Functional Encryption for Turing Machines with Dynamic Bounded Collusion from LWE

Shweta Agrawal

Monosij Maitra

Narasimha Sai Vempati

Shota Yamada

IIT Madras

TU Darmstadt

AIST Japan

TECHNISCHE UNIVERSITÄT DARMSTADT

AIST
NATIONAL INSTITUTE OF ADVANCED INDUSTRIAL SCIENCE AND TECHNOLOGY (AIST)

Functional Encryption [SW05, BSW11, O'N10]

Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)

Decrypt$(CT, SK_f) = f(m)$

Functional Encryption [SW05, BSW11, O'N10]

Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)

Decrypt(CT, SK_f) = $f(m)$

Functional Encryption [SW05, BSW11, O'N10]

Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)

Decrypt$(CT, SK_f) = f(m)$

Functional Encryption [SW05, BSW11, O'N10]

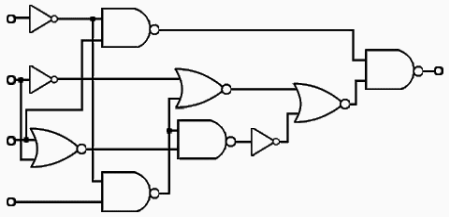
Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)



Decrypt$(CT, SK_f) = f(m)$

Functional Encryption [SW05, BSW11, O'N10]

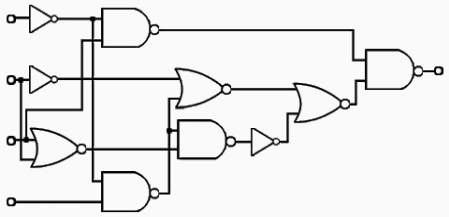
Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)



Decrypt(CT, SK_f) = $f(m)$

Functional Encryption [SW05, BSW11, O'N10]

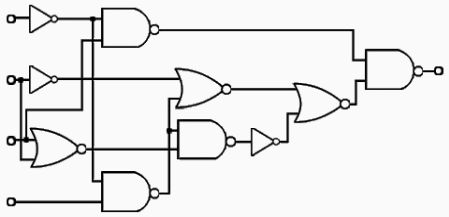
Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)



Security:

Any PPT adv. learns only $f(m)$ and nothing else.

Decrypt(CT, SK_f) = $f(m)$

Functional Encryption [SW05, BSW11, O'N10]

$\mathbf{Setup}(1^\lambda) : (\mathrm{PK}, \mathrm{MSK})$

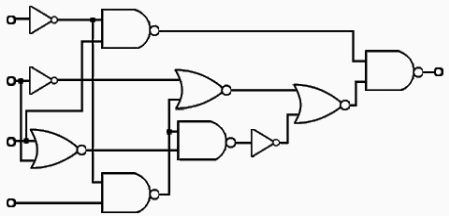
$\mathrm{CT} \leftarrow \mathbf{Encrypt}(\mathrm{PK}, m)$
$\approx$
$\mathrm{CT}^* \leftarrow \mathrm{SIM}(\mathrm{PK}, \{\mathrm{SK}_{f_i}, f_i(m)\}, 1^{|m|})$

Ciphertext

$\mathrm{CT} = \mathbf{Encrypt}(\mathrm{PK}, m)$

Secret Key

$\mathrm{SK}_f = \mathbf{KeyGen}(\mathrm{MSK}, f)$



SK_{f_1}
SK_{f_2}
$\vdots$
SK_{f_Q}

$f_1(m)$
$f_2(m)$
$\vdots$
$f_Q(m)$

$\mathbf{Decrypt}(\mathrm{CT}, \mathrm{SK}_f) = f(m)$

Functional Encryption [SW05, BSW11, O'N10]

Setup(1^λ) : (PK, MSK)

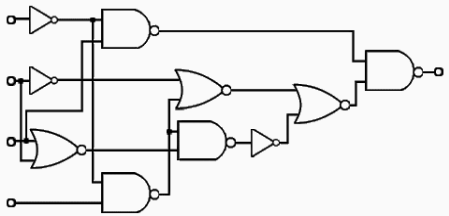
$\mathrm{CT} \leftarrow \mathbf{Encrypt}(\mathrm{PK}, m)$
$\approx$
$\mathrm{CT}^* \leftarrow \mathrm{SIM}(\mathrm{PK}, \{\mathrm{SK}_{f_i}, f_i(m)\}, 1^{|m|})$

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)



SK_{f_1}
SK_{f_2}
$\vdots$
SK_{f_Q}

$f_1(m)$
$f_2(m)$
$\vdots$
$f_Q(m)$

Q = unbounded poly
Collusion resistant

Decrypt(CT, SK_f) = $f(m)$

Functional Encryption [SW05, BSW11, O'N10]

$\mathbf{Setup}(1^\lambda) : (\mathrm{PK}, \mathrm{MSK})$

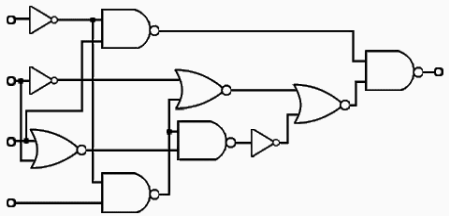
$\mathrm{CT} \leftarrow \mathbf{Encrypt}(\mathrm{PK}, m)$
$\approx$
$\mathrm{CT}^* \leftarrow \mathrm{SIM}(\mathrm{PK}, \{\mathrm{SK}_{f_i}, f_i(m)\}, 1^{|m|})$

Ciphertext

$\mathrm{CT} = \mathbf{Encrypt}(\mathrm{PK}, m)$

Secret Key

$\mathrm{SK}_f = \mathbf{KeyGen}(\mathrm{MSK}, f)$



SK_{f_1}
SK_{f_2}
$\vdots$
SK_{f_Q}

$f_1(m)$
$f_2(m)$
$\vdots$
$f_Q(m)$

$\mathbf{Decrypt}(\mathrm{CT}, \mathrm{SK}_f) = f(m)$

Q = unbounded poly
Collusion resistant

Full SIM security impossible [BSW11, AGVW13]

Functional Encryption [SW05, BSW11, O'N10]

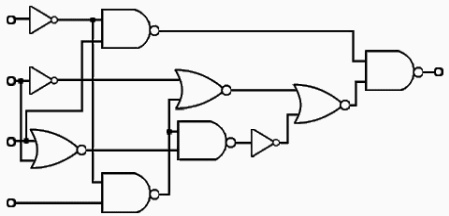
Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)



SK_{f_1}
SK_{f_2}
$\vdots$
SK_{f_Q}

$f_1(m)$
$f_2(m)$
$\vdots$
$f_Q(m)$

[Agr19, AJL+19, JLMS19, GJLS20, JLS20]

Decrypt(CT, SK_f) = $f(m)$

Q = unbounded poly
Collusion resistant

IND security

Functional Encryption [SW05, BSW11, O'N10]

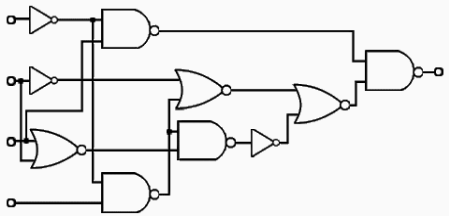
Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)



SK_{f_1} SK_{f_2} ⋮ SK_{f_Q} → $f_1(m)$ $f_2(m)$ ⋮ $f_Q(m)$

Q = unbounded poly

Collusion resistant

Decrypt(CT, SK_f) = $f(m)$

IND security

Functional Encryption [SW05, BSW11, O'N10]

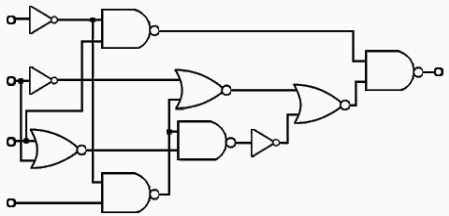
Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)



SK_{f_1}
SK_{f_2}
$\vdots$
SK_{f_Q}

$f_1(m)$
$f_2(m)$
$\vdots$
$f_Q(m)$

Q = unbounded poly
Collusion resistant

Decrypt(CT, SK_f) = $f(m)$

Complex constructions
Mixed assumptions

Cryptomania

$\vdots$

[Agr19, AJL+19, JLMS19, GJLS20, JLS20]

[AJ15, BV15, AJS15, BNP+16, LV16, Lin17, LT17, KS17, KNT18...]

Obfustopia

IND security

Functional Encryption [SW05, BSW11, O'N10]

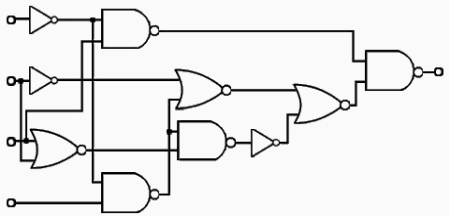
Setup(1^λ) : (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)



SK_{f_1}
SK_{f_2}
$\vdots$
SK_{f_Q}

$f_1(m)$
$f_2(m)$
$\vdots$
$f_Q(m)$

Q = *bounded* poly
Bounded collusion

SIM security *possible*: [GVW12, AR17, Agr17, AV19]

Decrypt(CT, SK_f) = $f(m)$

Functional Encryption [SW05, BSW11, O'N10]

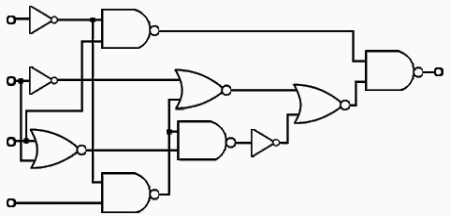
Setup$(1^\lambda, 1^Q)$: (PK, MSK)

Ciphertext

CT = **Encrypt**(PK, m)

Secret Key

SK_f = **KeyGen**(MSK, f)



SK_{f_1}
SK_{f_2}
$\vdots$
SK_{f_Q}

$f_1(m)$
$f_2(m)$
$\vdots$
$f_Q(m)$

Q = *bounded* poly
Bounded collusion

SIM security *possible*: [GVW12, AR17, Agr17, AV19]

Decrypt(CT, SK_f) = $f(m)$

Limitations of Prior Work

1

- *Bounded collusion* model
  - Prior work: [GVW12, AR16, Agr17, AV19]

Limitations of Prior Work

1

- *Bounded collusion* model
  - Prior work: [GVW12, AR16, Agr17, AV19]
  - Q: *fixed* at setup
  - *Inefficient*: Keys (& CT) *grow* with Q

Limitations of Prior Work

1

- *Bounded collusion* model
  - Prior work: [GVW12, AR16, Agr17, AV19]
  - Q: *fixed* at setup
  - *Inefficient*: Keys (& CT) *grow* with Q
  - *Same collusion-tolerance for all* CT

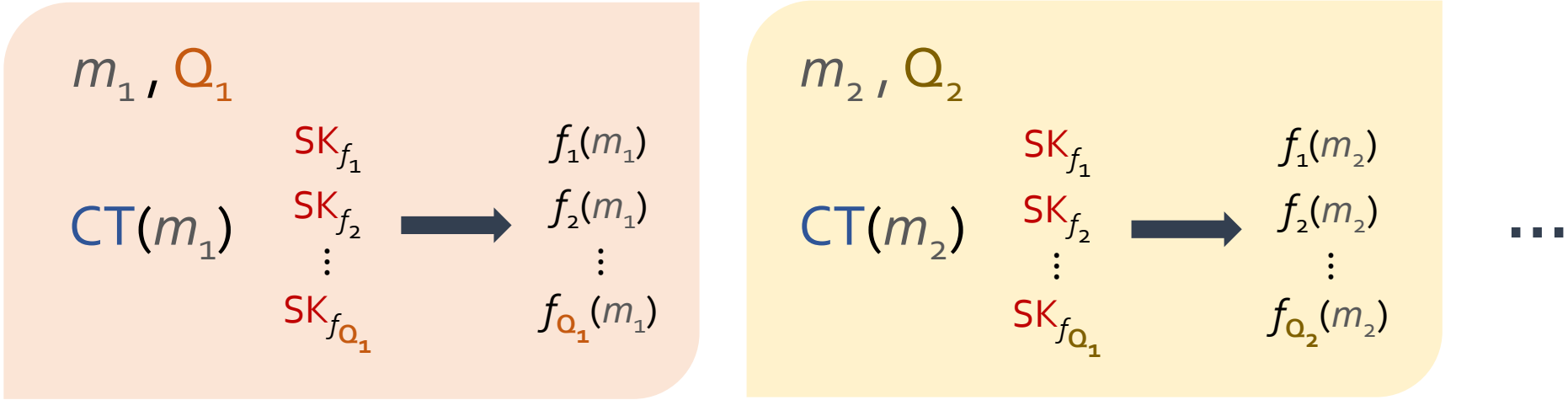
Limitations of Prior Work

1

- *Bounded collusion* model
  - Prior work: [GVW12, AR16, Agr17, AV19]
  - Q: *fixed* at setup
  - *Inefficient*: Keys (& CT) *grow* with Q
  - *Same collusion-tolerance for all* CT

Our Results

- *Dynamic bounded collusion* (*DBC*) model
  - *Stronger* model*: encryptor fixes* Q per CT



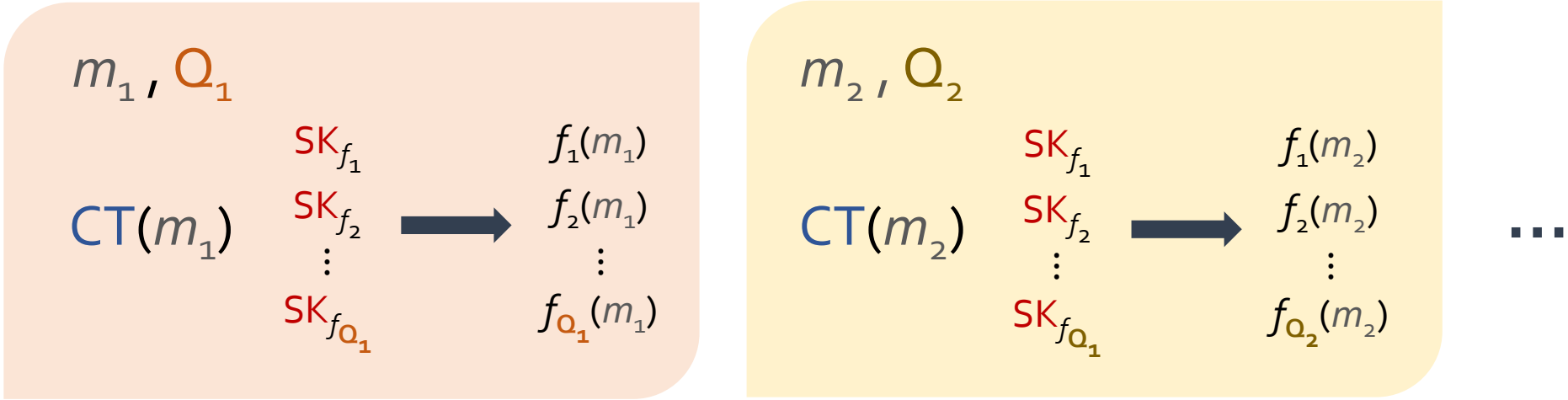
Limitations of Prior Work

1

- *Bounded collusion* model
  - Prior work: [GVW12, AR16, Agr17, AV19]
  - Q: *fixed* at setup
  - *Inefficient*: Keys (& CT) *grow* with Q
  - *Same collusion-tolerance for all* CT

Our Results

- *Dynamic bounded collusion* (*DBC*) model
  - *Stronger* model*: encryptor fixes* Q per CT
  - Time(**SetUp**, **KeyGen**) *independent* of Q
  - |CT| *grows linearly* with *choice* of Q



Limitations of Prior Work

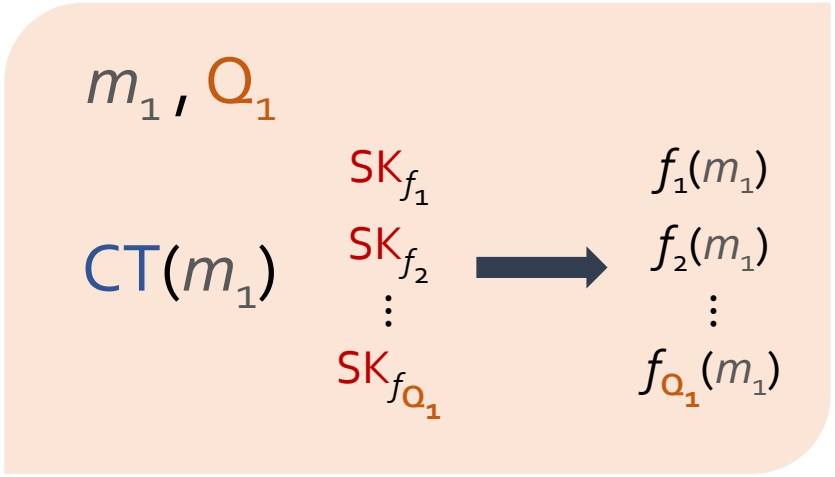
1

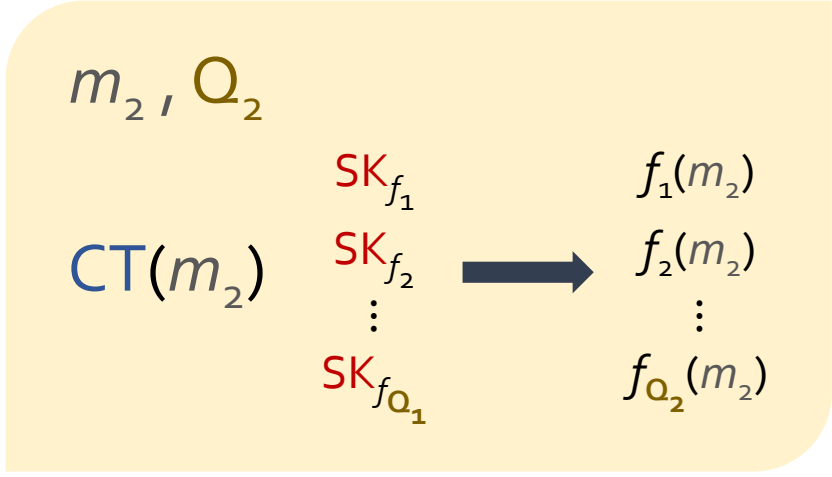
- *Bounded collusion* model
  - Prior work: [GVW12, AR16, Agr17, AV19]
  - Q: *fixed* at setup
  - *Inefficient*: Keys (& CT) *grow* with Q
  - *Same collusion-tolerance for all* CT

Our Results

- *Dynamic bounded collusion* (*DBC*) model
  - *Stronger* model*: encryptor fixes* Q per CT
  - Time(**SetUp**, **KeyGen**) *independent* of Q
  - |CT| *grows linearly* with *choice* of Q
  - From *IBE*

Based on IBE



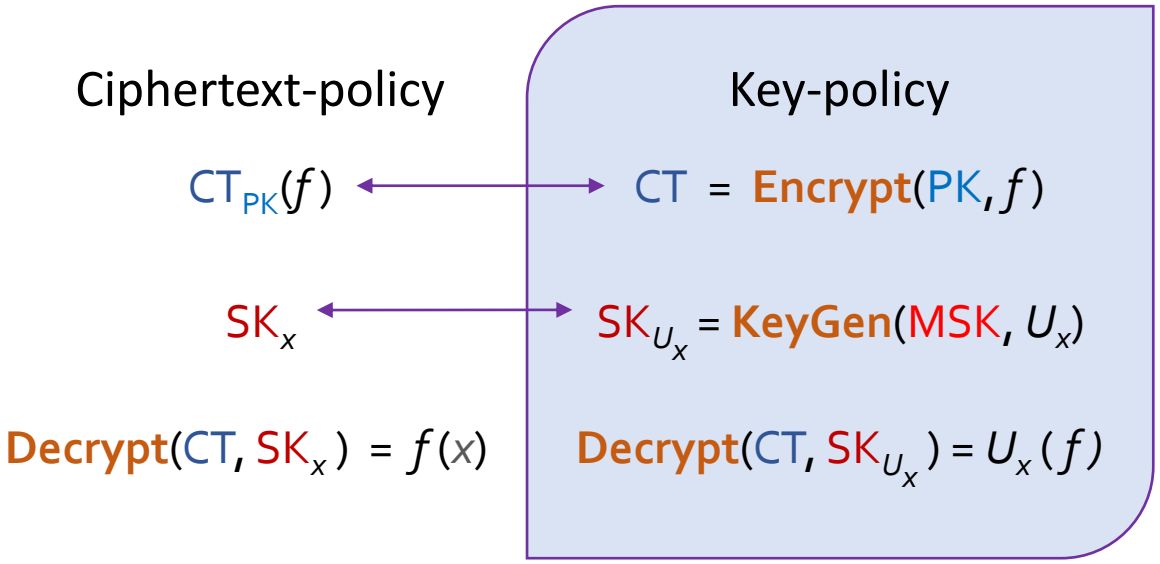


...

