Honest Majority MPC with $\tilde{O}(|C|)$ Communication in Minicrypt

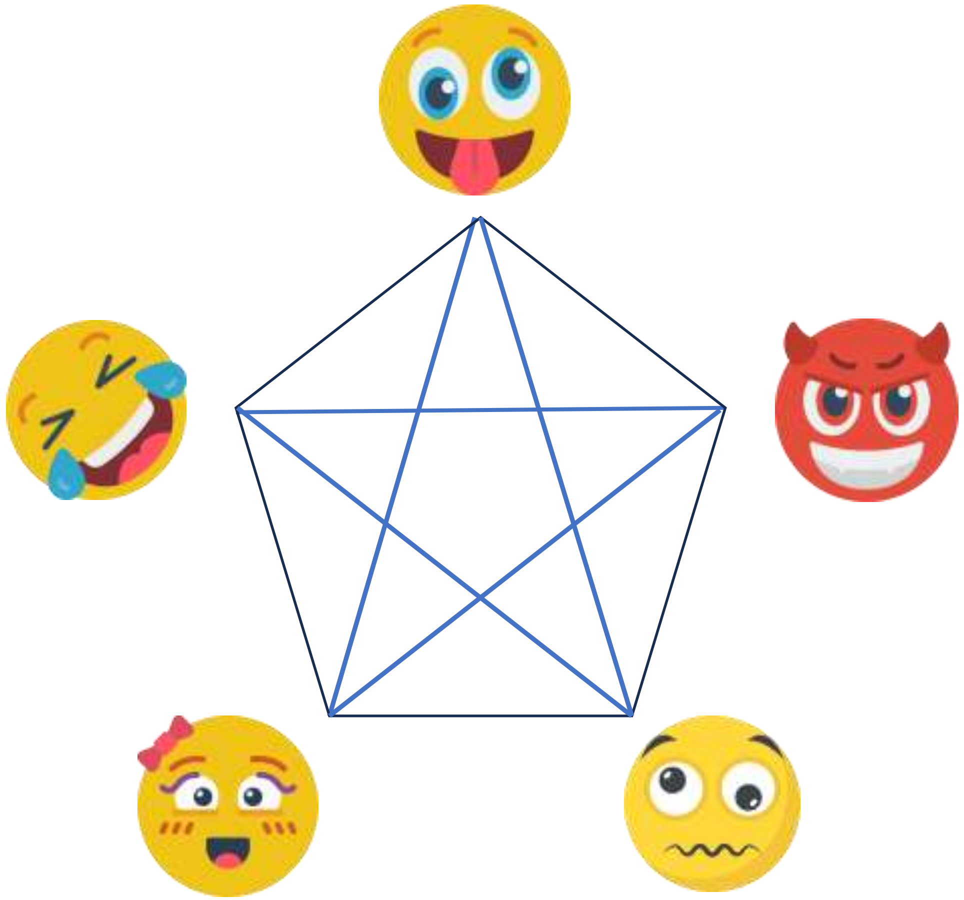
Yifan Song

Tsinghua University & Shanghai Qi Zhi Institute

Xiaxi Ye

Tsinghua University

Multiparty Computation



Setting

- n parties
- t corrupted parties
- Honest majority: $n = 2t + 1$
- Synchronous network

Communication Complexity

Reference	Communication	Corruption threshold	Security
[DN07, GIP+14, CGH+14…]	$O(\|C\| \cdot n)$	$t = (n-1)/2$	Information-theoretic
[GPS21]	$O(\|C\|)$	$t = (0.5-\epsilon) \cdot n$	Information-theoretic

$|C|$: circuit size, n: number of parties, counted by field elements

Communication Complexity

Reference	Communication	Corruption threshold	Security
[DN07, GIP+14, CGH+14…]	$O(\|C\| \cdot n)$	$t = (n-1)/2$	Information-theoretic
[GPS21]	$O(\|C\|)$	$t = (0.5 - \epsilon) \cdot n$	Information-theoretic

$|C|$: circuit size, n: number of parties, counted by field elements

Is it possible to construct an information-theoretic MPC protocol in standard honest majority setting with $t = (n-1)/2$ achieving communication of $O(C)$ field elements?

Negative Evidence from [DLN19]

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \ldots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \ldots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Theorem [DLN19]. Let $n = 2t + 1$. Any statistically t-private and statistically correct protocol for $IP_{n,I}$ communicates at least $\frac{In(t-1)}{2} - negl$ elements.

