , g^{\boxed{\boldsymbol{f}_1^T \dots \mathbf{0} \dots \boldsymbol{f}_\ell^T}\boxed{\boldsymbol{r}}\, \boldsymbol{s_i}}$$

$$\cdot\, g^{\boxed{\mathbf{0} \dots \boldsymbol{f}_i^T \dots \mathbf{0}}\boxed{\boldsymbol{r}}} \quad \cdot\, g^{\boxed{\mathbf{0} \dots \boldsymbol{f}_i^T \dots \mathbf{0}}\boxed{\boldsymbol{r}}\, \boldsymbol{s_i} + \boldsymbol{f}_i^T \cdot \boldsymbol{m}}$$

SNARGs for Local Correctness

Prove this Part via SNARGs for Linear relations from sub-exp DDH
[Choudhuri-Garg-Jain-J-Zhang'23]

$$g^{\boxed{\boldsymbol{f}_1^T \ldots \mathbf{0} \ldots \boldsymbol{f}_\ell^T}\boxed{\boldsymbol{r}}}, \; g^{\boxed{\boldsymbol{f}_1^T \ldots \mathbf{0} \ldots \boldsymbol{f}_\ell^T}\boxed{\boldsymbol{r}}\, s_i}$$

$$\cdot\, g^{\boxed{\mathbf{0} \ldots \boldsymbol{f}_i^T \ldots \mathbf{0}}\boxed{\boldsymbol{r}}} \qquad \cdot\, g^{\boxed{\mathbf{0} \ldots \boldsymbol{f}_i^T \ldots \mathbf{0}}\boxed{\boldsymbol{r}}\, s_i \,+\, \boldsymbol{f}_i^T \cdot \boldsymbol{m}}$$

SNARGs for Local Correctness

Prove this Part via SNARGs for Linear relations from sub-exp DDH
[Choudhuri-Garg-Jain-J-Zhang'23]

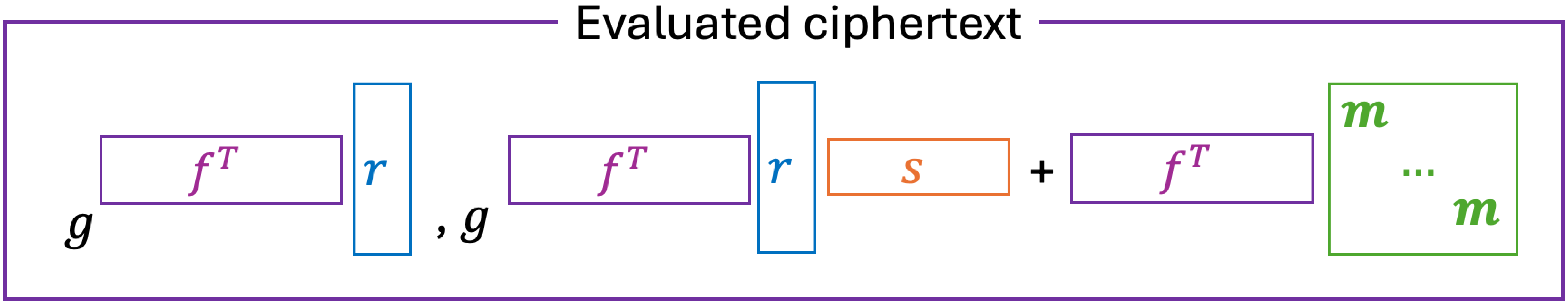
$$g^{\boxed{\boldsymbol{f}_1^T \dots \boldsymbol{0} \dots \boldsymbol{f}_\ell^T}\boxed{\boldsymbol{r}}} \;,\; g^{\boxed{\boldsymbol{f}_1^T \dots \boldsymbol{0} \dots \boldsymbol{f}_\ell^T}\boxed{\boldsymbol{r}}\; s_i}$$

$$\cdot\; g^{\boxed{\boldsymbol{0} \dots \boldsymbol{f}_i^T \dots \boldsymbol{0}}\boxed{\boldsymbol{r}}} \qquad \cdot\; g^{\boxed{\boldsymbol{0} \dots \boldsymbol{f}_i^T \dots \boldsymbol{0}}\boxed{\boldsymbol{r}}\; s_i \,+\, \boldsymbol{f}_i^T \cdot \boldsymbol{m}}$$

The verifier can compute by itself in time poly(input arity of i-th output)

Bootstrap from Linear Functions to TC^0 (high-level)