Limitations of Prior Work

2

- All prior work build *key-policy FE*
  - Build CPFE via *universal circuits*: $U_m(.)$
  - *Inefficient*, supports *bounded size* circuits

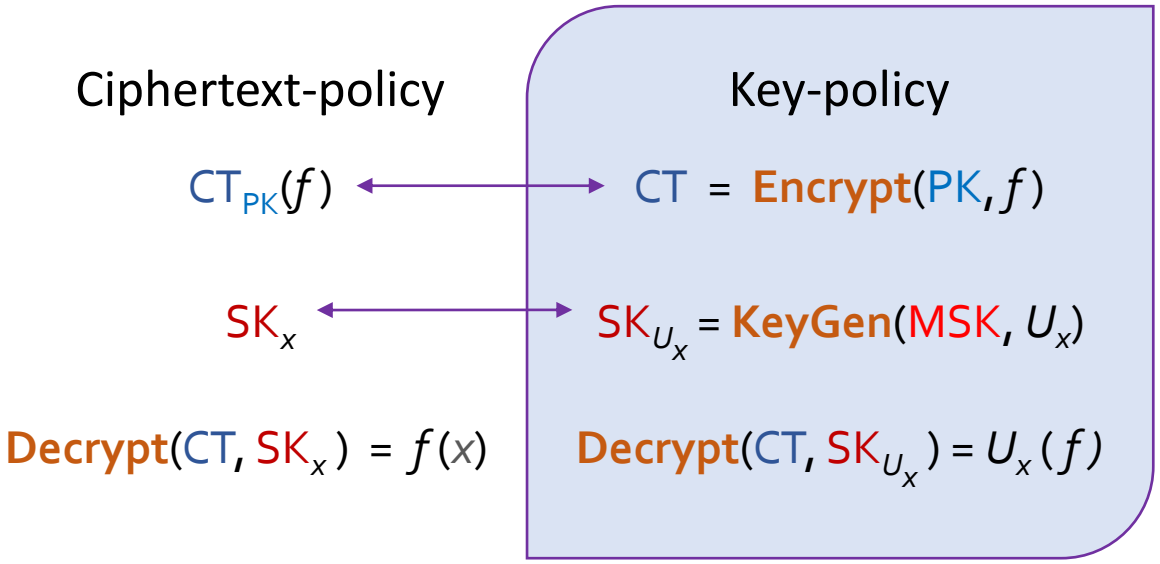


Our Results

Limitations of Prior Work

- All prior work build *key-policy FE*
  - Build CPFE via *universal circuits*: $U_m(.)$
  - *Inefficient*, supports *bounded size* circuits
  - *Exception*: [SS10] - *1-key* secure

2



Our Results

Limitations of Prior Work

- All prior work build *key-policy FE*
  - Build CPFE via *universal circuits*: $U_m(.)$
  - *Inefficient*, supports *bounded size* circuits
  - *Exception*: [SS10] - *1-key* secure

2

Our Results

- *DBC* CPFE with *various tradeoffs* -

Limitations of Prior Work

- All prior work build *key-policy FE*
  - Build CPFE via *universal circuits*: $U_m(.)$
  - *Inefficient*, supports *bounded size* circuits
  - *Exception*: [SS10] - *1-key* secure

Our Results

- *DBC* CPFE with *various tradeoffs* -
  - (*IND-CPA*/SIM-RSO) *IBE* ⇒ (*NA* / *AD*)-*SIM*, (*Unbounded* / *Bounded*) size circuits

Limitations of Prior Work

- All prior work build *key-policy FE*
  - Build CPFE via *universal circuits*: $U_m(.)$
  - *Inefficient*, supports *bounded size* circuits
  - *Exception*: [SS10] - *1-key* secure

2

Our Results

- *DBC* CPFE with *various tradeoffs* -
  - (*IND-CPA*/SIM-RSO) *IBE* ⇒ (*NA* / *AD*)-*SIM*, (*Unbounded* / *Bounded*) size circuits

IND-CPA ⇒ SIM-RSO [KT18]

Limitations of Prior Work

- All prior work build *key-policy FE*
  - Build CPFE via *universal circuits*: $U_m(.)$
  - *Inefficient*, supports *bounded size* circuits
  - *Exception*: [SS10] - *1-key* secure

Our Results

- *DBC* CPFE with *various tradeoffs* -
  - (*IND-CPA*/SIM-RSO) *IBE* ⇒ (*NA* / *AD*)-*SIM*, (*Unbounded* / *Bounded*) size circuits
  - IBE: *necessary* for DBC

Limitations of Prior Work

- All prior work build *key-policy FE*
  - Build CPFE via *universal circuits*: $U_m(.)$
  - *Inefficient*, supports *bounded size* circuits
  - *Exception*: [SS10] - *1-key* secure

Our Results

- *DBC* CPFE with *various tradeoffs* -
  - (*IND-CPA*/SIM-RSO) *IBE* ⇒ (*NA* / *AD*)-*SIM*, (*Unbounded* / *Bounded*) size circuits
  - IBE: *necessary* for DBC
- *DBC*, *AD-SIM*, *succinct* CP/KP-FE from *LWE*

Unbounded size, bounded depth & output circuits

Limitations of Prior Work

2

- All prior work build *key-policy FE*
  - Build CPFE via *universal circuits*: $U_m(.)$
  - *Inefficient*, supports *bounded size* circuits
  - *Exception*: [SS10] - *1-key* secure

Our Results

- *DBC* CPFE with *various tradeoffs* -
  - (*IND-CPA*/SIM-RSO) *IBE* ⇒ (*NA* / *AD*)-*SIM*, (*Unbounded* / *Bounded*) size circuits
  - IBE: *necessary* for DBC
- *DBC*, *AD-SIM*, *succinct* CP/KP-FE from *LWE*

Stronger security for [GKP+13b]

Limitations of Prior Work

- Model of computation

3

Our Results

Limitations of Prior Work

Our Results

- Model of computation (*Uniform*: TM,FA)

Circuits have *fixed* input sizes
Incurs *worst-case* runtime

3

Limitations of Prior Work

- Model of computation (*Uniform*: TM,FA)
  - *PK*-FE [GKP+13a]: *non-standard* assumption
  - *PK*-FE [AS17]: *1-key*, *LWE*
  - *SK*-FE for *FA* [AMY19]: *LWE*

3

Our Results

Limitations of Prior Work

- Model of computation (*Uniform*: TM,FA)
  - *PK*-FE [GKP+13a]: *non-standard* assumption
  - *PK*-FE [AS17]: *1-key*, *LWE*
  - *SK*-FE for *FA* [AMY19]: *LWE*
  - *Bounded collusion*

3

Our Results

Limitations of Prior Work

- Model of computation (*Uniform*: TM,FA)
  - *PK*-FE [GKP+13a]: *non-standard* assumption
  - *PK*-FE [AS17]: *1-key*, *LWE*
  - *SK*-FE for *FA* [AMY19]: *LWE*
  - *Bounded collusion*

3

Our Results

- *PK*-FE for *TM*/*NL* from *LWE*

Limitations of Prior Work

3

- Model of computation (*Uniform*: TM,FA)
  - *PK*-FE [GKP+13a]: *non-standard* assumption
  - *PK*-FE [AS17]: *1-key*, *LWE*
  - *SK*-FE for *FA* [AMY19]: *LWE*
  - *Bounded collusion*

Our Results

- *PK*-FE for *TM*/*NL* from *LWE*
  - Both *KP*/*CP*, satisfies *DBC*
  - *TM*: *NA-SIM*, *NL*: *AD-SIM* → [AMY19, AS17]

Limitations of Prior Work

3

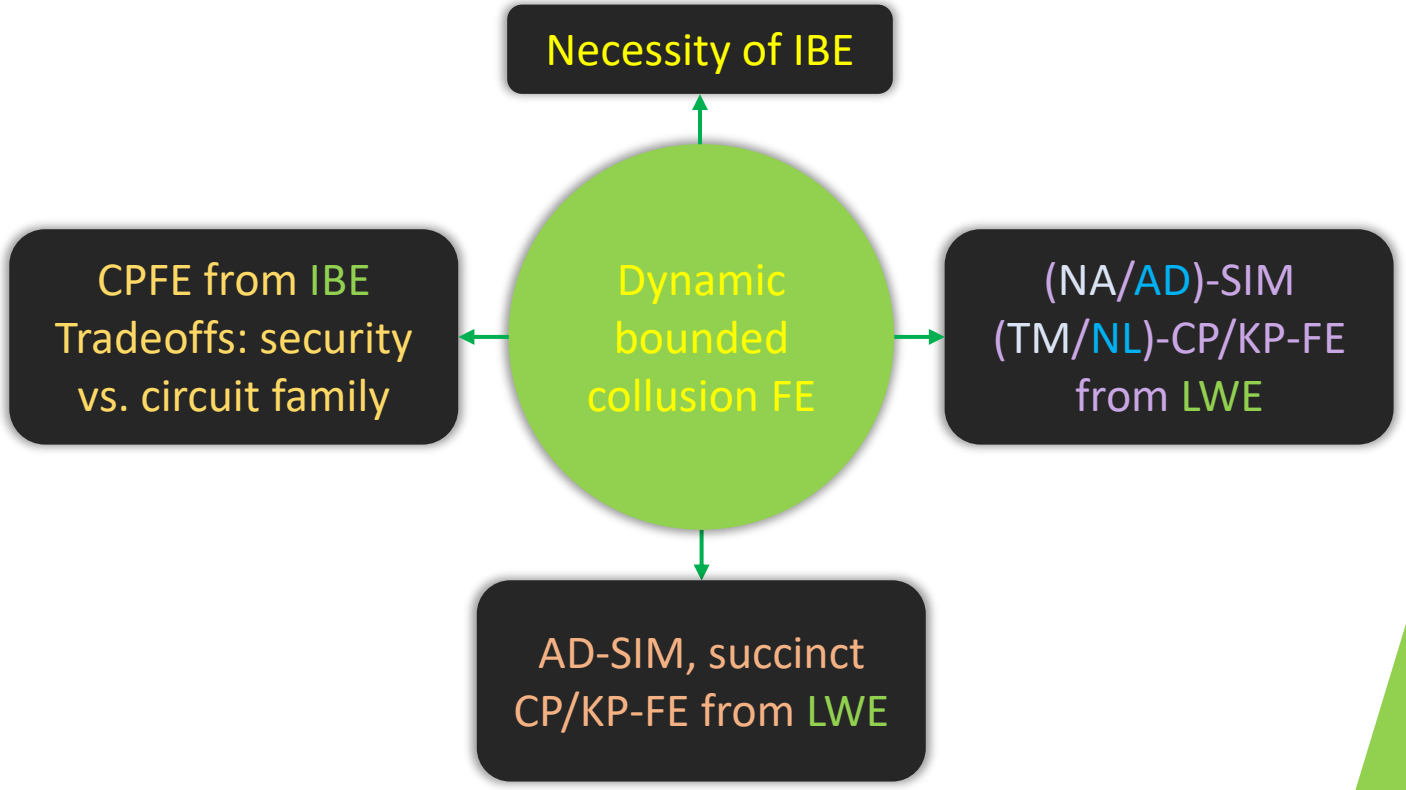
- Model of computation (*Uniform*: TM,FA)
  - *PK*-FE [GKP+13a]: *non-standard* assumption
  - *PK*-FE [AS17]: *1-key*, *LWE*
  - *SK*-FE for *FA* [AMY19]: *LWE*
  - *Bounded collusion*

Our Results

- *PK*-FE for *TM*/*NL* from *LWE*
  - Both *KP*/*CP*, satisfies *DBC*
  - *TM*: *NA-SIM*, *NL*: *AD-SIM* → [AMY19, AS17]
  - |CT| *grows* with runtime t

    t is *not* a global bound, can *vary per input*.

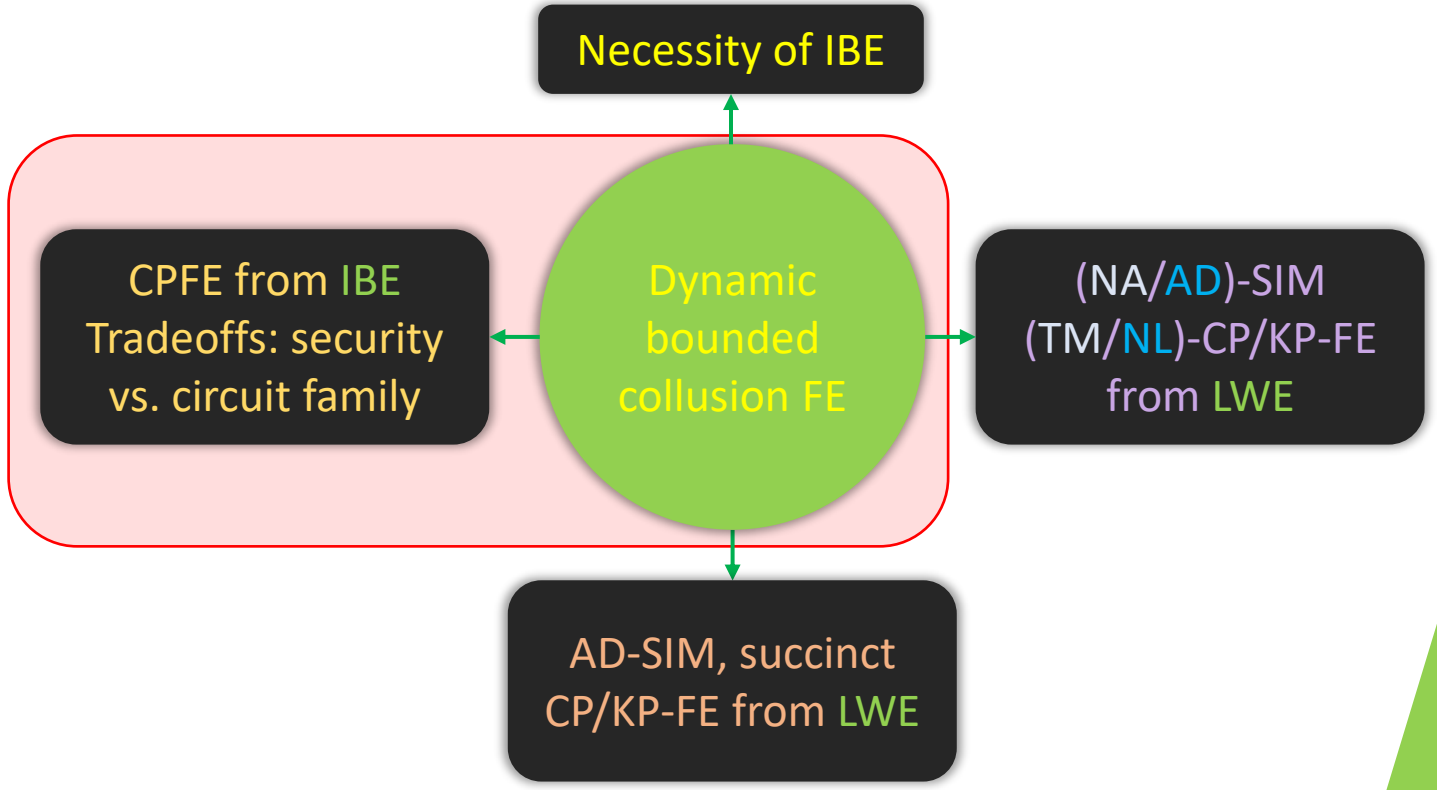
Summary of Results



Summary of Results

Concurrent Work:

- [GGLW21] also introduced DBC
- *Similar techniques*, but for KPFE:
  - IBE + existing KPFE [GVW12, AV19]

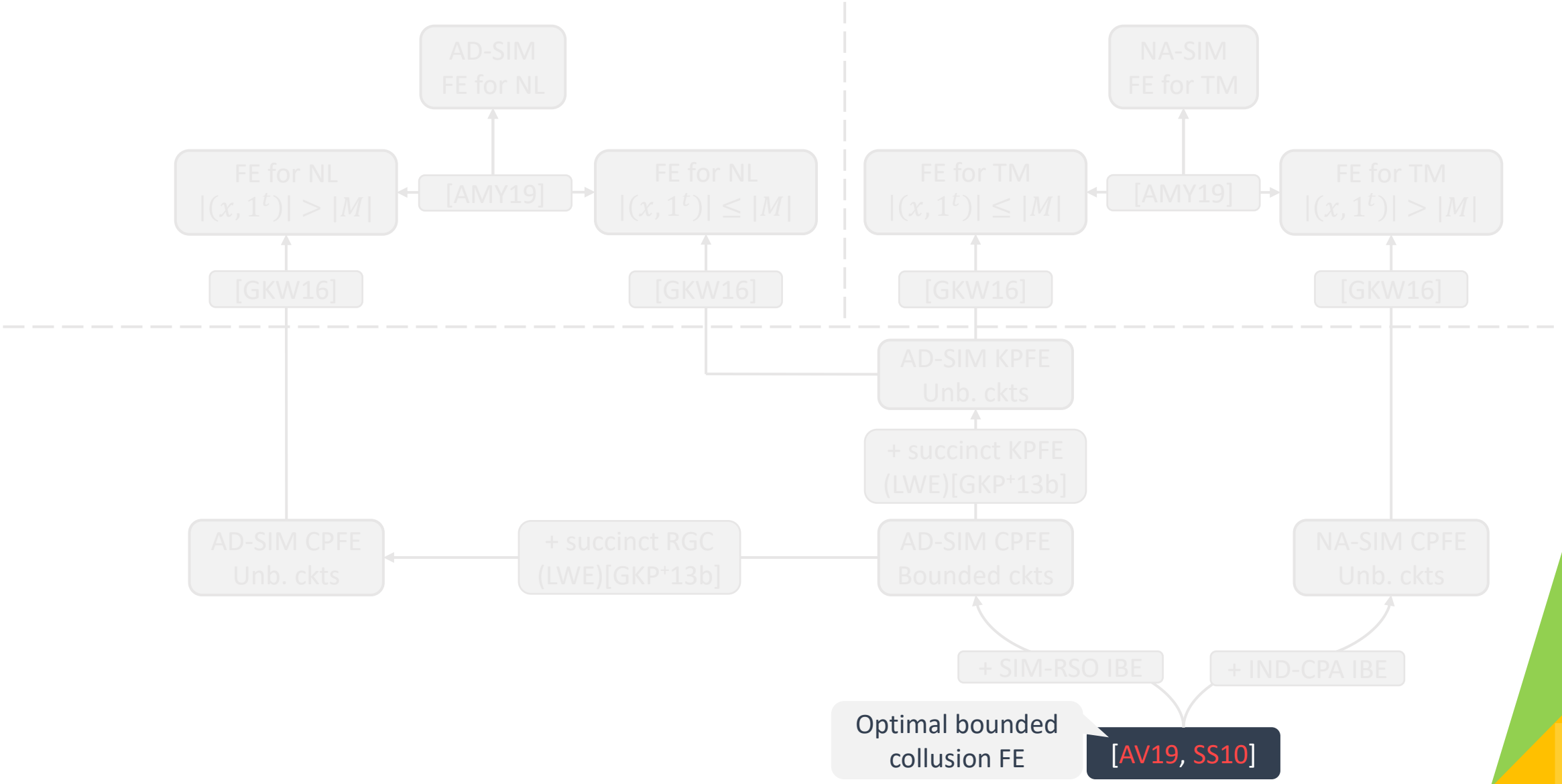


Techniques

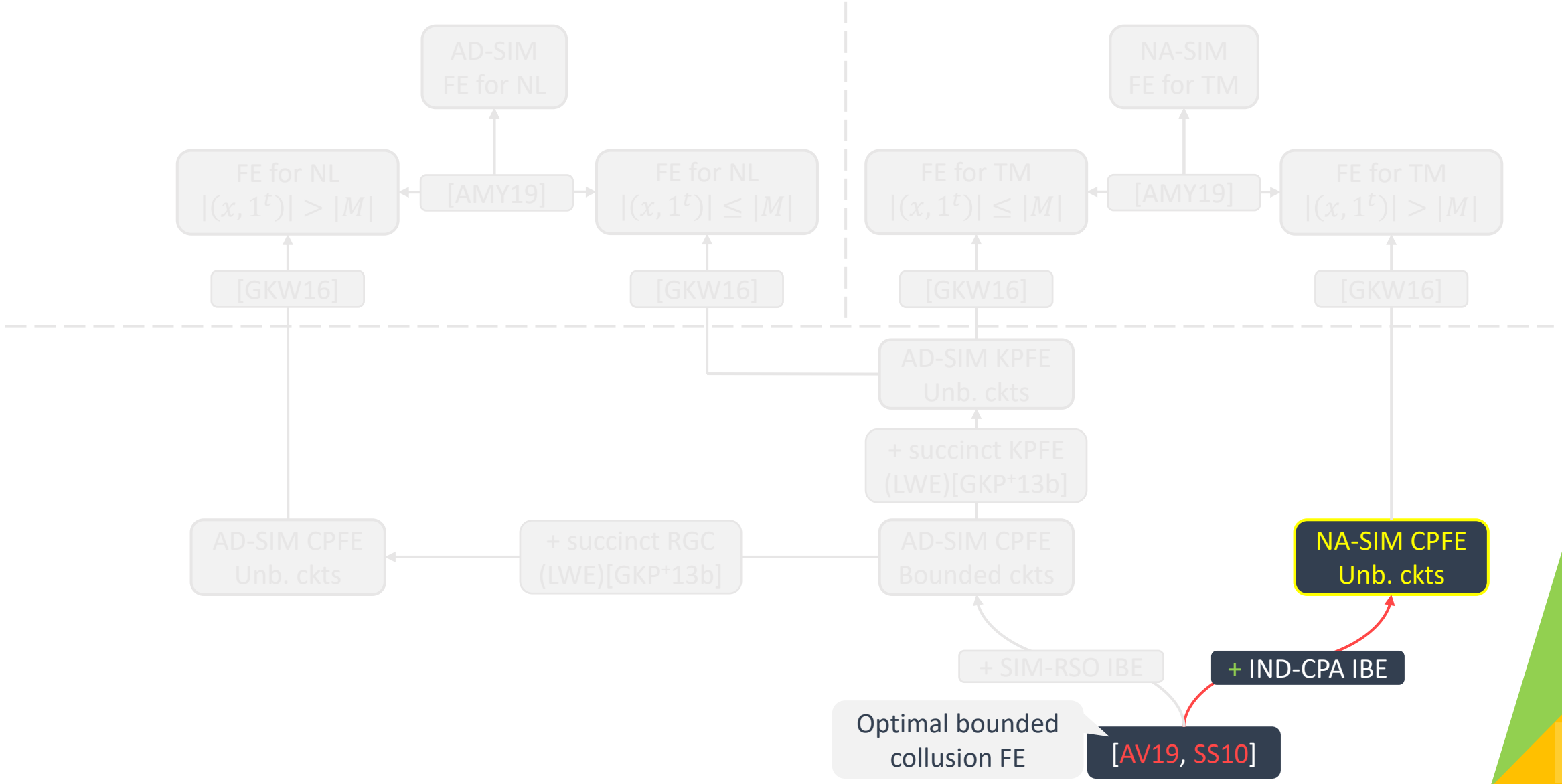
Roadmap

Optimal bounded collusion FE [AV19, SS10]

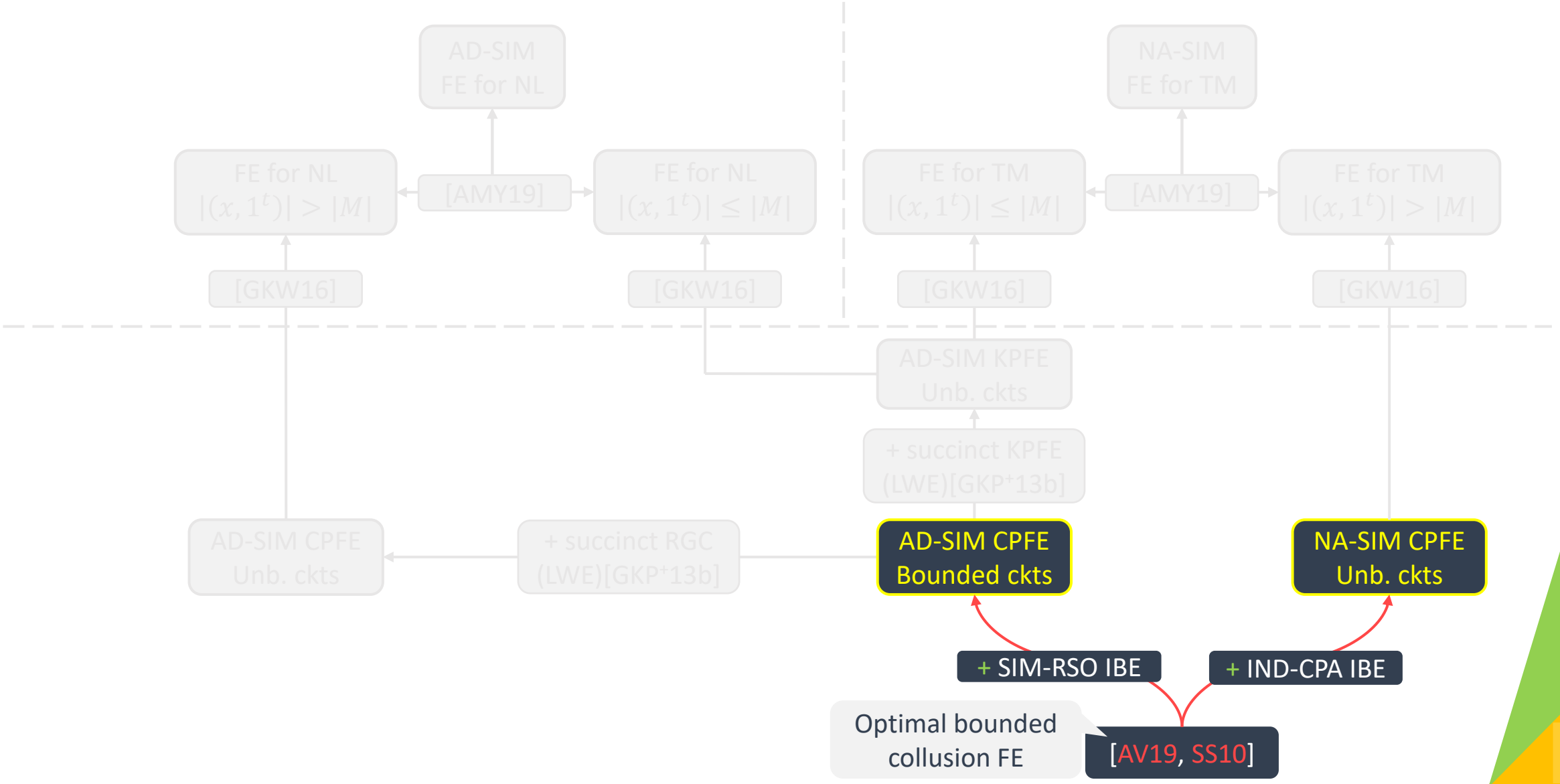
Roadmap


AD-SIM
FE for NL
NA-SIM
FE for TM
FE for NL
$|(x,1^t)| > |M|$
[AMY19]
FE for NL
$|(x,1^t)| \leq |M|$
FE for TM
$|(x,1^t)| \leq |M|$
[AMY19]
FE for TM
$|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
[GKW16]
AD-SIM KPFE
Unb. ckts
+ succinct KPFE
(LWE)[GKP+13b]
AD-SIM CPFE
Unb. ckts
+ succinct RGC
(LWE)[GKP+13b]
AD-SIM CPFE
Bounded ckts
NA-SIM CPFE
Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded
collusion FE
[AV19, SS10]

