Multi-key Fully-Homomorphic Encryption in the Plain Model

Prabhanjan Ananth
University of California, Santa Barbara

Abhishek Jain
Johns Hopkins University

Zhengzhong Jin
Johns Hopkins University

Giulio Malavolta
Carnegie Mellon University
University of California, Berkeley

Multi-key Fully-Homomorphic Encryption [LTV12]

Multi-key Fully-Homomorphic Encryption [LTV12]

Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Multi-key Fully-Homomorphic Encryption [LTV12]



Public Keys:	$\mathbf{pk}_1$	$\mathbf{pk}_2$	$\cdots$	$\mathbf{pk}_N$
Ciphertexts:	m_1	m_2		m_N

Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C

Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \ldots, m_N)$

Multi-key Fully-Homomorphic Encryption [LTV12]



Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \ldots, m_N)$

Decryption Protocol:

Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys:	$\mathbf{pk}_1$	$\mathbf{pk}_2$	$\cdots$	$\mathbf{pk}_N$
Ciphertexts:	m_1	m_2		m_N

C $\qquad$ $C(m_1, m_2, \ldots, m_N)$

Decryption Protocol:

Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \dots, m_N)$

Decryption Protocol:

…

Multi-key Fully-Homomorphic Encryption [LTV12]

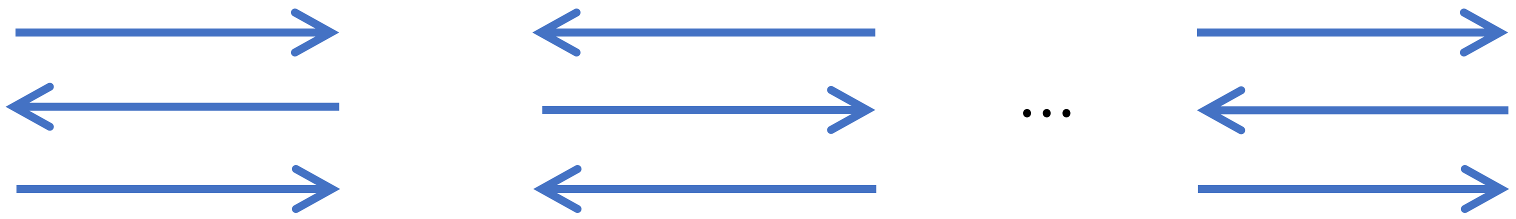
Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ … $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \ldots, m_N)$

Decryption Protocol: …

Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys:	$\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$
Ciphertexts:	m_1 m_2 m_N
	C $C(m_1, m_2, \ldots, m_N)$
Decryption Protocol:	
Recovery:	$C(m_1, m_2, \ldots, m_N)$

Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ … $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \ldots, m_N)$ **(Compact)**

Decryption Protocol: …

Recovery: $C(m_1, m_2, \ldots, m_N)$

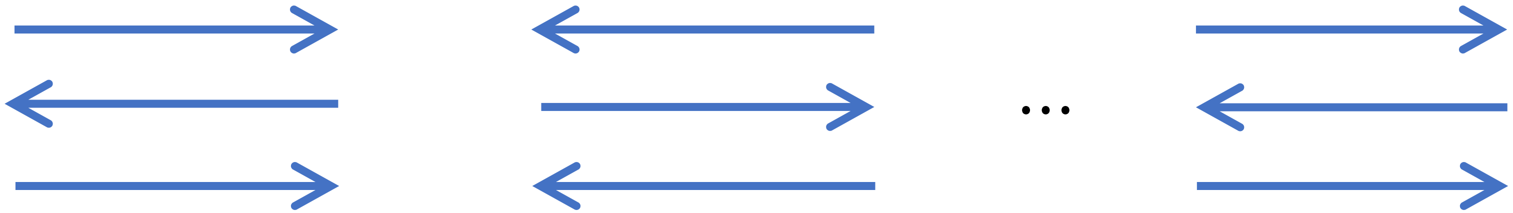
Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys: $\mathbf{pk}_1$ $\quad$ $\mathbf{pk}_2$ $\quad \cdots \quad$ $\mathbf{pk}_N$

Ciphertexts: m_1 $\quad$ m_2 $\quad$ m_N

C $\quad$ $C(m_1, m_2, \dots, m_N)$ **(Compact)**

Decryption Protocol:



Recovery: $C(m_1, m_2, \dots, m_N)$

- **Security:** adversary can learn nothing beyond $C(m_1, m_2, \dots, m_N)$.

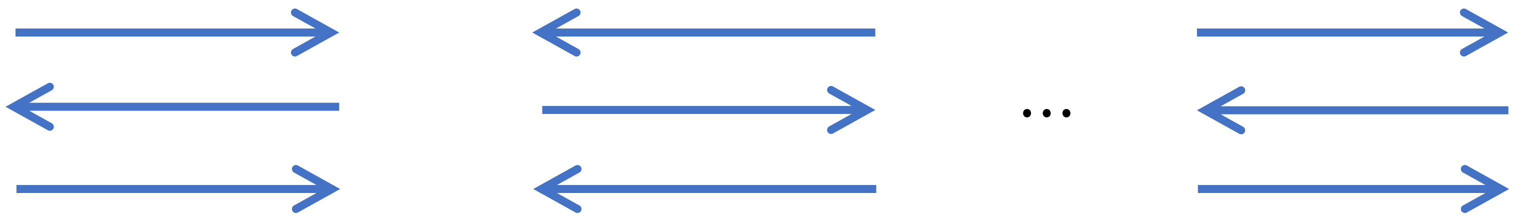
Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \dots, m_N)$

Decryption Protocol:



$\cdots$

Recovery: $C(m_1, m_2, \dots, m_N)$

- (Implicit) **Reusability**: decryption can run for different $C(m_1, m_2, \dots, m_N)$ without re-generating the public keys/ciphertexts.

Multi-key Fully-Homomorphic Encryption [LTV12]

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

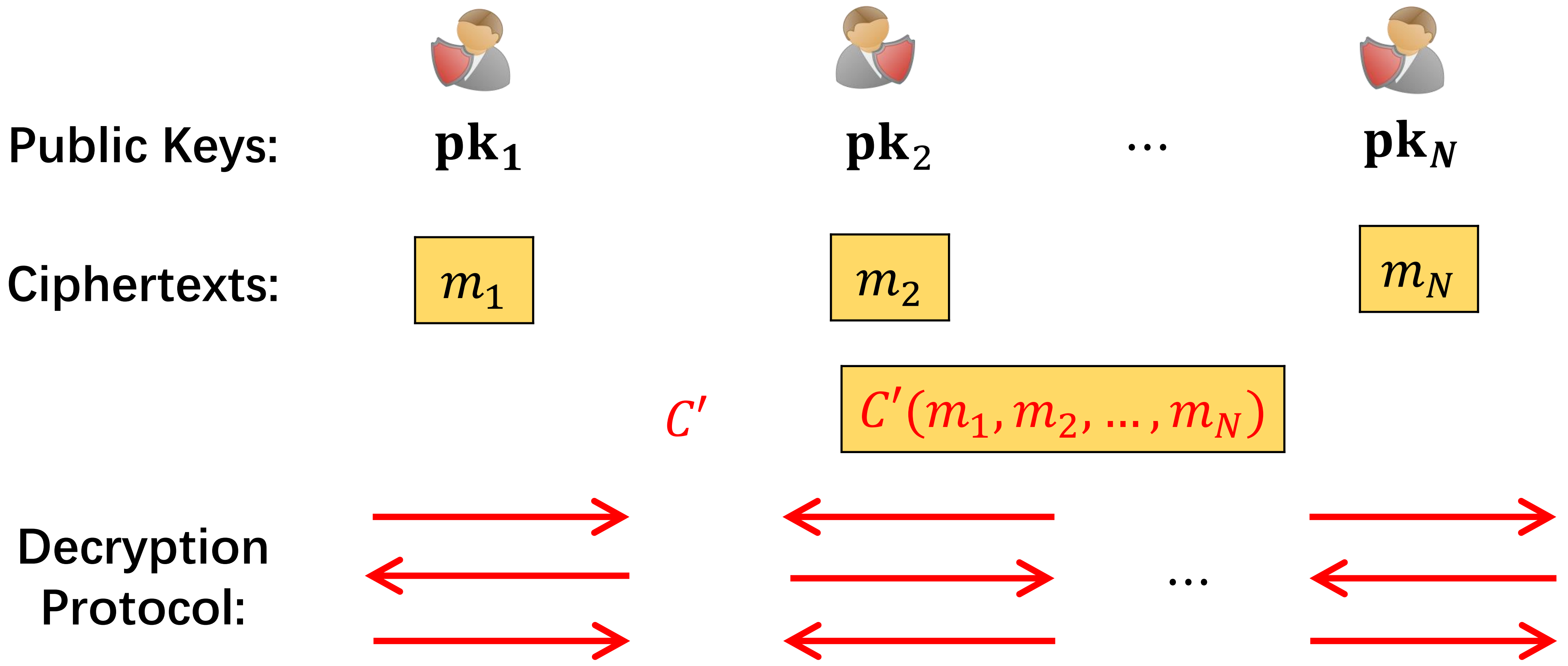
Ciphertexts: m_1 m_2 m_N

C' $C'(m_1, m_2, \dots, m_N)$

$\cdots$

- (Implicit) **Reusability**: decryption can run for different $C(m_1, m_2, \dots, m_N)$ without re-generating the public keys/ciphertexts.

Multi-key Fully-Homomorphic Encryption [LTV12]



- (Implicit) **Reusability**: decryption can run for different $C(m_1, m_2, \ldots, m_N)$ without re-generating the public keys/ciphertexts.

Multi-key Fully-Homomorphic Encryption [LTV12]



Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

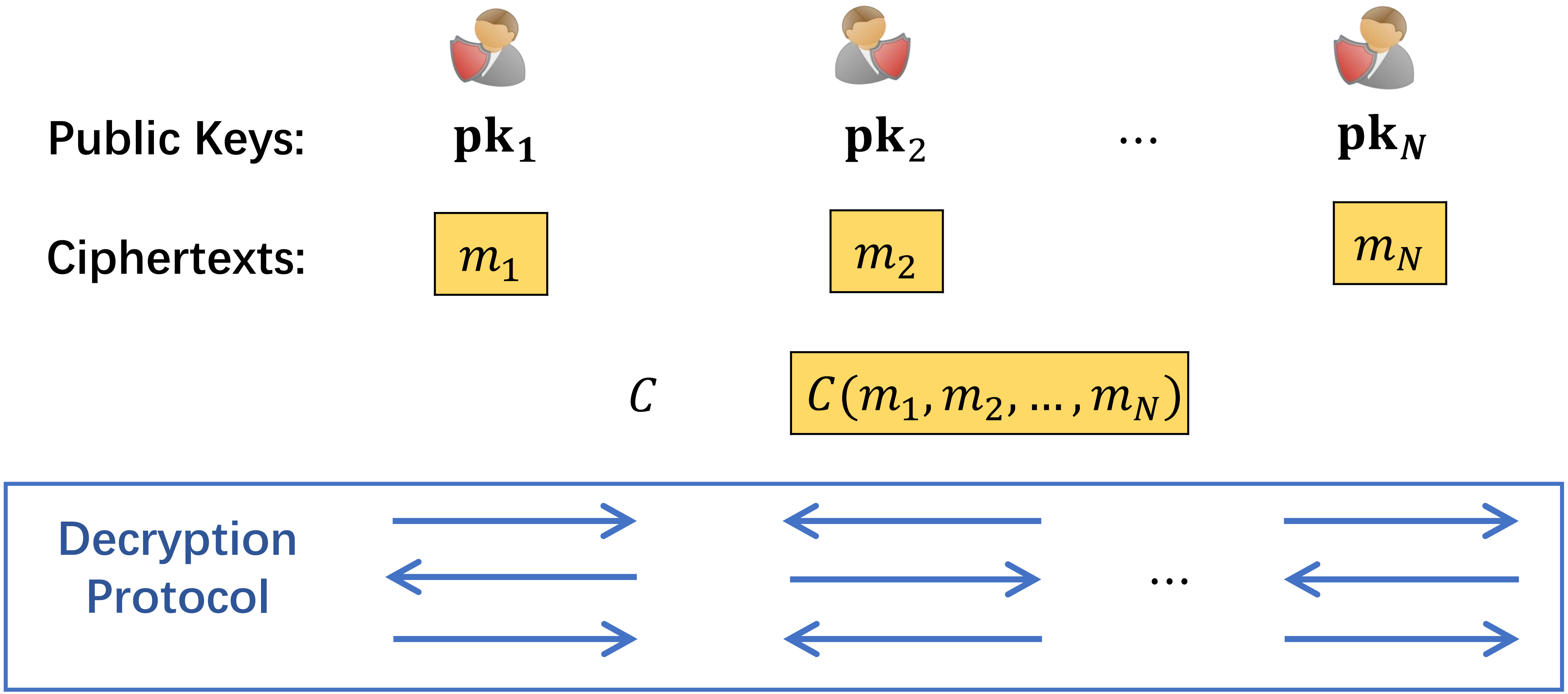
C' $C'(m_1, m_2, \dots, m_N)$

Decryption Protocol: $\cdots$

Recovery: $C'(m_1, m_2, \dots, m_N)$

- (Implicit) **Reusability**: decryption can run for different $C(m_1, m_2, \dots, m_N)$ without re-generating the public keys/ciphertexts.

MK-FHE with 1-Round Decryption [MW16]



MK-FHE with 1-Round Decryption [MW16]

				
Public Keys:	$\mathbf{pk}_1$	$\mathbf{pk}_2$	$\cdots$	$\mathbf{pk}_N$
Ciphertexts:	m_1	m_2		m_N

C $\quad$ $C(m_1, m_2, \dots, m_N)$

1-round Decryption: → ← ⋯ →

MK-FHE with 1-Round Decryption [MW16]

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \dots, m_N)$

1-round Decryption: $\cdots$

Public Recovery: $C(m_1, m_2, \dots, m_N)$

Applications

Applications

- 2-round multiparty computation [MW16]
- Spooky encryption [DHRW16]
- Homomorphic secret sharing [BGI16, BGI17]
- obfuscation & functional encryption combiners [AJNSY16, AJS17]
- Multiparty obfuscation [HIJKSY17]
- Homomorphic time-lock puzzles [MT19,BDGM19]
- Ad-hoc multi-input functional encryption [ACFGOT20]
- ……

Applications

- 2-round multiparty computation [MW16]
- Spooky encryption [DHRW16]
- Homomorphic secret sharing [BGI16, BGI17]
- obfuscation & functional encryption combiners [AJNSY16, AJS17]
- Multiparty obfuscation [HIJKSY17]
- Homomorphic time-lock puzzles [MT19,BDGM19]
- Ad-hoc multi-input functional encryption [ACFGOT20]
- ……

Prior works on
Multi-key FHE with 1-round decryption

- [CM15, MW16, BP16, PS16] need a trusted setup.
- [DHRW16] sub-exponentially secure indistinguishable obfuscation.

In the plain model, does Multi-key FHE with 1-round decryption exist?

Our Results

Our Results

1. Multi-key FHE with 1-round decryption in the plain model from Learning with Error (LWE), Ring-LWE, and Decisional Small Polynomial Ratio problem.
   - O(1)-party Multi-key FHE from only LWE.

Our Results

1. Multi-key FHE with 1-round decryption in the plain model from Learning with Error (LWE), Ring-LWE, and Decisional Small Polynomial Ratio problem.
   - O(1)-party Multi-key FHE from only LWE.

2. Multiparty Homomorphic Encryption (a weaker notion of MK-FHE) from LWE.

Multiparty Homomorphic Encryption: A weakening of MK-FHE



Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \ldots, m_N)$

Partial Decryption: $\longrightarrow$ $\longleftarrow$ $\cdots$ $\longrightarrow$

Public Recovery: **Partial Decryptions** $\rightarrow C(m_1, m_2, \ldots, m_N)$

Multiparty Homomorphic Encryption: A weakening of MK-FHE

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \ldots, m_N)$

Partial Decryption: → ← $\cdots$ →

Public Recovery: C, **Partial Decryptions** $\rightarrow C(m_1, m_2, \ldots, m_N)$

Multiparty Homomorphic Encryption: A weakening of MK-FHE

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \ldots, m_N)$

Partial Decryption: $\longrightarrow$ $\longleftarrow$ $\cdots$ $\longrightarrow$

Public Recovery: C, **Partial Decryptions** $\rightarrow C(m_1, m_2, \ldots, m_N)$

- It implies 2-round reusable multiparty computation with compact communication complexity.

Multiparty Homomorphic Encryption: A weakening of MK-FHE

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

Multiparty Homomorphic Encryption: A weakening of MK-FHE

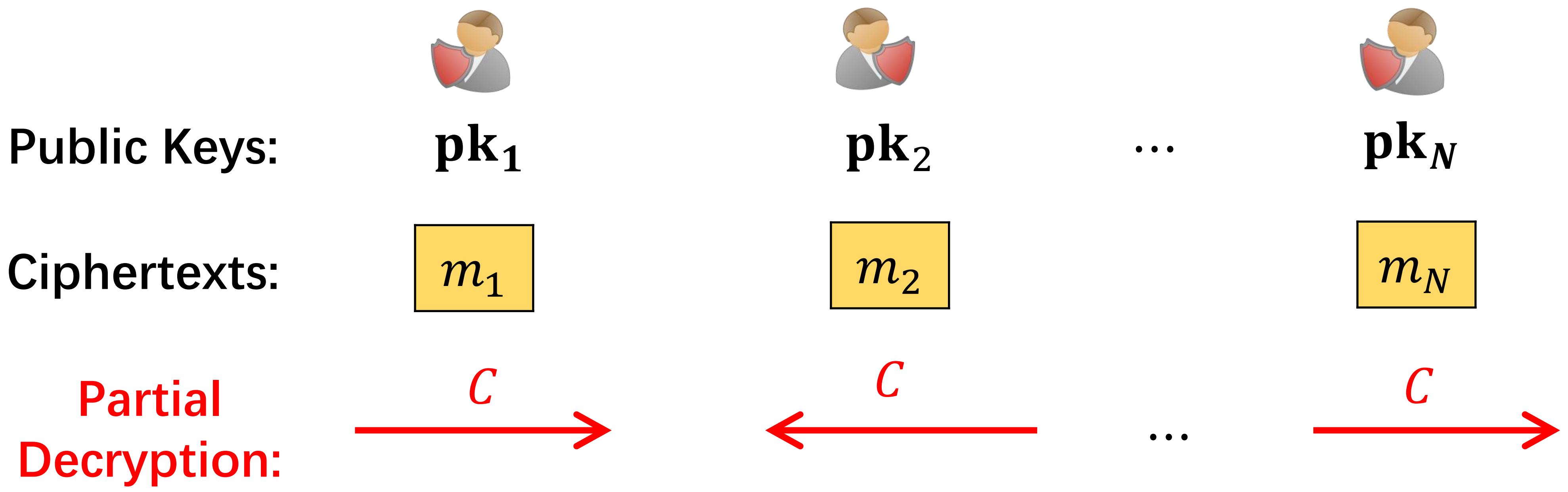
  

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ … $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

Partial Decryption: C C … C

Multiparty Homomorphic Encryption: A weakening of MK-FHE



Multiparty Homomorphic Encryption: A weakening of MK-FHE

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ … $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

Partial Decryption: C → C ← … C →

Public Recovery: C, **Partial Decryptions** $\rightarrow C(m_1, m_2, \ldots, m_N)$

Multiparty Homomorphic Encryption: A weakening of MK-FHE

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ $\cdots$ $\mathbf{pk}_N$

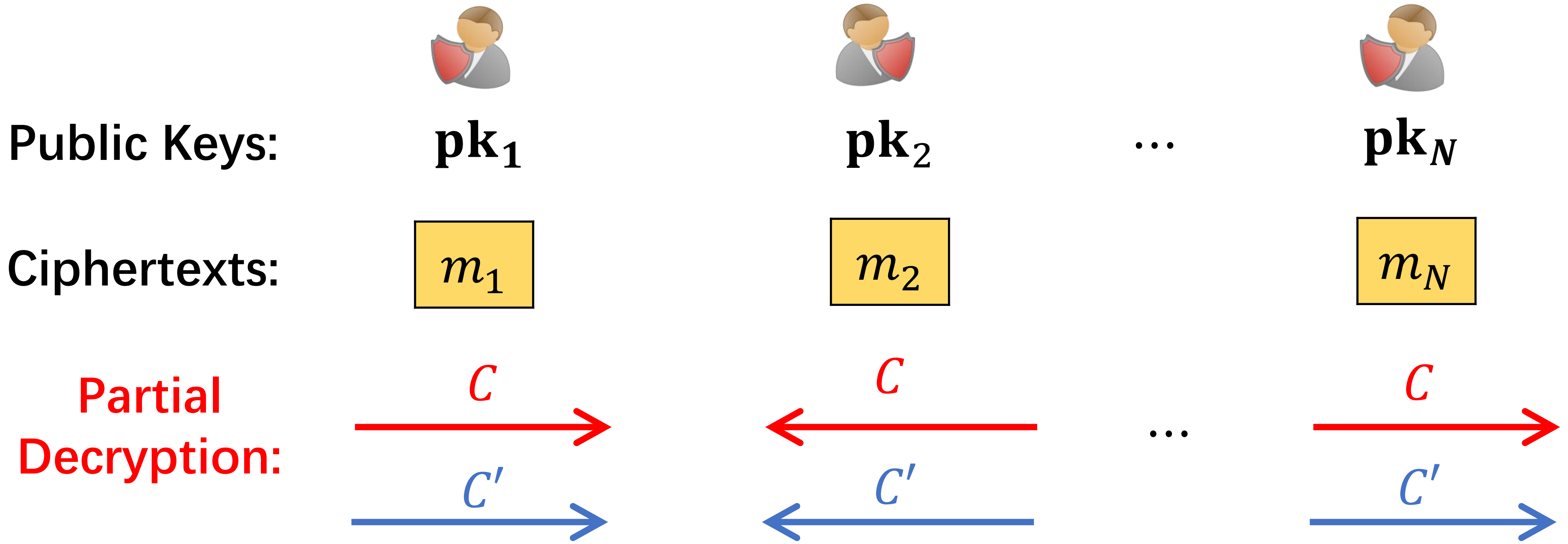
Ciphertexts: m_1 m_2 m_N

Partial Decryption: C → C ← $\cdots$ C →

Public Recovery: C, **Partial Decryptions** $\rightarrow C(m_1, m_2, \ldots, m_N)$

- **Reusability:** public keys can be reused for different circuits.
- **Compactness:** communication complexity is independent of the circuit.