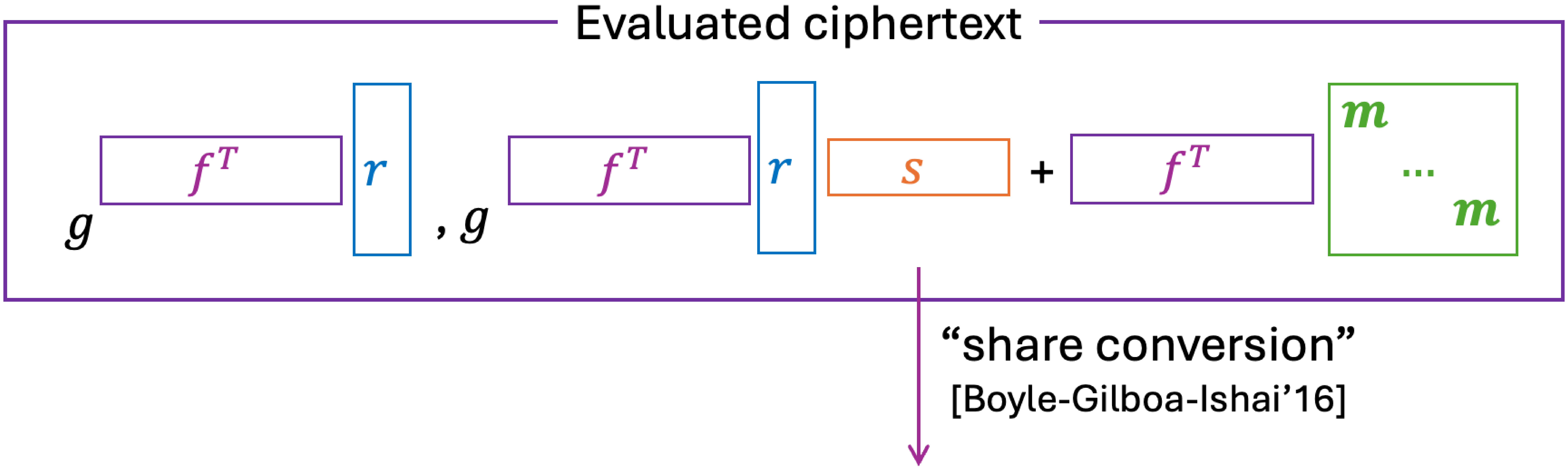
Bootstrap from Linear Functions to TC^0 (high-level)



Evaluated ciphertext

$$g^{\boxed{f^T}\boxed{r}},\ g^{\boxed{f^T}\boxed{r}\boxed{s} + \boxed{f^T}\boxed{\begin{matrix} m & & \\ & \cdots & \\ & & m \end{matrix}}}$$

Bootstrap from Linear Functions to TC^0 (high-level)

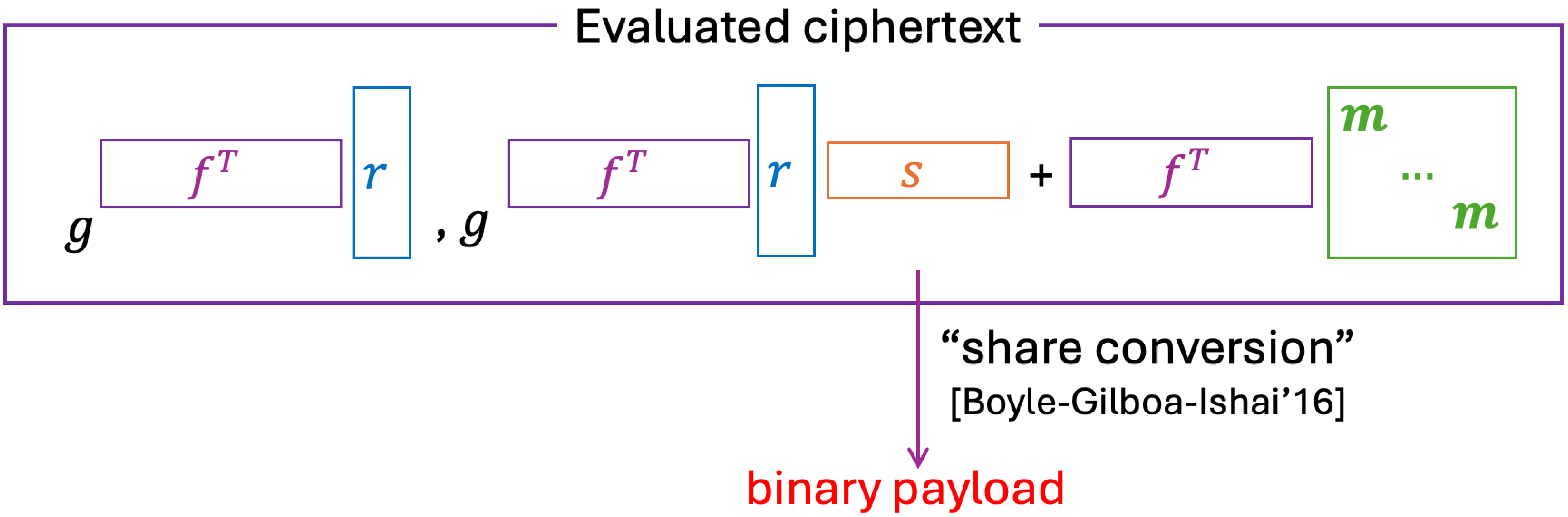


Evaluated ciphertext

“share conversion”