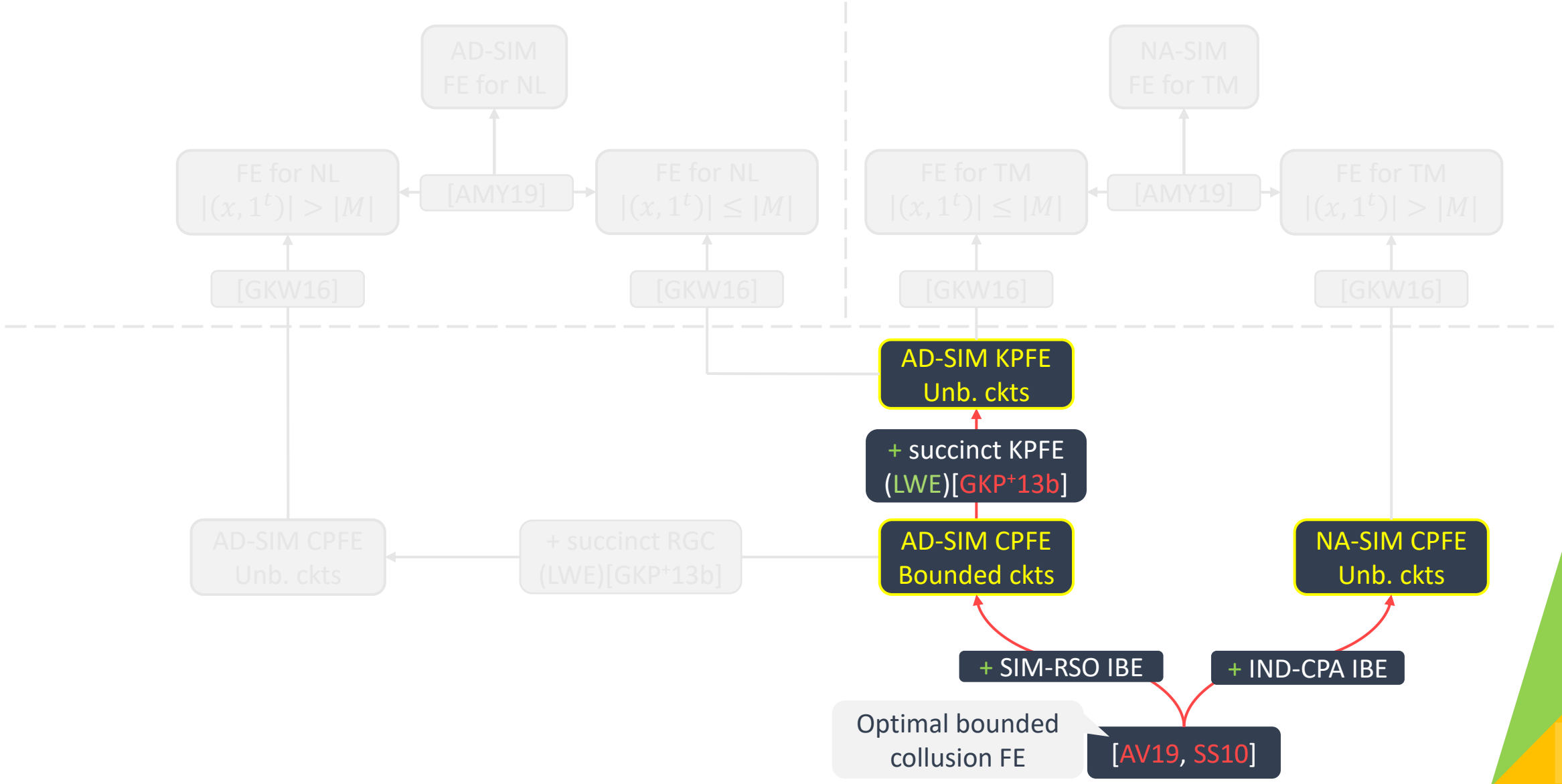
Roadmap


AD-SIM FE for NL
NA-SIM FE for TM
FE for NL $|(x,1^t)| > |M|$
[AMY19]
FE for NL $|(x,1^t)| \leq |M|$
FE for TM $|(x,1^t)| \leq |M|$
[AMY19]
FE for TM $|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
[GKW16]
AD-SIM KPFE Unb. ckts
+ succinct KPFE (LWE)[GKP+13b]
AD-SIM CPFE Unb. ckts
+ succinct RGC (LWE)[GKP+13b]
AD-SIM CPFE Bounded ckts
NA-SIM CPFE Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded collusion FE
[AV19, SS10]

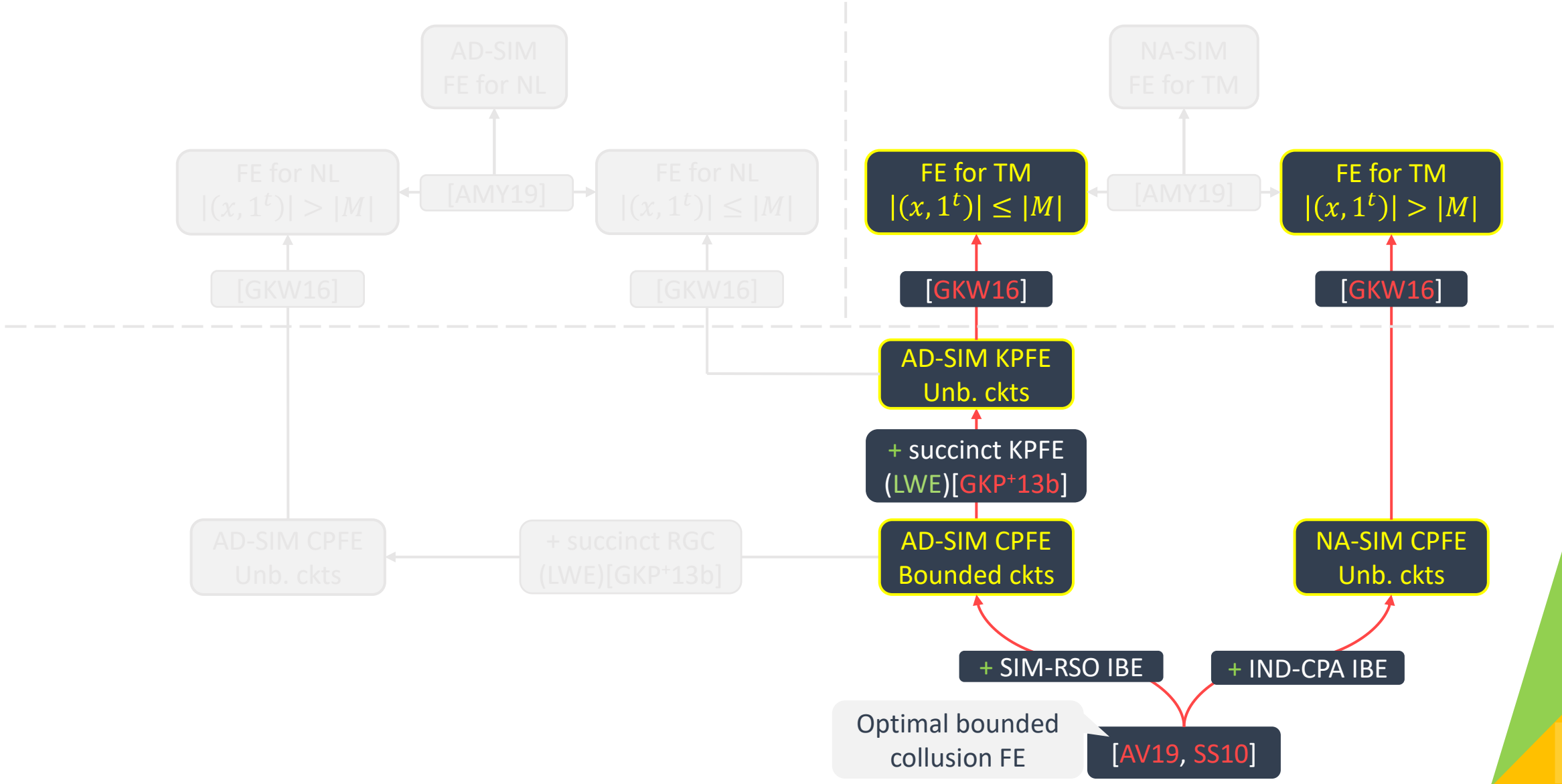
Roadmap


AD-SIM FE for NL
FE for NL $|(x,1^t)| > |M|$
[AMY19]
FE for NL $|(x,1^t)| \leq |M|$
NA-SIM FE for TM
FE for TM $|(x,1^t)| \leq |M|$
[AMY19]
FE for TM $|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
[GKW16]
AD-SIM KPFE Unb. ckts
+ succinct KPFE (LWE)[GKP+13b]
AD-SIM CPFE Unb. ckts
+ succinct RGC (LWE)[GKP+13b]
AD-SIM CPFE Bounded ckts
NA-SIM CPFE Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded collusion FE
[AV19, SS10]

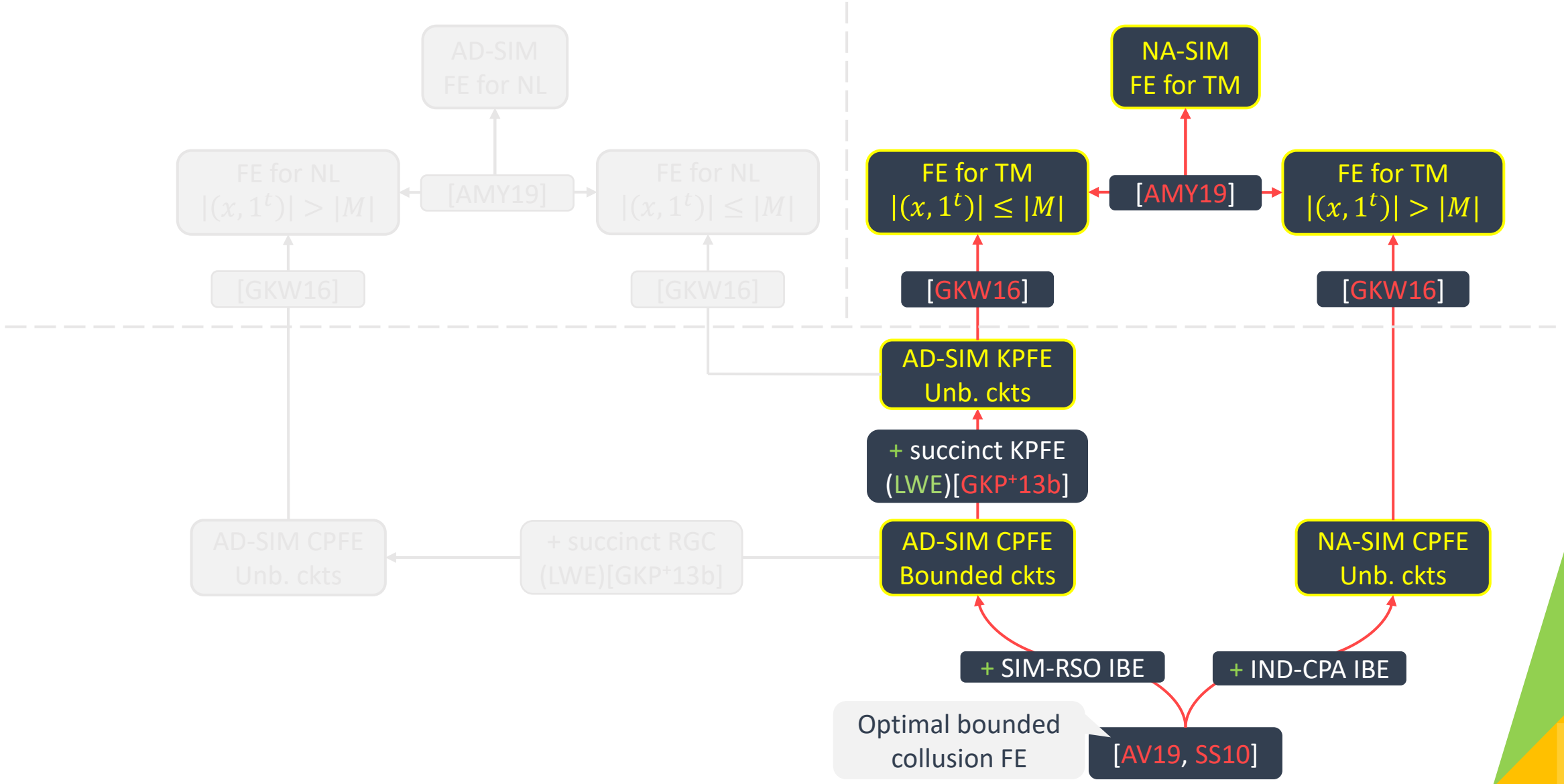
Roadmap


AD-SIM
FE for NL
NA-SIM
FE for TM
FE for NL
$|(x,1^t)| > |M|$
[AMY19]
FE for NL
$|(x,1^t)| \leq |M|$
FE for TM
$|(x,1^t)| \leq |M|$
[AMY19]
FE for TM
$|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
[GKW16]
AD-SIM KPFE
Unb. ckts
+ succinct KPFE
(LWE)[GKP+13b]
AD-SIM CPFE
Unb. ckts
+ succinct RGC
(LWE)[GKP+13b]
AD-SIM CPFE
Bounded ckts
NA-SIM CPFE
Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded
collusion FE
[AV19, SS10]

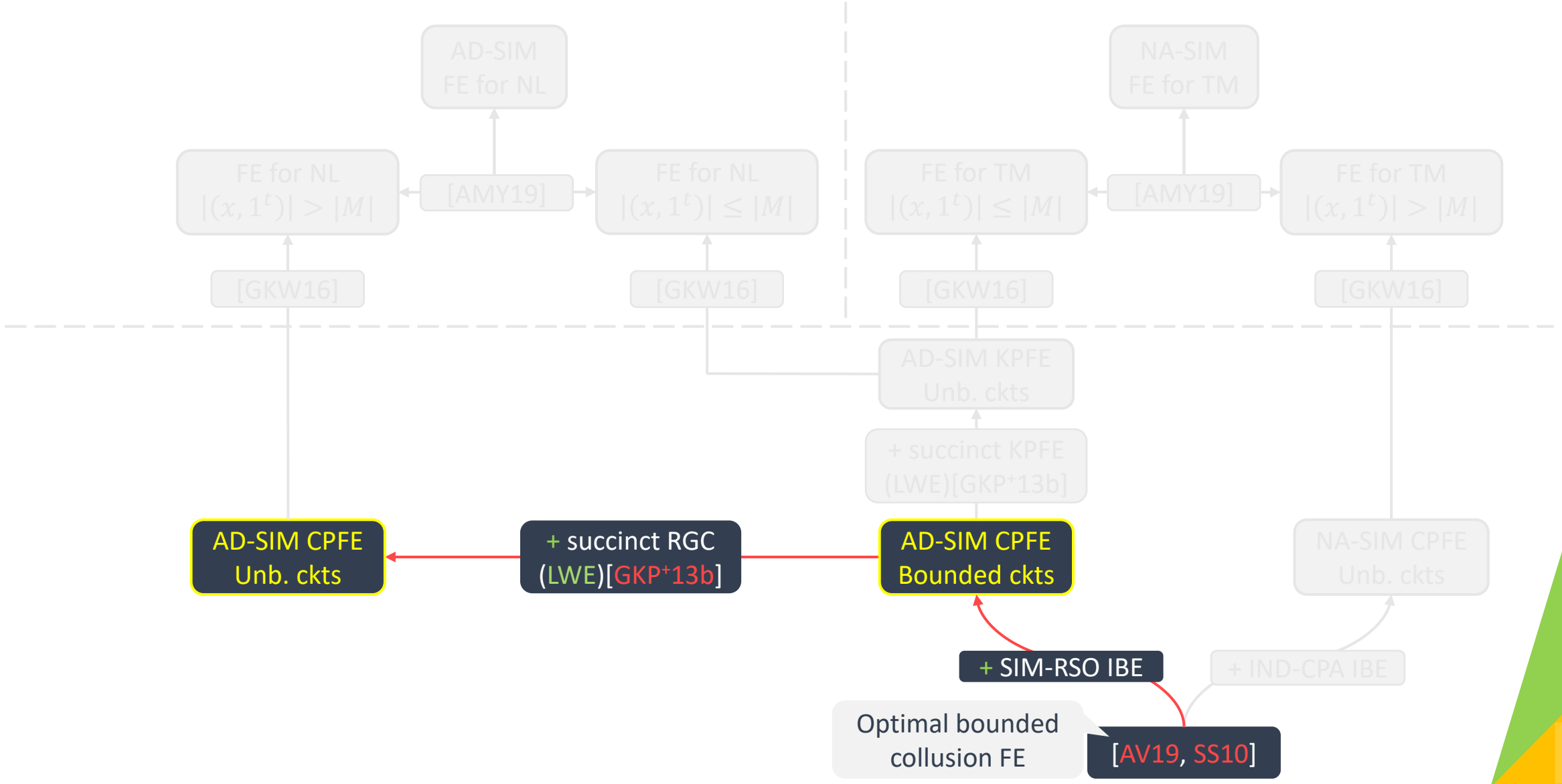
Roadmap


AD-SIM FE for NL
NA-SIM FE for TM
FE for NL $|(x,1^t)| > |M|$
[AMY19]
FE for NL $|(x,1^t)| \leq |M|$
FE for TM $|(x,1^t)| \leq |M|$
[AMY19]
FE for TM $|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
AD-SIM KPFE Unb. ckts
+ succinct KPFE (LWE)[GKP+13b]
AD-SIM CPFE Unb. ckts
+ succinct RGC (LWE)[GKP+13b]
AD-SIM CPFE Bounded ckts
NA-SIM CPFE Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded collusion FE
[AV19, SS10]

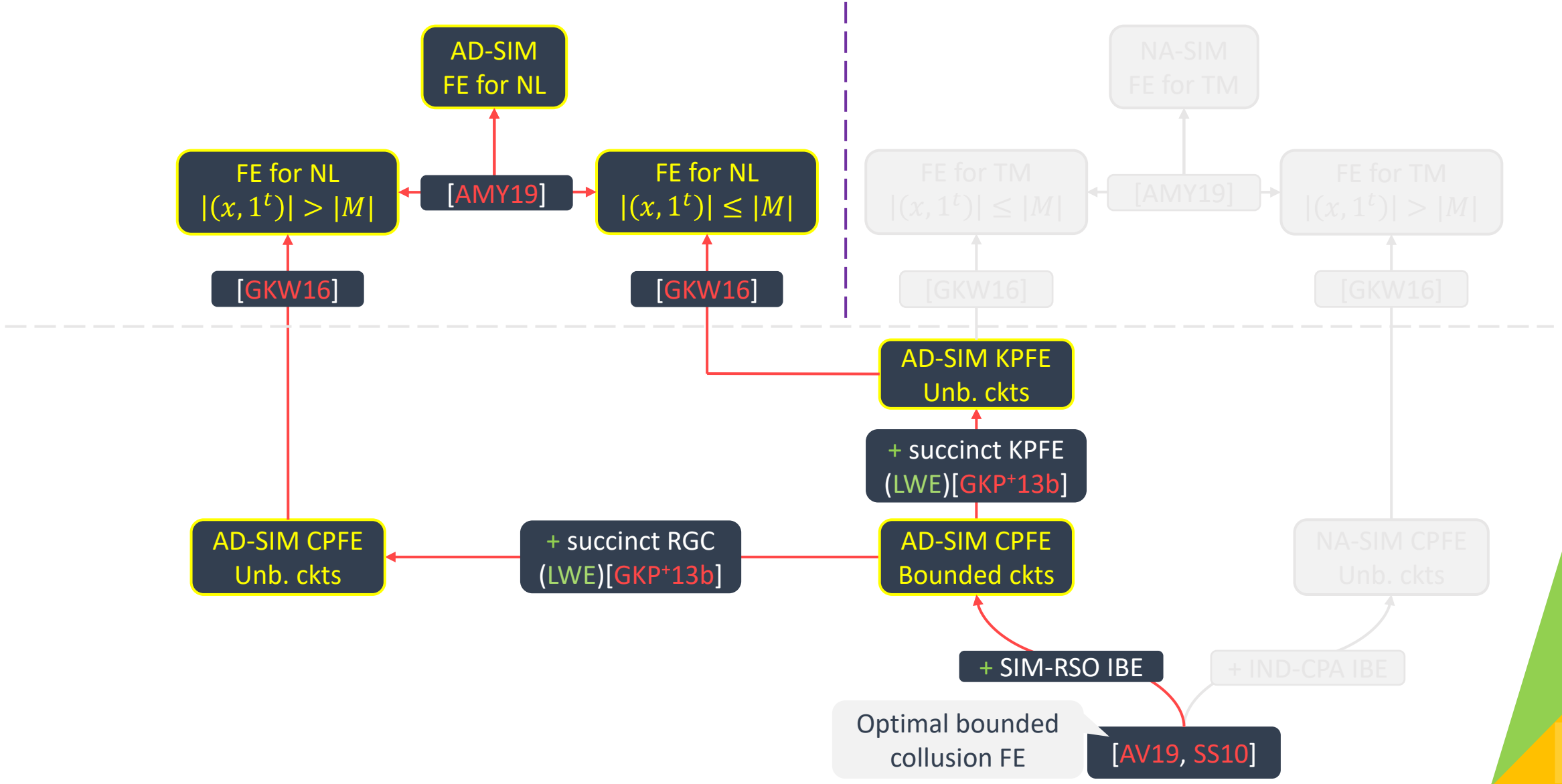
Roadmap


AD-SIM
FE for NL
FE for NL
$|(x,1^t)| > |M|$
[AMY19]
FE for NL
$|(x,1^t)| \leq |M|$
[GKW16]
[GKW16]
AD-SIM CPFE
Unb. ckts
+ succinct RGC
(LWE)[GKP+13b]
NA-SIM
FE for TM
FE for TM
$|(x,1^t)| \leq |M|$
[AMY19]
FE for TM
$|(x,1^t)| > |M|$
[GKW16]
[GKW16]
AD-SIM KPFE
Unb. ckts
+ succinct KPFE
(LWE)[GKP+13b]
AD-SIM CPFE
Bounded ckts
NA-SIM CPFE
Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded
collusion FE
[AV19, SS10]