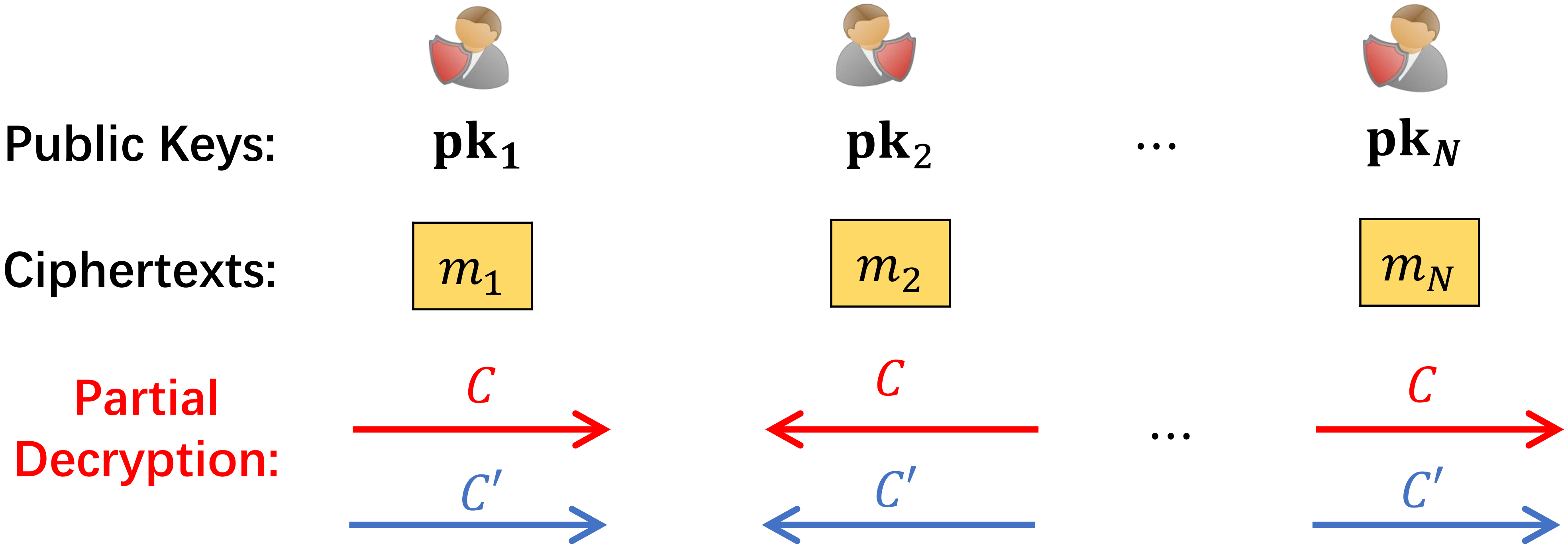
Multiparty Homomorphic Encryption: A weakening of MK-FHE



Public Recovery: C, **Partial Decryptions** $\rightarrow C(m_1, m_2, \ldots, m_N)$

- **Reusability:** public keys can be reused for different circuits.
- **Compactness:** communication complexity is independent of the circuit.

Multiparty Homomorphic Encryption: A weakening of MK-FHE



Public Recovery: C, **Partial Decryptions** $\rightarrow C(m_1, m_2, \ldots, m_N)$

- **Reusability:** public keys can be reused for different circuits.
- **Compactness:** communication complexity is independent of the circuit.

Multiparty Homomorphic Encryption: A weakening of MK-FHE

Public Keys: $\mathbf{pk}_1$ $\mathbf{pk}_2$ … $\mathbf{pk}_N$

Ciphertexts: m_1 m_2 m_N

Partial Decryption: C C' C C' … C C'

Public Recovery: C, **Partial Decryptions** $\rightarrow C(m_1, m_2, \ldots, m_N)$

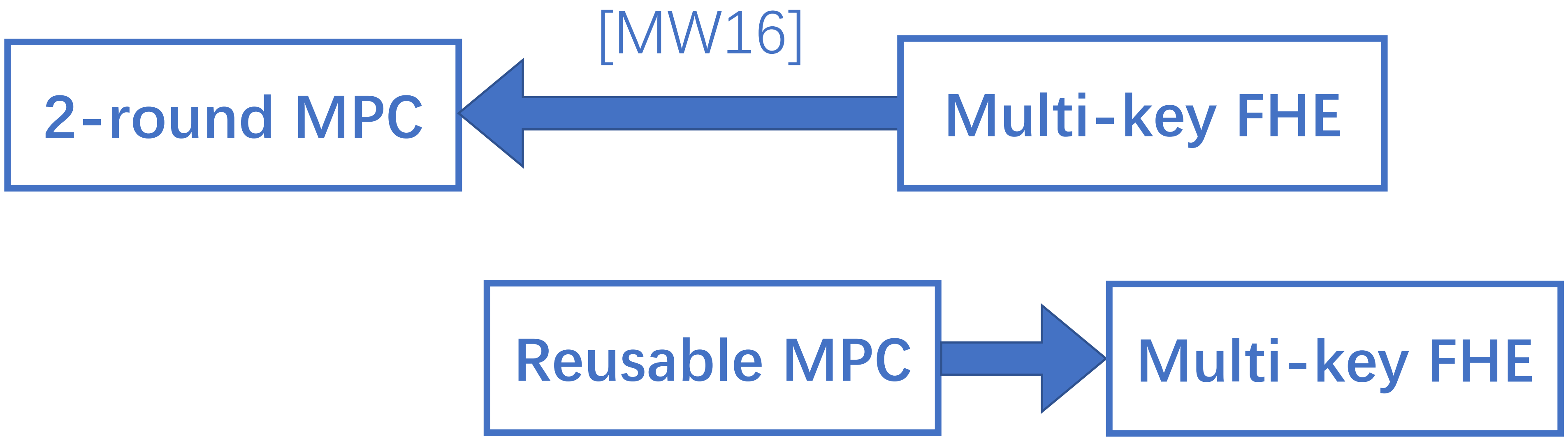
- **Reusability:** public keys can be reused for different circuits.
- **Compactness:** communication complexity is independent of the circuit.

- It implies 2-round Multiparty Computation.

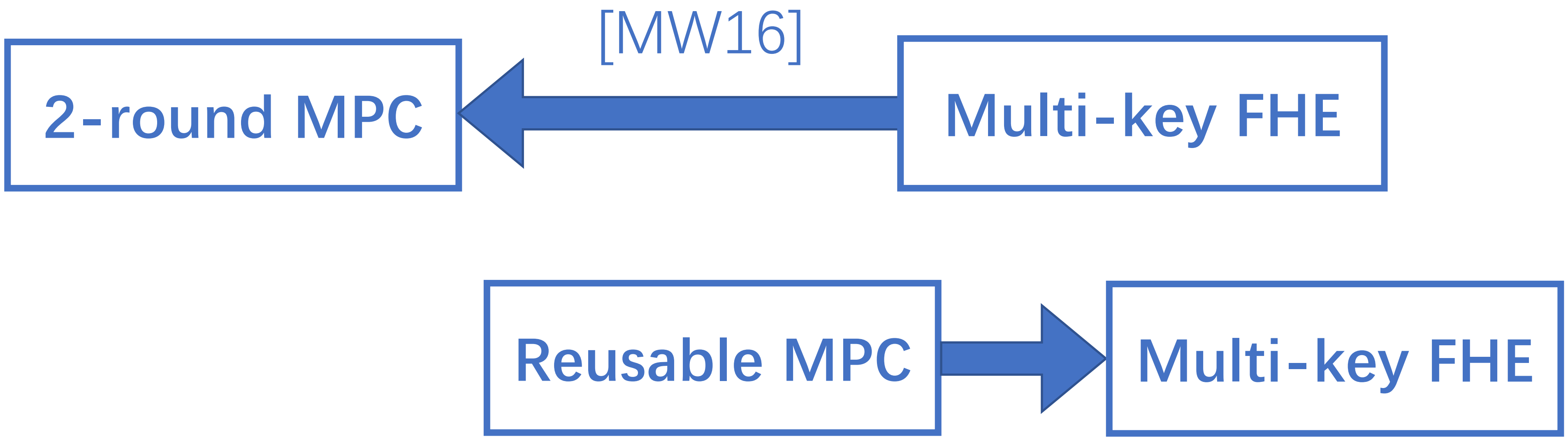
Our Approach

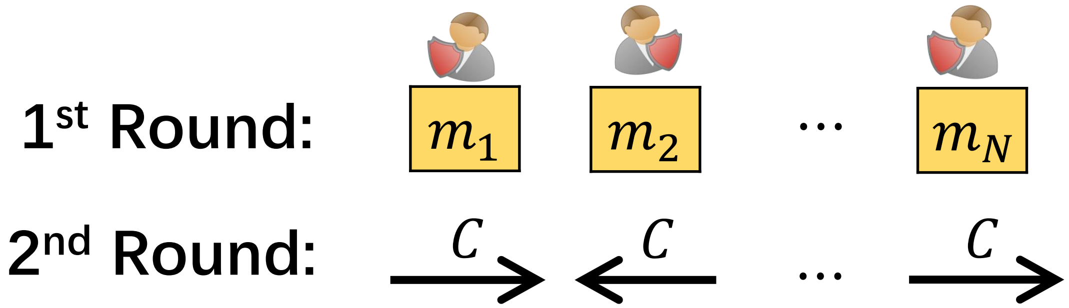
Our Approach

Our Approach

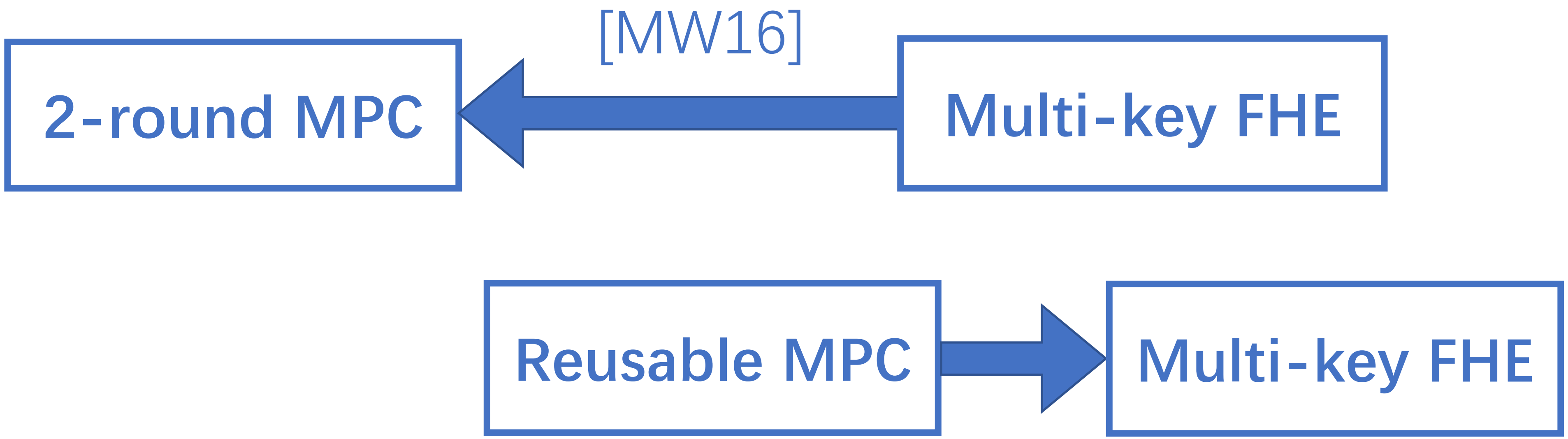


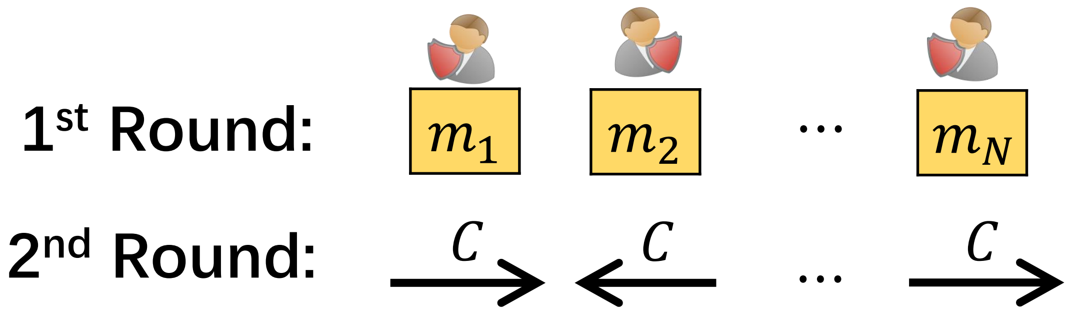
Our Approach





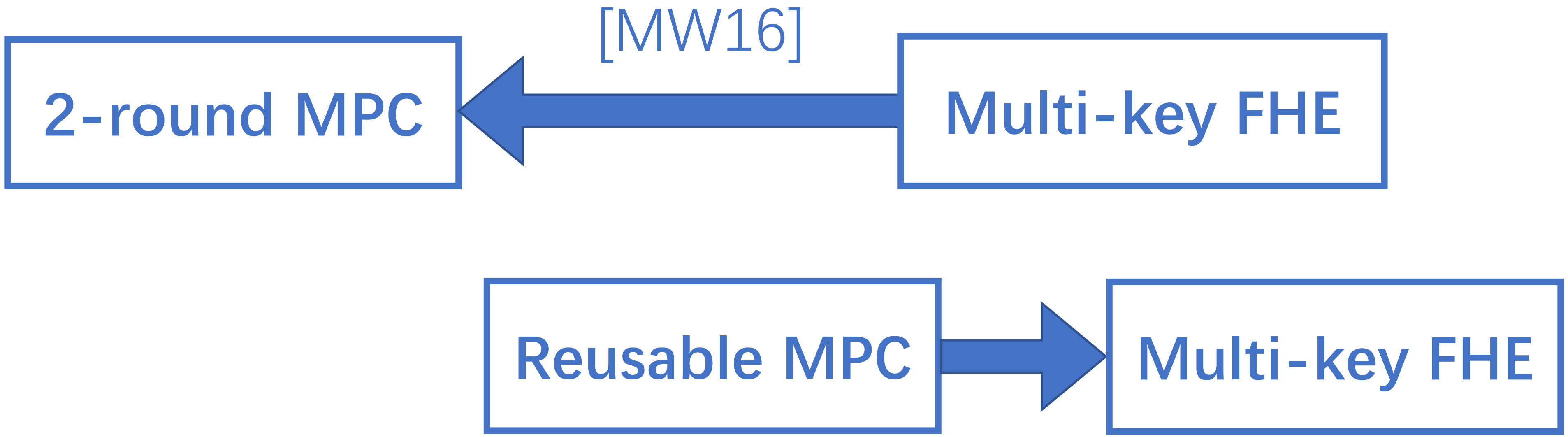
Our Approach

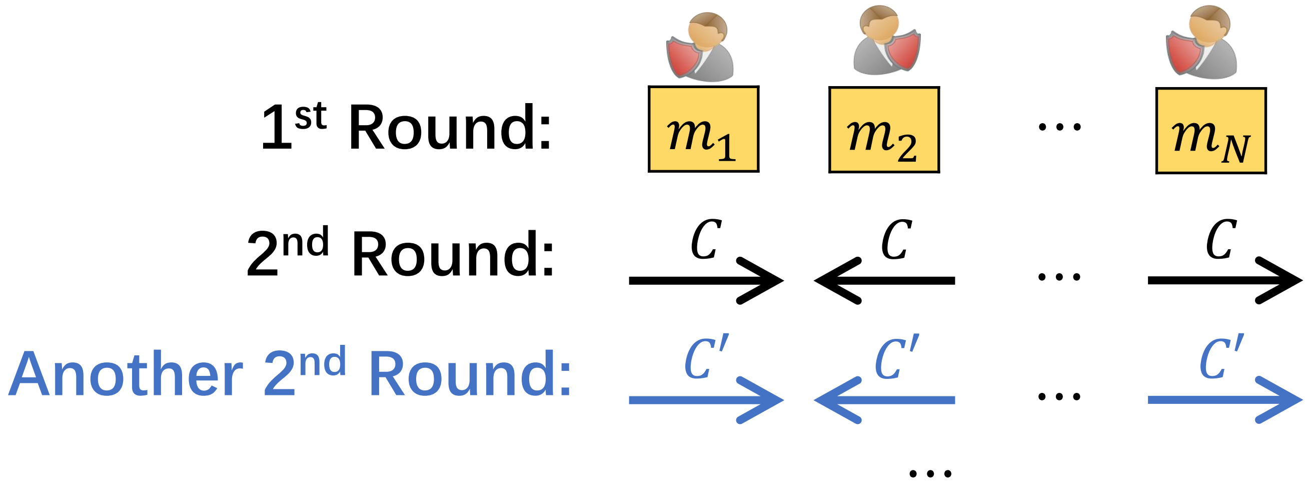




- **Reusability**: 1st round is reusable.
- **See Also**:
  - [BL20], from bilinear maps.
  - [BGMM20], from DDH or Succinct 1st msg MPC

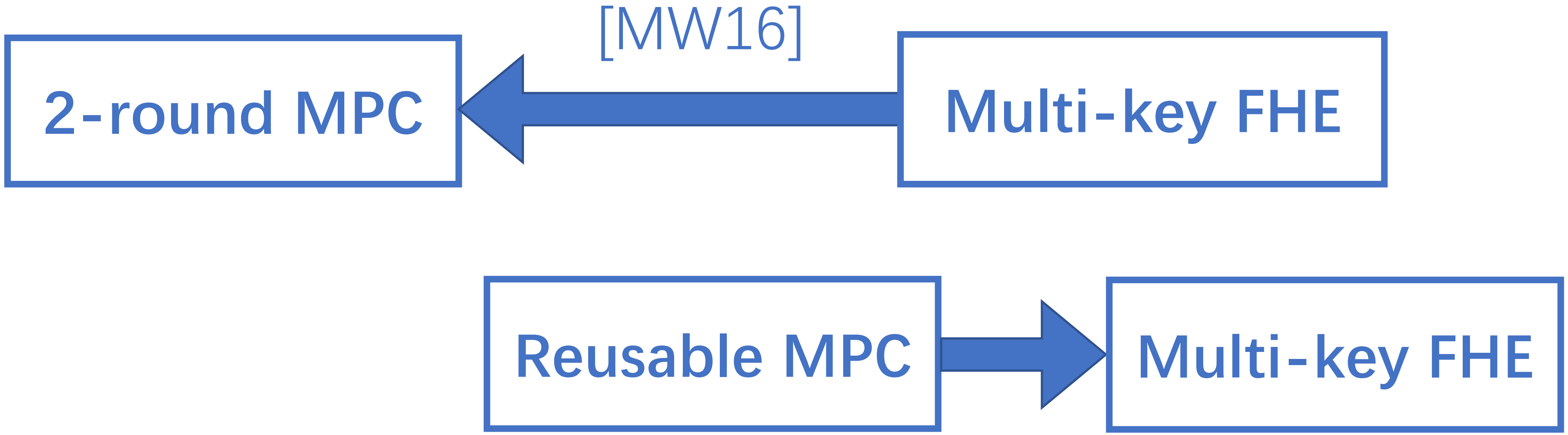
Our Approach

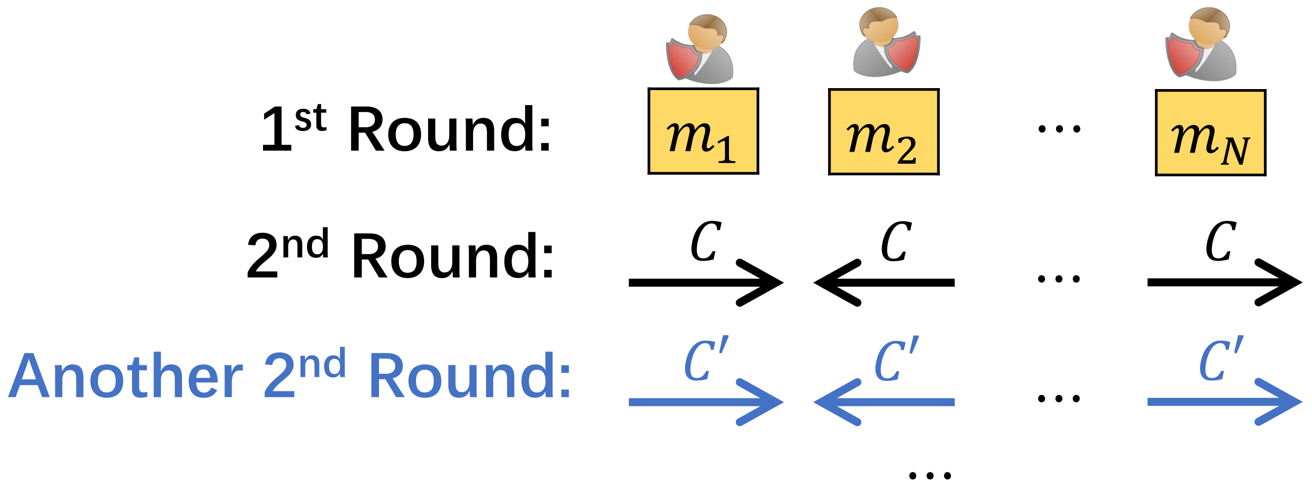




- **Reusability**: 1st round is reusable.
- **See Also**:
  - [BL20], from bilinear maps.
  - [BGMM20], from DDH or Succinct 1st msg MPC