[Boyle-Gilboa-Ishai’16]

Bootstrap from Linear Functions to TC^0 (high-level)



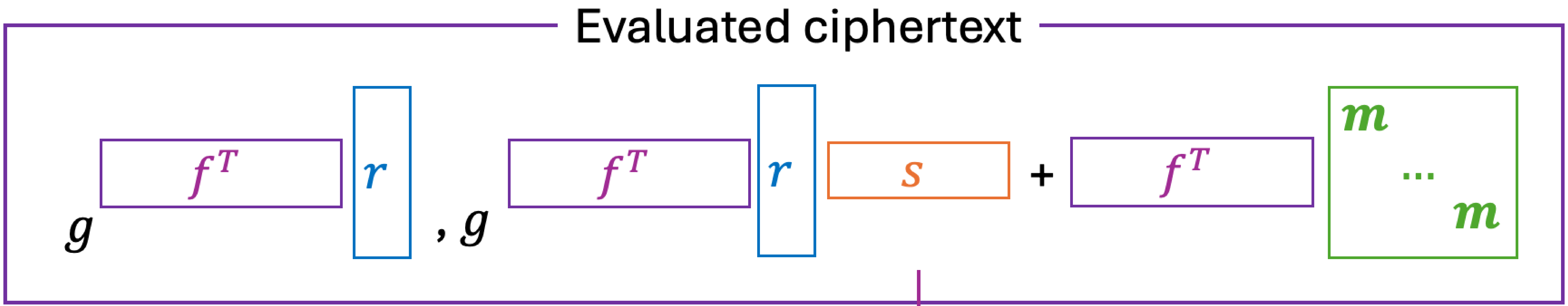
Evaluated ciphertext

“share conversion”

[Boyle-Gilboa-Ishai’16]

binary payload

Bootstrap from Linear Functions to TC^0 (high-level)



Evaluated ciphertext

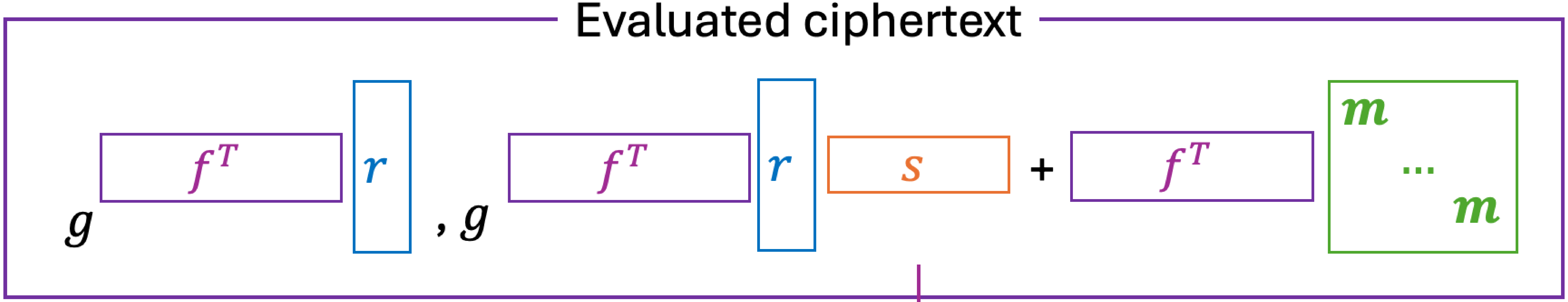
"share conversion"

[Boyle-Gilboa-Ishai'16]

Additive decryption: Decryption = binary payload ⊕ BGI(header^s)

Bootstrap from Linear Functions to TC^0 (high-level)

Evaluated ciphertext



“share conversion”

[Boyle-Gilboa-Ishai’16]

Additive decryption: Decryption = binary payload ⊕ BGI(header^s)

→ s-HE for a layer of Threshold Gates → s-HE for full TC^0

[Jain-J’21] techniques

Summary of Results

Summary of Results

- Sometimes-decryptable HE for TC^0 from sub-exp DDH

Summary of Results

- Sometimes-decryptable HE for TC^0 from sub-exp DDH
- Applications:
  - SNARGs from sub-exp DDH for languages that has poly-size TC^0 Frege proof of non-membership
  - Monotone-Policy BARGs from sub-exp DDH

Summary of Results

- Sometimes-decryptable HE for TC^0 from sub-exp DDH
- Applications:
  - SNARGs from sub-exp DDH for languages that has poly-size TC^0 Frege proof of non-membership
  - Monotone-Policy BARGs from sub-exp DDH

Take away

Can replace FHE in “proof-system applications” (e.g. NIZK/SNARG) to achieve constructions from DDH in pairing-free groups!

Thank you!

Q & A