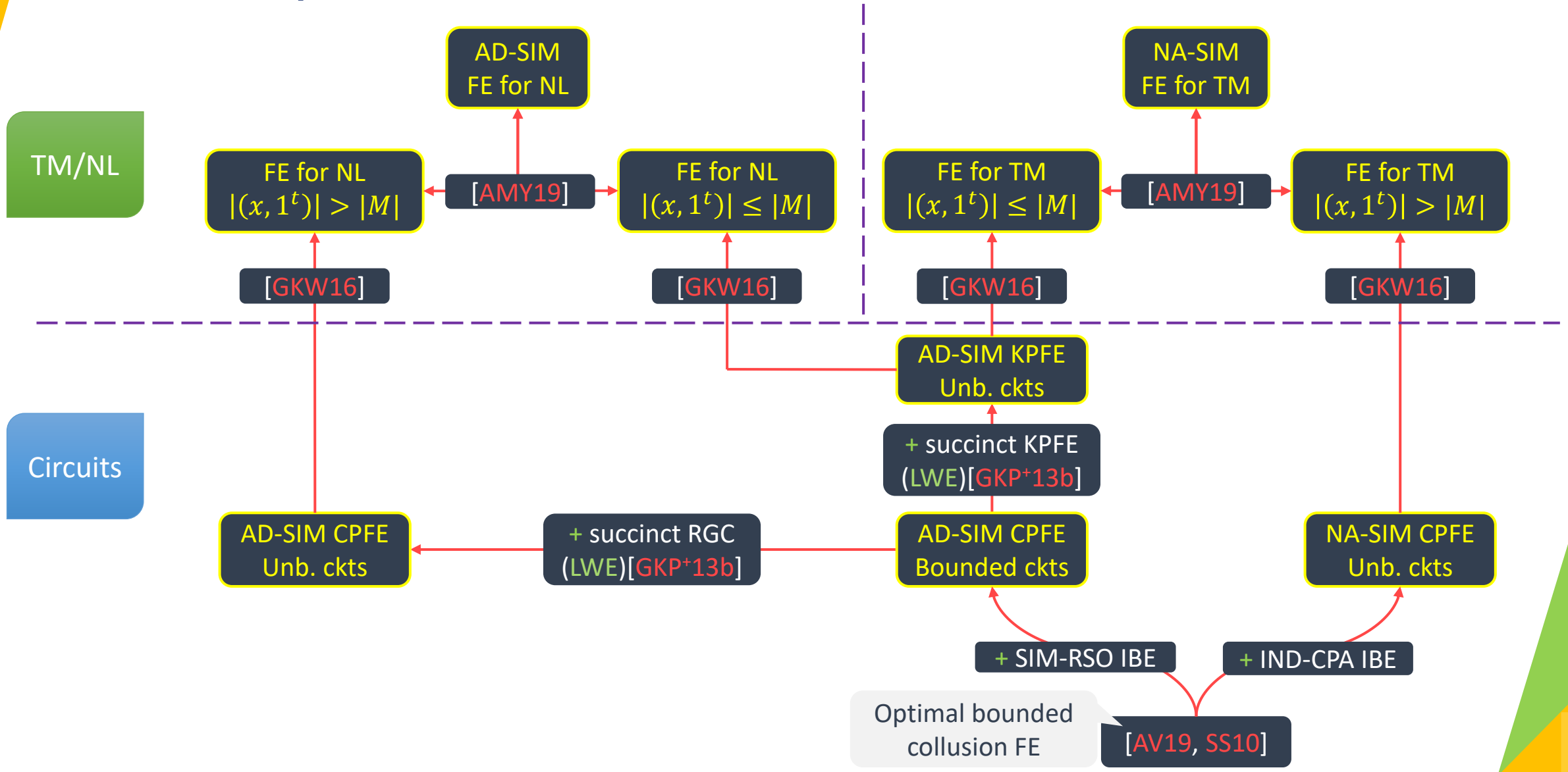
Roadmap


AD-SIM
FE for NL
NA-SIM
FE for TM
FE for NL
$|(x,1^t)| > |M|$
[AMY19]
FE for NL
$|(x,1^t)| \leq |M|$
FE for TM
$|(x,1^t)| \leq |M|$
[AMY19]
FE for TM
$|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
[GKW16]
AD-SIM KPFE
Unb. ckts
+ succinct KPFE
(LWE)[GKP+13b]
AD-SIM CPFE
Unb. ckts
+ succinct RGC
(LWE)[GKP+13b]
AD-SIM CPFE
Bounded ckts
NA-SIM CPFE
Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded
collusion FE
[AV19, SS10]

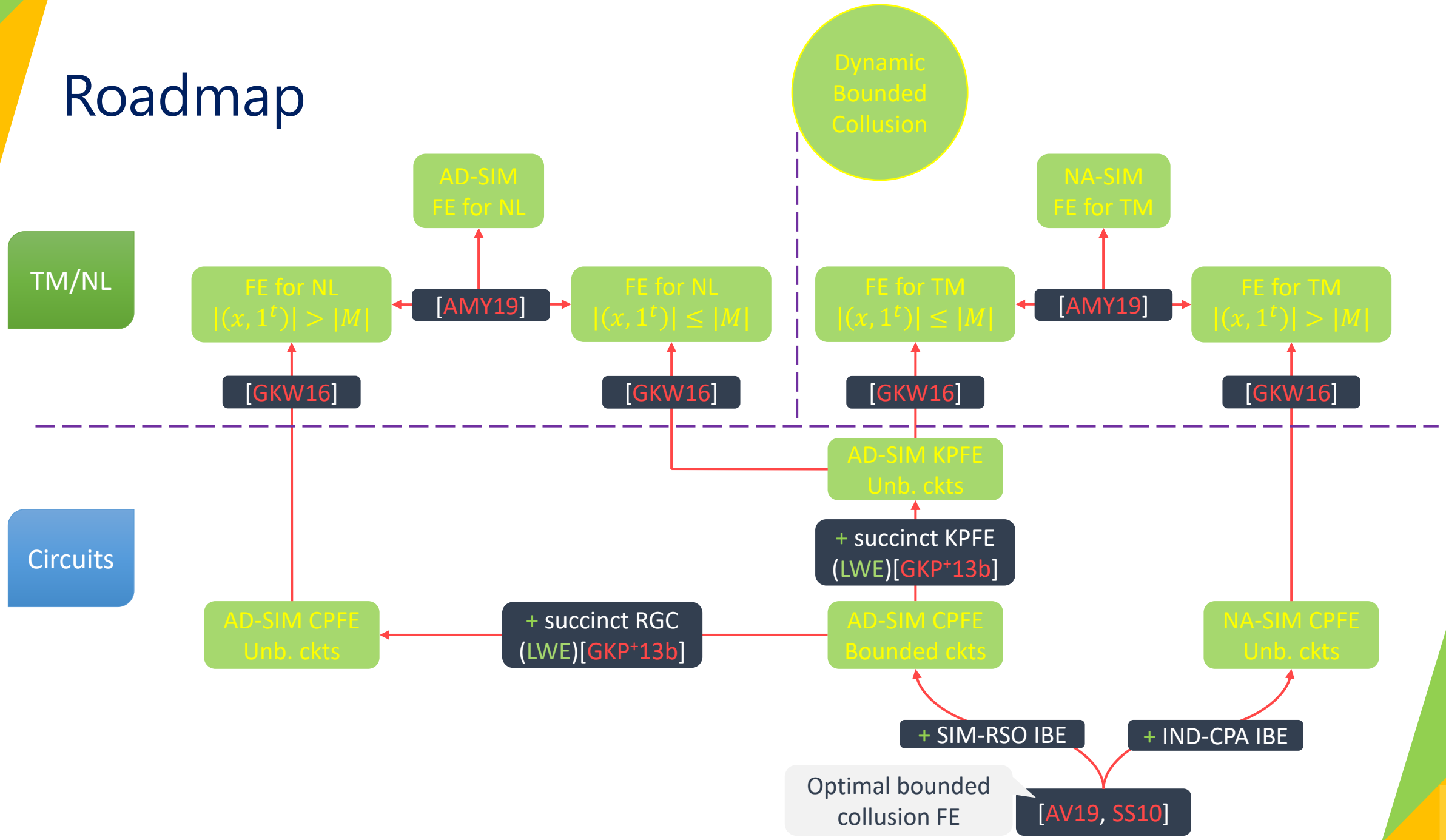
Roadmap


AD-SIM
FE for NL
FE for NL
$|(x, 1^t)| > |M|$
[AMY19]
FE for NL
$|(x, 1^t)| \leq |M|$
[GKW16]
[GKW16]
NA-SIM
FE for TM
FE for TM
$|(x, 1^t)| \leq |M|$
[AMY19]
FE for TM
$|(x, 1^t)| > |M|$
[GKW16]
[GKW16]
AD-SIM KPFE
Unb. ckts
+ succinct KPFE
(LWE)[GKP+13b]
AD-SIM CPFE
Unb. ckts
+ succinct RGC
(LWE)[GKP+13b]
AD-SIM CPFE
Bounded ckts
NA-SIM CPFE
Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded
collusion FE
[AV19, SS10]

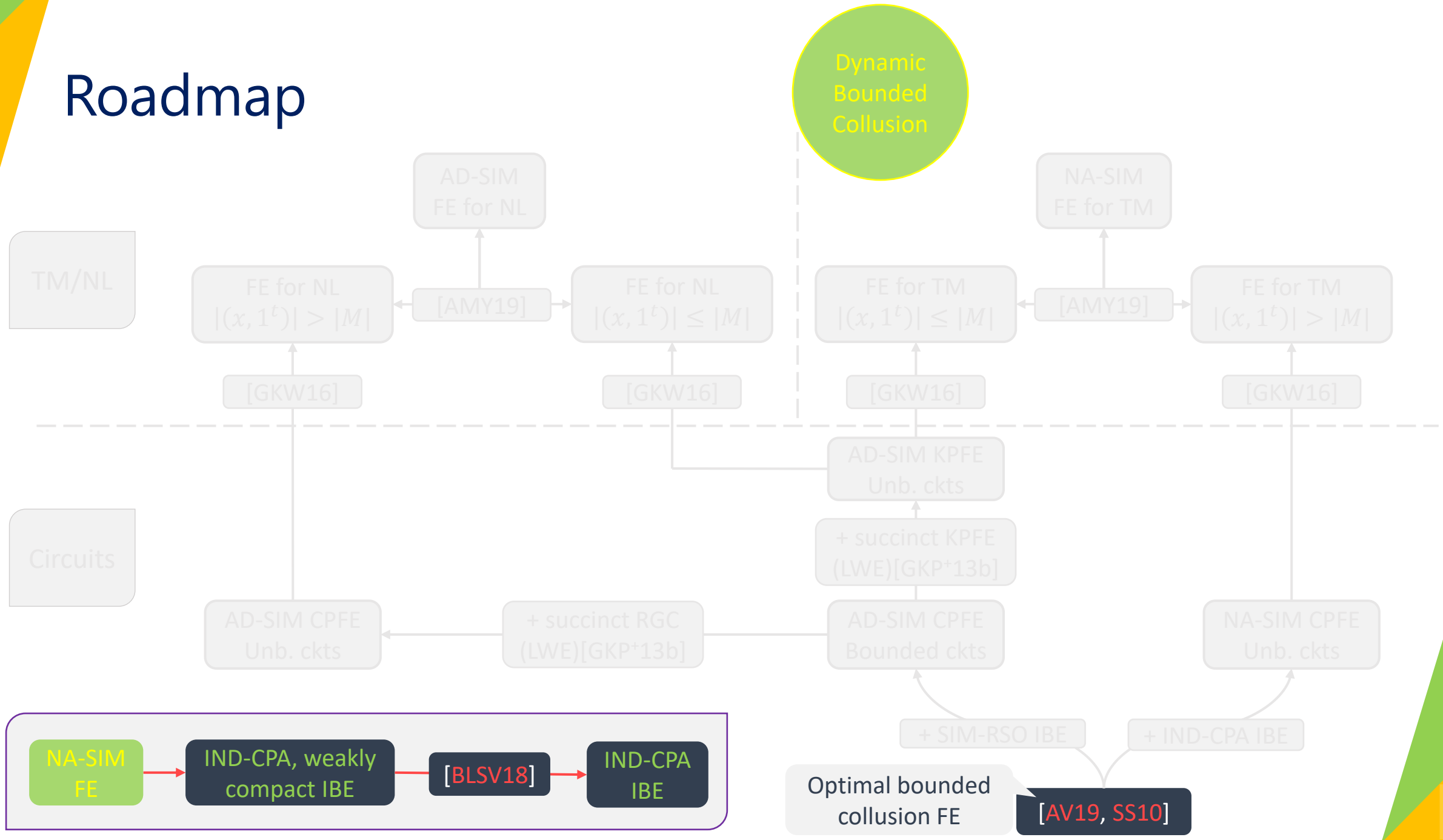
Roadmap


TM/NL
AD-SIM FE for NL
NA-SIM FE for TM
FE for NL $|(x,1^t)| > |M|$
[AMY19]
FE for NL $|(x,1^t)| \leq |M|$
FE for TM $|(x,1^t)| \leq |M|$
[AMY19]
FE for TM $|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
[GKW16]
Circuits
AD-SIM KPFE Unb. ckts
+ succinct KPFE (LWE)[GKP+13b]
AD-SIM CPFE Unb. ckts
+ succinct RGC (LWE)[GKP+13b]
AD-SIM CPFE Bounded ckts
NA-SIM CPFE Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded collusion FE
[AV19, SS10]

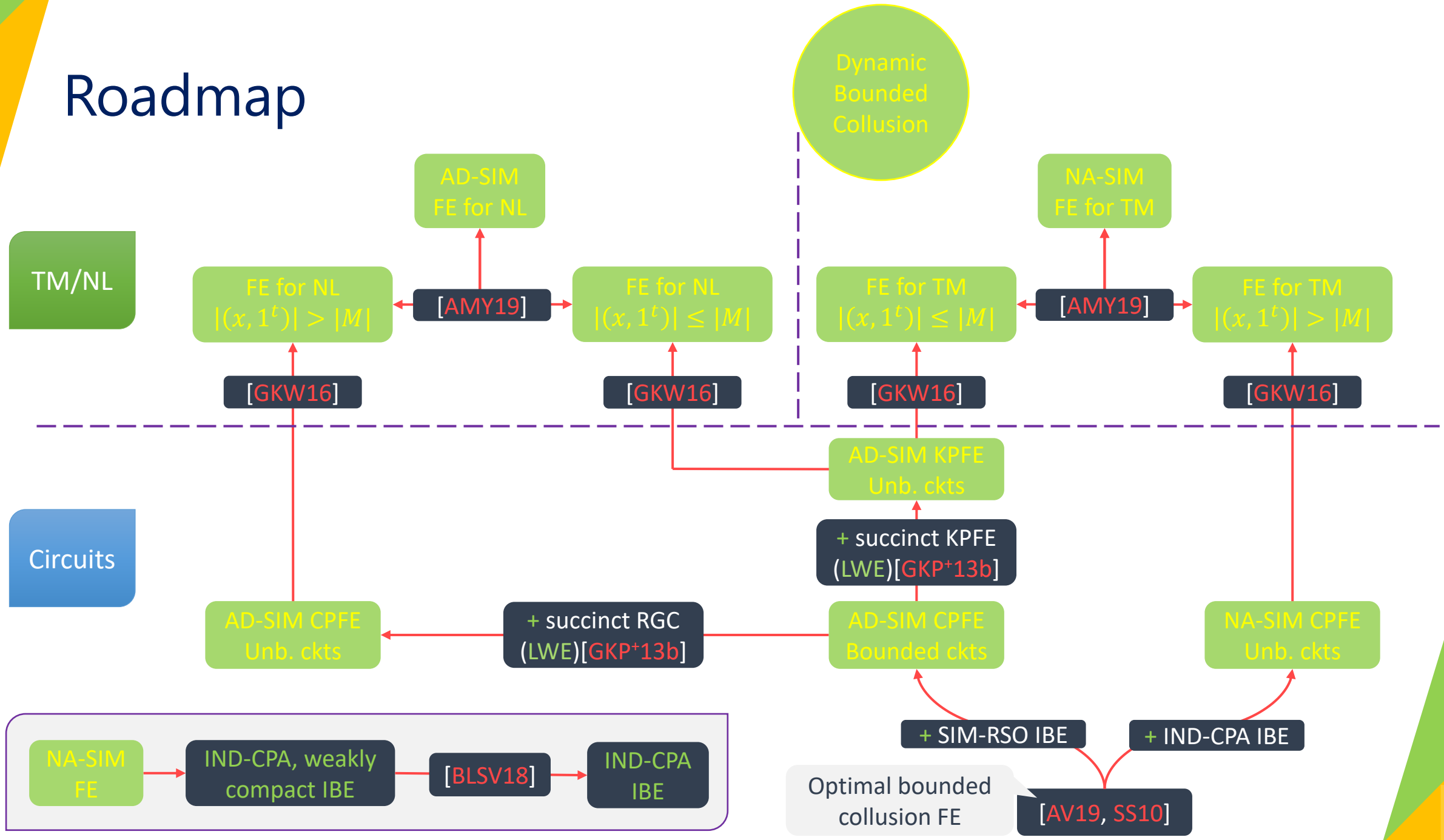
Roadmap


Dynamic Bounded Collusion
AD-SIM FE for NL
NA-SIM FE for TM
TM/NL
FE for NL $|(x,1^t)| > |M|$
[AMY19]
FE for NL $|(x,1^t)| \leq |M|$
FE for TM $|(x,1^t)| \leq |M|$
[AMY19]
FE for TM $|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
[GKW16]
AD-SIM KPFE Unb. ckts
+ succinct KPFE (LWE)[GKP+13b]
Circuits
AD-SIM CPFE Unb. ckts
+ succinct RGC (LWE)[GKP+13b]
AD-SIM CPFE Bounded ckts
NA-SIM CPFE Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded collusion FE
[AV19, SS10]

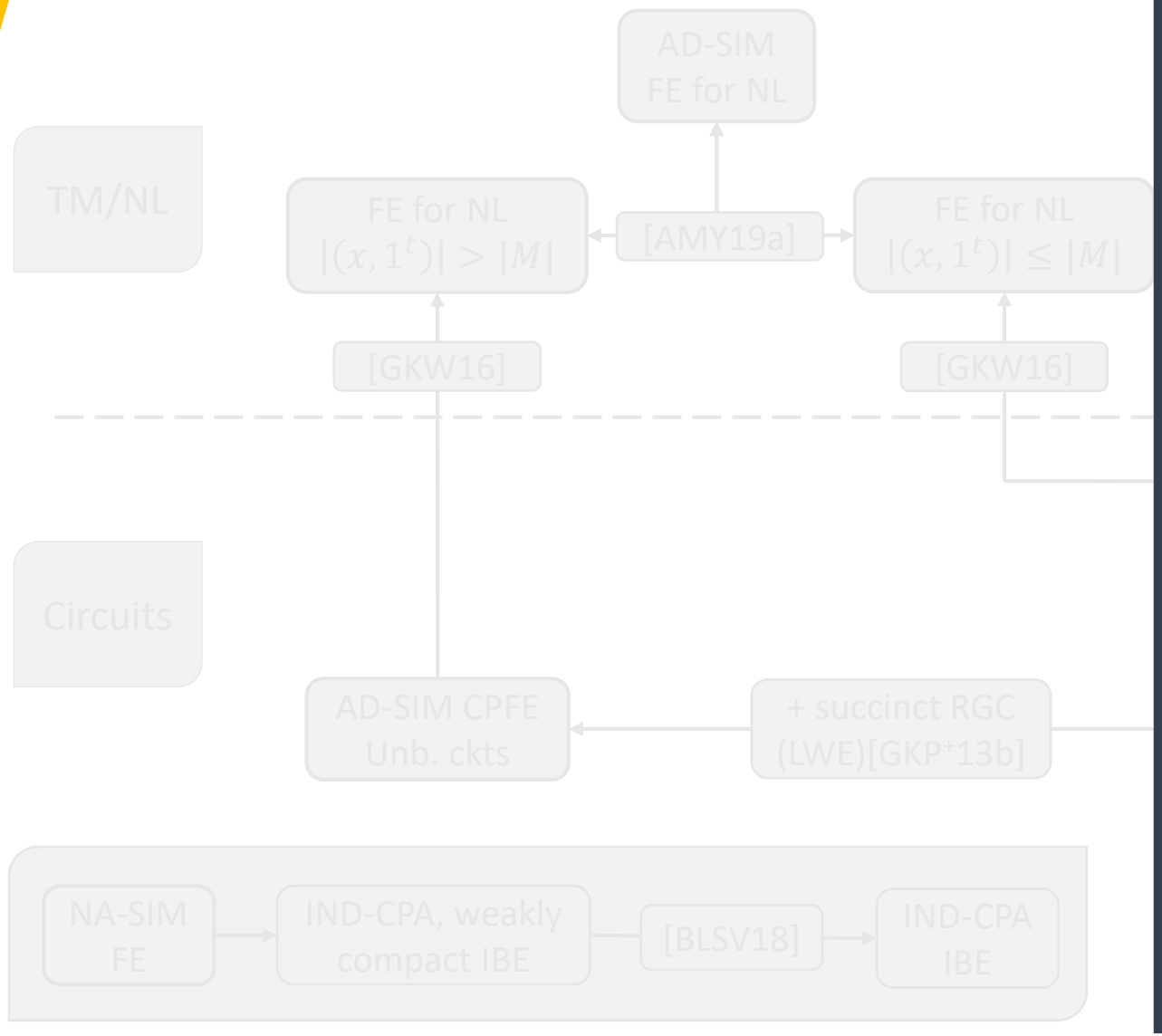
Roadmap

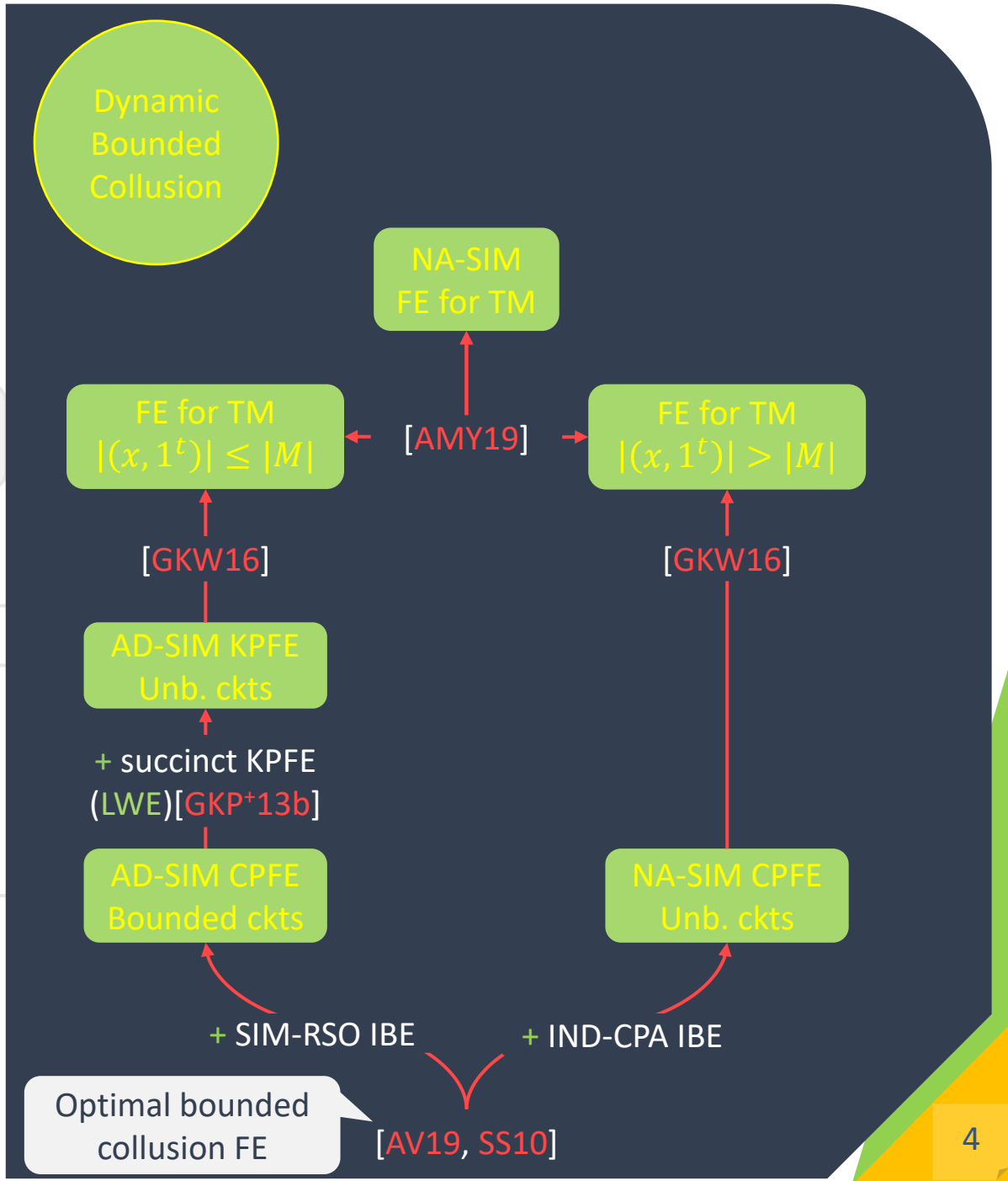

Dynamic Bounded Collusion
AD-SIM FE for NL
NA-SIM FE for TM
TM/NL
FE for NL $|(x,1^t)| > |M|$
[AMY19]
FE for NL $|(x,1^t)| \leq |M|$
FE for TM $|(x,1^t)| \leq |M|$
[AMY19]
FE for TM $|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
[GKW16]
AD-SIM KPFE Unb. ckts
Circuits
+ succinct KPFE (LWE)[GKP+13b]
AD-SIM CPFE Unb. ckts
+ succinct RGC (LWE)[GKP+13b]
AD-SIM CPFE Bounded ckts
NA-SIM CPFE Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
NA-SIM FE
IND-CPA, weakly compact IBE
[BLSV18]
IND-CPA IBE
Optimal bounded collusion FE
[AV19, SS10]