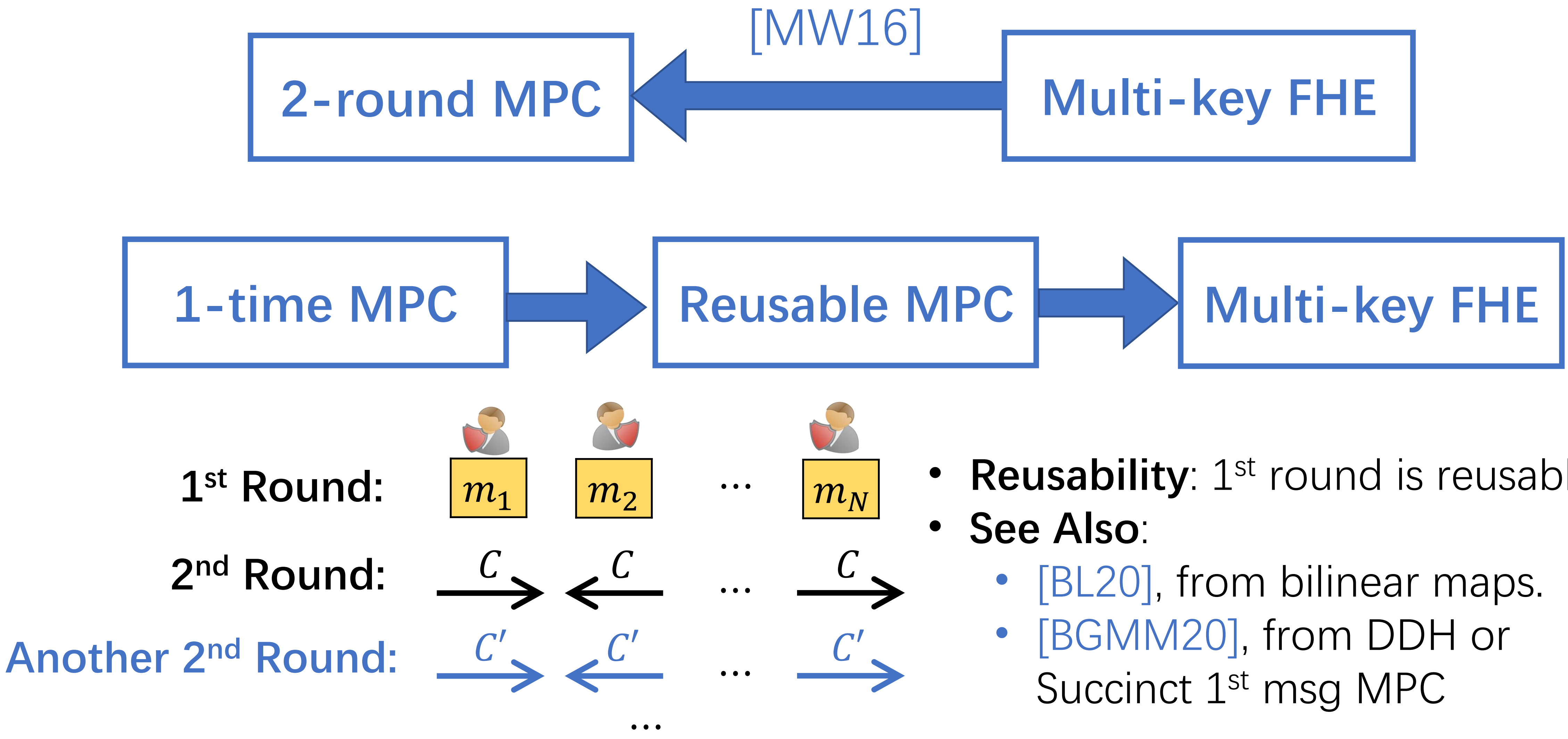
Our Approach





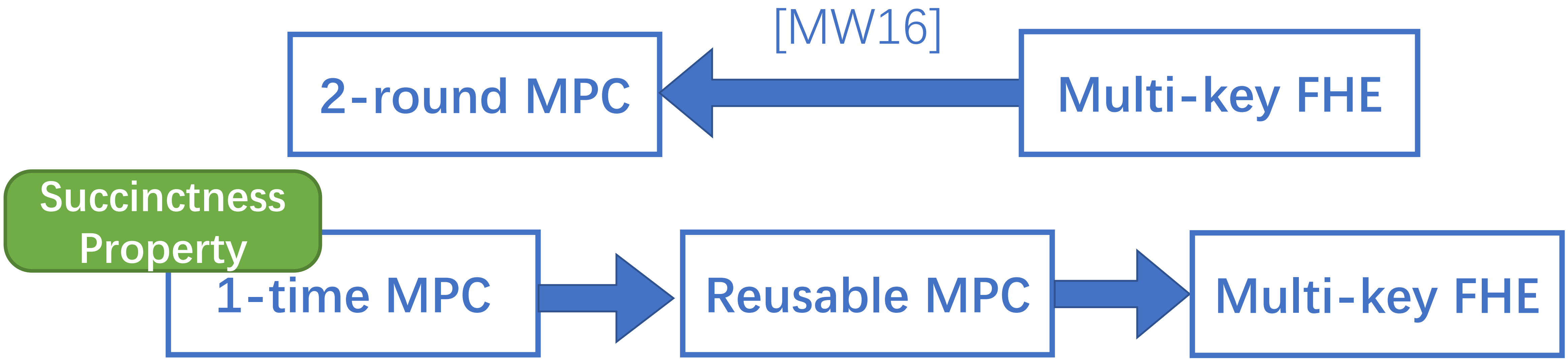
- **Reusability**: 1st round is reusable.
- **See Also**:
  - [BL20], from bilinear maps.
  - [BGMM20], from DDH or Succinct 1st msg MPC

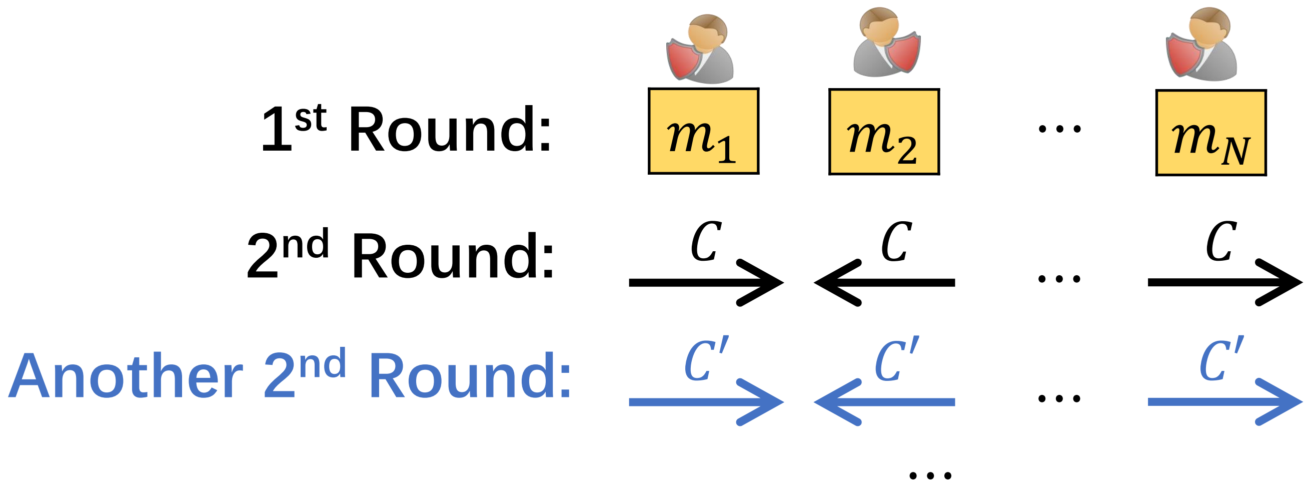
Our Approach



- **Reusability**: 1st round is reusable.
- **See Also**:
  - [BL20], from bilinear maps.
  - [BGMM20], from DDH or Succinct 1st msg MPC

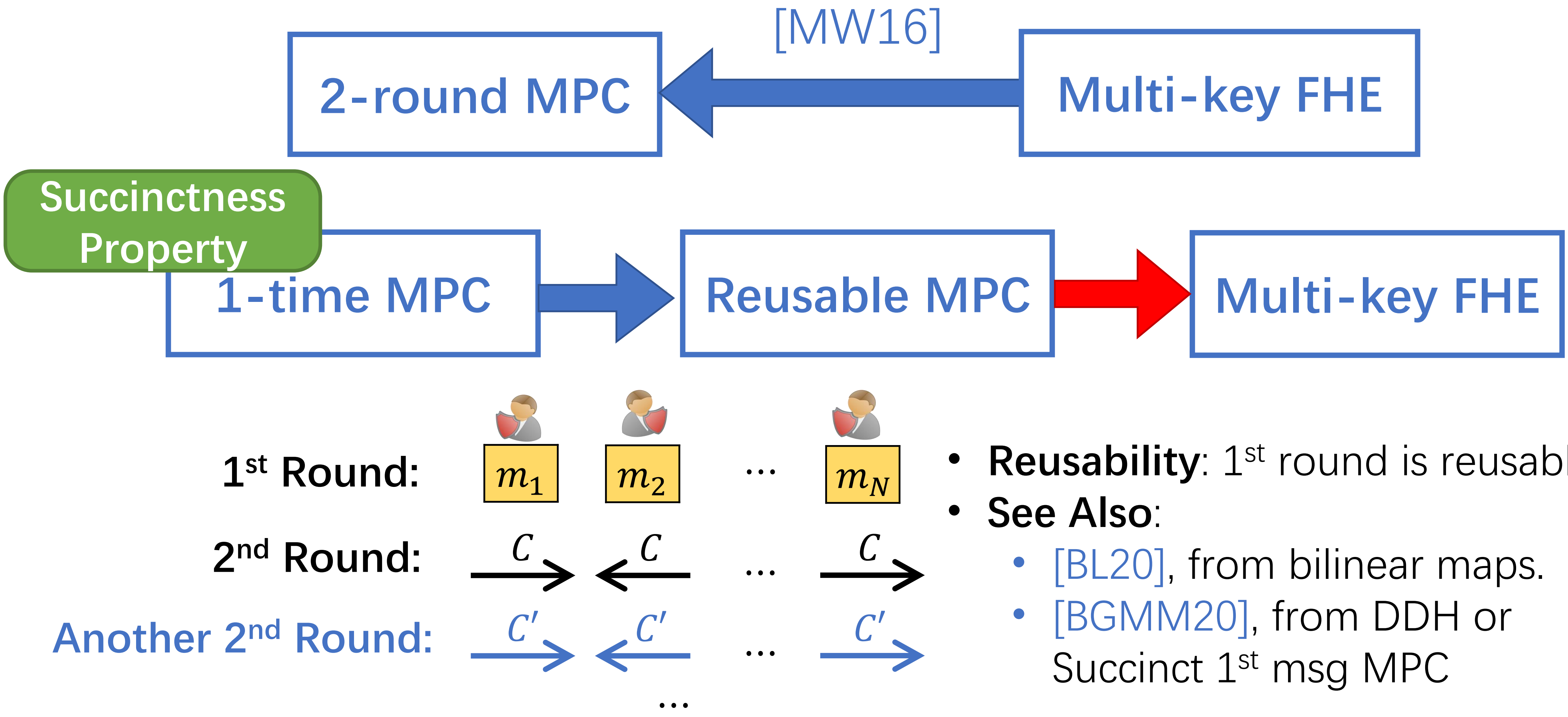
Our Approach





- **Reusability**: 1st round is reusable.
- **See Also**:
  - [BL20], from bilinear maps.
  - [BGMM20], from DDH or Succinct 1st msg MPC

Our Approach



- **Reusability**: 1st round is reusable.
- **See Also**:
  - [BL20], from bilinear maps.
  - [BGMM20], from DDH or Succinct 1st msg MPC

Reusable MPC → Multi-key FHE

Reusable MPC → Multi-key FHE

- [LTV12] is in plain mode, but has a multi-round decryption protocol.

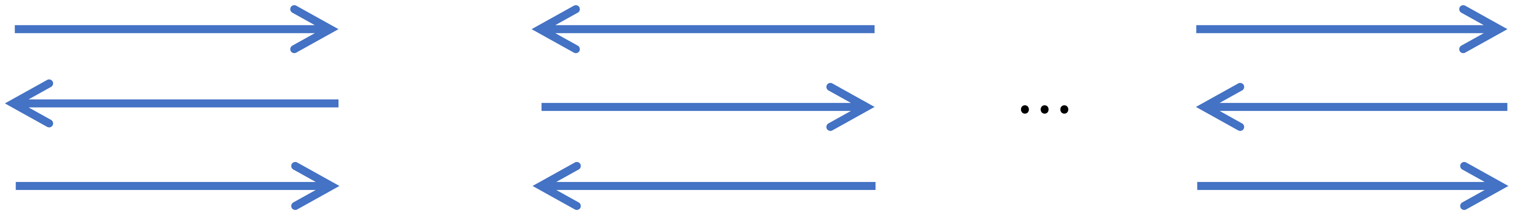
Reusable MPC → Multi-key FHE

- [LTV12] is in plain mode, but has a multi-round decryption protocol.

Public Keys:	$\mathbf{pk}_1$	$\mathbf{pk}_2$	$\cdots$	$\mathbf{pk}_N$
Ciphertexts:	m_1	m_2		m_N

C $\quad$ $C(m_1, m_2, \ldots, m_N)$

Multi-round Decryption:



$C(m_1, m_2, \ldots, m_N)$

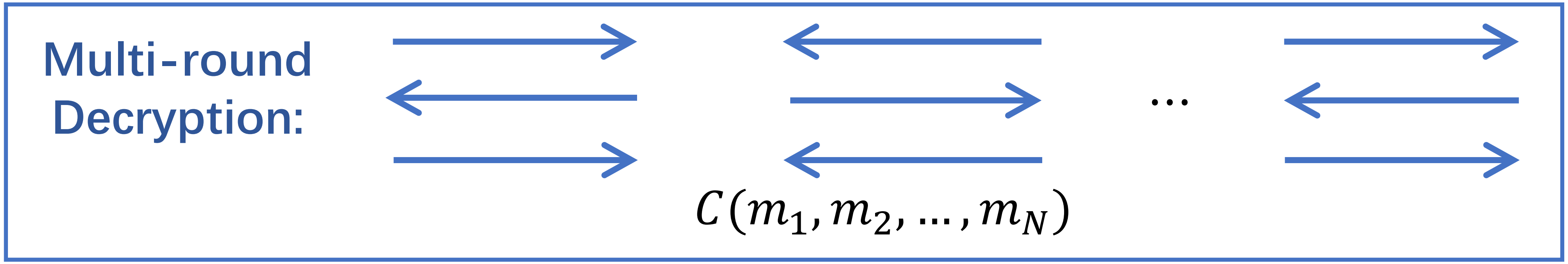
Reusable MPC → Multi-key FHE

- [LTV12] is in plain mode, but has a multi-round decryption protocol.

Public Keys:	$\mathbf{pk}_1$	$\mathbf{pk}_2$	⋯	$\mathbf{pk}_N$
Ciphertexts:	m_1	m_2		m_N

C $\quad C(m_1, m_2, \dots, m_N)$

- Run reusable MPC for $\mathbf{Dec}(\cdot)$.



Reusable MPC → Multi-key FHE

- [LTV12] is in plain mode, but has a multi-round decryption protocol.

Public Keys & Reusable MPC 1st Round

$\mathbf{pk}_1$ sk_1

$\mathbf{pk}_2$ sk_1 …

$\mathbf{pk}_N$ sk_N

Ciphertexts: m_1 m_2 m_N

C $C(m_1, m_2, \dots, m_N)$

- Run reusable MPC for $\mathrm{Dec}(\cdot)$.

Reusable MPC 2nd Round → ← … →

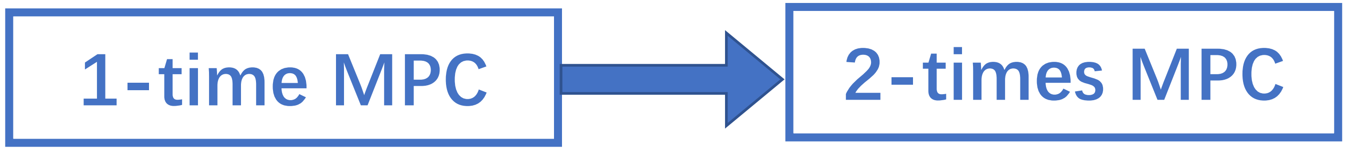
$C(m_1, m_2, \dots, m_N)$

Our Approach

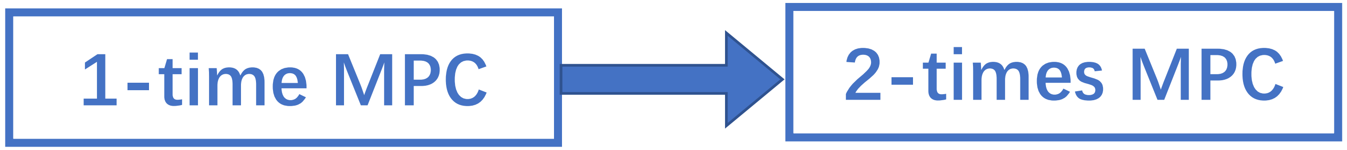
Our Approach

Reusable MPC: A Self-Synthesis Approach

Reusable MPC: A Self-Synthesis Approach

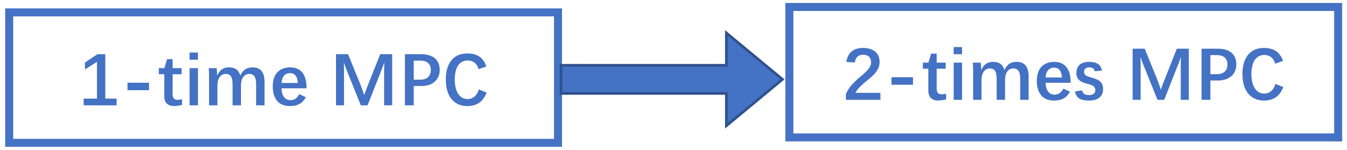


Reusable MPC: A Self-Synthesis Approach

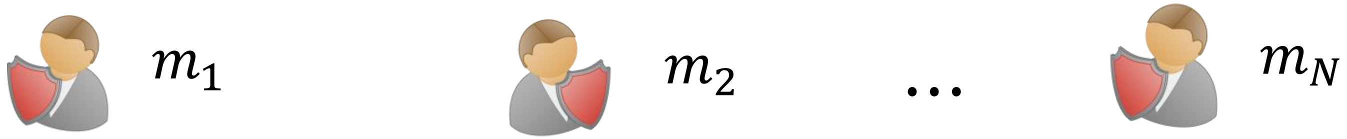


- Use 1-time MPC to generate 2 sets of fresh new 1st round messages

Reusable MPC: A Self-Synthesis Approach



- Use 1-time MPC to generate 2 sets of fresh new 1st round messages

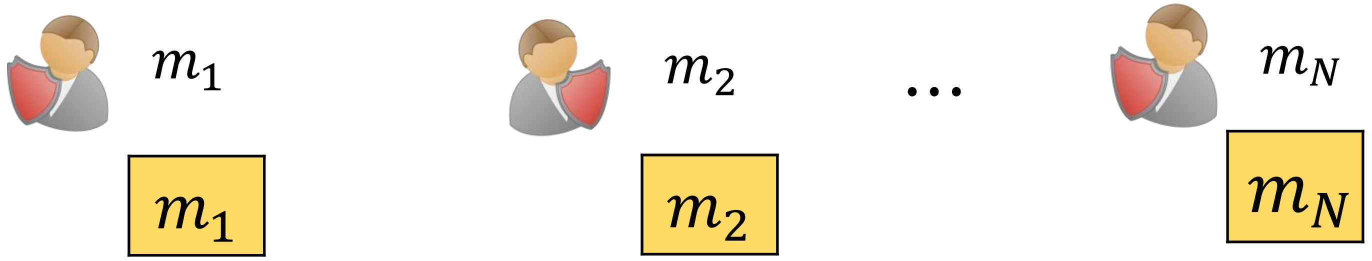


Reusable MPC: A Self-Synthesis Approach

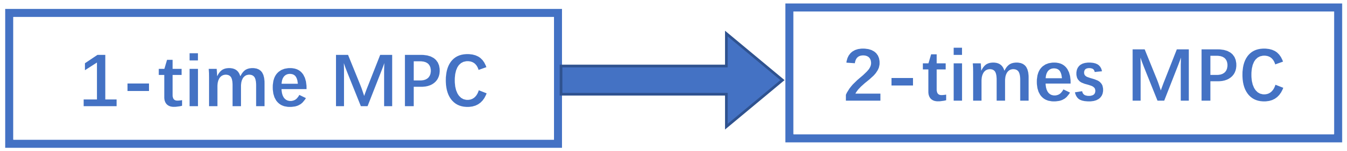
1-time MPC → **2-times MPC**

- Use 1-time MPC to generate 2 sets of fresh new 1st round messages

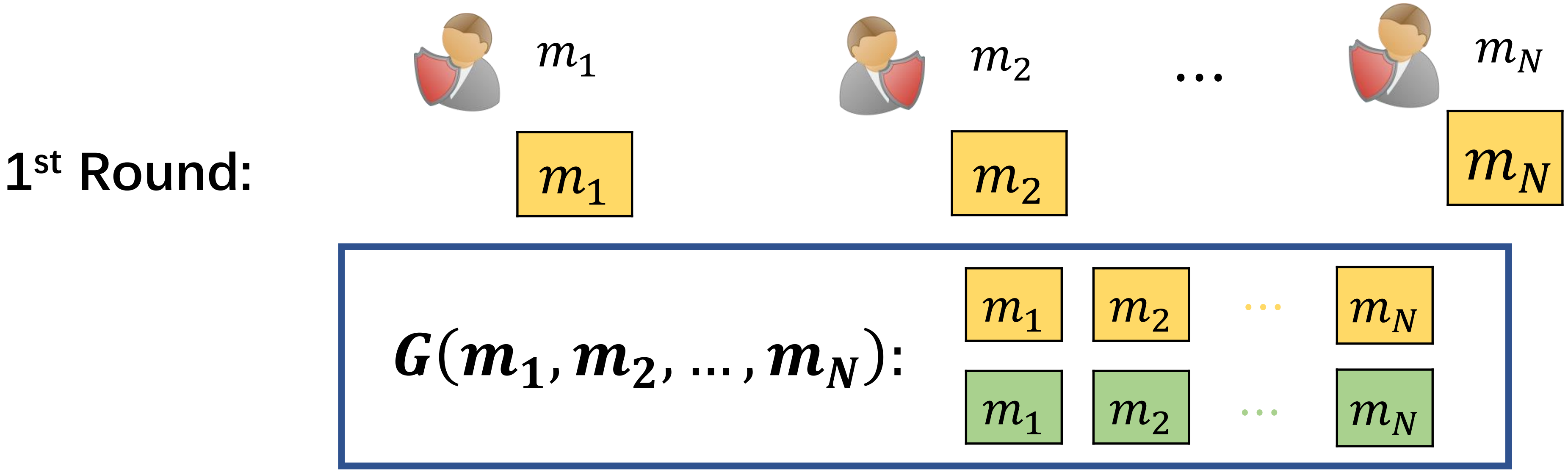
1st Round:



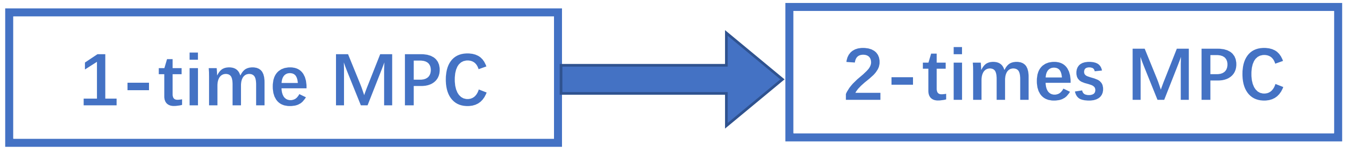
Reusable MPC: A Self-Synthesis Approach



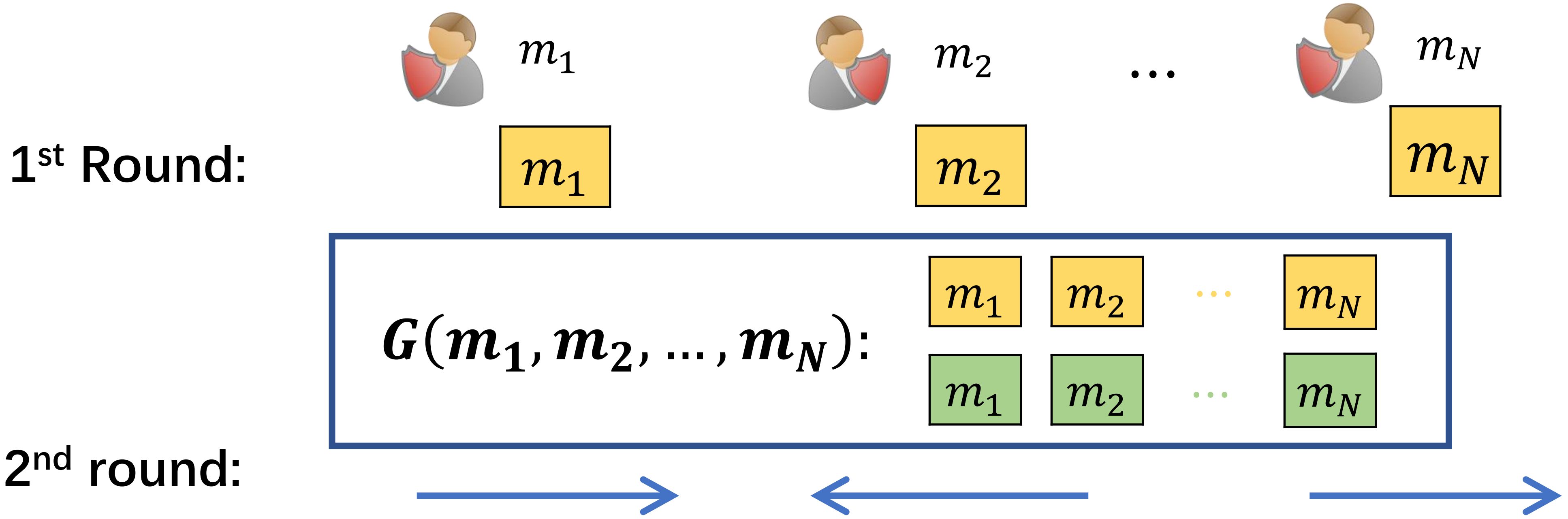
- Use 1-time MPC to generate 2 sets of fresh new 1st round messages



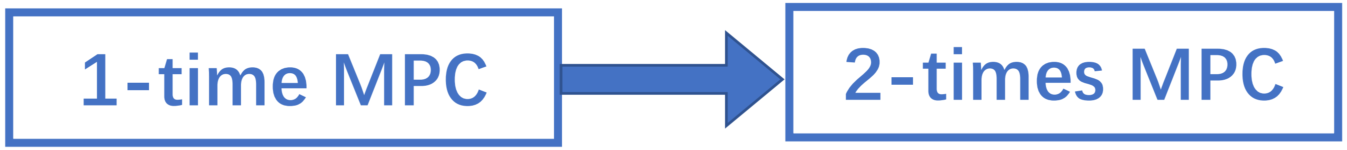
Reusable MPC: A Self-Synthesis Approach



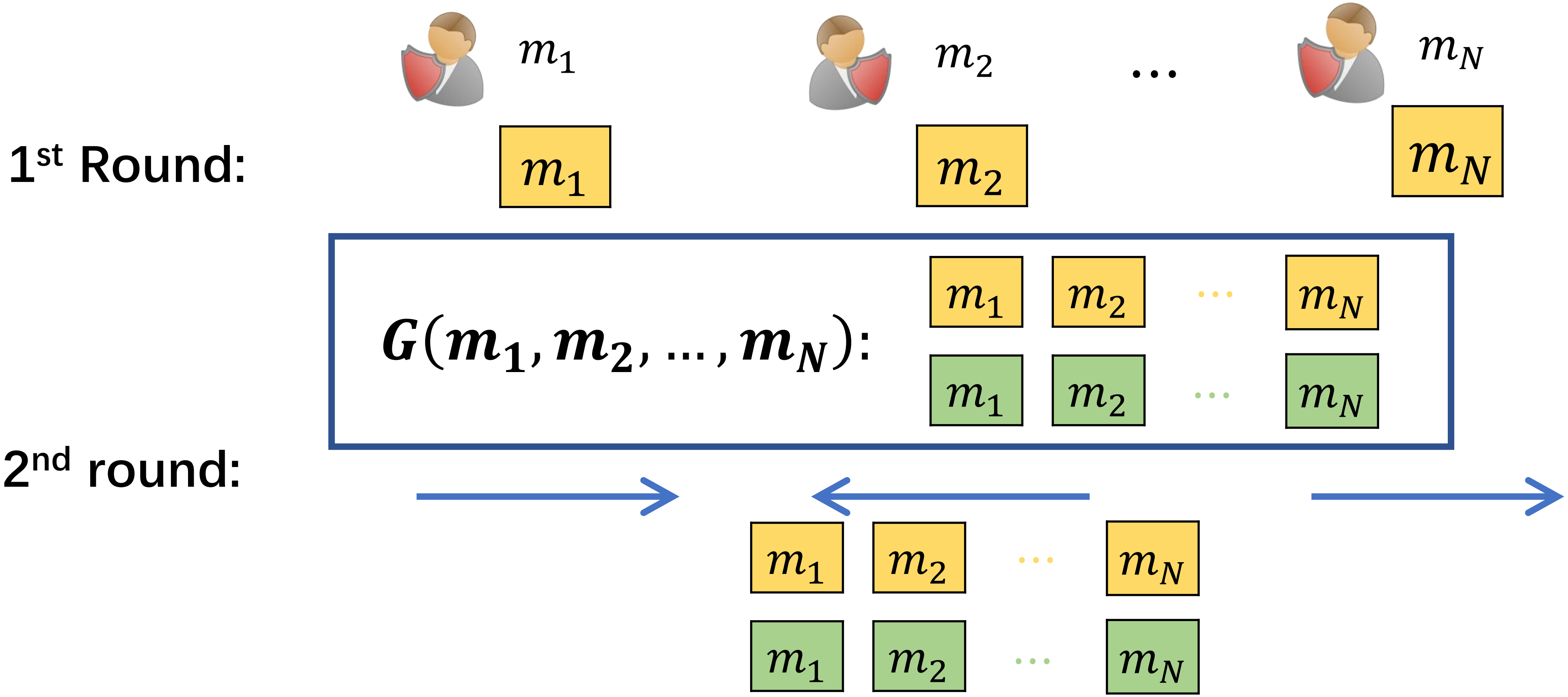
- Use 1-time MPC to generate 2 sets of fresh new 1st round messages



Reusable MPC: A Self-Synthesis Approach



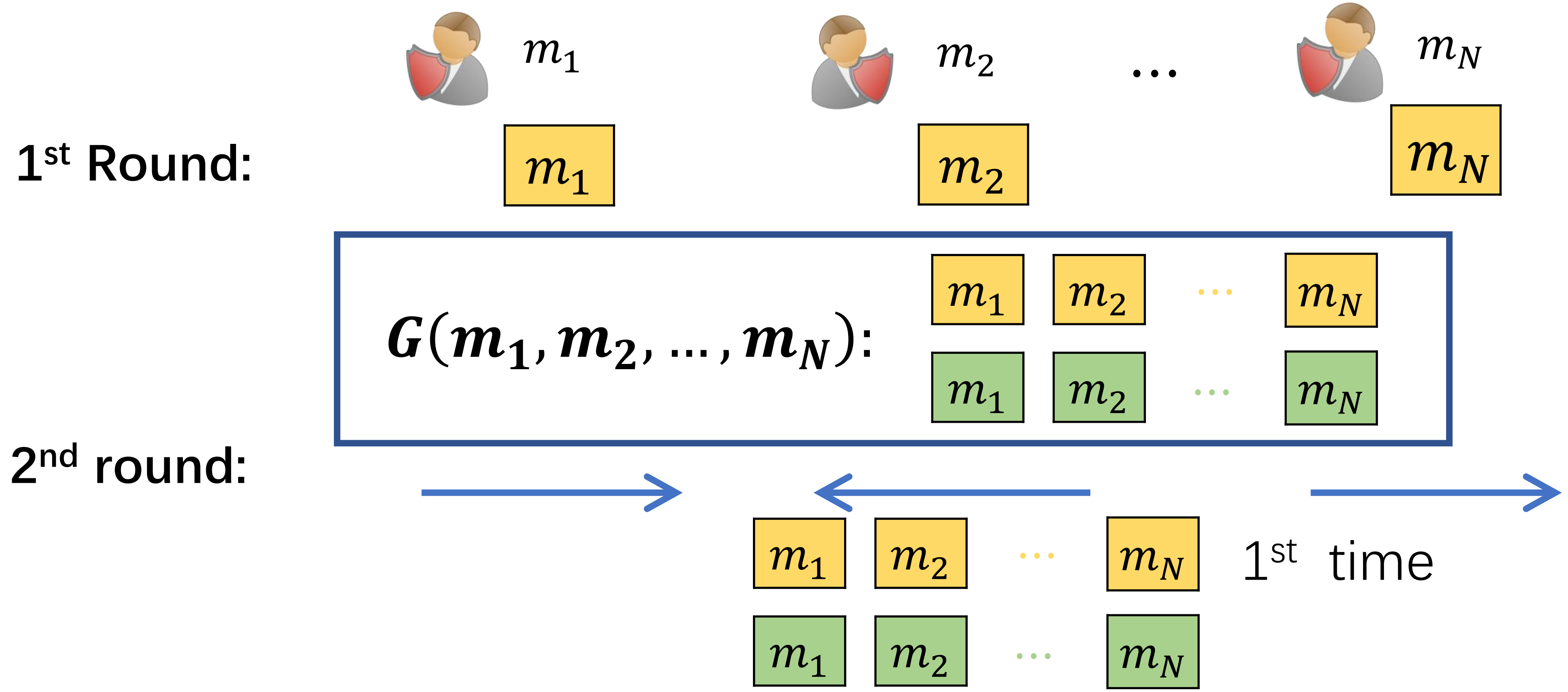
- Use 1-time MPC to generate 2 sets of fresh new 1st round messages



Reusable MPC: A Self-Synthesis Approach

1-time MPC → **2-times MPC**

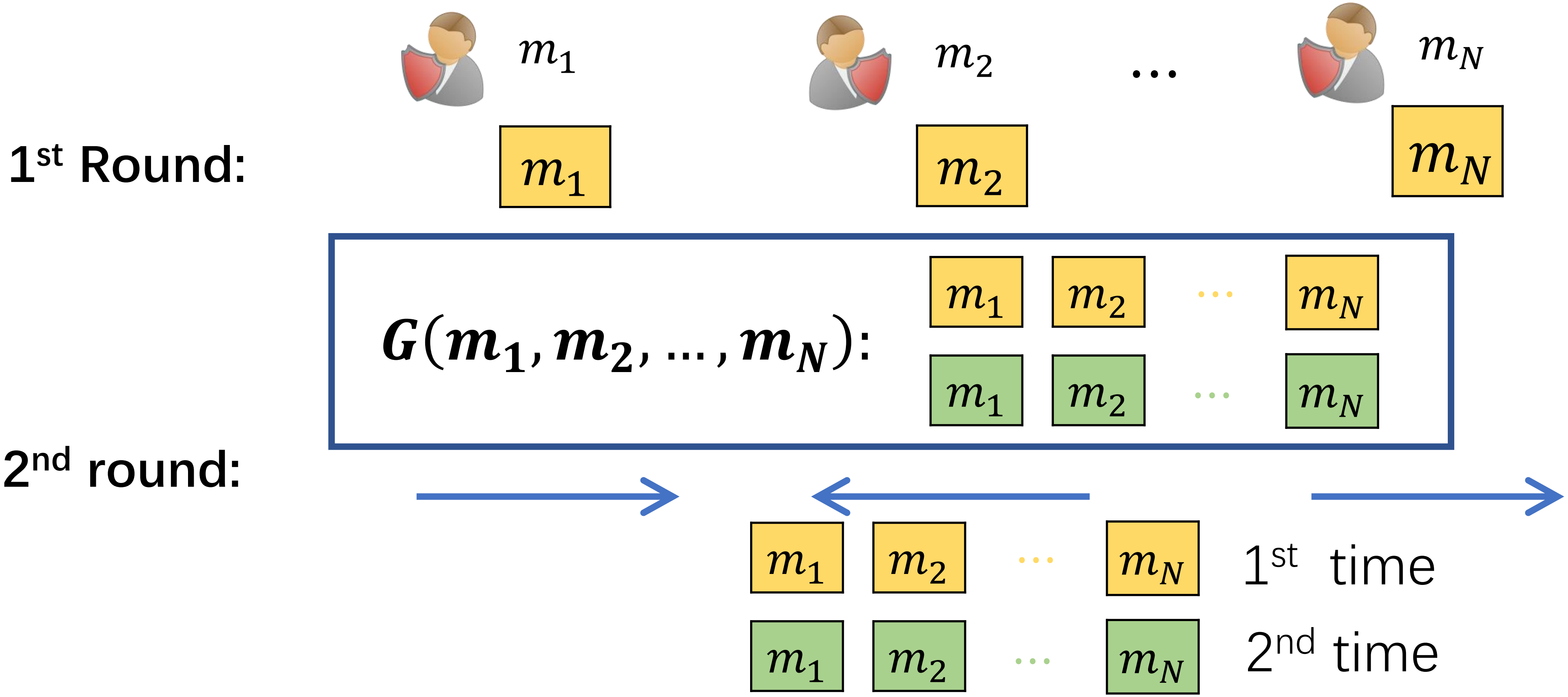
- Use 1-time MPC to generate 2 sets of fresh new 1st round messages



Reusable MPC: A Self-Synthesis Approach

1-time MPC → **2-times MPC**

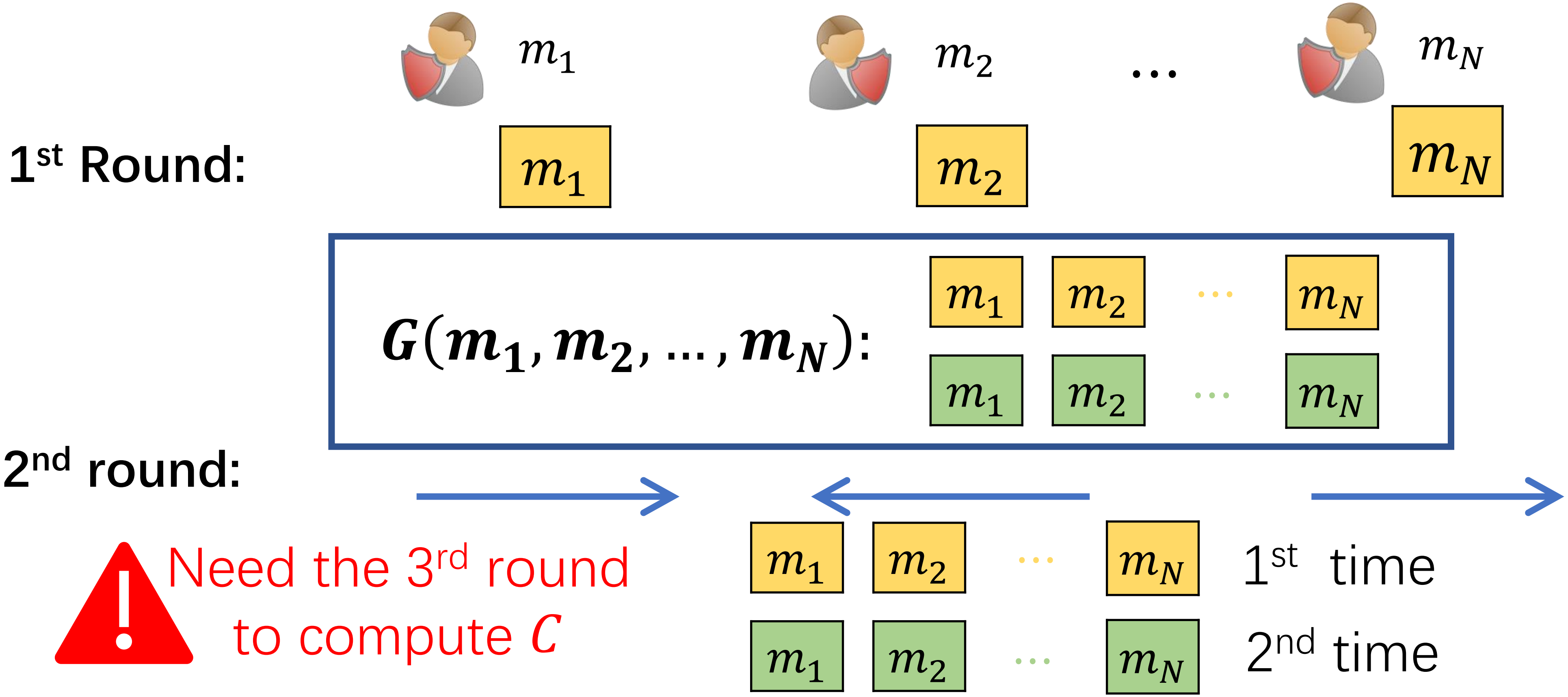
- Use 1-time MPC to generate 2 sets of fresh new 1st round messages



Reusable MPC: A Self-Synthesis Approach

1-time MPC → **2-times MPC**

- Use 1-time MPC to generate 2 sets of fresh new 1st round messages



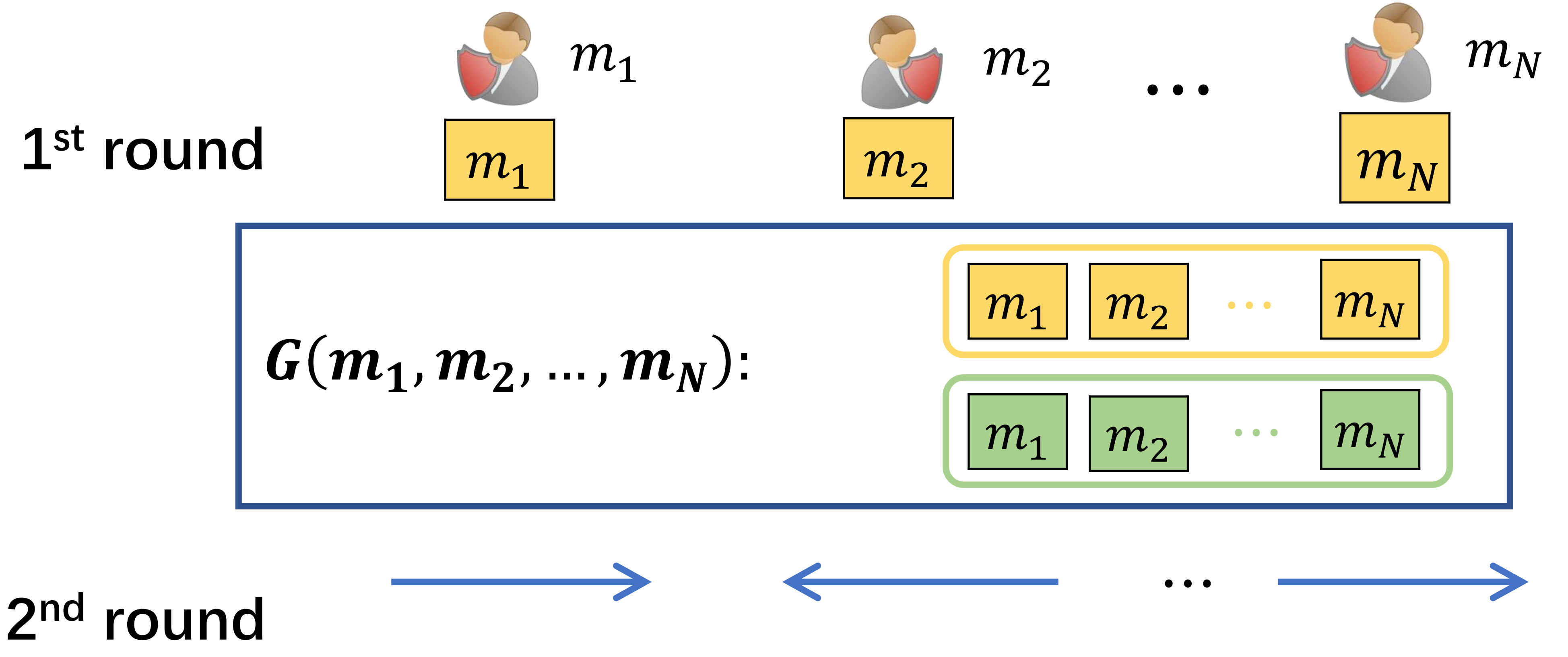
Round Compression to Rescue

Round Compression to Rescue

- Garble the 3rd round next message function $\mathbf{Next}^i$ to compress to 2 rounds.