Negative Evidence from [DLN19]

Circuit size: $|C| = O(I \cdot n)$

Input size: $I_C = O(I \cdot n)$

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \dots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \dots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Theorem [DLN19]. Let $n = 2t + 1$. Any statistically t-private and statistically correct protocol for $IP_{n,I}$ communicates at least $\frac{In(t-1)}{2} - negl$ elements.

Negative $\boxed{\text{Evidence}}$ from [DLN19]

Circuit size: $|C| = O(I \cdot n)$

Input size: $I_C = O(I \cdot n)$

$IP_{n,I}$

1. $y \leftarrow IP\left(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \dots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \dots \overrightarrow{x_{2,t}}\right)$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Theorem [DLN19]. Let $n = 2t + 1$. Any statistically t-private and statistically correct protocol for $IP_{n,I}$ communicates at least $\frac{In(t-1)}{2} - negl$ elements.

Negative $\boxed{\text{Evidence}}$ from [DLN19]

Circuit size: $|C| = O(I \cdot n)$

Input size: $I_C = O(I \cdot n)$

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \ldots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \ldots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Theorem [DLN19]. Let $n = 2t + 1$. Any statistically t-private and statistically correct protocol for $IP_{n,I}$ communicates at least $\frac{In(t-1)}{2} - negl$ elements.

This does **NOT** rule out the case of MPC protocols with communication $\Omega(I_C \cdot n)$ but $o(|C| \cdot n)$.

Communication Complexity

Reference	Communication	Corruption threshold	Security
[DN07, GIP+14, CGH+14…]	$O(\|C\| \cdot n)$	$t = (n-1)/2$	Information-theoretic
[GPS21]	$O(\|C\|)$	$t = (0.5 - \epsilon) \cdot n$	Information-theoretic

$|C|$: circuit size, n: number of parties, counted by field elements

What assumptions suffice to build an MPC protocol in honest majority setting with $t = (n-1)/2$ achieving communication of $O(C)$ field elements?

Communication Complexity

Reference	Communication	Corruption threshold	Security
[DN07, GIP+14, CGH+14…]	$O(\|C\| \cdot n)$	$t = (n-1)/2$	Information-theoretic
[GPS21]	$O(\|C\|)$	$t = (0.5 - \epsilon) \cdot n$	Information-theoretic

Honest majority

MPC

Oblivious linear evaluation (OLE)

s, u

$s \cdot t = u + v$

t, v

Efficient

communication and

small **data size**?