Roadmap


Dynamic Bounded Collusion
AD-SIM FE for NL
NA-SIM FE for TM
TM/NL
FE for NL $|(x,1^t)| > |M|$
[AMY19]
FE for NL $|(x,1^t)| \leq |M|$
FE for TM $|(x,1^t)| \leq |M|$
[AMY19]
FE for TM $|(x,1^t)| > |M|$
[GKW16]
[GKW16]
[GKW16]
[GKW16]
AD-SIM KPFE Unb. ckts
Circuits
+ succinct KPFE (LWE)[GKP+13b]
AD-SIM CPFE Unb. ckts
+ succinct RGC (LWE)[GKP+13b]
AD-SIM CPFE Bounded ckts
NA-SIM CPFE Unb. ckts
NA-SIM FE
IND-CPA, weakly compact IBE
[BLSV18]
IND-CPA IBE
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded collusion FE
[AV19, SS10]

Roadmap


AD-SIM
FE for NL
TM/NL
FE for NL
$|(x,1^t)| > |M|$
[AMY19a]
FE for NL
$|(x,1^t)| \leq |M|$
[GKW16]
[GKW16]
Circuits
AD-SIM CPFE
Unb. ckts
+ succinct RGC
(LWE)[GKP+13b]
NA-SIM
FE
IND-CPA, weakly
compact IBE
[BLSV18]
IND-CPA
IBE

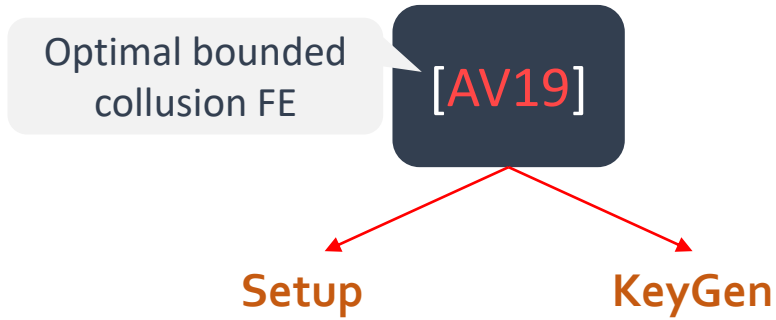

Dynamic
Bounded
Collusion
NA-SIM
FE for TM
FE for TM
$|(x,1^t)| \leq |M|$
[AMY19]
FE for TM
$|(x,1^t)| > |M|$
[GKW16]
[GKW16]
AD-SIM KPFE
Unb. ckts
+ succinct KPFE
(LWE)[GKP+13b]
AD-SIM CPFE
Bounded ckts
NA-SIM CPFE
Unb. ckts
+ SIM-RSO IBE
+ IND-CPA IBE
Optimal bounded
collusion FE
[AV19, SS10]

Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.

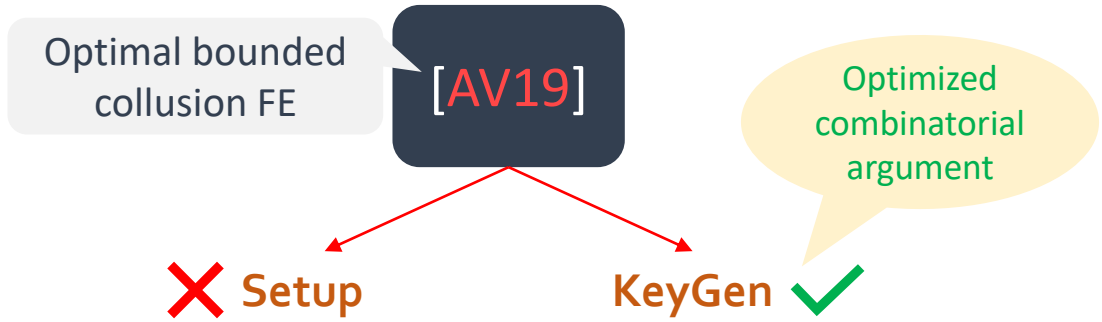
Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).



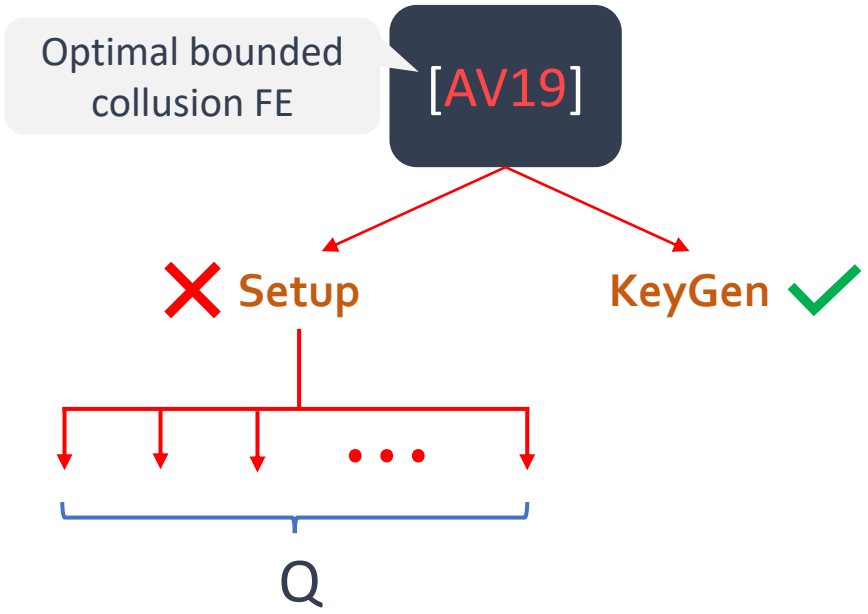
Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly*(Q) . . .



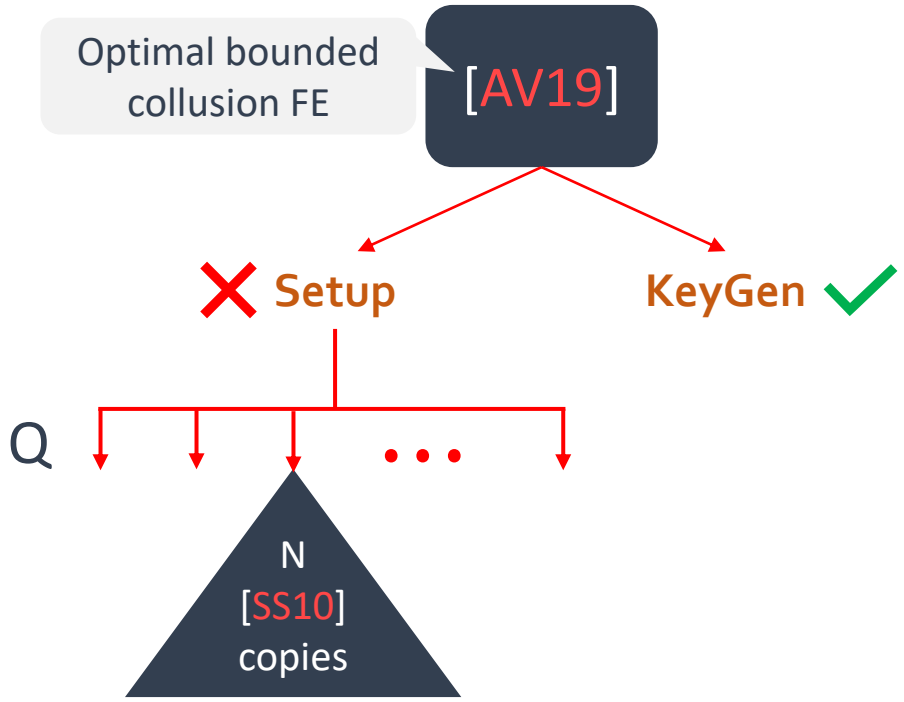
Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly*(Q) . . .



Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly*(Q) . . .



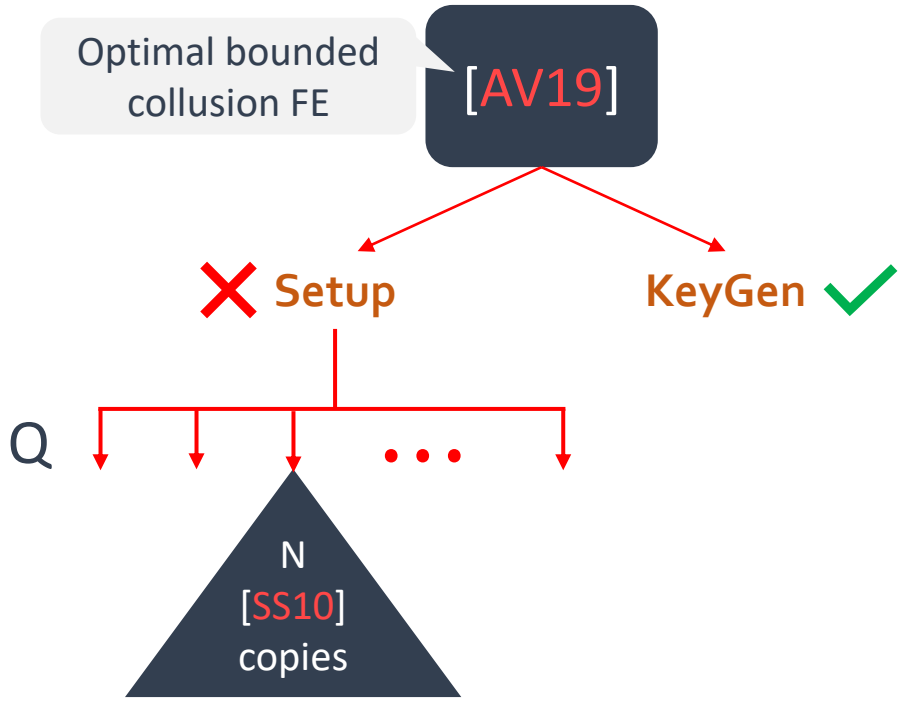
Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly*(Q) . . .

[SS10]: inputs $x \in \{0,1\}^n$

Setup: 2n PKE keys $\{pk_{i,b}, sk_{i,b}\}$

KeyGen(x): $sk_x = \{ sk_{i,x_i} \}$



Dynamic Bounded Collusion CPFE

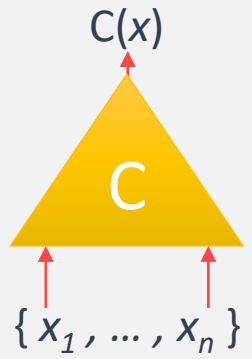
- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly*(Q) . . .

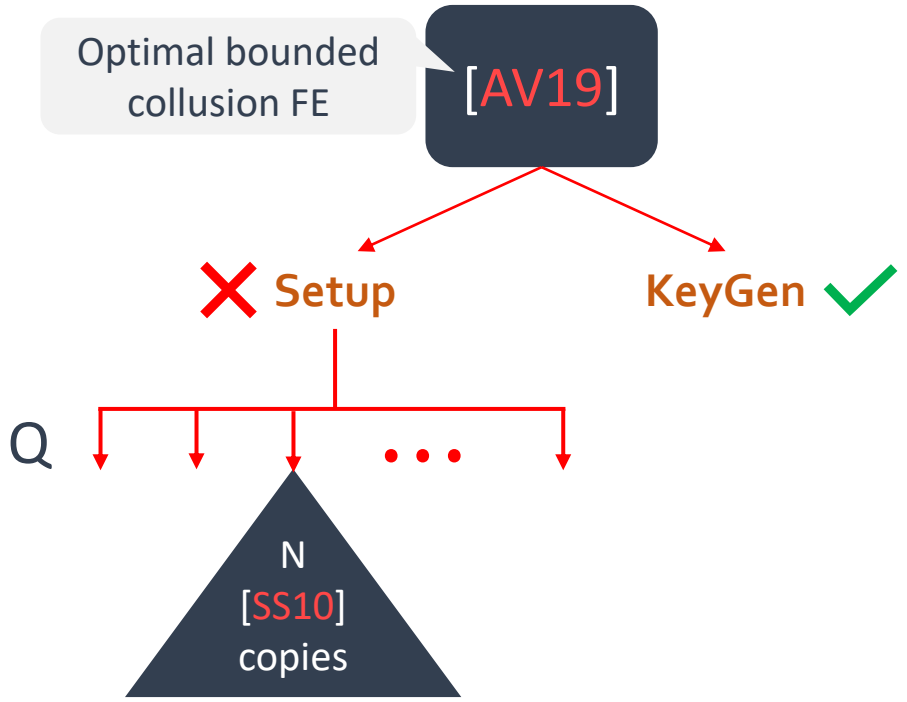
[SS10]: inputs $x \in \{0,1\}^n$

Setup: 2n PKE keys $\{pk_{i,b}, sk_{i,b}\}$

KeyGen(x): $sk_x = \{ sk_{i,x_i} \}$

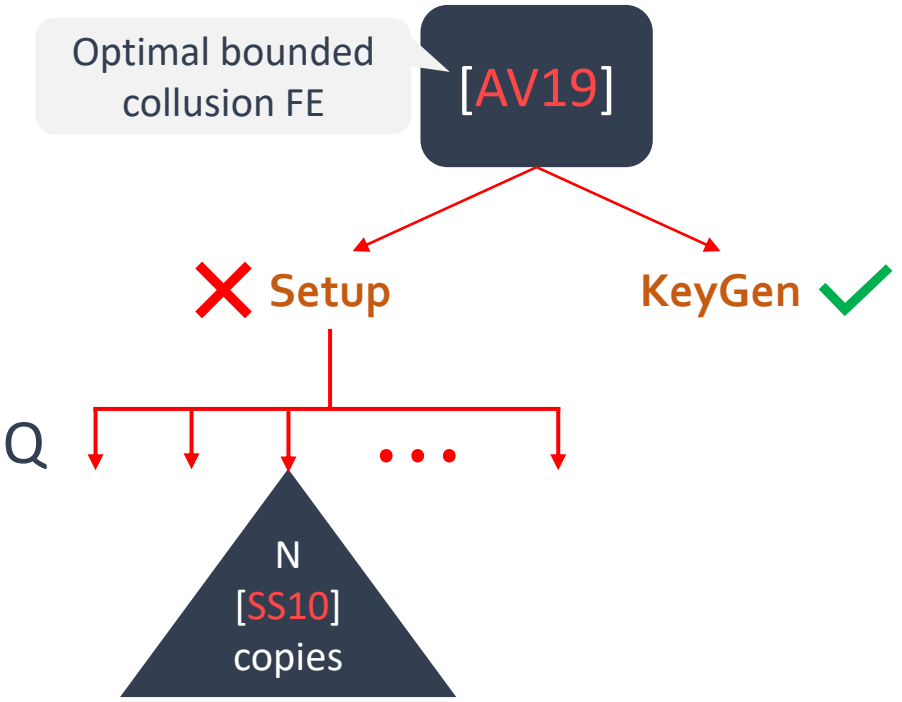
Encrypt(C):





Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly*(Q) . . .

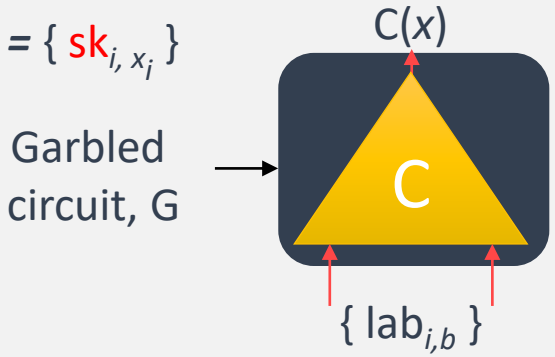


[SS10]: inputs $x \in \{0,1\}^n$

Setup: 2n PKE keys $\{pk_{i,b}, sk_{i,b}\}$

KeyGen(x): $sk_x = \{ sk_{i,x_i} \}$

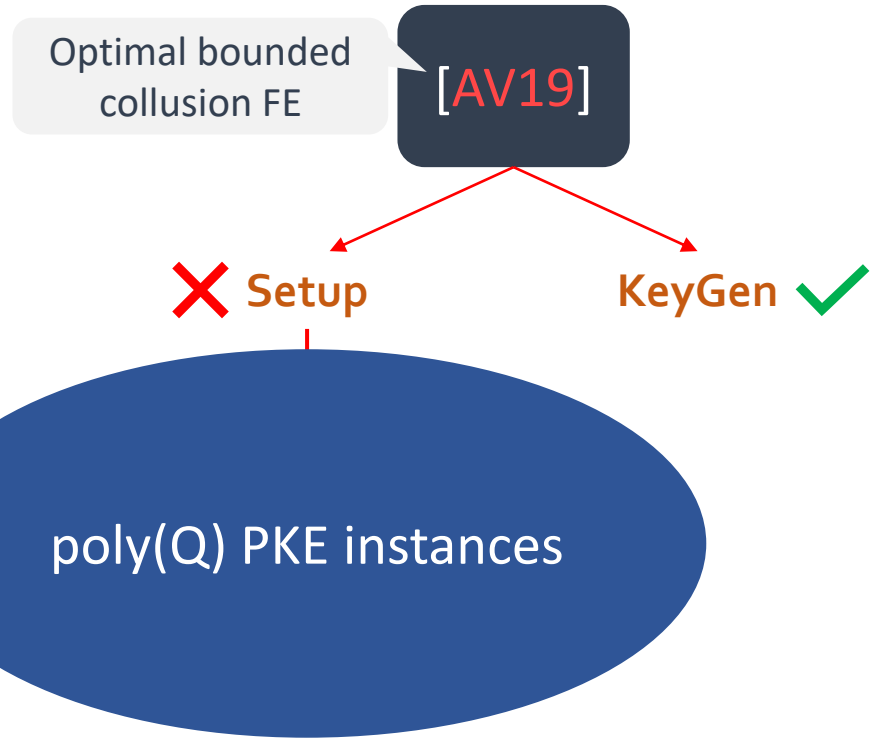
Encrypt(C):



PKE. CT = $\{ lab_{i,b} \}$

Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly*(Q) . . .

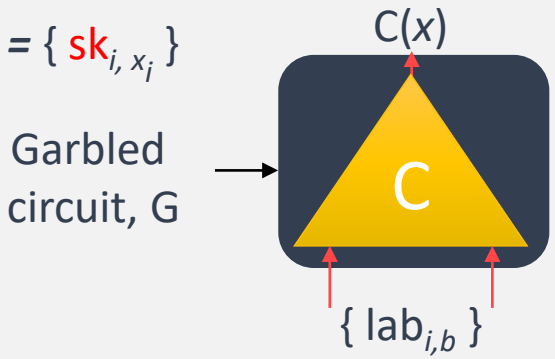


[SS10]: inputs $x \in \{0,1\}^n$

Setup: 2n PKE keys $\{pk_{i,b}, sk_{i,b}\}$

KeyGen(x): $sk_x = \{ sk_{i,x_i} \}$

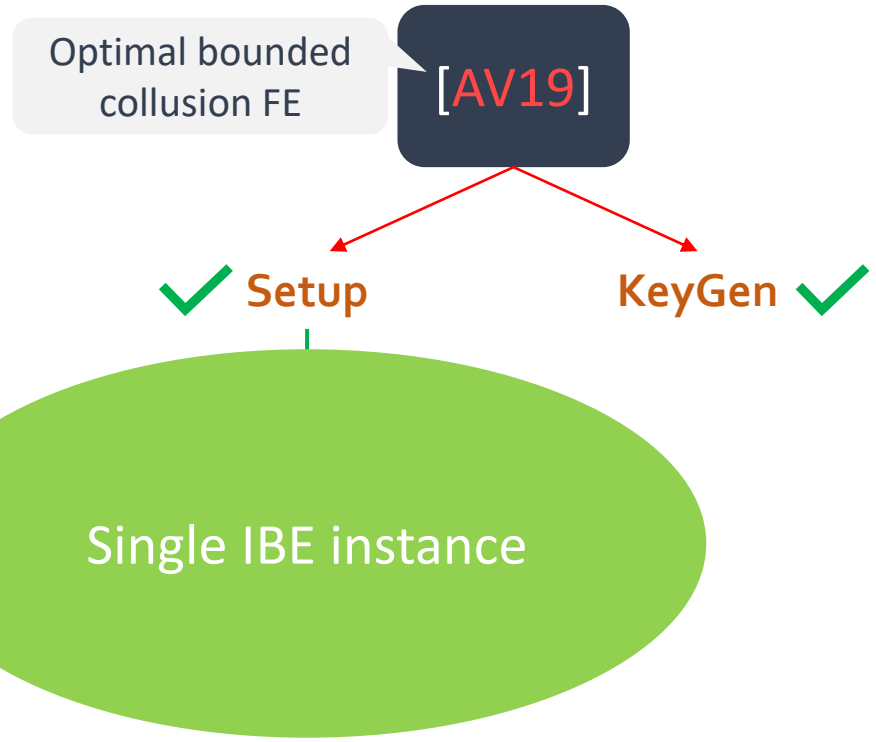
Encrypt(C):



PKE. CT = $\{ \boxed{lab_{i,b}} \}$

Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly*(Q) . . .