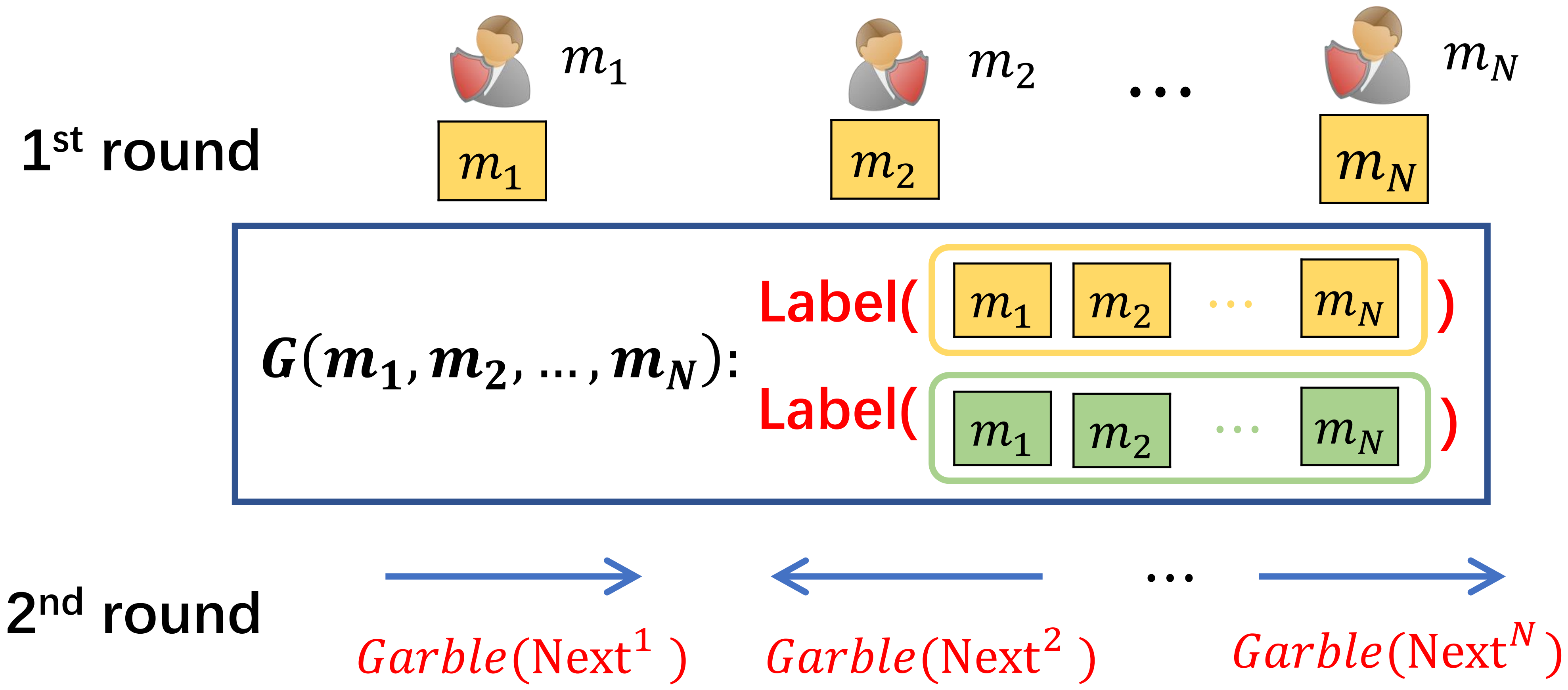
Round Compression to Rescue

- Garble the 3rd round next message function Next^i to compress to 2 rounds.



Round Compression to Rescue

- Garble the 3rd round next message function Next^i to compress to 2 rounds.



Full-Fledged Tree-Based Approach

Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

Full-Fledged Tree-Based Approach

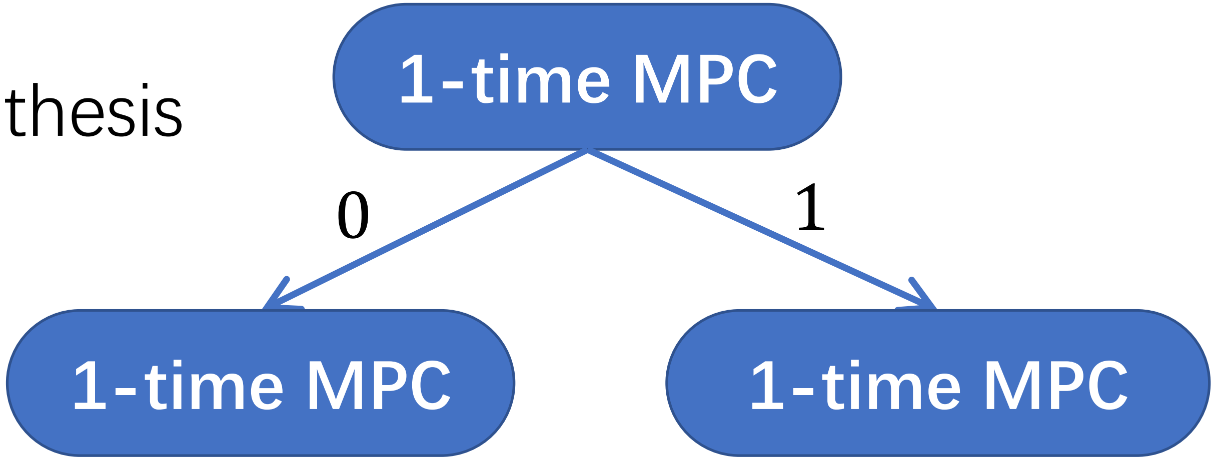
- From **2-times** usable to **reusable:**

  Recursively apply the self-synthesis

Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

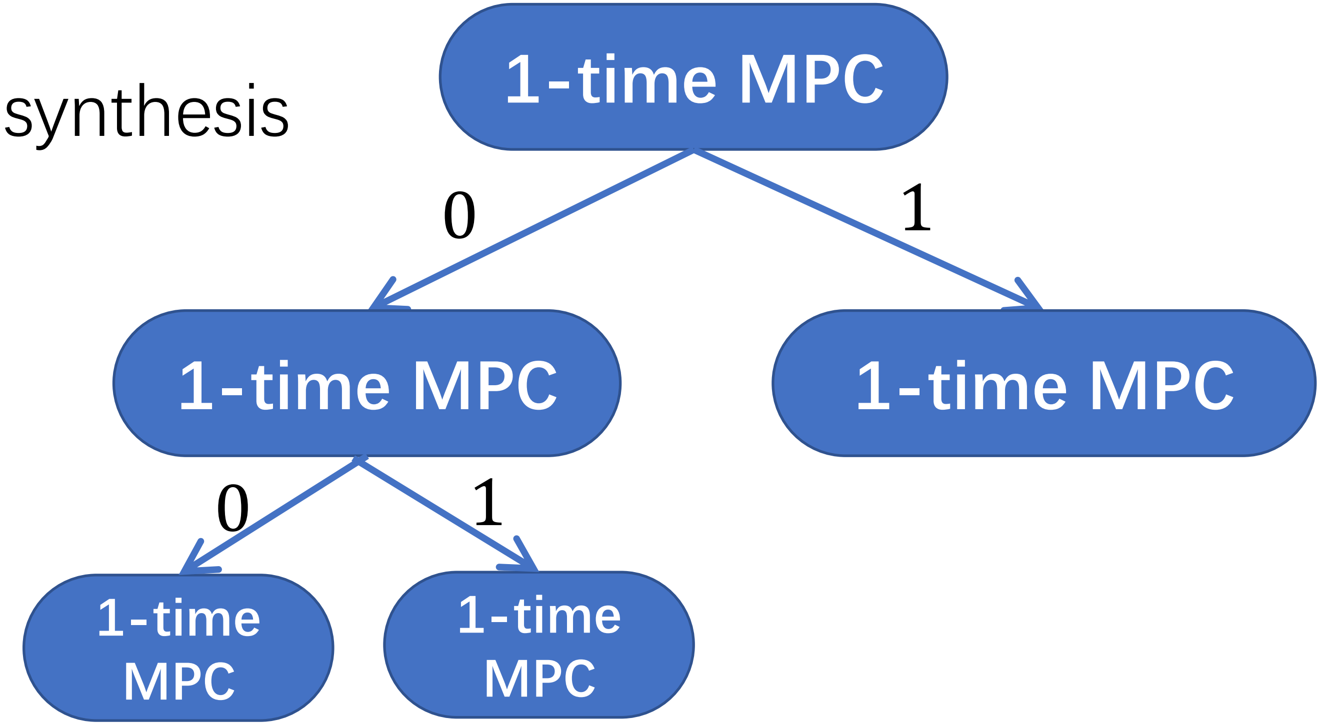
  Recursively apply the self-synthesis



Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

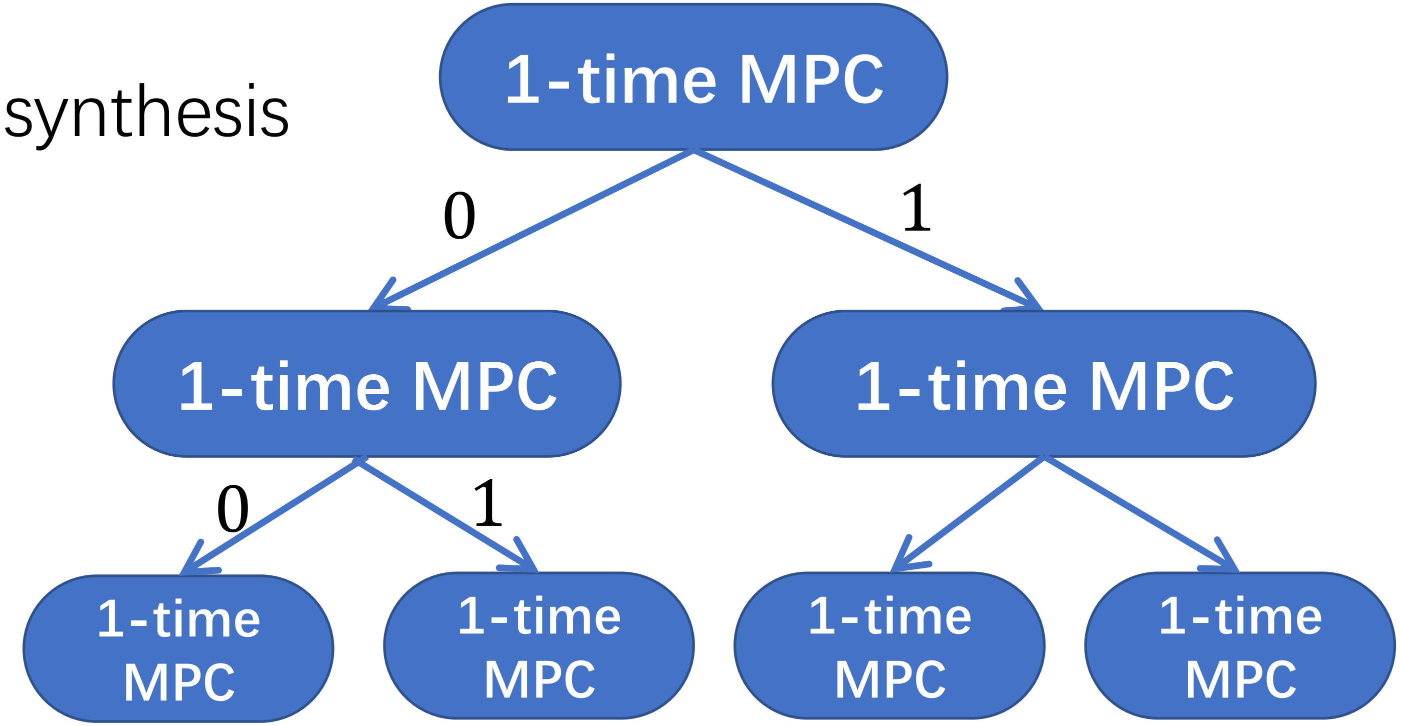
  Recursively apply the self-synthesis



Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

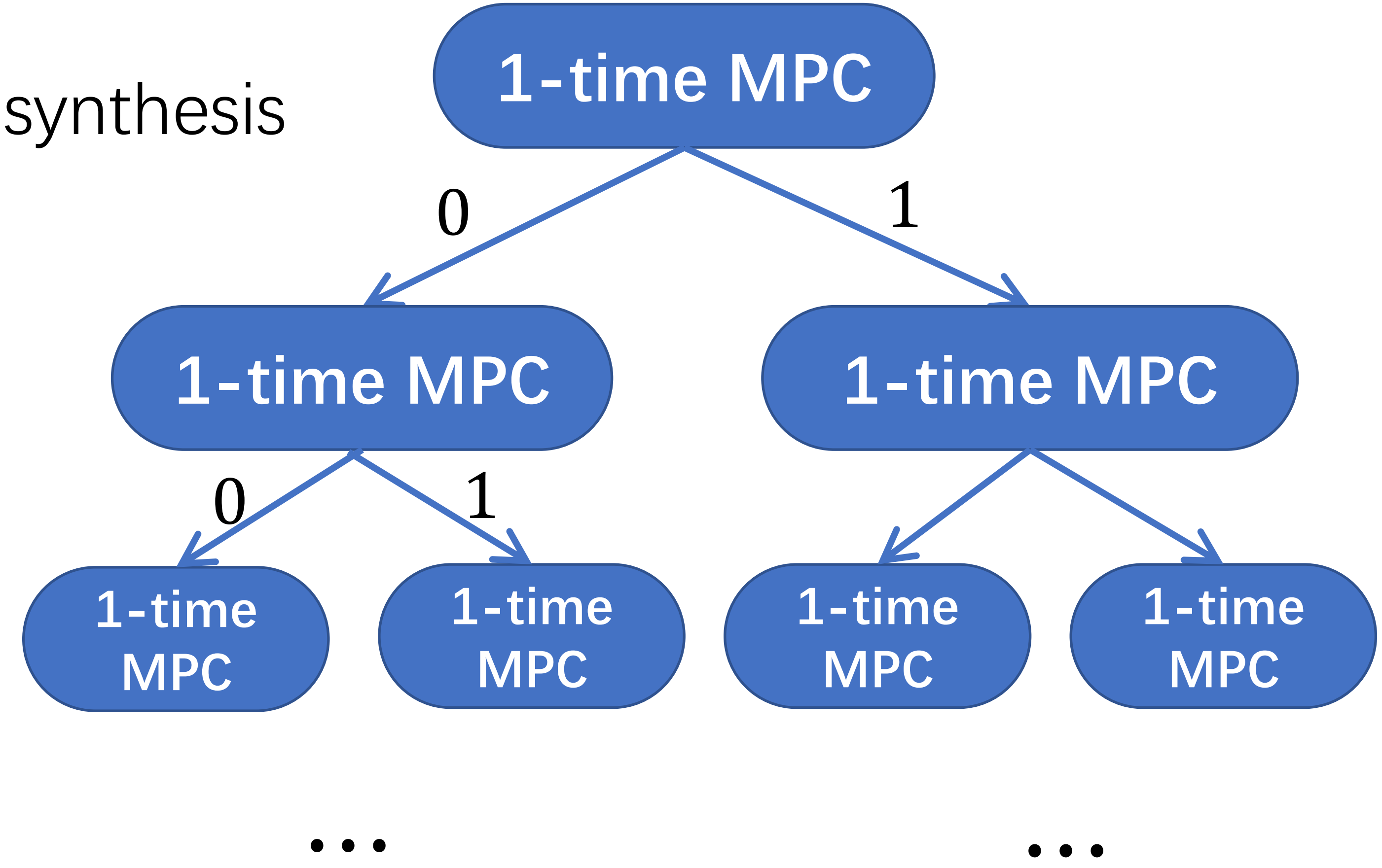
  Recursively apply the self-synthesis



Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

  Recursively apply the self-synthesis

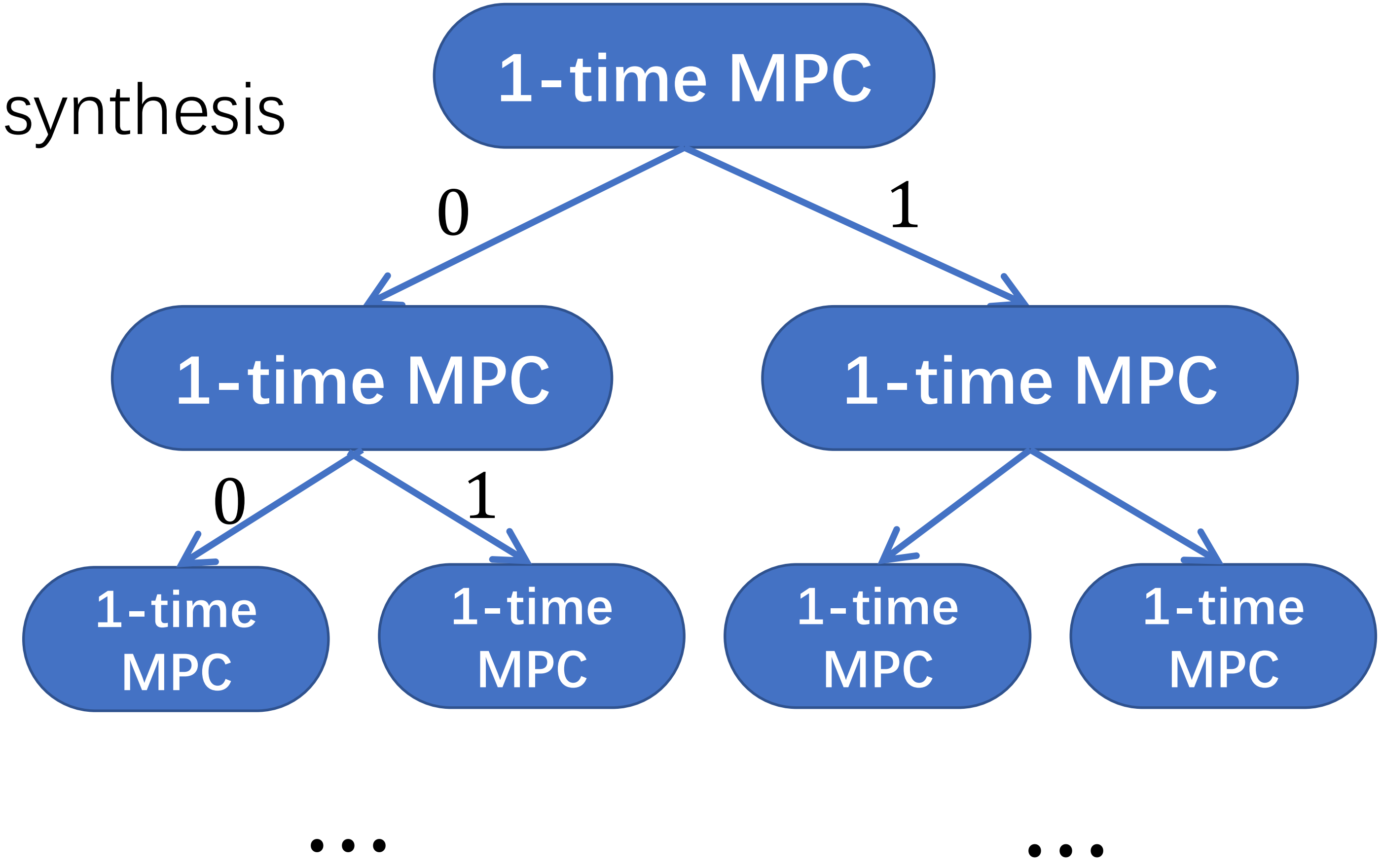


Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

  Recursively apply the self-synthesis

  Given C, walk down the tree according to C.