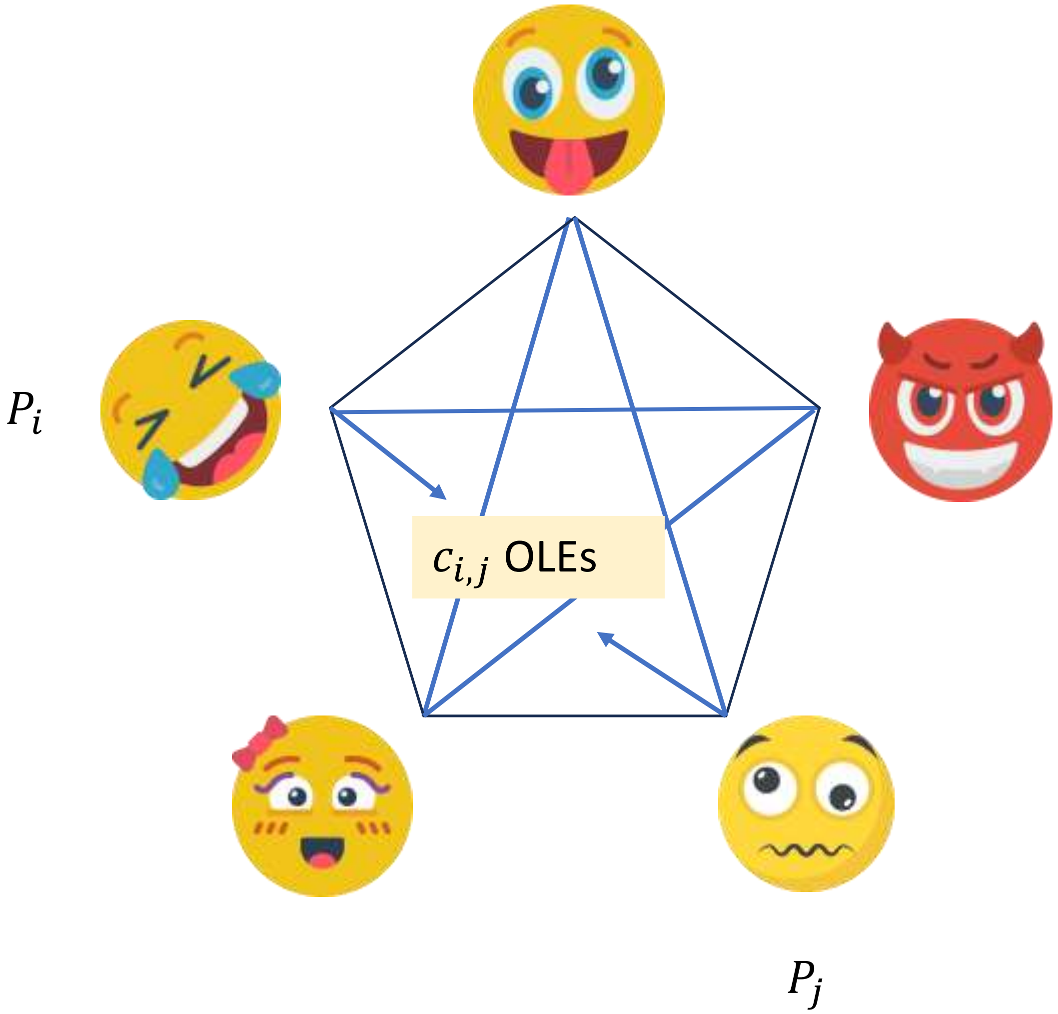
Our Results – positive result

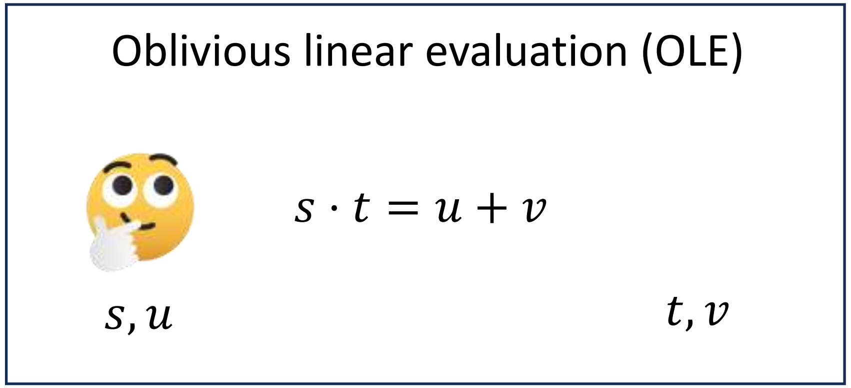
Reference	Communication	Corruption threshold	Security
[DN07, GIP+14, CGH+14…]	$O(\lvert C\rvert \cdot n)$	$t = (n-1)/2$	Information-theoretic
[GPS21]	$O(\lvert C\rvert)$	$t = (0.5-\epsilon)\cdot n$	Information-theoretic
Our result	$O(\lvert C\rvert) + O(\lvert C\rvert)$ OLEs	$t = (n-1)/2$	Information-theoretic

Theorem 1 (Informal).

Let n denote the number of parties and $t = \frac{n-1}{2}$ denote the number of corrupted parties. There exists an information-theoretic MPC protocol in OLE-hybrid model which computes an arithmetic circuit C with malicious security and at the cost of $O\big(|C| + D\cdot n + poly(n)\big)$ field elements of communication plus $O(|C| + D\cdot n + poly(n))$ invocations of OLE-hybrid functionalities, where D is the circuit depth.

Our Results – negative result





Theorem 2.

Let $n = 2t + 1$. There does **NOT** exist any statistically t-private and statistically correct protocol preparing N random OLE correlations following **any pattern** with communication of $o(N \cdot n)$ elements.

Our Results – positive result

Reference	Communication	Corruption threshold	Security
[DN07, GIP+14, CGH+14…]	$O(\|C\| \cdot n)$	$t = (n-1)/2$	Information-theoretic
[GPS21]	$O(\|C\|)$	$t = (0.5 - \epsilon) \cdot n$	Information-theoretic
Our result	$O(\|C\|) + O(\|C\|)$ OLEs	$t = (n-1)/2$	Information-theoretic
	$\tilde{O}(\|C\|)$		ROM

Theorem 3.