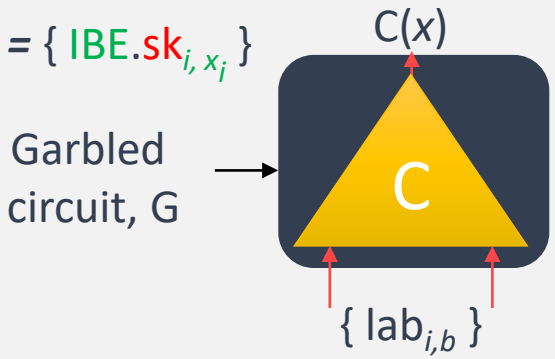


[SS10]: inputs $x \in \{0,1\}^n$

Setup: IBE = (mpk, msk)

KeyGen(x): sk_x = { $IBE.sk_{i, x_i}$ }

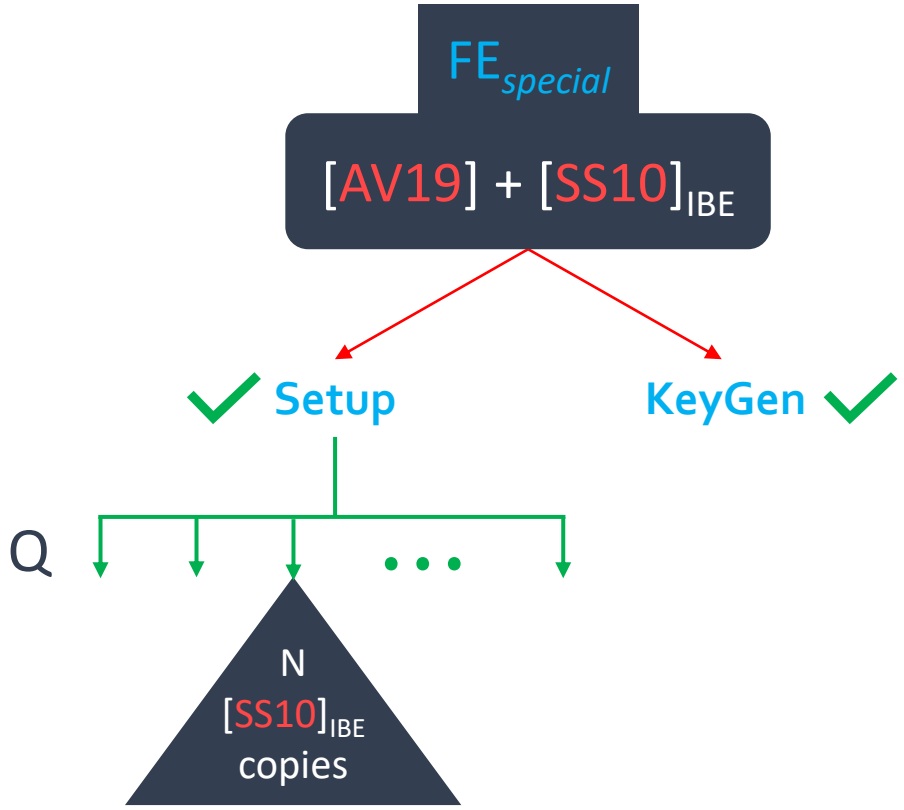
Encrypt(C):



IBE. CT = { $lab_{i,b}$ }, $id_i = (i, x_i)$

Dynamic Bounded Collusion CPFE

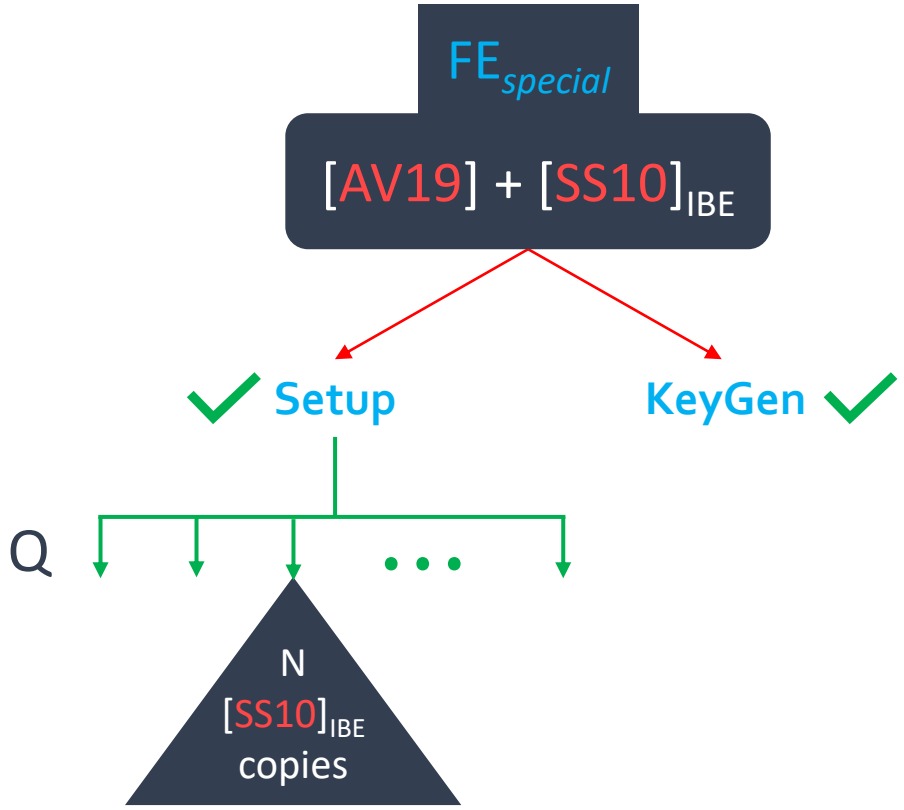
- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly-log*(Q)



Dynamic Bounded Collusion CPFE

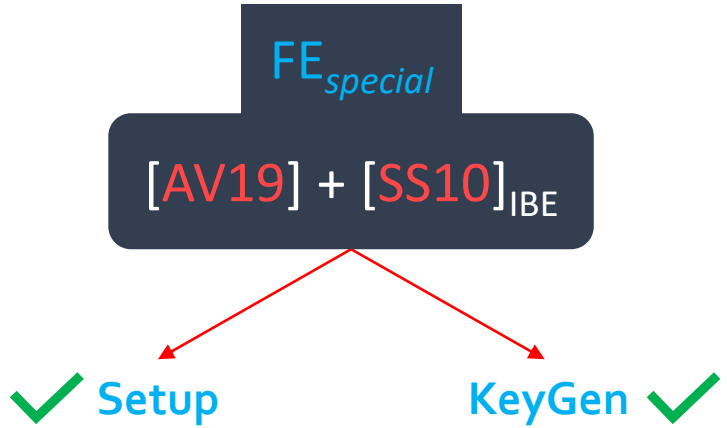
- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly-log*(Q)

- $FE_{special}$ (**Setup**, **KeyGen**) *still need to get rid of* Q . . .



Dynamic Bounded Collusion CPFE

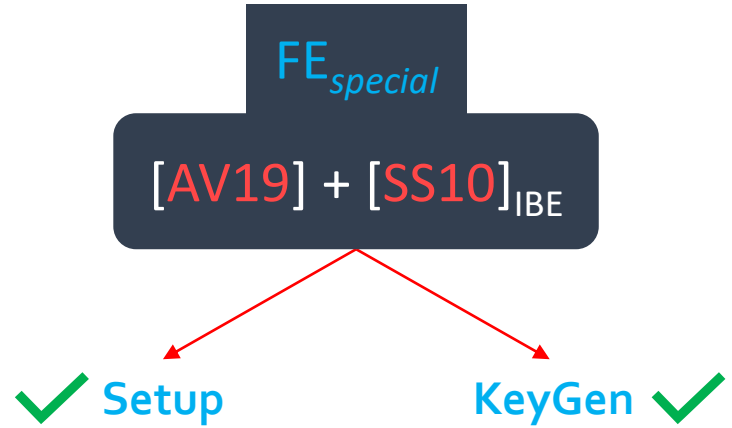
- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly-log*(Q)



- $FE_{special}$ (**Setup**, **KeyGen**) *still need to get rid of* Q . . .
- "*Power of 2*" trick[GKP+13]: Run λ $FE_{special}$ *parallely*

Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly-log*(Q)

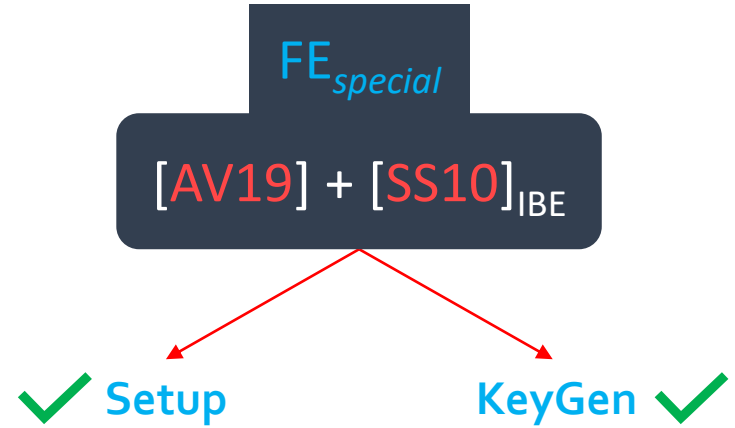


- $FE_{special}$ (**Setup**, **KeyGen**) *still need to get rid of* Q . . .
- "*Power of 2*" trick[GKP+13]: Run λ $FE_{special}$ *parallely*

$CPFE_{DBC}$	**Setup**	**KeyGen**
$FE_{special}$	Setup(2)	KeyGen(2)
	$Setup(2^2)$	$KeyGen(2^2)$
	⋮	⋮
	$Setup(2^\lambda)$	$KeyGen(2^\lambda)$

Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly-log*(Q)

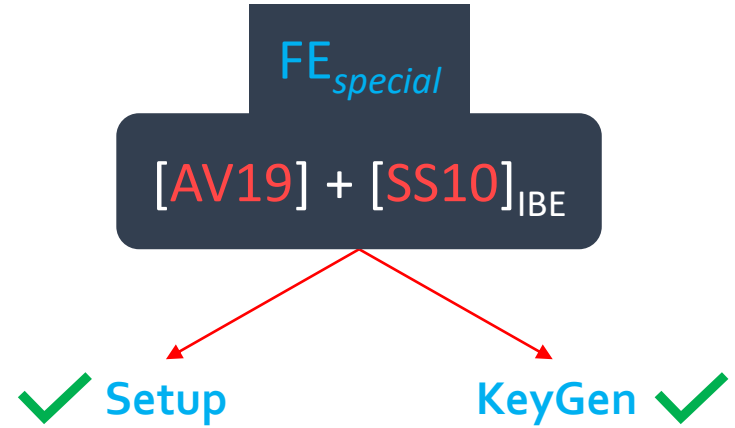


- $FE_{special}$ (**Setup**, **KeyGen**) *still need to get rid of* Q . . .
- "*Power of 2*" trick[GKP+13]: Run λ $FE_{special}$ *parallely*

$CPFE_{DBC}$	**Setup**	**KeyGen**	**Encrypt**$(x,1^Q)$	**Decrypt**
$FE_{special}$	Setup(2) Setup(2^2) ⋮ Setup(2^λ)	KeyGen(2) KeyGen(2^2) ⋮ KeyGen(2^λ)	Encrypt(2^i,x) $2^{i-1} < Q \leq 2^i$	Decrypt with 2^i-th subsystem

Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly-log*(Q)



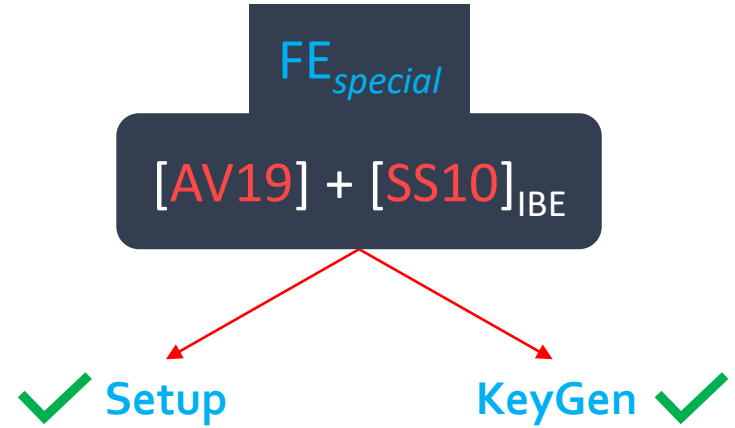
- $FE_{special}$ (**Setup**, **KeyGen**) *still need to get rid of* Q . . .
- "*Power of 2*" trick[GKP+13]: Run λ $FE_{special}$ *parallely*

$CPFE_{DBC}$	**Setup**	**KeyGen**	**Encrypt**$(x,1^Q)$	**Decrypt**
$FE_{special}$	Setup(2) Setup(2^2) ⋮ Setup(2^λ)	KeyGen(2) KeyGen(2^2) ⋮ KeyGen(2^λ)	Encrypt(2^i,x) $2^{i-1} < Q \leq 2^i$	Decrypt with 2^i-th subsystem

Time(**Setup**, **KeyGen**) = *poly-log*(Q) ⇒ $CPFE_{DBC}$ *efficient*

Dynamic Bounded Collusion CPFE

- Time(**Setup**, **KeyGen**) must be *independent* of Q.
- Weaker need: Time(**Setup**, **KeyGen**) = *poly-log*(Q).
- [AV19]: Time(**KeyGen**) = *poly-log*(Q)
- [AV19]: Time(**SetUp**) = *poly-log*(Q)



- $FE_{special}$ (**Setup**, **KeyGen**) *still need to get rid of* Q . . .
- "*Power of 2*" trick[GKP+13]: Run λ $FE_{special}$ *parallely*

$CPFE_{DBC}$	**Setup**	**KeyGen**	**Encrypt**$(x,1^Q)$	**Decrypt**
$FE_{special}$	Setup(2) Setup(2^2) ⋮ Setup(2^λ)	KeyGen(2) KeyGen(2^2) ⋮ KeyGen(2^λ)	Encrypt(2^i,x) $2^{i-1} < Q \leq 2^i$	Decrypt with 2^i-th subsystem

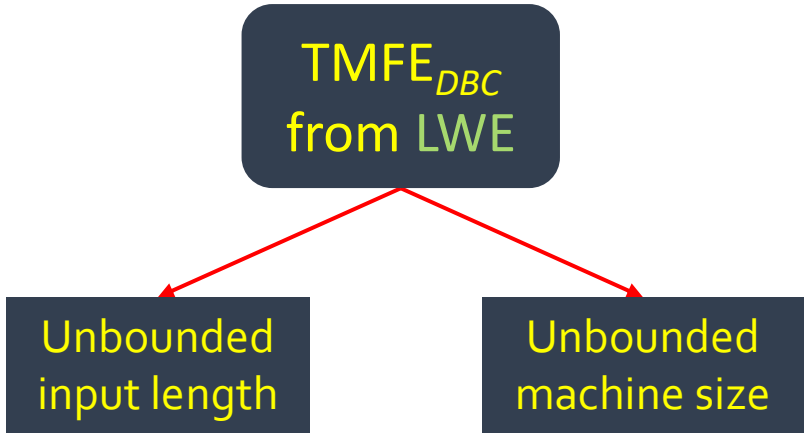
Time(**Setup**, **KeyGen**) = *poly-log*(Q) ⇒ $CPFE_{DBC}$ *efficient*

IND-CPA IBE ⇒ *NA*-SIM, *unbounded* circuits

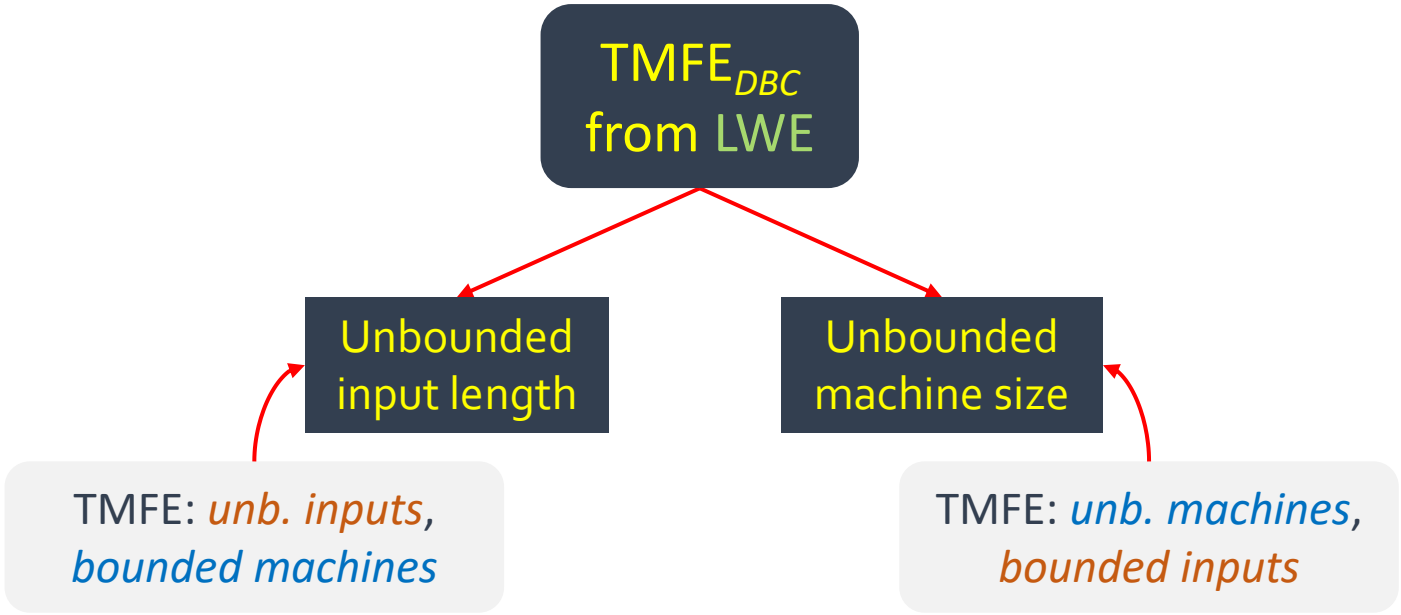
SIM-RSO IBE ⇒ AD-SIM, *bounded* circuits

NA-SIM FE ⇒ IND-CPA IBE

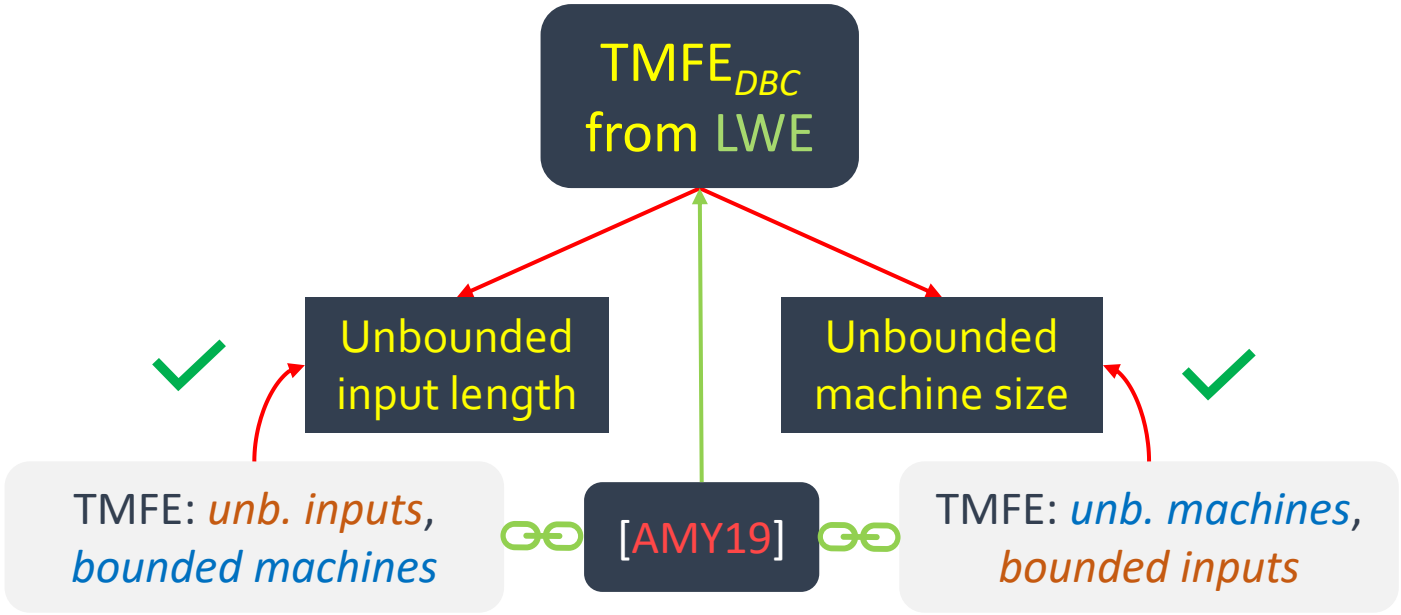
Dynamic Bounded Collusion TM-FE



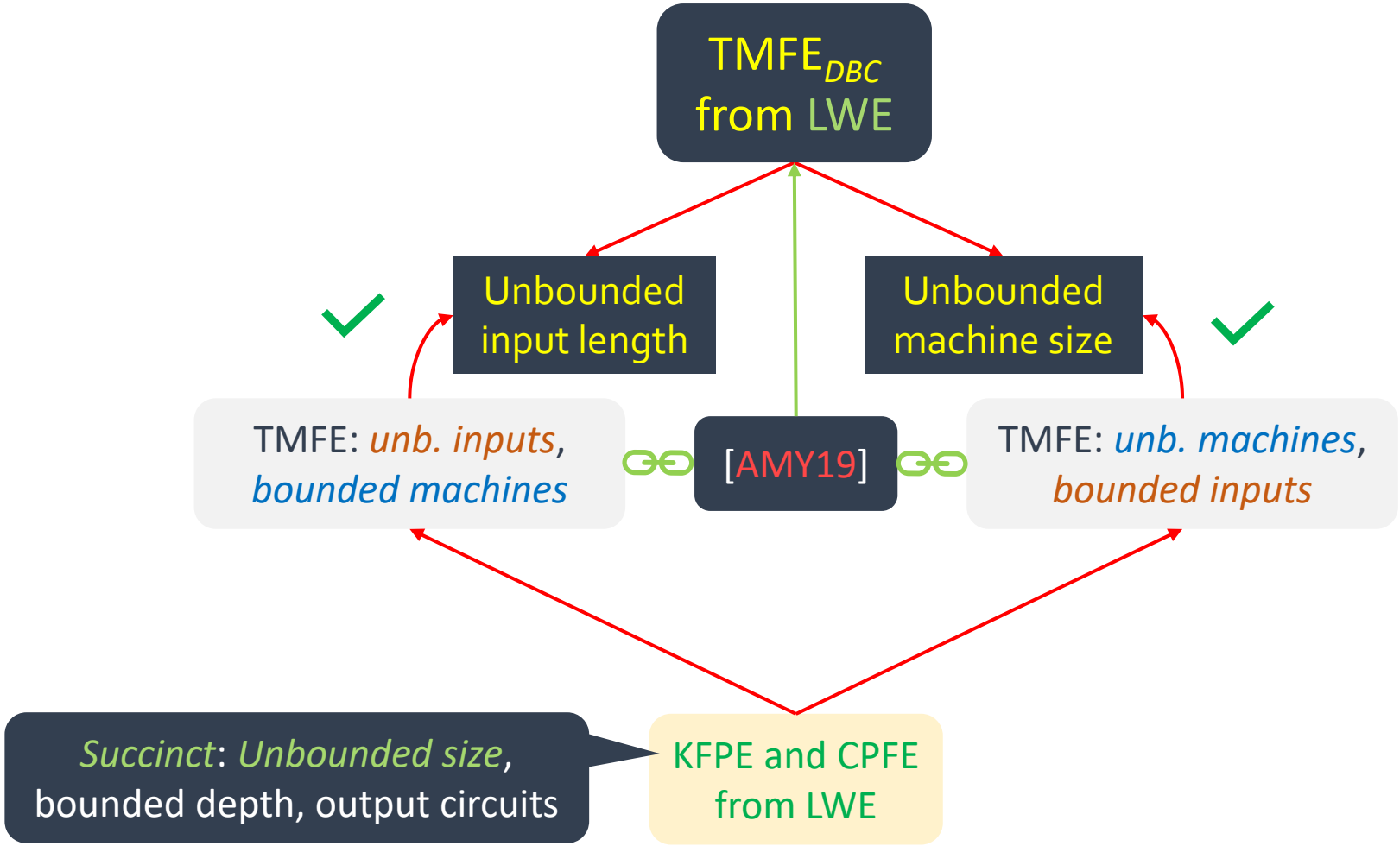
Dynamic Bounded Collusion TM-FE



Dynamic Bounded Collusion TM-FE



Dynamic Bounded Collusion TM-FE



Dynamic Bounded Collusion, Succinct KPFE

DBC CPFE

IND-CPA IBE ⇒ *NA*-SIM, *unbounded* circuits

SIM-RSO IBE ⇒ AD-SIM, *bounded* circuits

NA-SIM FE ⇒ IND-CPA IBE

[GKP+13b]
NA-SIM, *1-key* KPFE
succinct, from LWE

Dynamic Bounded Collusion, Succinct KPFE

DBC CPFE

IND-CPA IBE ⇒ *NA*-SIM, *unbounded* circuits

SIM-RSO IBE ⇒ AD-SIM, *bounded* circuits

NA-SIM FE ⇒ IND-CPA IBE

DBC, *AD-SIM* CPFE
bounded circuits

[GKP+13b]
NA-SIM, *1-key* KPFE
succinct, from LWE

AD-SIM, *DBC* KPFE
succinct, from LWE
(Improves [GKP+13b])