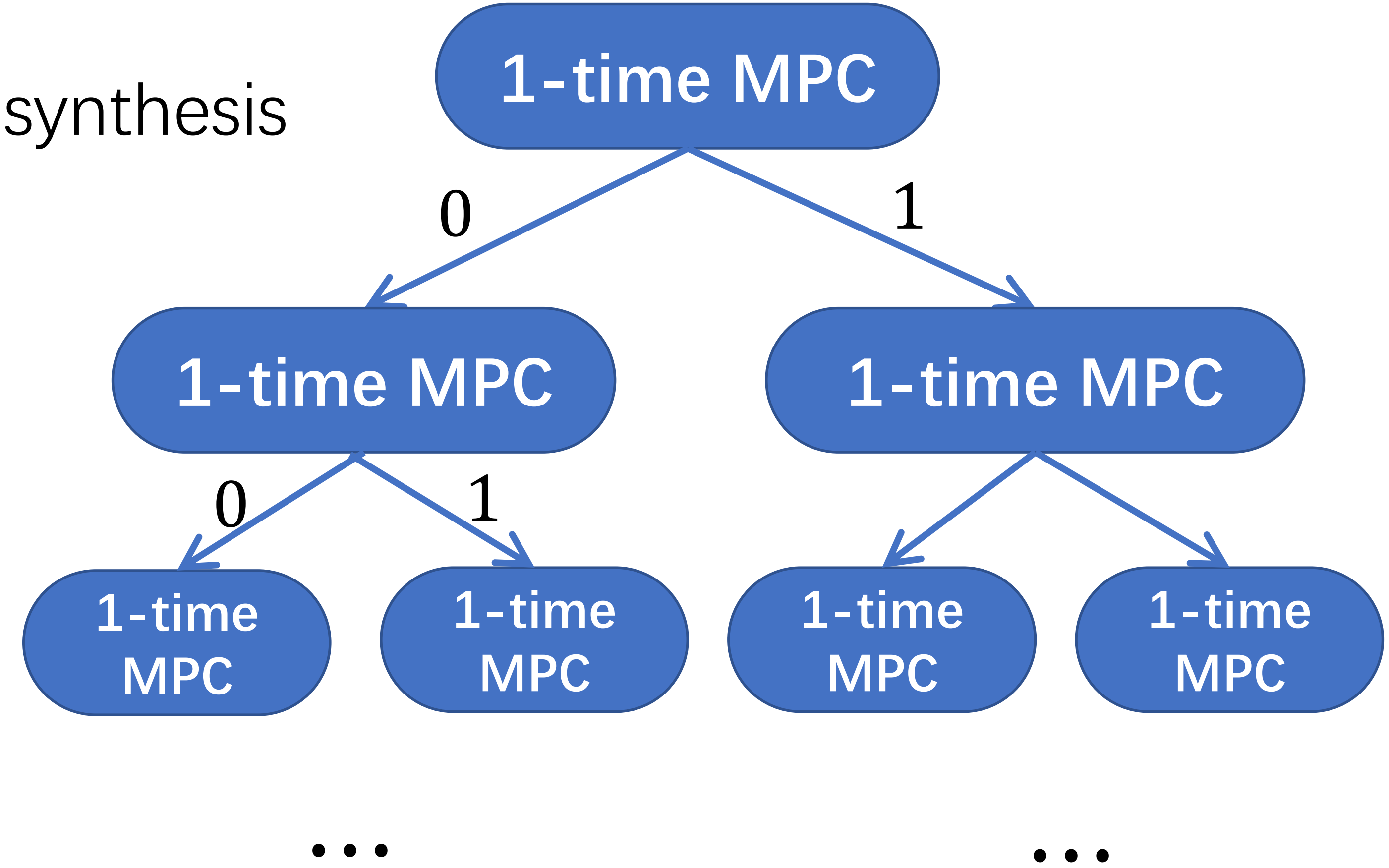
Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

  Recursively apply the self-synthesis

  Given C, walk down the tree according to C.

  e.g. $C = 01 \ldots$



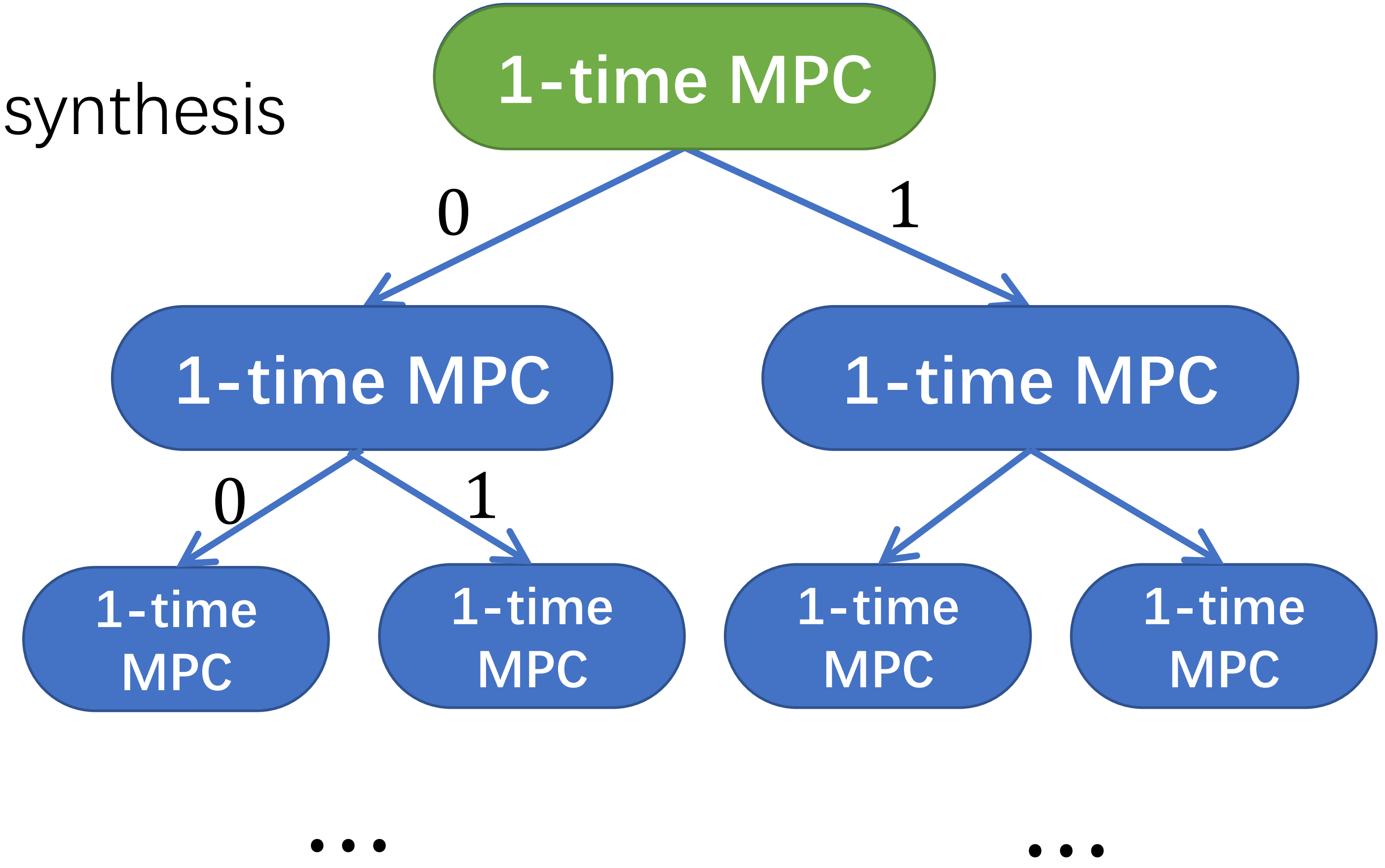
Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

  Recursively apply the self-synthesis

  Given C, walk down the tree according to C.

  e.g. $C = 01 \ldots$



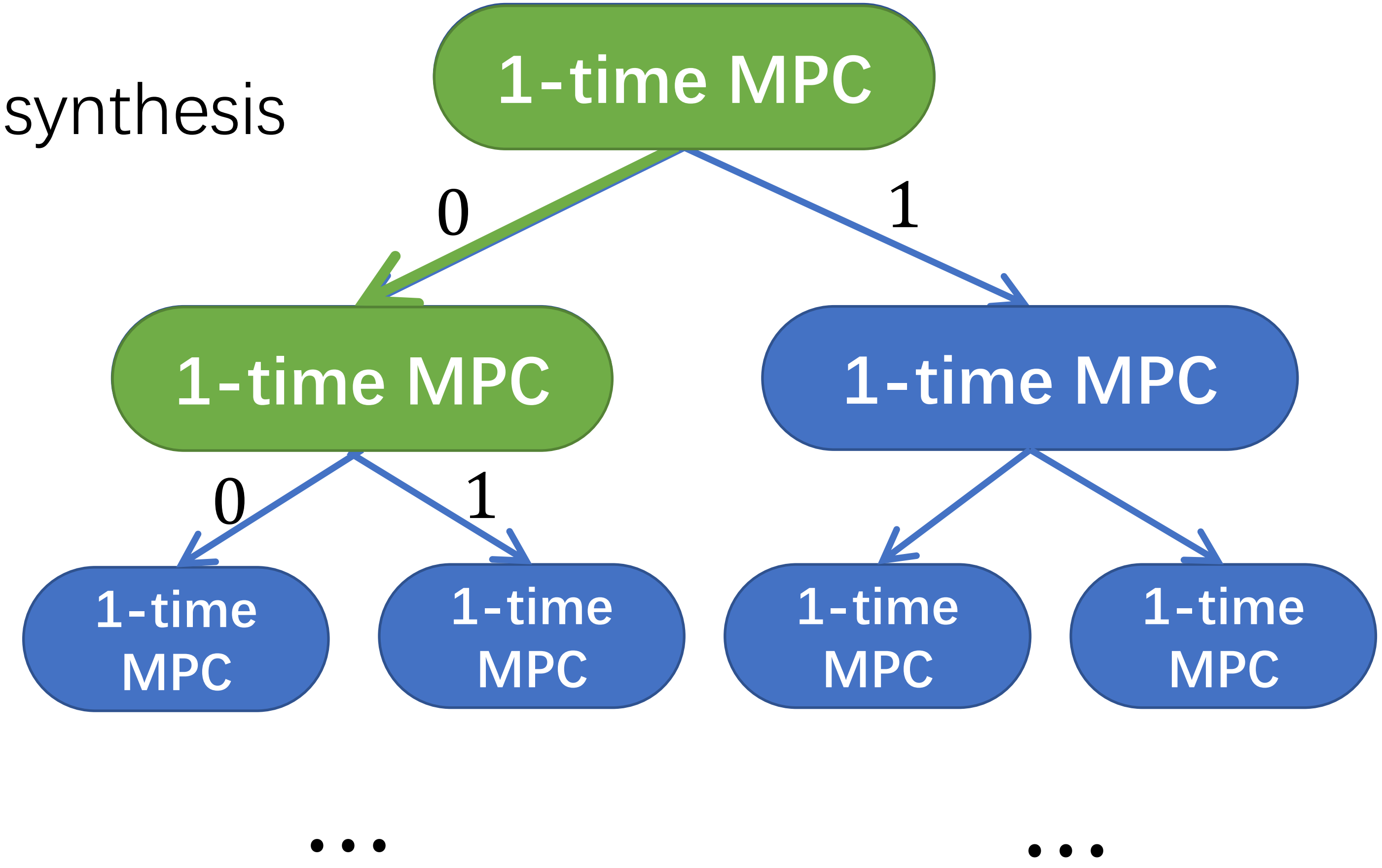
Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

  Recursively apply the self-synthesis

  Given C, walk down the tree according to C.

  e.g. $C = 01\ldots$



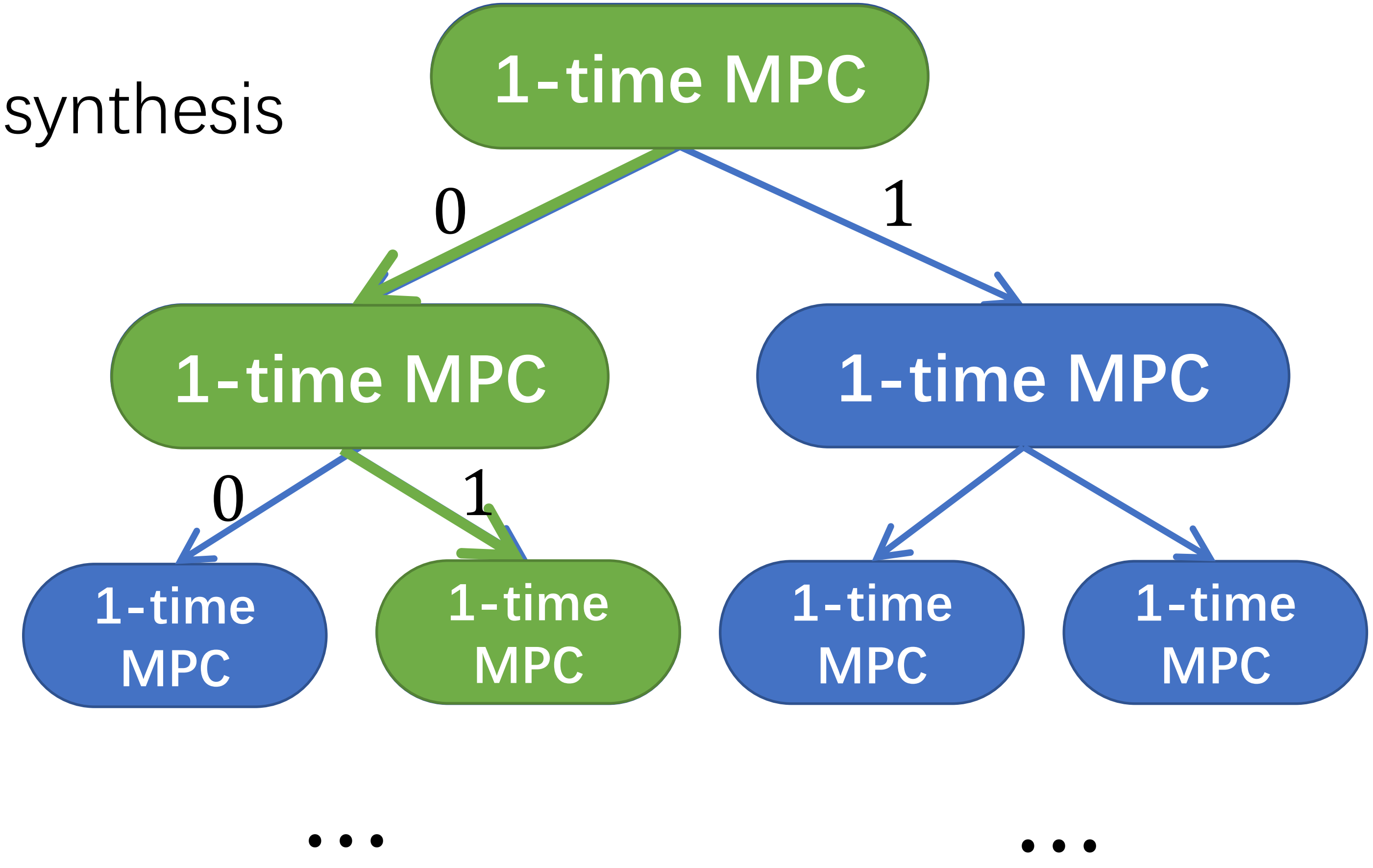
Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

  Recursively apply the self-synthesis

  Given C, walk down the tree according to C.

  e.g. $C = 01 \ldots$



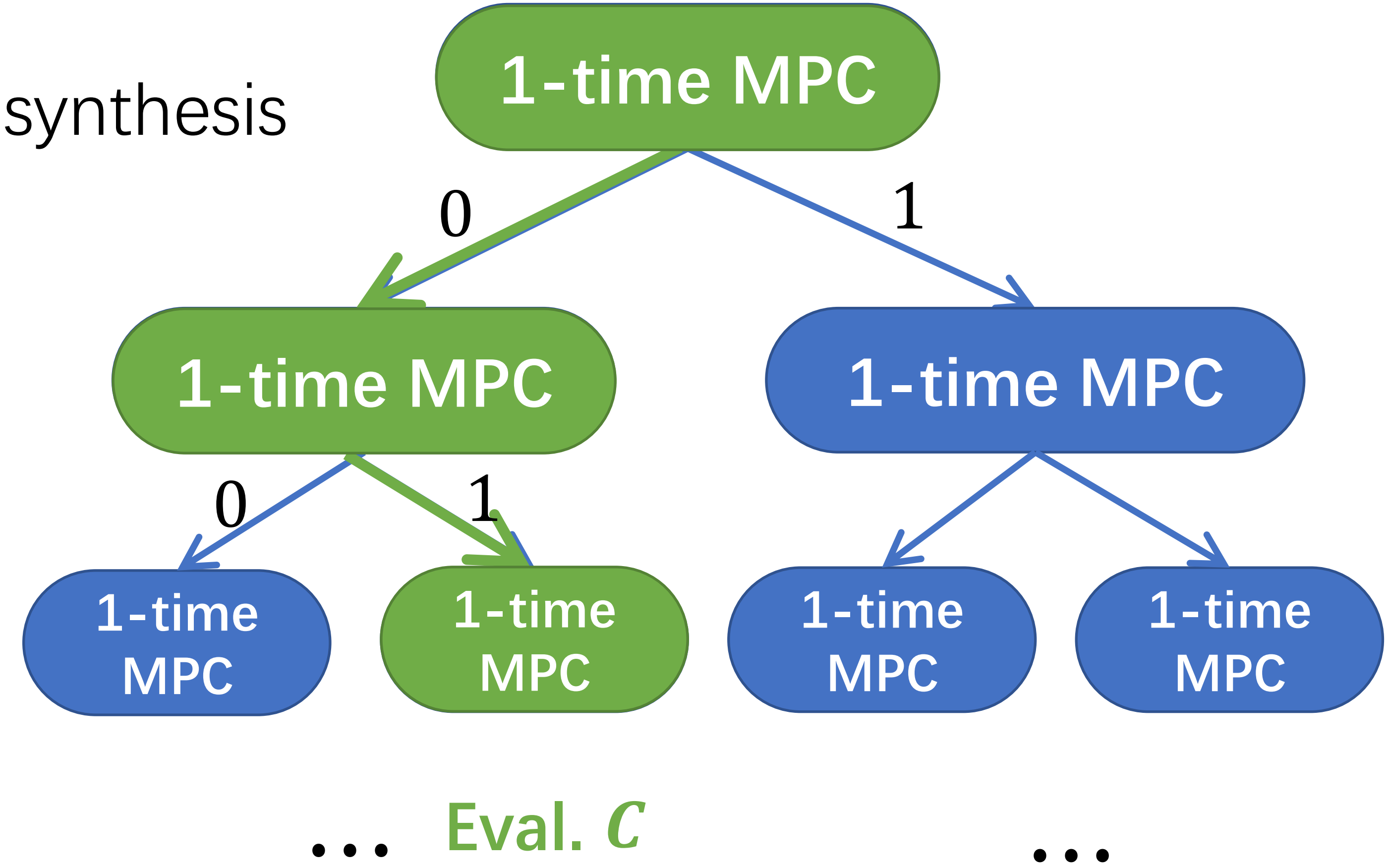
Full-Fledged Tree-Based Approach

- From **2-times** usable to **reusable:**

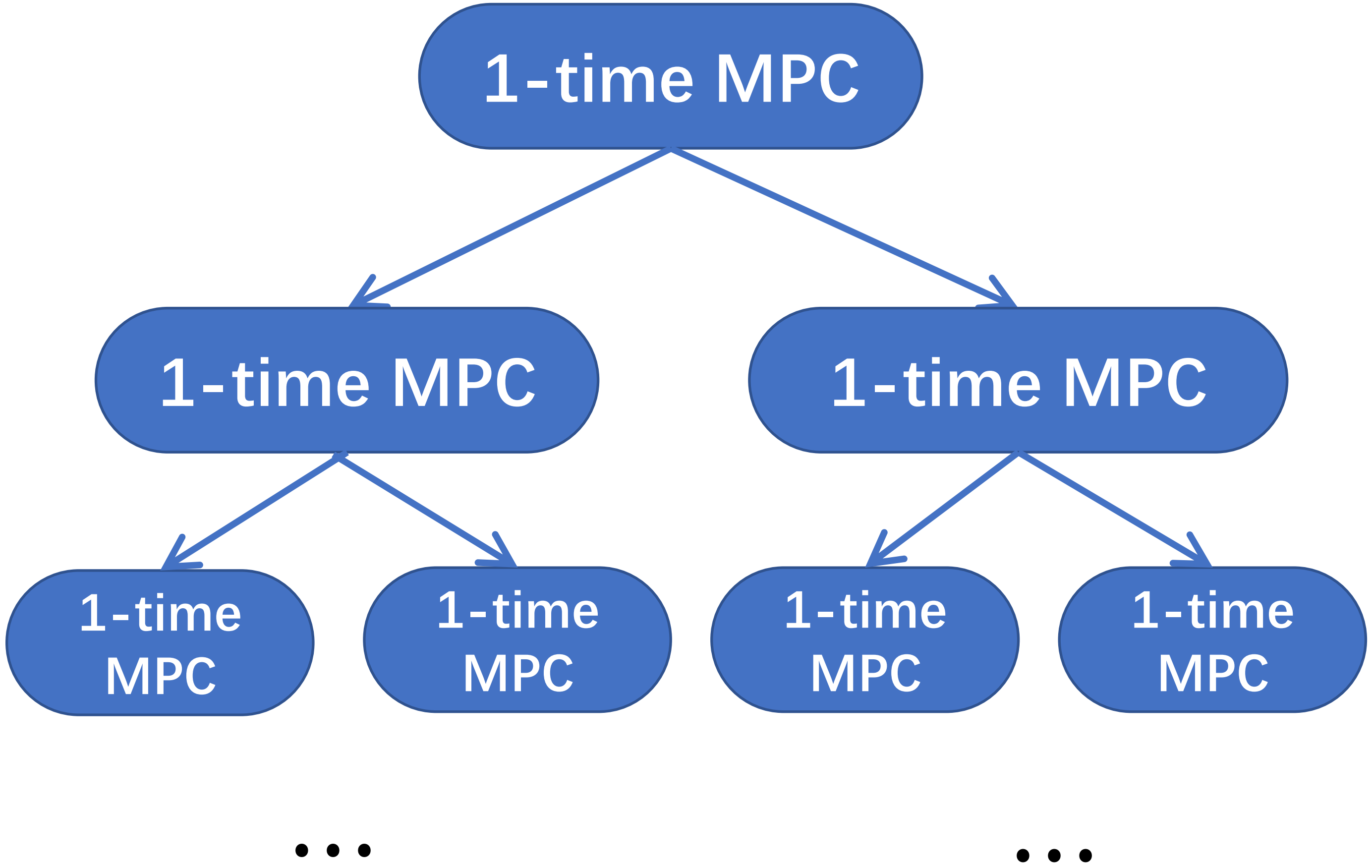
  Recursively apply the self-synthesis

  Given C, walk down the tree according to C.