Let n denote the number of parties and $t = \frac{n-1}{2}$ denote the number of corrupted parties. Let κ be the security parameter and $\mathbb{F}$ be a finite field of size $|\mathbb{F}| \geq 2^\kappa$ with each element of ℓ bits length. For an arithmetic circuit C, there exists an MPC protocol in the random oracle model which computes C with malicious security and communicates $O\big((|C| + D \cdot n + poly(n)) \cdot (\ell + \kappa) + n \cdot \kappa^2\big)$ field elements, where D is the circuit depth.

Outline

- **Honest majority MPC with information-theoretic security in OLE-hybrid model**
- Negative results
  - communication lower bound for OLE preparation in information-theoretic setting
- Preparing OLE correlations in Minicrypt

Starting point – preprocessing data of [GPS22]

Theorem [GPS22]. For an arithmetic circuit C over a finite field $\mathbb{F}$ of size $|\mathbb{F}| \geq |C| + n$, and for all constant $\epsilon \geq 0$ and $t = (1 - \epsilon) \cdot n$, there is a semi-honest information-theoretic MPC which computes C with $O(|C|)$ elements of both preprocessing data and communication complexity.

IT MPC with $O(C)$ communication

[GPS22]

$k = O(n)$

$\mathcal{F}_{prep}$

For each group of k multiplication gates,

Sample a random packed Beaver triple $([\boldsymbol{a}]_{n-k}, [\boldsymbol{b}]_{n-k}, [\boldsymbol{c}]_{n-k})$.

Starting point – preproces PS22]

Honest majority is a special case.

Theorem [GPS22]. For an arithmetic circuit C over a finite field $\mathbb{F}$ of size $|\mathbb{F}| \geq |C| + n$, and for all constant $\epsilon \geq 0$ and $\boxed{t = (1-\epsilon) \cdot n}$, there is a semi-honest information-theoretic MPC which computes C with $O(|C|)$ elements of both preprocessing data and communication complexity.

IT MPC with $O(C)$ communication

[GPS22]

$k = O(n)$

$\mathcal{F}_{prep}$

For each group of k multiplication gates,

Sample a random packed Beaver triple $([\boldsymbol{a}]_{n-k}, [\boldsymbol{b}]_{n-k}, [\boldsymbol{c}]_{n-k})$.

Starting point – preproces … GPS22]

Honest majority is a special case.

Theorem [GPS22]. For an arithmetic circuit C over a finite field $\mathbb{F}$ of size $|\mathbb{F}| \geq |C| + n$, and for all constant $\epsilon \geq 0$ and $\boxed{t = (1-\epsilon) \cdot n}$, there is a semi-honest information-theoretic MPC which computes C with $O(|C|)$ elements of both preprocessing data and communication complexity.

IT MPC with $O(C)$ communication

[GPS22]

$k = O(n)$

$\mathcal{F}_{prep}$

For each group of k multiplication gates,

Sample a random packed Beaver triple $([\boldsymbol{a}]_{n-k}, [\boldsymbol{b}]_{n-k}, [\boldsymbol{c}]_{n-k})$.

Packed Beaver triples with $O(n)$ communication per packed triple.

Packed triple generation – packed triple extraction

[CP17, GLS24]

Packed Triple extraction

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

- $N = 2\ell + 1$

  packed triples

- $T = \gamma \cdot N$ of them

  are known by

  corrupted parties

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$d = t + k$

$k = O(n)$

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

Packed Triple extraction

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

- $N = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

$d = t + k$
$k = O(n)$

2 Packed Triple extraction

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

$d = t + k$

$k = O(n)$

- $N = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

2 Packed Triple extraction

3 Output random triples

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

$\ell + 1 - T$ random packed triples

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

- $N = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

$d = t + k$
$k = O(n)$

2 Packed Triple extraction

3 Output random triples

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

$\ell + 1 - T$ random packed triples

$O(\ell \cdot n)$ elements

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

- $N = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

If we have strong honest majority, i.e., $\gamma < \frac{1}{2}$

2 Packed Triple extraction

3 Output random triples

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

$\ell + 1 - T$ random packed triples

$O(\ell \cdot n)$ elements

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

- $N = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

If we have strong honest majority, i.e., $\gamma < \frac{1}{2}$

2 Packed Triple extraction

3 Output random triples

$[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$\left(\frac{1}{2} - \gamma\right) \cdot N$ random packed triples