Dynamic Bounded Collusion, Succinct KPFE

Setup(1^λ):	$\text{CPFE}_{DBC}.(\text{PK}, \text{MSK})$

DBC, *AD-SIM* CPFE bounded circuits

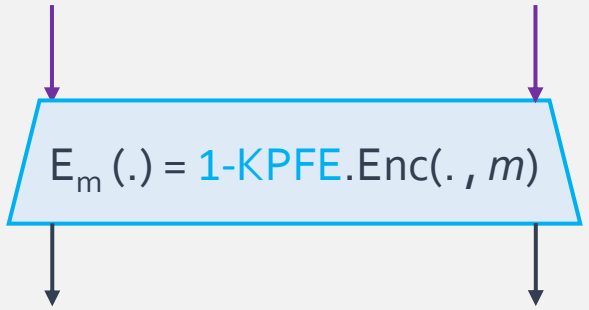
[GKP+13b]
NA-SIM, *1-key* KPFE
succinct, from LWE

AD-SIM, *DBC* KPFE
succinct, from LWE
(Improves [GKP+13b])

Dynamic Bounded Collusion, Succinct KPFE

Setup(1^λ): $\text{CPFE}_{DBC}.(\text{PK}, \text{MSK})$

Encrypt(PK, m, 1^Q):



DBC, *AD-SIM* CPFE bounded circuits

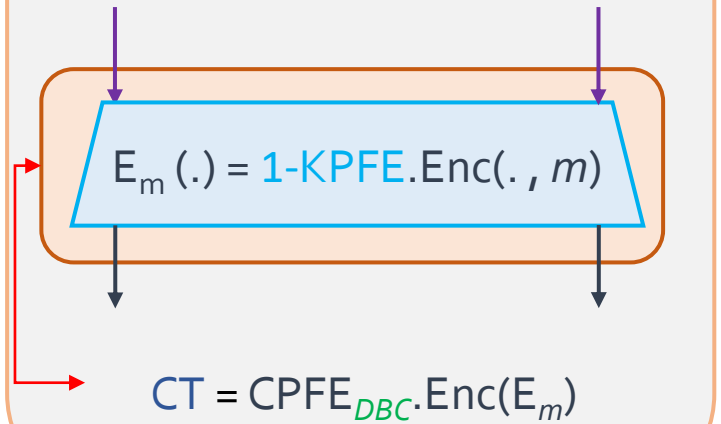
[GKP+13b]
NA-SIM, *1-key* KPFE
succinct, from LWE

AD-SIM, *DBC* KPFE
succinct, from LWE
(Improves [GKP+13b])

Dynamic Bounded Collusion, Succinct KPFE

Setup(1^λ): $\text{CPFE}_{DBC}.(\text{PK}, \text{MSK})$

Encrypt(PK, m, 1^Q):



DBC, *AD-SIM* CPFE bounded circuits

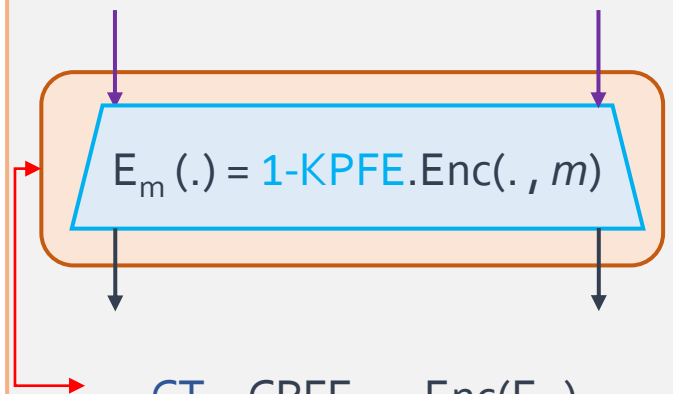
[GKP+13b]
NA-SIM, *1-key* KPFE
succinct, from LWE

AD-SIM, *DBC* KPFE
succinct, from LWE
(Improves [GKP+13b])

Dynamic Bounded Collusion, Succinct KPFE

Setup(1^λ): CPFE_{DBC}.(PK, MSK)

Encrypt(PK, m, 1^q):



KeyGen(MSK, f):

1. Sample 1-KPFE(PK, MSK)
2. Get 1-KPFE.sk_f with MSK

$\text{SK}_f = (\text{sk}_{PK}, \text{1-KPFE.sk}_f)$

DBC, *AD-SIM* CPFE bounded circuits

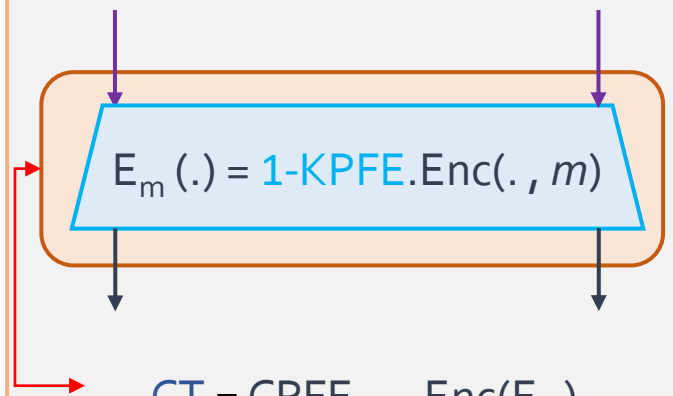
[GKP+13b]
NA-SIM, *1-key* KPFE
succinct, from LWE

AD-SIM, *DBC* KPFE
succinct, from LWE
(Improves [GKP+13b])

Dynamic Bounded Collusion, Succinct KPFE

Setup(1^λ): $\text{CPFE}_{DBC}.(\text{PK}, \text{MSK})$

Encrypt(PK, m, 1^Q):



$\text{CT} = \text{CPFE}_{DBC}.\text{Enc}(E_m)$

KeyGen(MSK, f):

1. Sample 1-KPFE(PK, MSK)
2. Get 1-KPFE.sk_f with MSK

3. $\text{sk}_{\text{PK}} \leftarrow \text{CPFE}_{DBC}.\text{KeyGen}(\text{MSK}, \text{PK})$

$\text{SK}_f = (\text{sk}_{\text{PK}}, \text{1-KPFE.sk}_f)$

DBC, *AD-SIM* CPFE bounded circuits

[GKP+13b]
NA-SIM, *1-key* KPFE succinct, from LWE

AD-SIM, *DBC* KPFE succinct, from LWE
(Improves [GKP+13b])

Dynamic Bounded Collusion, Succinct KPFE

Setup(1^λ): $\text{CPFE}_{DBC}.(\text{PK}, \text{MSK})$

Encrypt(PK, m, 1^Q):

… 1-KPFE.PK …

$E_m(.) = \text{1-KPFE.Enc}(., m)$

… 1-KPFE.ct …

$\text{CT} = \text{CPFE}_{DBC}.\text{Enc}(E_m)$

KeyGen(MSK, f):

1. Sample 1-KPFE(PK, MSK)
2. Get 1-KPFE.sk_f with MSK

3. $\text{sk}_{\text{PK}} \leftarrow \text{CPFE}_{DBC}.\text{KeyGen}(\text{MSK}, \text{PK})$

$\text{SK}_f = (\text{sk}_{\text{PK}}, \text{1-KPFE.sk}_f)$

Decrypt(CT, SK_f):

1. $\text{CPFE}_{DBC}.\text{Dec} \rightarrow \text{1-KPFE.ct}$
2. $\text{1-KPFE.Dec} \rightarrow f(m)$

DBC, *AD-SIM* CPFE bounded circuits

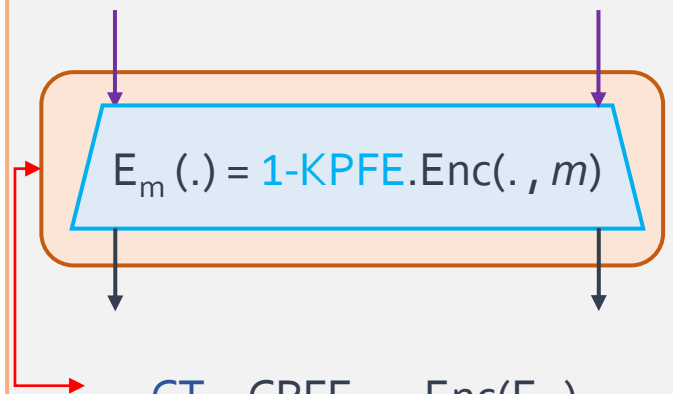
[GKP+13b]
NA-SIM, *1-key* KPFE succinct, from LWE

AD-SIM, *DBC* KPFE succinct, from LWE
(Improves [GKP+13b])

Dynamic Bounded Collusion, Succinct KPFE

Setup(1^λ): $\text{CPFE}_{DBC}.(\text{PK}, \text{MSK})$

Encrypt(PK, m, 1^Q):



KeyGen(MSK, f):

1. Sample 1-KPFE(PK, MSK)
2. Get 1-KPFE.sk_f with MSK

3. $\text{sk}_{\text{PK}} \leftarrow \text{CPFE}_{DBC}.\text{KeyGen}(\text{MSK}, \text{PK})$

$\text{SK}_f = (\text{sk}_{\text{PK}}, \text{1-KPFE.sk}_f)$

Decrypt(CT, SK_f):

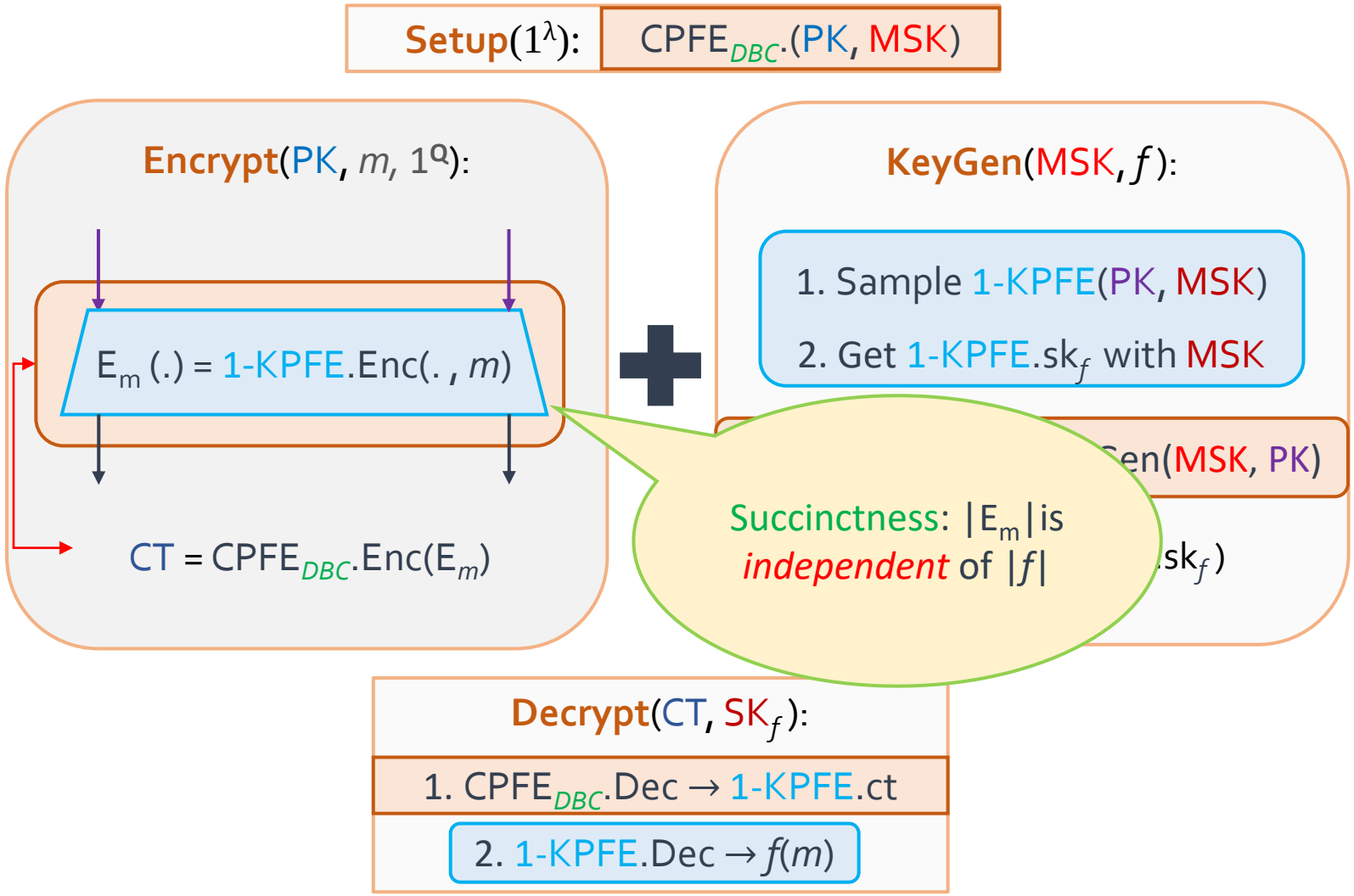
1. $\text{CPFE}_{DBC}.\text{Dec} \rightarrow \text{1-KPFE.ct}$
2. $\text{1-KPFE.Dec} \rightarrow f(m)$

DBC, *AD-SIM* CPFE bounded circuits

[GKP+13b]
NA-SIM, *1-key* KPFE succinct, from LWE

AD-SIM, *DBC* KPFE succinct, from LWE
(Improves [GKP+13b])

Dynamic Bounded Collusion, Succinct KPFE



DBC, *AD-SIM* CPFE bounded circuits

[GKP+13b] *NA-SIM*, *1-key* KPFE succinct, from LWE

AD-SIM, *DBC* KPFE succinct, from LWE (Improves [GKP+13b])

Dynamic Bounded Collusion, Succinct CPFE

DBC, *AD-SIM* CPFE bounded circuits

[GKP+13b] *Succinct* RGC from LWE

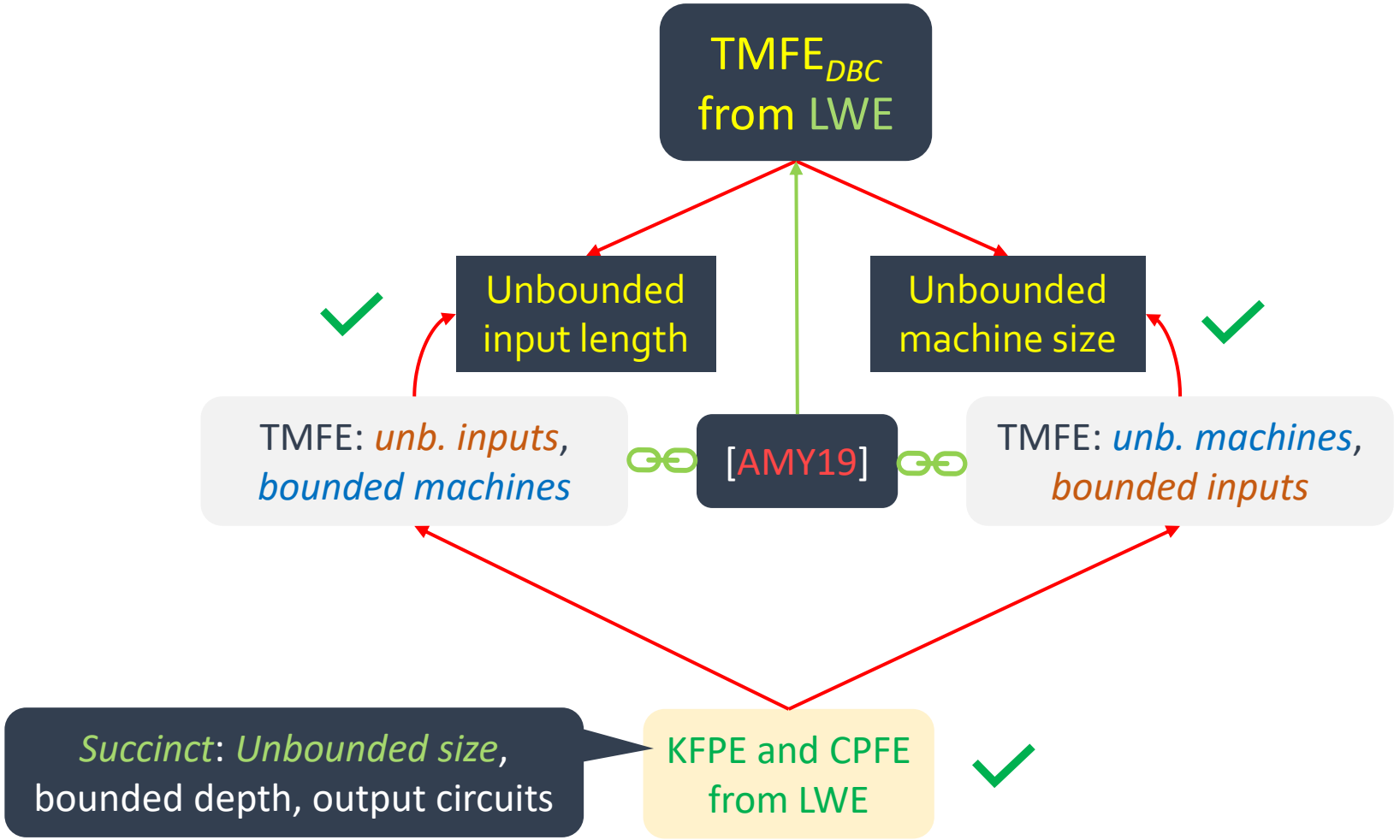
AD-SIM, *DBC* CPFE succinct, from LWE

Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$

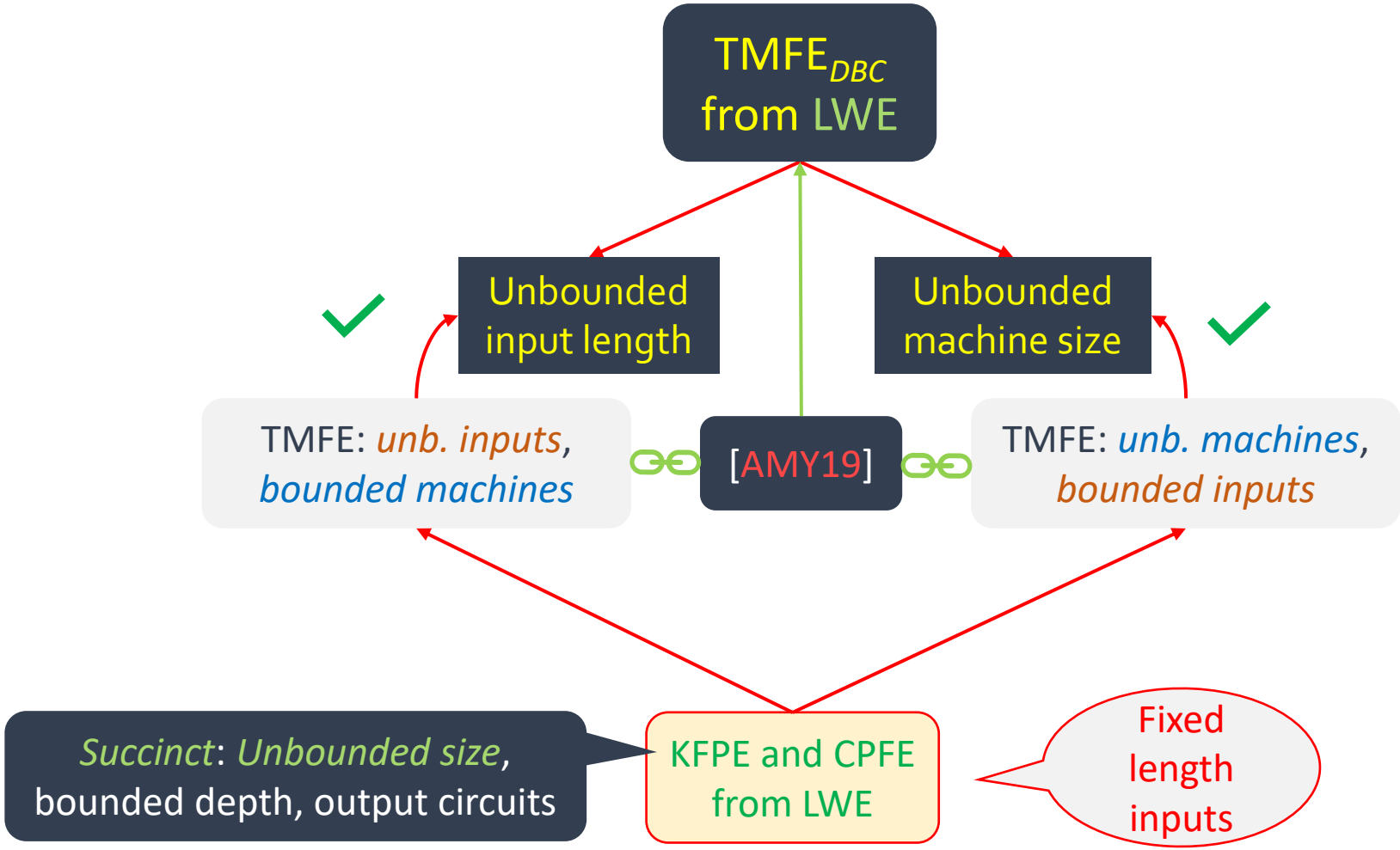


Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$



Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$

TMFE_{DBC} from LWE

$\ell > s$

CPFE

… … CPFE_i … …

$\ell \leq s$

KPFE

… … KPFE_i … …

Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$

TMFE$_{DBC}$ from LWE

$\ell > s$

CPFE

… … CPFE$_i$ … …

Circuit $U_{i,x,t}$ (M):