  e.g. $C = 01 \ldots$

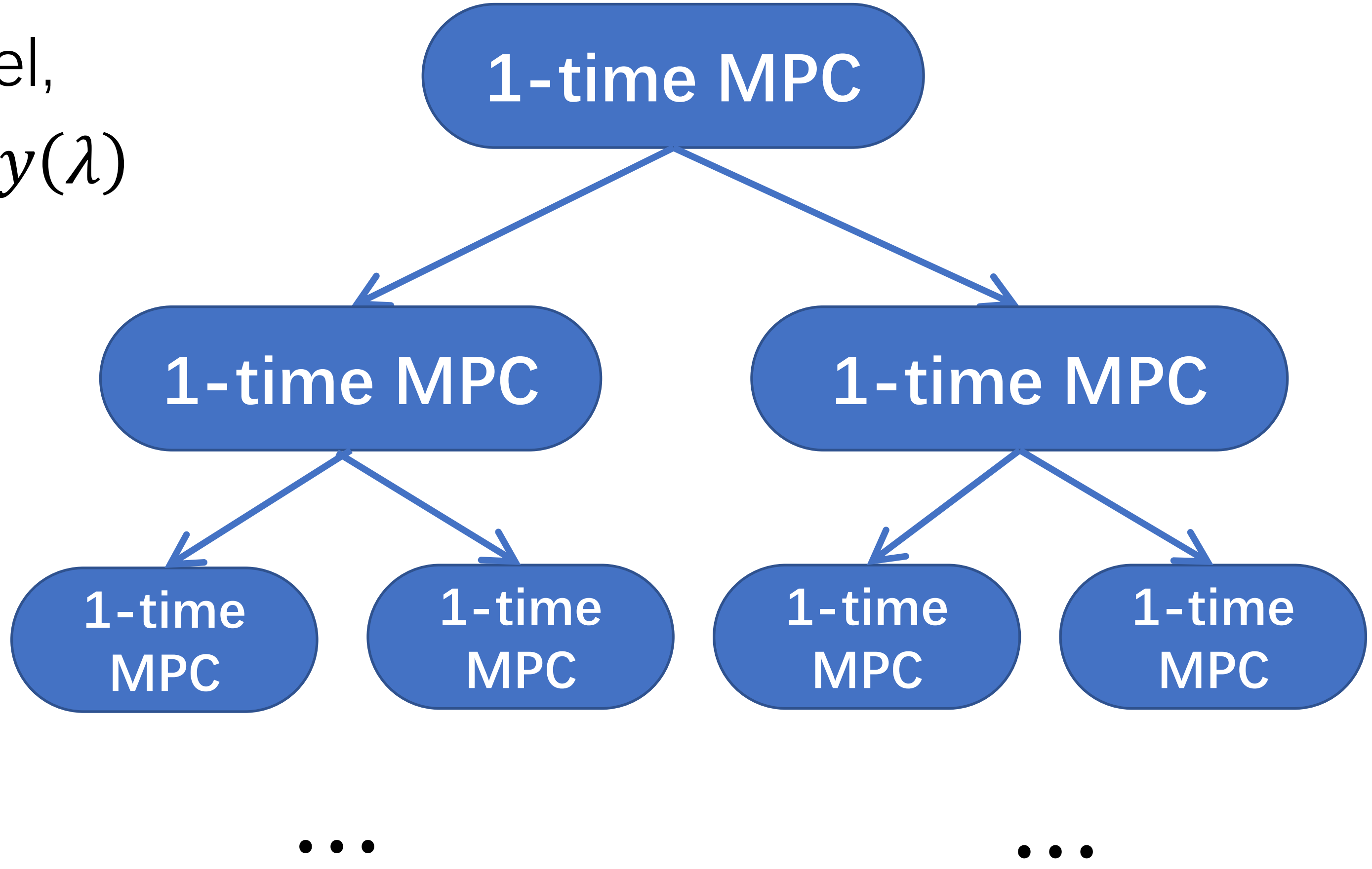


Time Complexity Blow Up



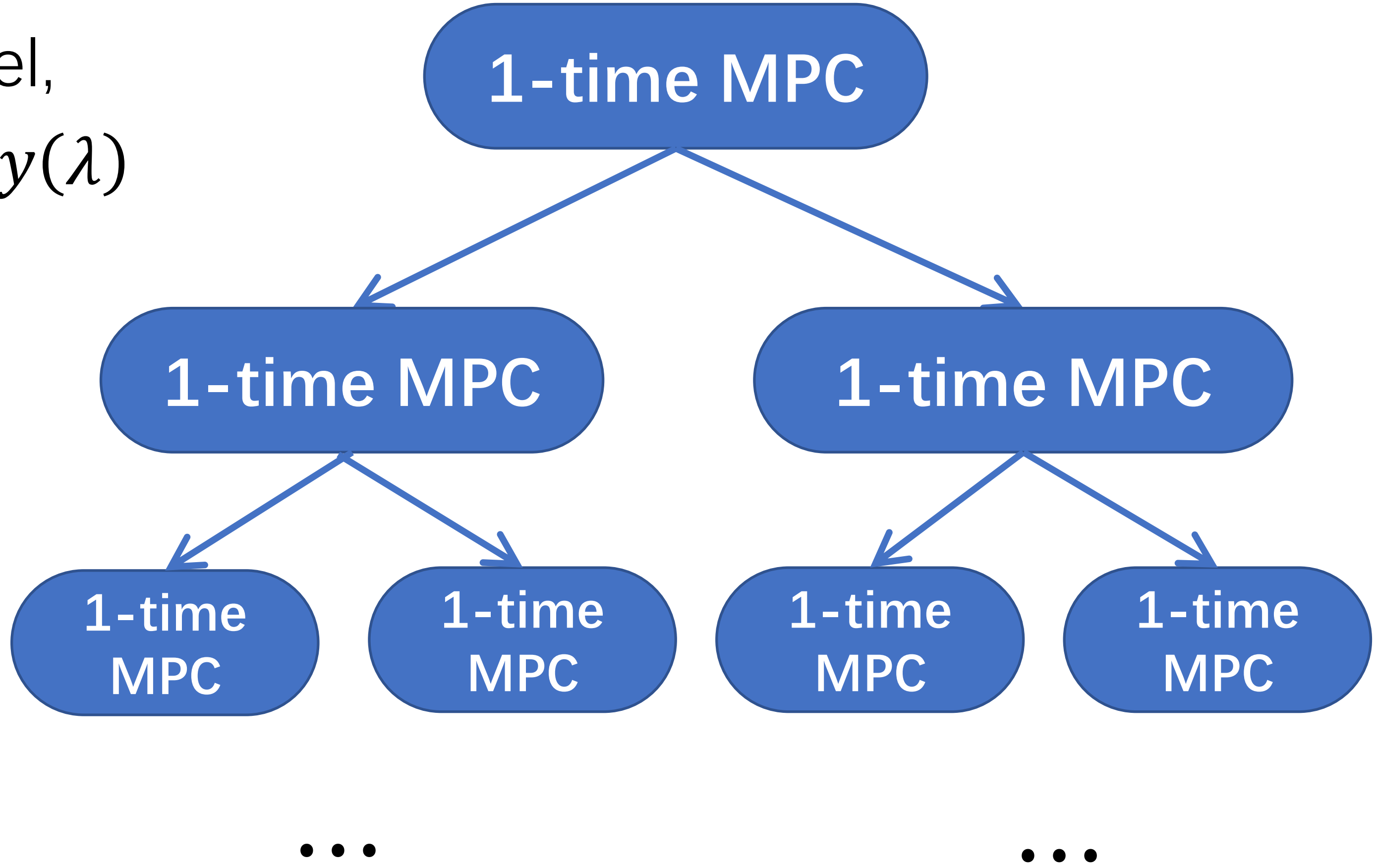
Time Complexity Blow Up

- For 1-time MPC in plain model,
  Time(1st Round) $\approx |C| \cdot poly(\lambda)$



Time Complexity Blow Up

- For 1-time MPC in plain model,
  Time(1st Round) $\approx |C| \cdot poly(\lambda)$

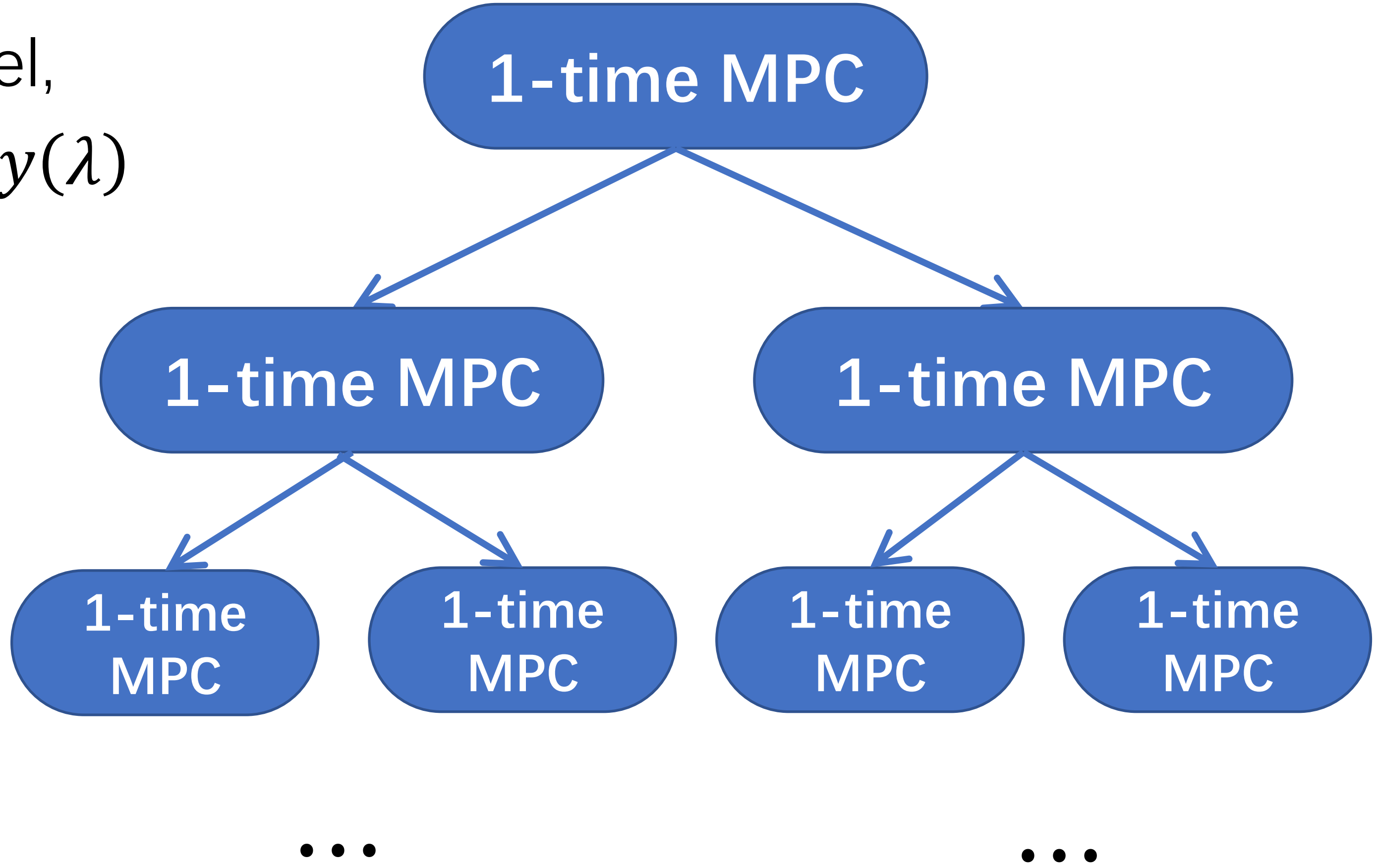


Time Complexity Blow Up

- For 1-time MPC in plain model,

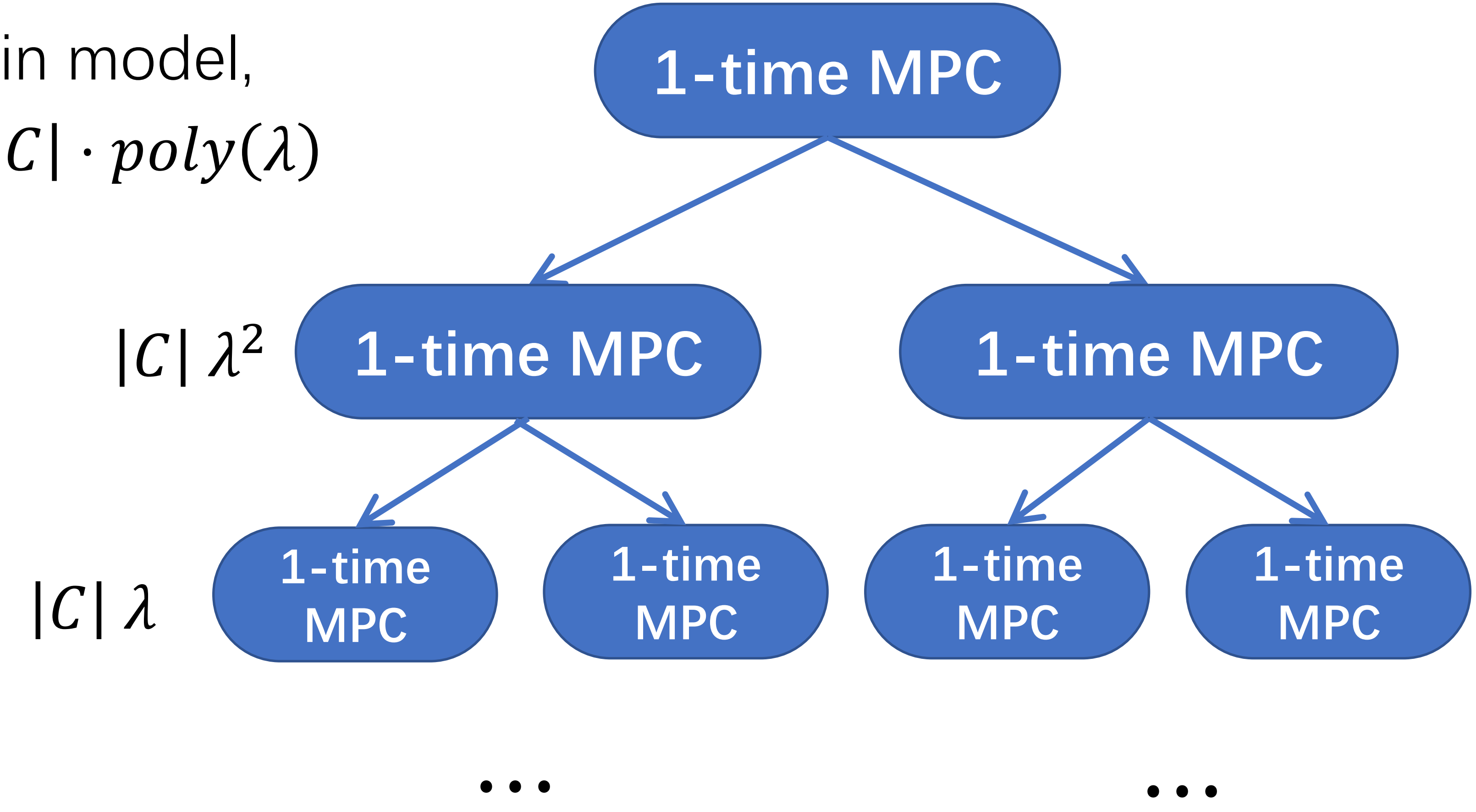
  Time(1st Round) $\approx |C| \cdot poly(\lambda)$

$|C|\ \lambda$



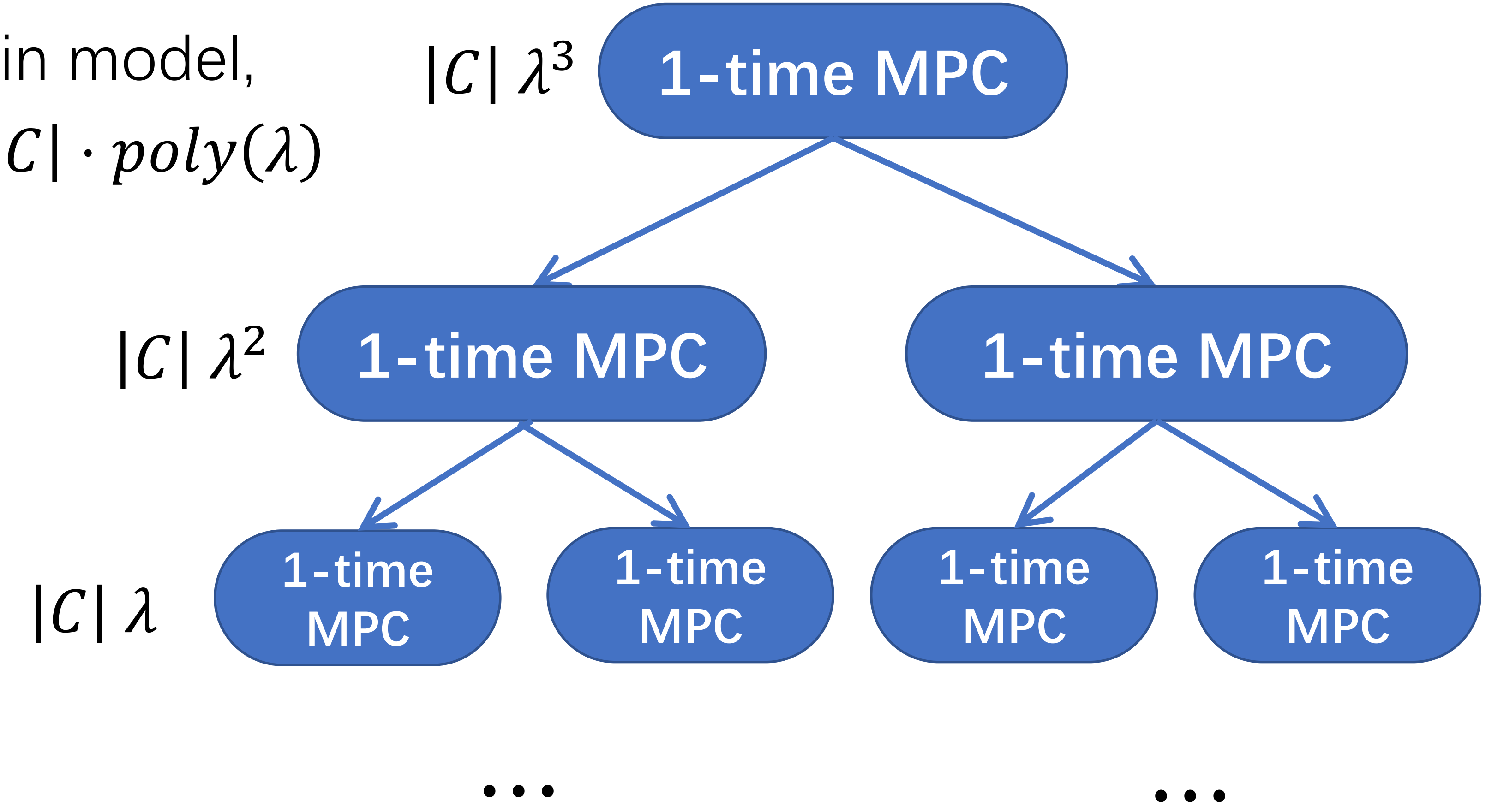
Time Complexity Blow Up

- For 1-time MPC in plain model,
  Time(1st Round) $\approx |C| \cdot poly(\lambda)$



Time Complexity Blow Up

- For 1-time MPC in plain model,
  Time(1st Round) $\approx |C| \cdot poly(\lambda)$