$O(\ell \cdot n)$ elements

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$

- $N = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

If we have strong honest majority, i.e., $\gamma < \frac{1}{2}$

2 Packed Triple extraction

3 Output random triples

	$[b]_d$	$[c]_d$

$\left(\frac{1}{2} - \gamma\right) \cdot N$ random packed triples

$O(\ell \cdot n)$ elements

$O(n)$ elements per packed triple

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

- $N = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

If we have strong honest majority, i.e., $\gamma < \frac{1}{2}$

2 Packed Triple extraction

3 Output random triples

$[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$\left(\frac{1}{2} - \gamma\right) \cdot N$ random packed triples

$O(\ell \cdot n)$ elements

$O(n)$ elements per packed triple

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

- $N = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

Each party distributes a packed triple.

2 Packed Triple extraction

3 Output random triples

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

$\ell + 1 - T$ random packed triples

$O(\ell \cdot n)$ elements

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

- $N = 2t + 1$ packed triples
- $T = t$ of them are known by corrupted parties

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

Each party distributes a packed triple.

2 Packed Triple extraction

3 Output random triples

$[\boldsymbol{a}]_d$ $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$\ell + 1 - T$ random packed triples

$O(\ell \cdot n)$ elements

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

- $N = 2t + 1$ packed triples
- $T = t$ of them are known by corrupted parties

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

Each party distributes a packed triple.

2 Packed Triple extraction

3 Output random triples

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

$t + 1 - t = 1$ random packed triples

$O(\ell \cdot n)$ elements

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$

- $\boxed{N = 2t + 1}$ packed triples
- $\boxed{T = t}$ of them are known by corrupted parties

Each party distributes a packed triple.

2 Packed Triple extraction

3 Output random triples

$[a]_d$	$[b]_d$	$[c]_d$

$\boxed{t + 1 - t = 1}$ random packed triples

$O(\ell \cdot n)$ elements

$O(t \cdot n)$ per packed triple

Packed triple generation – virtual parties [Bra87]

Each pair of two parties simulates a virtual party.

Packed triple generation – virtual parties [Bra87]

Each pair of two parties simulates a virtual party.

A committee containing an honest party acts as an honest virtual party.

Packed triple generation – virtual parties [Bra87]

Each pair of two parties simulates a virtual party.

$\frac{3}{4} \cdot n^2$ honest virtual parties

$\Rightarrow \gamma = \frac{1}{4} < \frac{1}{2}$

A committee containing an honest party acts as an honest virtual party.

Packed triple generation – virtual parties [Bra87]

Each pair of two parties simulates a virtual party.

$\frac{3}{4} \cdot n^2$ honest virtual parties

$$\Rightarrow \gamma = \frac{1}{4} < \frac{1}{2}$$

A committee containing an honest party acts as an honest virtual party.

Packed triple generation – virtual parties [Bra87]

A committee containing an honest party acts as an honest virtual party.

P_0

a_0, x_0
b_0, y_0

$\mathcal{F}_{OLE}$

b_1, x_1
a_1, y_1

P_1

$$a_0 * b_1 = x_0 + x_1$$

$$a_1 * b_0 = y_0 + y_1$$

$*$ - element-wise multiplication

Packed triple generation – virtual parties [Bra87]

A committee containing an honest party acts as an honest virtual party.