Run $M(x)$ for t steps

$ct_i = Enc_i(U_{i,x,t}, 1^Q), \forall i \in [\ell]$

$\ell \leq s$

KPFE

… … KPFE$_i$ … …

Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$

TMFE$_{DBC}$ from LWE

$\ell > s$

CPFE

… … CPFE$_i$ … …

Circuit $U_{i,x,t}$ (M):

Run $M(x)$ for t steps

$ct_i = Enc_i(U_{i,x,t}, 1^Q), \forall i \in [\ell]$

$KeyGen_s(M)$

$\ell \leq s$

KPFE

… … KPFE$_i$ … …

Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$

TMFE$_{DBC}$ from LWE

$\ell > s$

CPFE

… … CPFE$_i$ … …

Circuit $U_{i,x,t}$ (M):

Run $M(x)$ for t steps

$ct_i = Enc_i(U_{i,x,t}, 1^Q), \forall i \in [\ell]$

$KeyGen_s(M)$

$M(x) = Dec(sk_s, ct_s)$

$\ell \leq s$

KPFE

… … KPFE$_i$ … …

Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$

TMFE$_{DBC}$ from LWE

$\ell > s$

CPFE

… … CPFE$_i$ … …

Circuit U$_{i,x,t}$ (M):

Run $M(x)$ for t steps

$ct_i = Enc_i (U_{i,x,t}, 1^Q), \forall i \in [\ell]$

$KeyGen_s (M)$

$M(x) = Dec(sk_s, ct_s)$

$\ell \leq s$

KPFE

… … KPFE$_i$ … …

Circuit U$_{i,M}$ (x, 1^t):

Run $M(x)$ for t steps

Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$

TMFE$_{DBC}$ from LWE

$\ell > s$

CPFE

… … CPFE$_i$ … …

Circuit $U_{i,x,t}$ (M):

Run $M(x)$ for t steps

$ct_i = Enc_i (U_{i,x,t}, 1^Q), \forall i \in [\ell]$

$KeyGen_s (M)$

$M(x) = Dec(sk_s, ct_s)$

$\ell \leq s$

KPFE

… … KPFE$_i$ … …

Circuit $U_{i,M}$ (x, 1^t):

Run $M(x)$ for t steps

$ct_\ell = Enc_\ell ((x,1^t), 1^Q)$

$KeyGen_i (U_{i,M}), \forall i \in [s]$

$M(x) = Dec(sk_\ell, ct_\ell)$

Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$

TMFE$_{DBC}$ from LWE

[AMY19]

$\ell > s$

CPFE

… … CPFE$_i$ … …

Circuit U$_{i,x,t}$ (M):

Run $M(x)$ for t steps

ct$_i$ = Enc$_i$ (U$_{i,x,t}$, 1^Q), $\forall i \in [\ell]$

KeyGen$_s$ (M)

M(x) = Dec(sk$_s$, ct$_s$)

$\ell \leq s$

KPFE

… … KPFE$_i$ … …

Circuit U$_{i,M}$ (x, 1^t):

Run $M(x)$ for t steps

ct$_\ell$ = Enc$_\ell$ ((x,1^t), 1^Q)

KeyGen$_i$ (U$_{i,M}$) , $\forall i \in [s]$

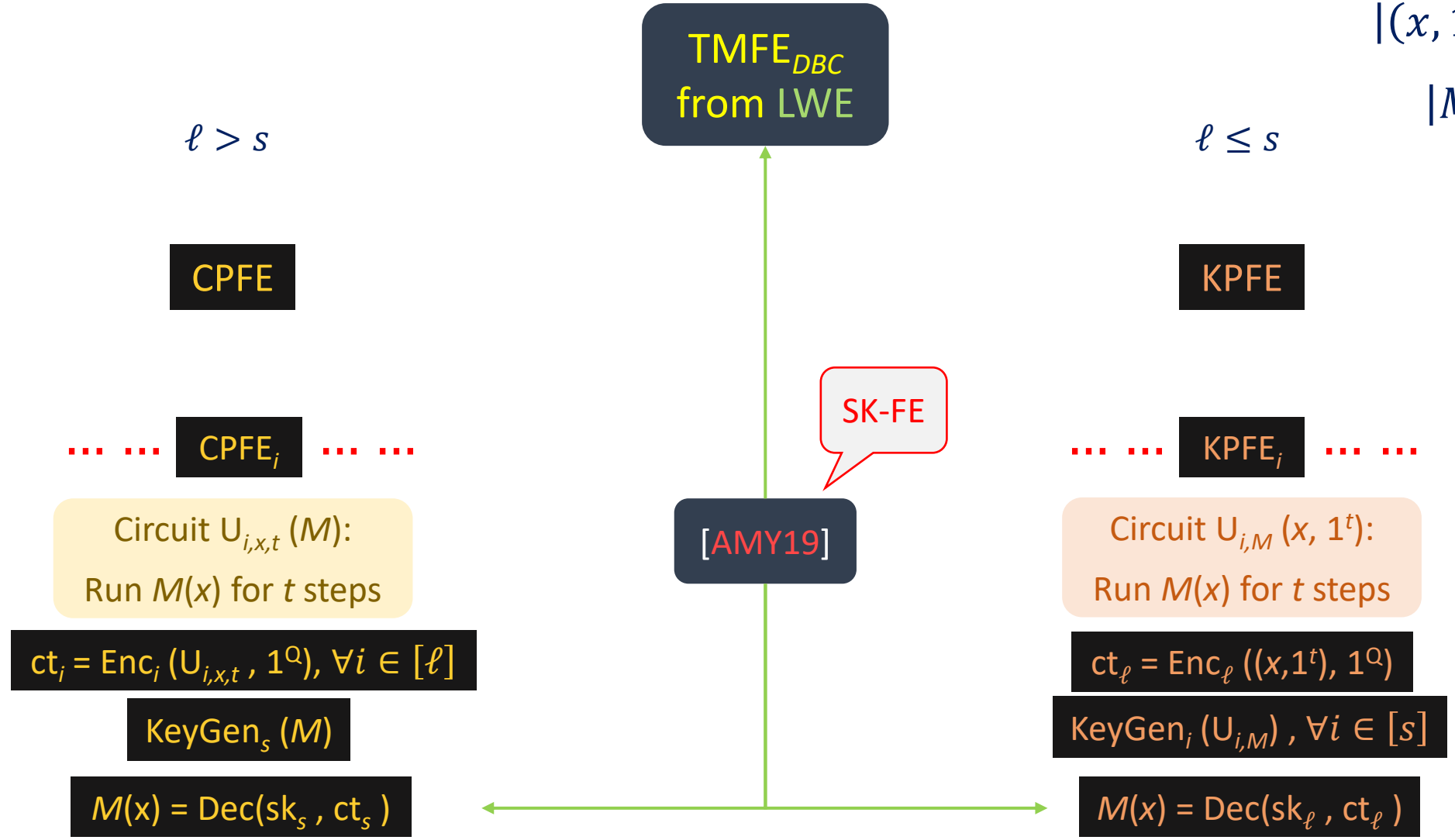
$M(x)$ = Dec(sk$_\ell$, ct$_\ell$)

Dynamic Bounded Collusion TM-FE

TM runtime = t

$|(x, 1^t)| = \ell,$

$|M| = s$

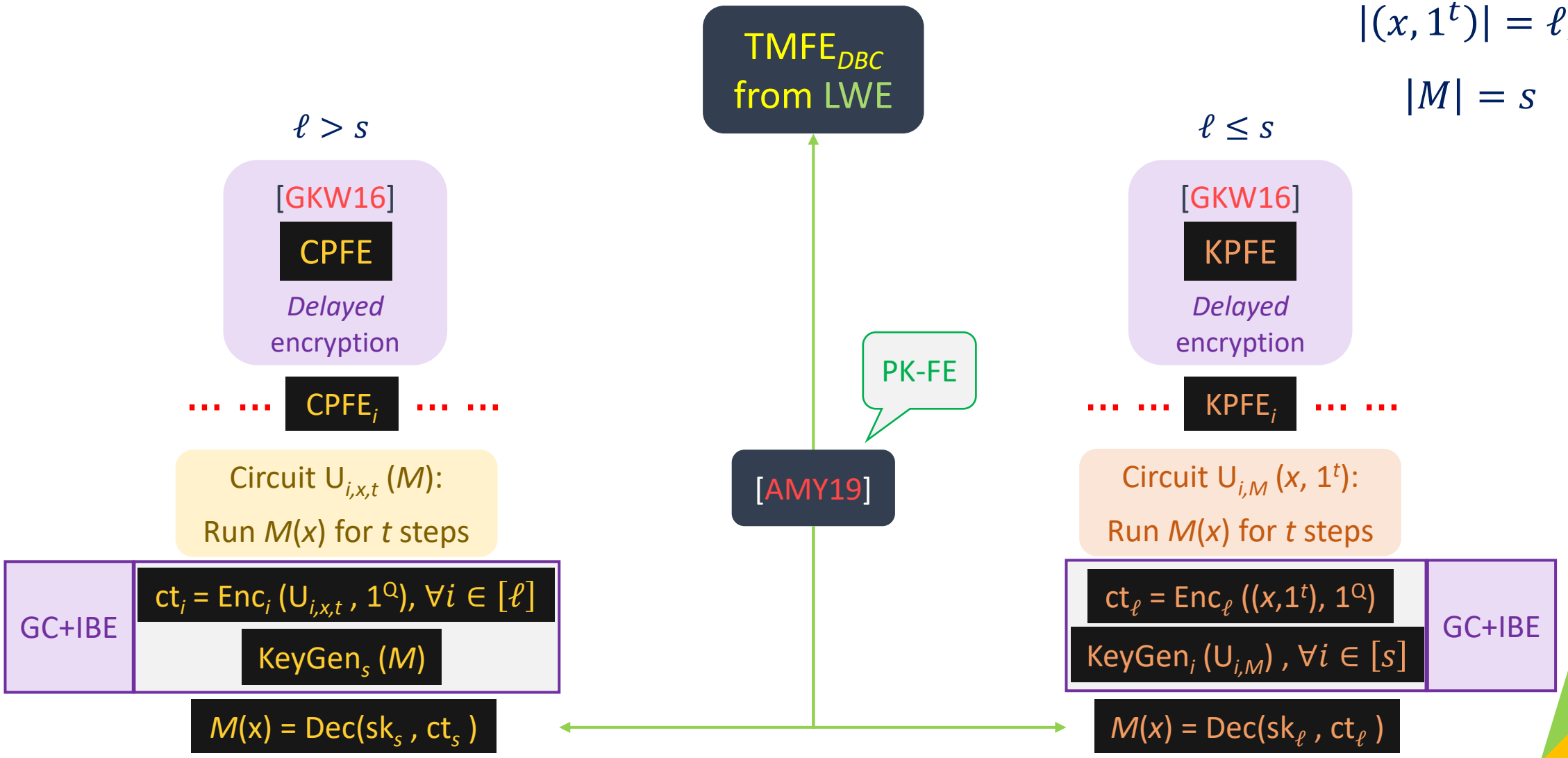


Dynamic Bounded Collusion TM-FE

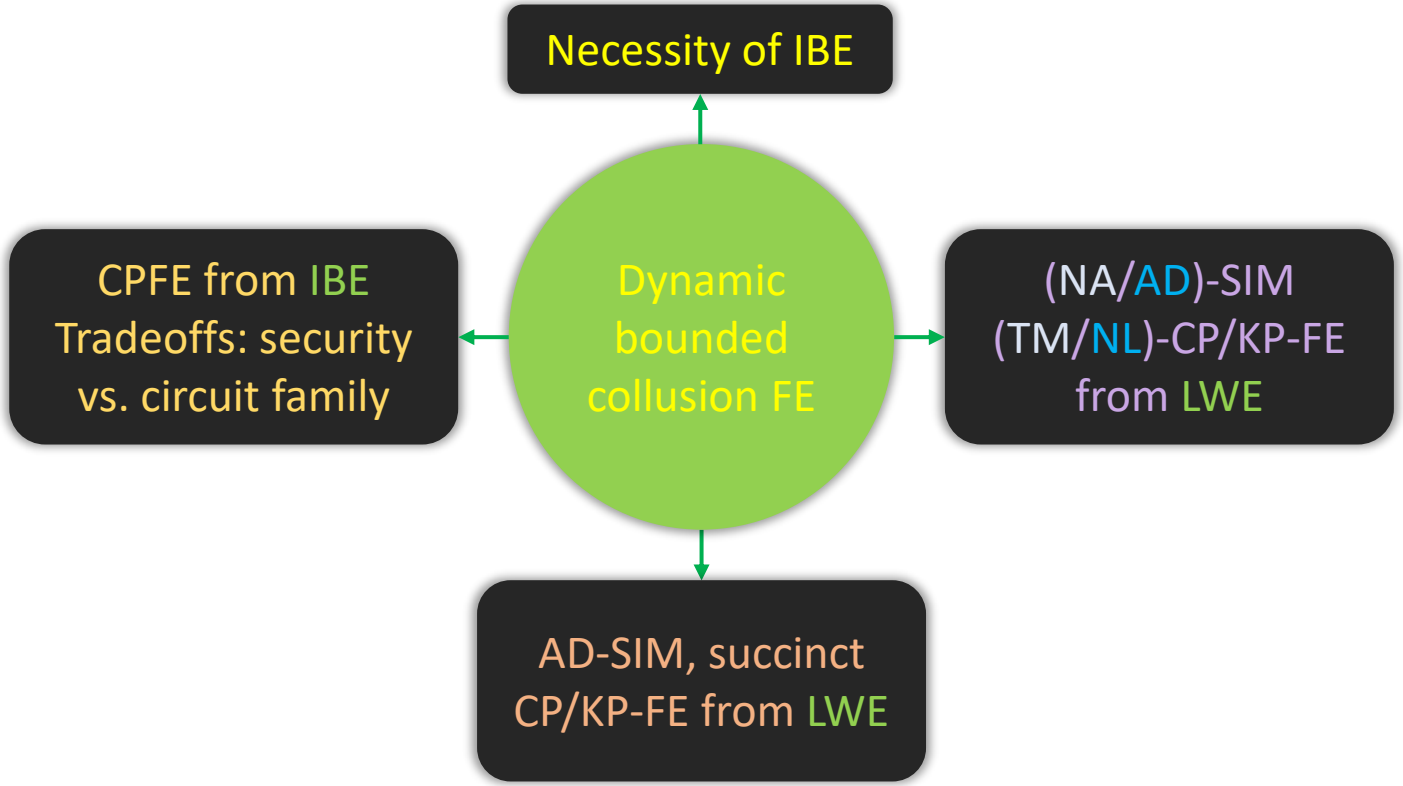
TM runtime = t

$|(x, 1^t)| = \ell,$

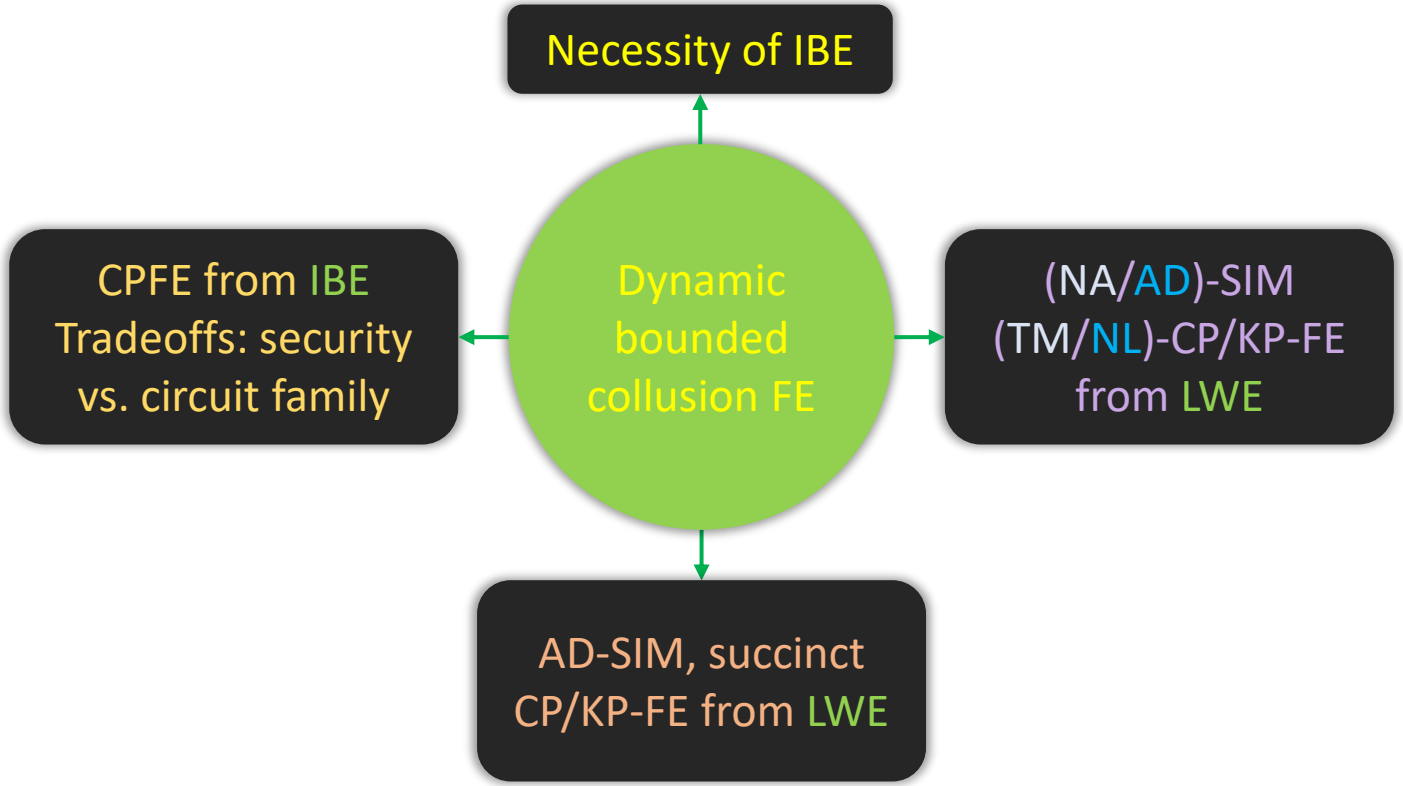
$|M| = s$



Summary and Open Problems



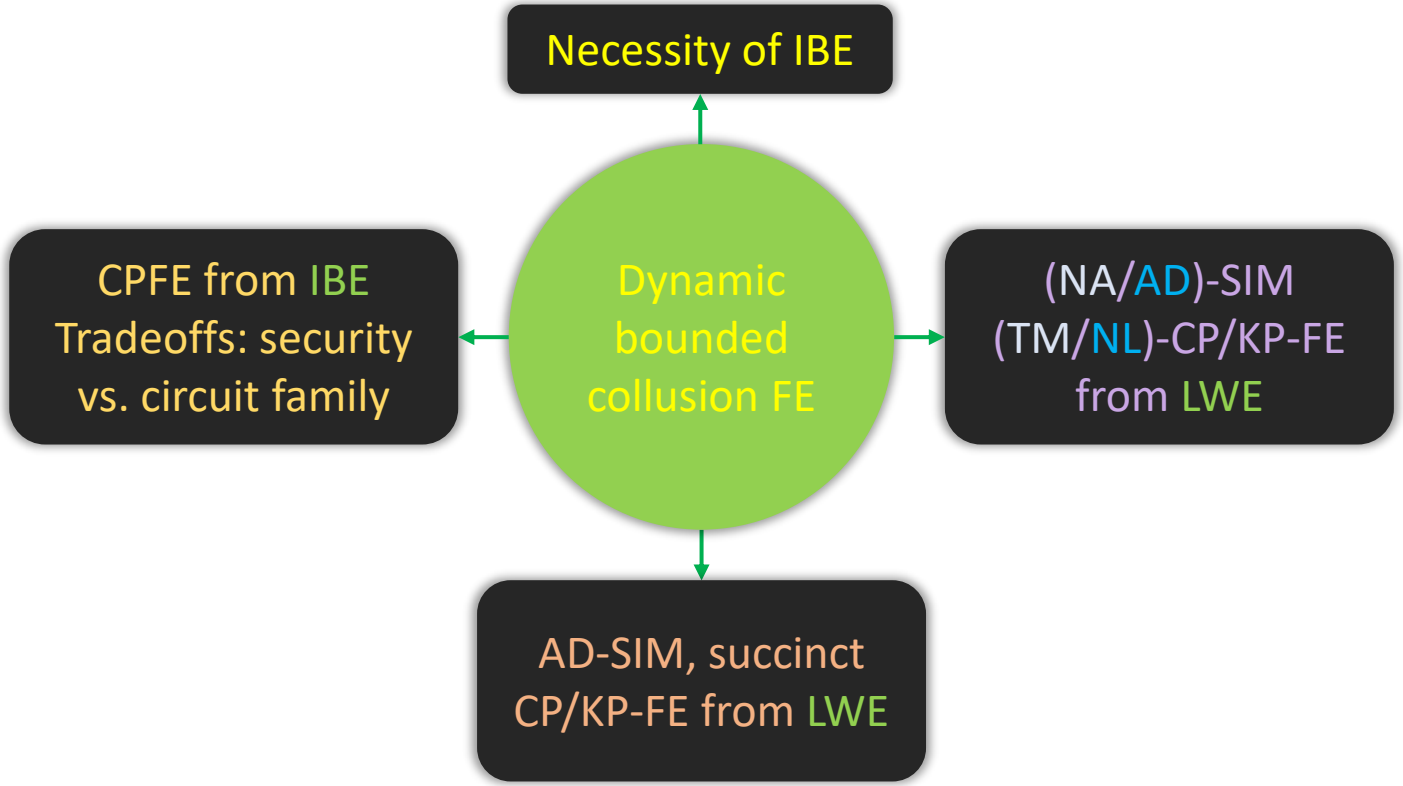
Summary and Open Problems



Open problems:

- DBC TM-FE with *AD-SIM* security
- *Remove runtime dependence* on |CT|
- *Collusion resistant ABE for TM* (*or NL*): needs *CP-ABE for unbounded depth circuits* (or *unbounded width circuits*).
- Other applications for our techniques

Summary and Open Problems



Open problems:

- DBC TM-FE with *AD-SIM* security
- *Remove runtime dependence* on |CT|
- *Collusion resistant ABE for TM* (*or NL*): needs *CP-ABE for unbounded depth circuits* (or *unbounded width circuits*).
- Other applications for our techniques

Thanks!