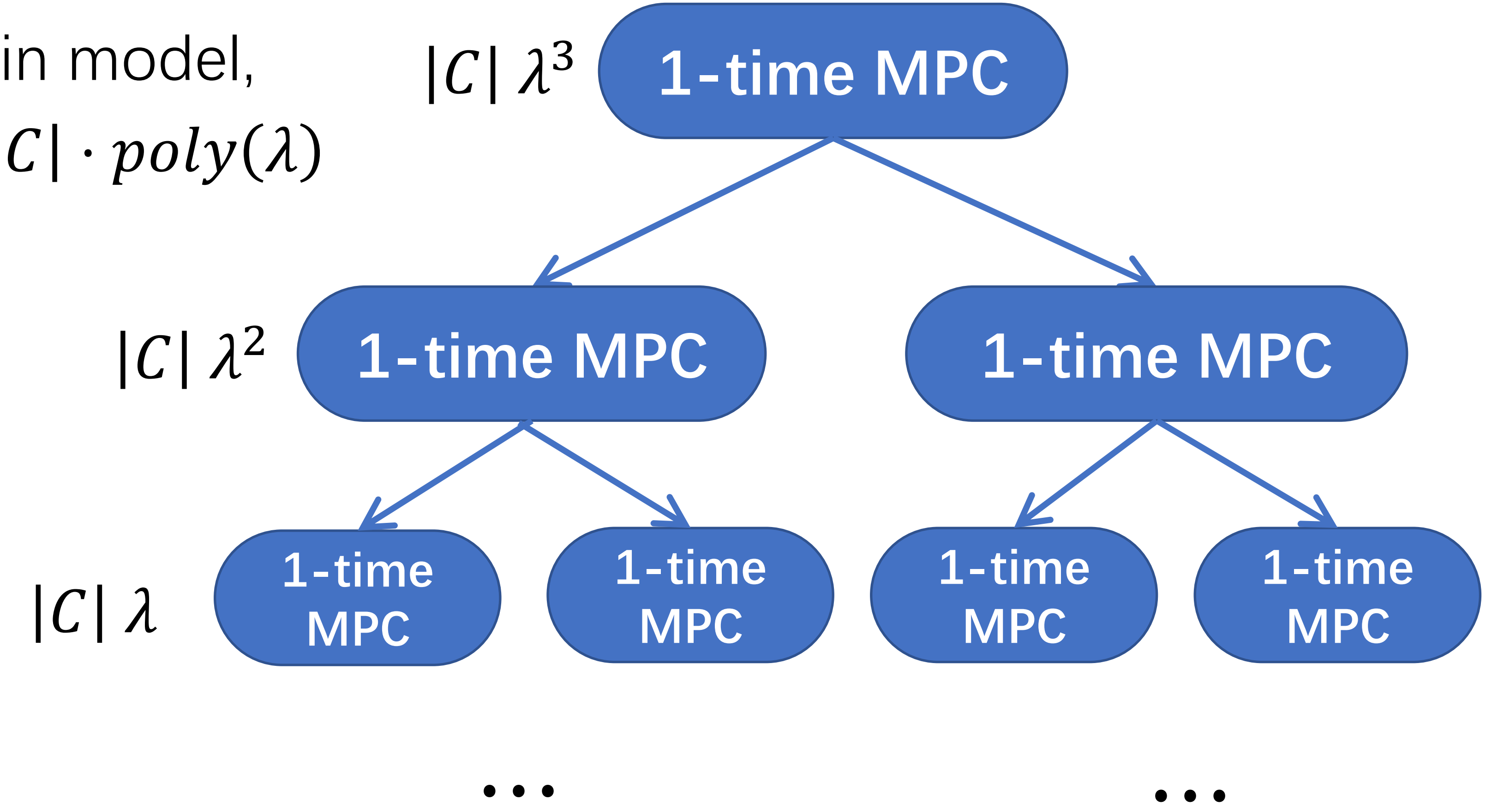
Time Complexity Blow Up

- For 1-time MPC in plain model,

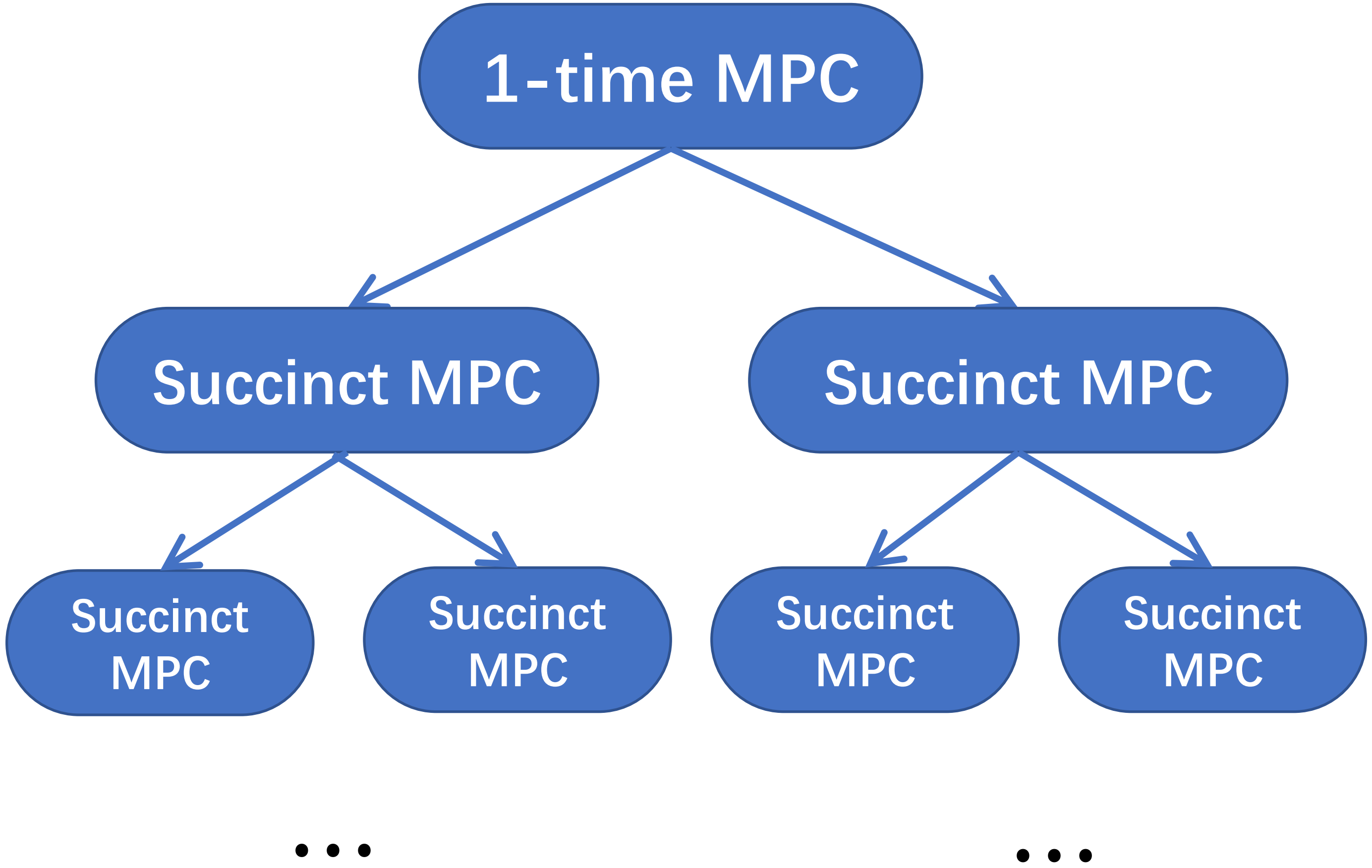
  Time(1^{st} Round) $\approx |C| \cdot poly(\lambda)$

  Time(Root node)

  is **exponential** in λ

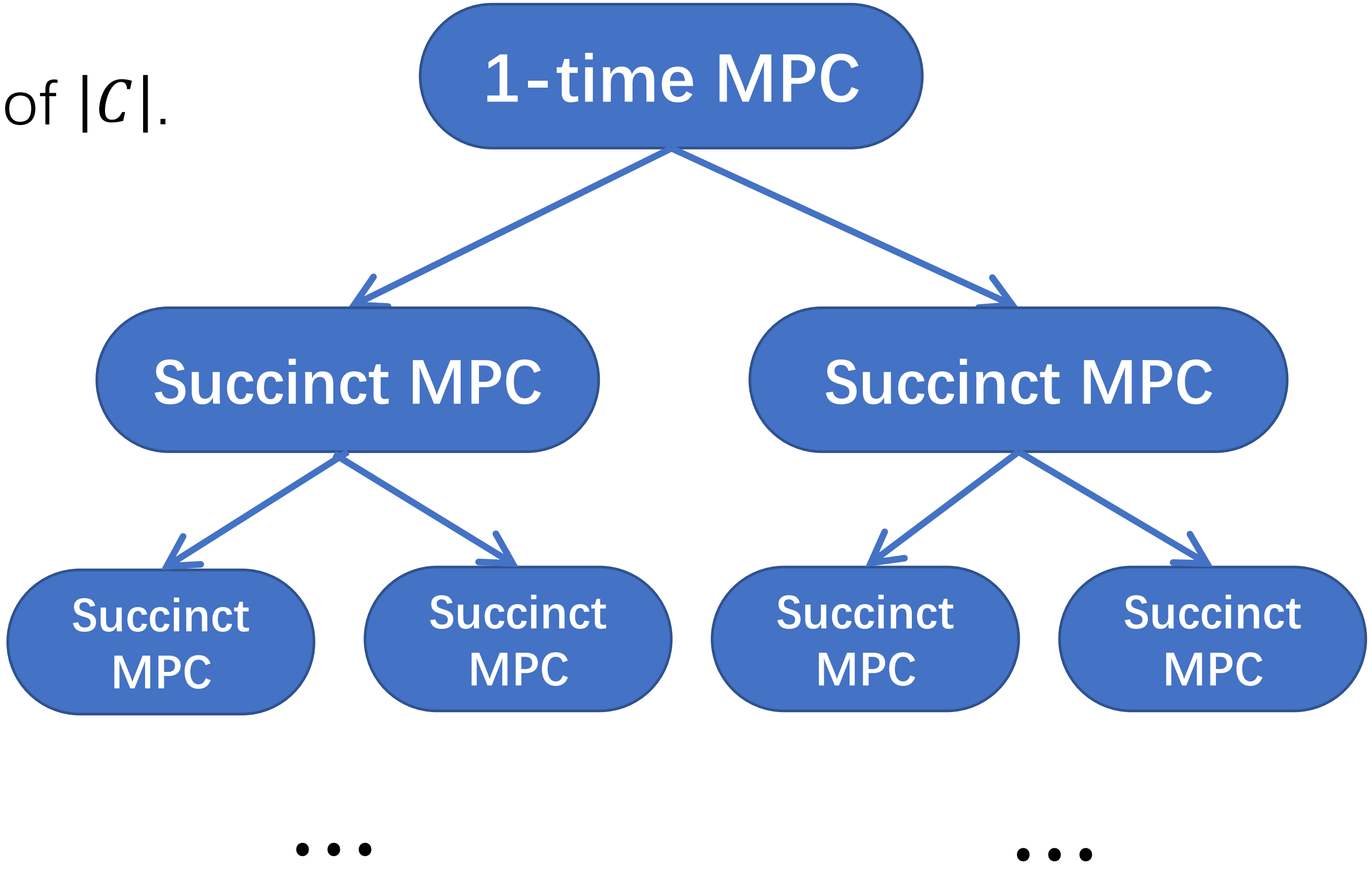


Necessary Condition for Recursion



Necessary Condition for Recursion

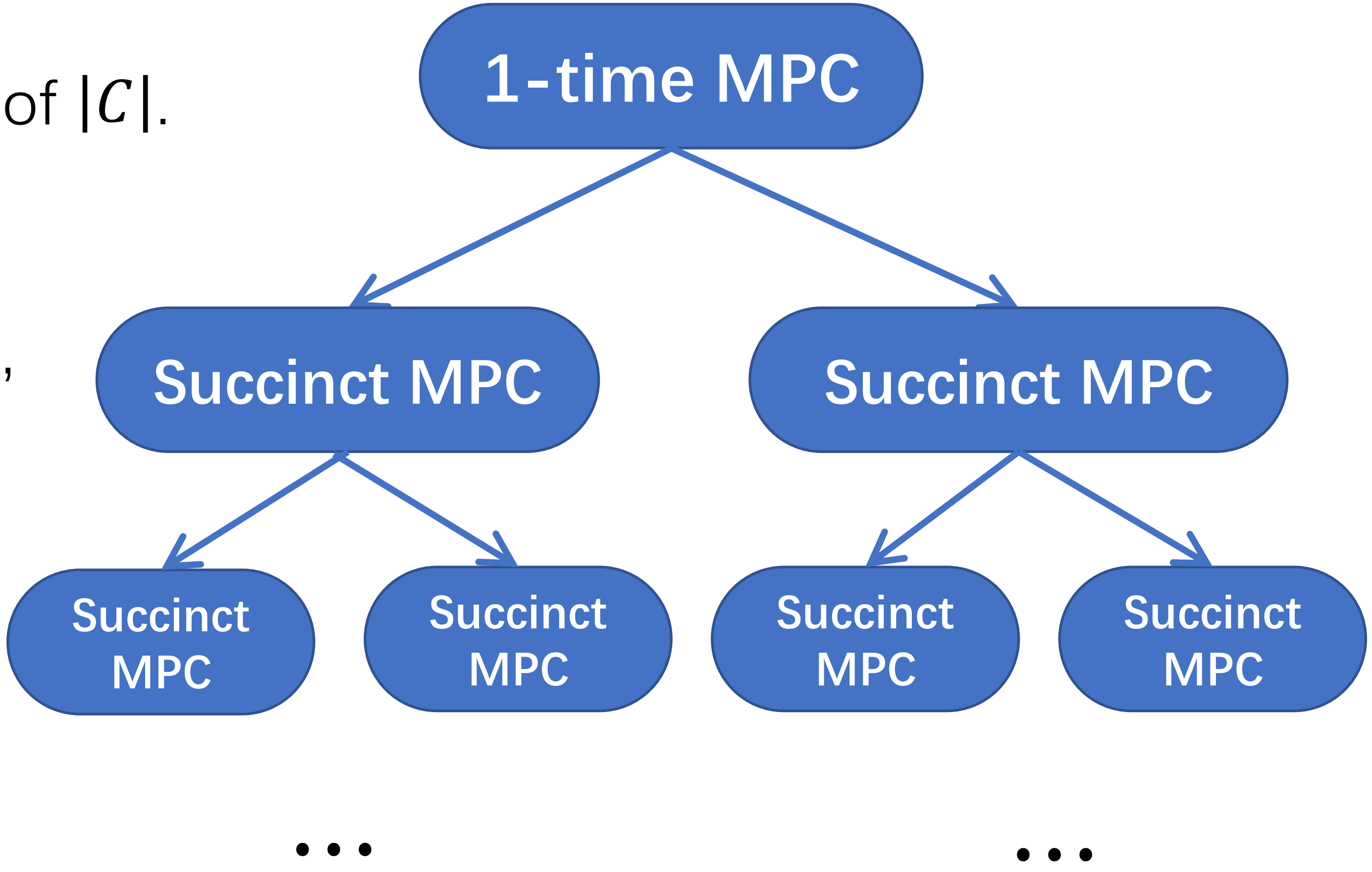
- **Succinct** 1-time MPC:
  Time(1st Round) is independent of $|C|$.



Necessary Condition for Recursion

- **Succinct** 1-time MPC:
  Time(1st Round) is independent of $|C|$.

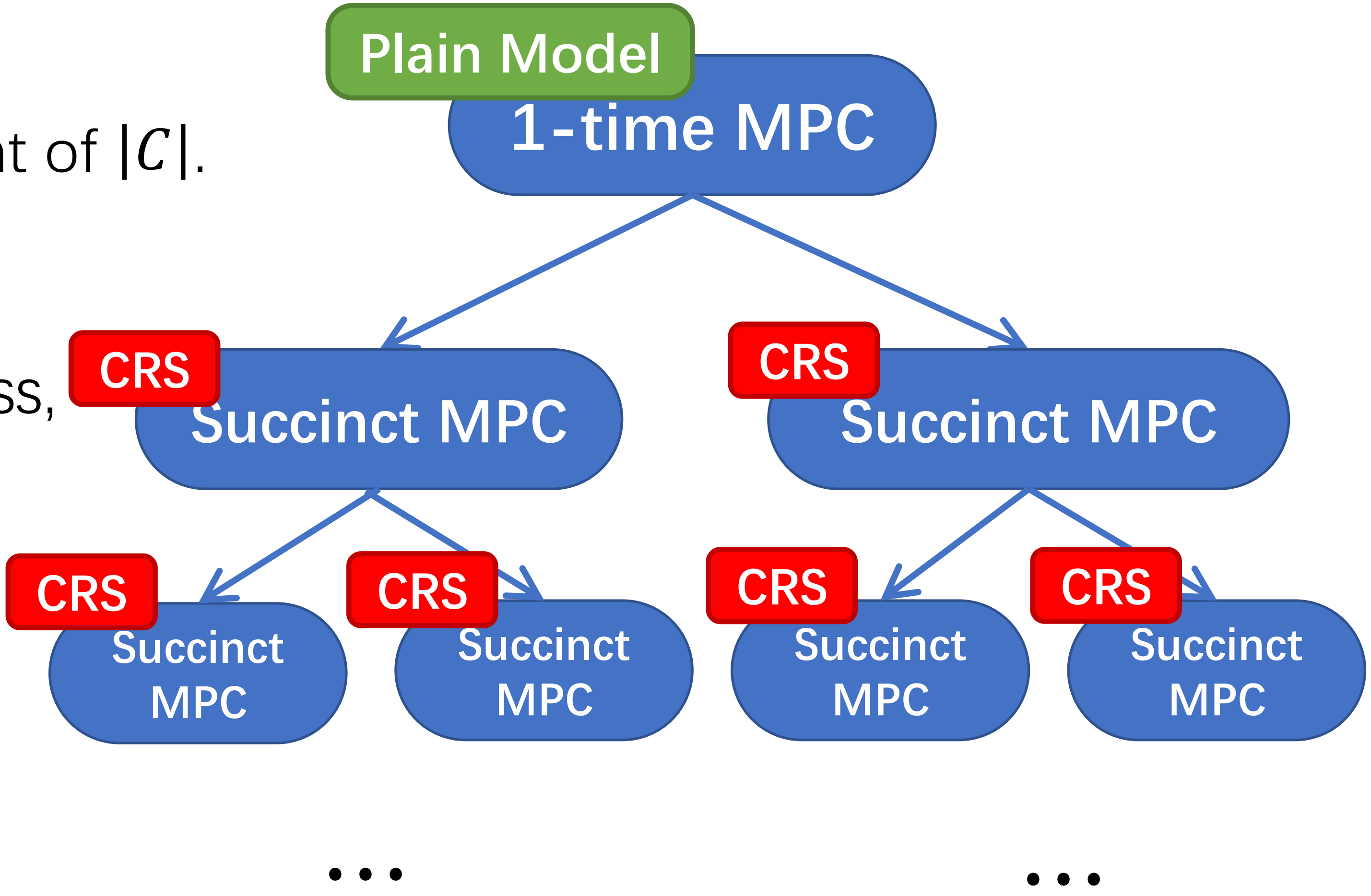
- [MW16] satisfies succinctness, but in **CRS** model.



Necessary Condition for Recursion

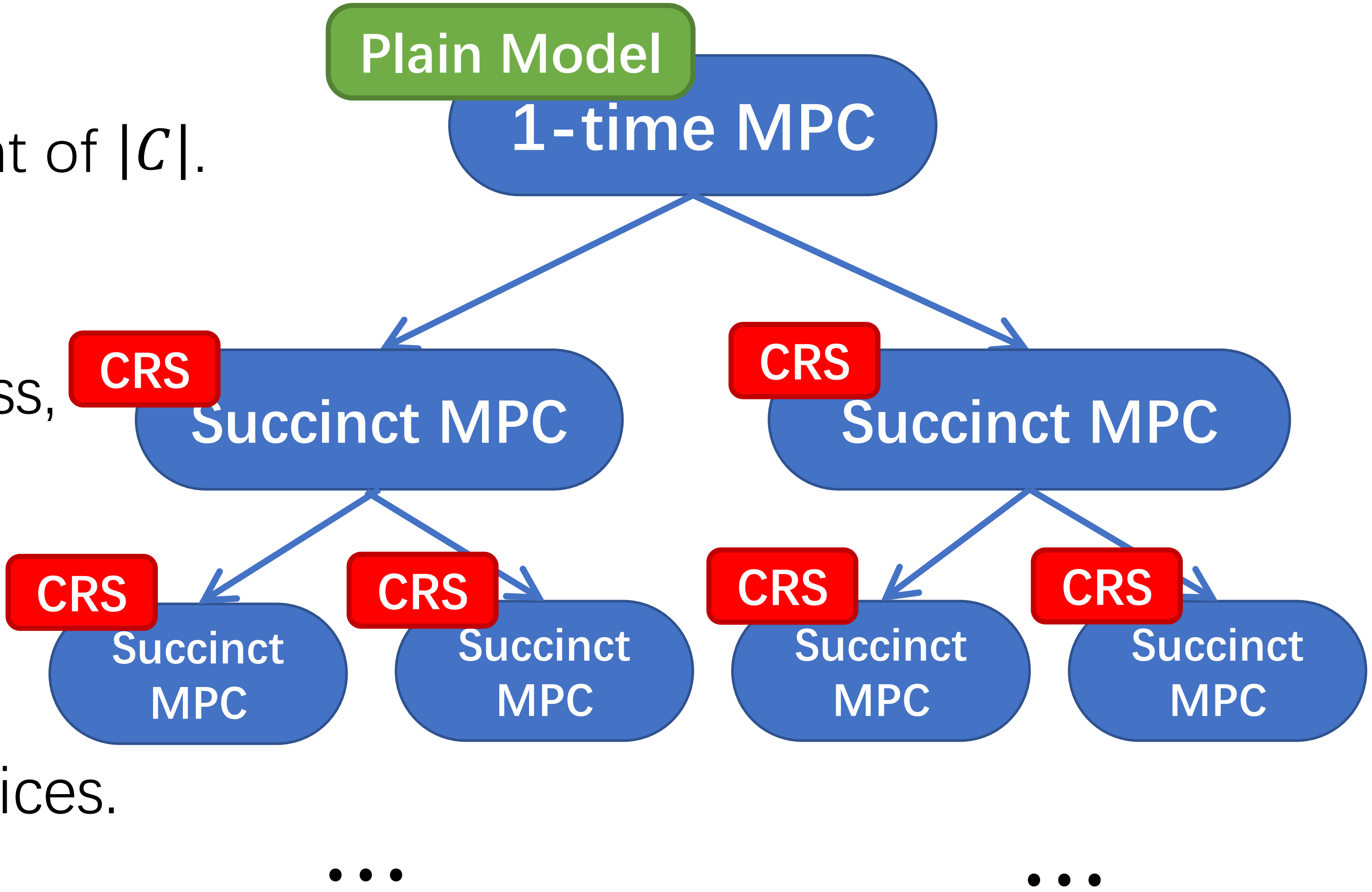
- **Succinct** 1-time MPC:
  Time(1st Round) is independent of $|C|$.

- [MW16] satisfies succinctness, but in **CRS** model.



Necessary Condition for Recursion

- **Succinct** 1-time MPC:
  Time(1st Round) is independent of $|C|$.
- [MW16] satisfies succinctness, but in **CRS** model.
- In fact succinct 1-time MPC in *preprocessing model* suffices.



Thank you!