P_0

$\boldsymbol{a_0, x_0}$
$\boldsymbol{b_0, y_0}$

$\leftarrow \mathcal{F}_{OLE} \rightarrow$

$\boldsymbol{b_1, x_1}$
$\boldsymbol{a_1, y_1}$

P_1

$$\boldsymbol{a_0} * \boldsymbol{b_1} = \boldsymbol{x_0} + \boldsymbol{x_1}$$

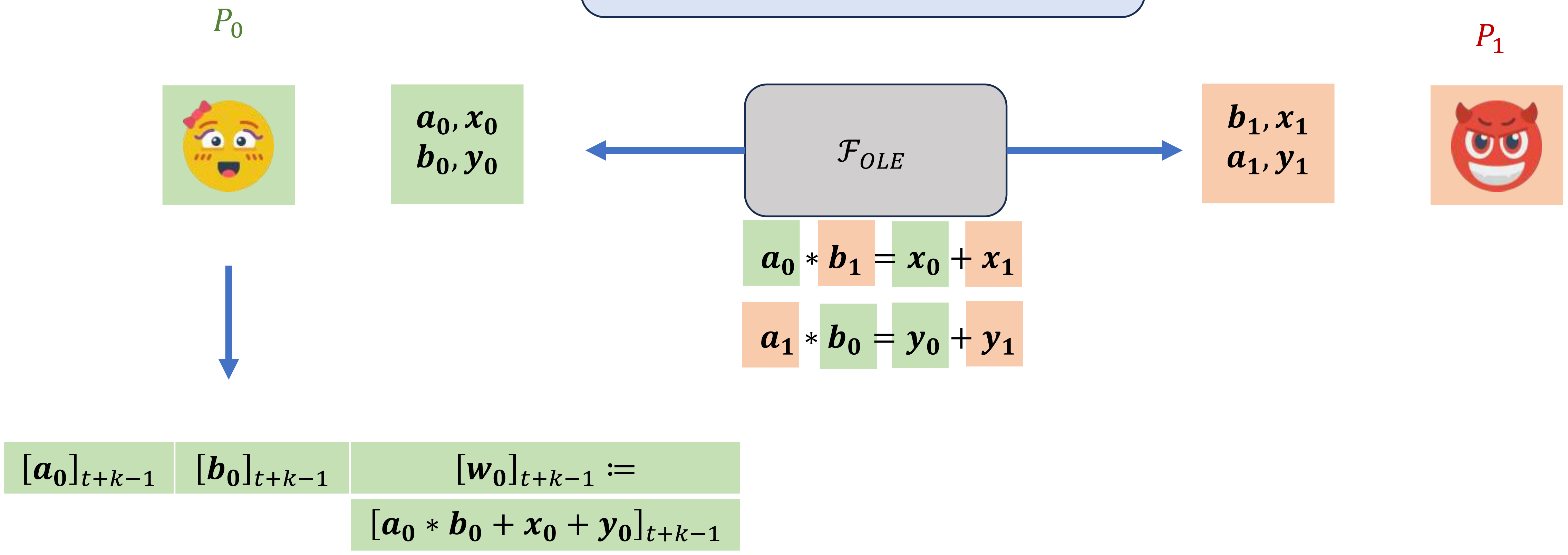
$$\boldsymbol{a_1} * \boldsymbol{b_0} = \boldsymbol{y_0} + \boldsymbol{y_1}$$

$$\boldsymbol{a} * \boldsymbol{b} = (\boldsymbol{a_0} + \boldsymbol{a_1}) * (\boldsymbol{b_0} + \boldsymbol{b_1}) = \underbrace{\boldsymbol{a_0} * \boldsymbol{b_0}}_{P_0} + \underbrace{\boldsymbol{a_0} * \boldsymbol{b_1} + \boldsymbol{a_1} * \boldsymbol{b_0}}_{\text{Additive shares}} + \underbrace{\boldsymbol{a_1} * \boldsymbol{b_1}}_{P_1}$$

$*$ - element-wise multiplication

Packed triple generation – virtual parties [Bra87]

A committee containing an honest party acts as an honest virtual party.



Packed triple generation – virtual parties [Bra87]

A committee containing an honest party acts as an honest virtual party.

P_0

$\boldsymbol{a_0}, \boldsymbol{x_0}$
$\boldsymbol{b_0}, \boldsymbol{y_0}$

$\mathcal{F}_{OLE}$

$\boldsymbol{a_0} * \boldsymbol{b_1} = \boldsymbol{x_0} + \boldsymbol{x_1}$

$\boldsymbol{a_1} * \boldsymbol{b_0} = \boldsymbol{y_0} + \boldsymbol{y_1}$

P_1

$\boldsymbol{b_1}, \boldsymbol{x_1}$
$\boldsymbol{a_1}, \boldsymbol{y_1}$

$[\boldsymbol{a_0}]_{t+k-1}$ $[\boldsymbol{b_0}]_{t+k-1}$ $[\boldsymbol{w_0}]_{t+k-1} \coloneqq [\boldsymbol{a_0} * \boldsymbol{b_0} + \boldsymbol{x_0} + \boldsymbol{y_0}]_{t+k-1}$

$[\boldsymbol{a_1}]_{t+k-1}$ $[\boldsymbol{b_1}]_{t+k-1}$ $[\boldsymbol{w_1}]_{t+k-1} \coloneqq [\boldsymbol{a_1} * \boldsymbol{b_1} + \boldsymbol{x_1} + \boldsymbol{y_1}]_{t+k-1}$

Packed triple generation – virtual parties [Bra87]

A committee containing an honest party acts as an honest virtual party.

P_0

$\boldsymbol{a_0, x_0}$
$\boldsymbol{b_0, y_0}$

$\mathcal{F}_{OLE}$

$\boldsymbol{a_0} * \boldsymbol{b_1} = \boldsymbol{x_0} + \boldsymbol{x_1}$

$\boldsymbol{a_1} * \boldsymbol{b_0} = \boldsymbol{y_0} + \boldsymbol{y_1}$

P_1

$\boldsymbol{b_1, x_1}$
$\boldsymbol{a_1, y_1}$

$[\boldsymbol{a_0}]_{t+k-1}$ $[\boldsymbol{b_0}]_{t+k-1}$ $[\boldsymbol{w_0}]_{t+k-1} \coloneqq [\boldsymbol{a_0} * \boldsymbol{b_0} + \boldsymbol{x_0} + \boldsymbol{y_0}]_{t+k-1}$

$[\boldsymbol{a_1}]_{t+k-1}$ $[\boldsymbol{b_1}]_{t+k-1}$ $[\boldsymbol{w_1}]_{t+k-1} \coloneqq [\boldsymbol{a_1} * \boldsymbol{b_1} + \boldsymbol{x_1} + \boldsymbol{y_1}]_{t+k-1}$

$[\boldsymbol{a}]_{t+k-1}$ $[\boldsymbol{b}]_{t+k-1}$ $[\boldsymbol{a} * \boldsymbol{b}]_{t+k-1}$

Packed triple generation – virtual parties [Bra87]

A committee containing an honest party acts as an honest virtual party.

P_0

$\boldsymbol{a_0}, \boldsymbol{x_0}$
$\boldsymbol{b_0}, \boldsymbol{y_0}$

$\mathcal{F}_{OLE}$

$\boldsymbol{a_0} * \boldsymbol{b_1} = \boldsymbol{x_0} + \boldsymbol{x_1}$

$\boldsymbol{a_1} * \boldsymbol{b_0} =$

$\boldsymbol{b_1}, \boldsymbol{x_1}$
$\boldsymbol{a_1}, \boldsymbol{y_1}$

P_1

Privacy: the secrets are unknown to the adversary.

$[\boldsymbol{a_0}]_{t+k-1}$ $[\boldsymbol{b_0}]_{t+k-1}$ $[\boldsymbol{w_0}]_{t+k-1} :=$ $[\boldsymbol{a_0} * \boldsymbol{b_0} + \boldsymbol{x_0} + \boldsymbol{y_0}]_{t+k-1}$

$[\boldsymbol{w_1}]_{t+k-1} :=$ $[\boldsymbol{a_1} * \boldsymbol{b_1} + \boldsymbol{x_1} + \boldsymbol{y_1}]_{t+k-1}$

$[\boldsymbol{a}]_{t+k-1}$ $[\boldsymbol{b}]_{t+k-1}$ $[\boldsymbol{a} * \boldsymbol{b}]_{t+k-1}$

Packed triple generation – virtual parties [Bra87]

A committee containing an honest party acts as an honest virtual party.

P_0

$\boldsymbol{a_0}, \boldsymbol{x_0}$
$\boldsymbol{b_0}, \boldsymbol{y_0}$

$\mathcal{F}_{OLE}$

$\boldsymbol{b_1}, \boldsymbol{x_1}$
$\boldsymbol{a_1}, \boldsymbol{y_1}$

P_1

$\boldsymbol{a_0} * \boldsymbol{b_1} = \boldsymbol{x_0} + \boldsymbol{x_1}$

$\boldsymbol{a_1} * \boldsymbol{b_0} =$

<u>Cost</u>: $2k$ invocations of $\mathcal{F}_{OLE}$ + $O(n)$ elements per packed triple

$[\boldsymbol{a_0}]_{t+k-1}$ $[\boldsymbol{b_0}]_{t+k-1}$ $[\boldsymbol{w_0}]_{t+k-1} :=$ $[\boldsymbol{a_0} * \boldsymbol{b_0} + \boldsymbol{x_0} + \boldsymbol{y_0}]_{t+k-1}$

$[\boldsymbol{w_1}]_{t+k-1} :=$ $[\boldsymbol{a_1} * \boldsymbol{b_1} + \boldsymbol{x_1} + \boldsymbol{y_1}]_{t+k-1}$

$[\boldsymbol{a}]_{t+k-1}$ $[\boldsymbol{b}]_{t+k-1}$ $[\boldsymbol{a} * \boldsymbol{b}]_{t+k-1}$

Packed triple generation – packed triple extraction

[CP17, GLS24]

❶ Triple distribution

- $\mathrm{N} = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

❷ Packed Triple extraction

❸ Output random triples

$\left(\frac{1}{2} - \gamma\right) \cdot N$ random packed triples

$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

$O(\ell \cdot n)$ elements

We have strong honest majority, i.e., $\gamma < \frac{1}{2}$

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

$[\boldsymbol{a}]_{\boldsymbol{d}}$	$[\boldsymbol{b}]_{\boldsymbol{d}}$	$[\boldsymbol{c}]_{\boldsymbol{d}}$
$[\boldsymbol{a}]_{\boldsymbol{d}}$	$[\boldsymbol{b}]_{\boldsymbol{d}}$	$[\boldsymbol{c}]_{\boldsymbol{d}}$
$[\boldsymbol{a}]_{\boldsymbol{d}}$	$[\boldsymbol{b}]_{\boldsymbol{d}}$	$[\boldsymbol{c}]_{\boldsymbol{d}}$
$[\boldsymbol{a}]_{\boldsymbol{d}}$	$[\boldsymbol{b}]_{\boldsymbol{d}}$	$[\boldsymbol{c}]_{\boldsymbol{d}}$
$[\boldsymbol{a}]_{\boldsymbol{d}}$	$[\boldsymbol{b}]_{\boldsymbol{d}}$	$[\boldsymbol{c}]_{\boldsymbol{d}}$

- $\mathrm{N} = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

We have strong honest majority, i.e., $\gamma < \frac{1}{2}$

2 Packed Triple extraction

3 Output random triples

$\left(\frac{1}{2} - \gamma\right) \cdot N$ $[\boldsymbol{b}]_{\boldsymbol{d}}$ $[\boldsymbol{c}]_{\boldsymbol{d}}$

random packed triples

$O(\ell \cdot n)$ elements

$O(n)$ elements plus $O(n)$ OLEs per packed triple

Packed triple generation – packed triple extraction

[CP17, GLS24]

1 Triple distribution

- $\mathrm{N} = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$
$[\boldsymbol{a}]_d$	$[\boldsymbol{b}]_d$	$[\boldsymbol{c}]_d$

We have strong honest majority, i.e., $\gamma < \frac{1}{2}$

2 Packed Triple extraction

3 Output random triples

$\left(\frac{1}{2} - \gamma\right) \cdot N$ random packed triples $[\boldsymbol{b}]_d$ $[\boldsymbol{c}]_d$

$O(\ell \cdot n)$ elements

Switch to **malicious** setting?

$O(n)$ elements plus $O(n)$ OLEs per packed triple

Packed triple generation – malicious security

[BY24]

1 Triple distribution

$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$

- $\mathrm{N} = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

2 Packed Triple extraction

3 Output random triples

$[a']_d$	$[b']_d$	$[c']_d$

$\ell + 1 - T$ random packed triples

Packed triple generation – malicious security

[BY24]

❶ Triple distribution

$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$

- $\mathrm{N} = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

❷ Packed Triple extraction

❸ Output random triples

$a' * b' \neq c'$

$[a']_d$	$[b']_d$	$[c']_d$

$\ell + 1 - T$ random packed triples

Packed triple generation – malicious security

[BY24]

1 Triple distribution

$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$
$[a]_d$	$[b]_d$	$[c]_d$

- $\mathrm{N} = 2\ell + 1$ packed triples
- $T = \gamma \cdot N$ of them are known by corrupted parties

2 Packed Triple extraction

3 Output random triples

$\boldsymbol{a}' * \boldsymbol{b}' \neq \boldsymbol{c}'$

$[\boldsymbol{a}']_d$	$[\boldsymbol{b}']_d$	$[\boldsymbol{c}']_d$

$\ell + 1 - T$ random packed triples

Triple verification to ensure: $\boldsymbol{a} * \boldsymbol{b} = \boldsymbol{c}$

Packed triple generation – malicious security

[GPS22, BY24]

$\mathcal{F}_{prep}$

For each group of k multiplication gates,

Sample a random packed Beaver triple $([\boldsymbol{a}]_{n-k}, [\boldsymbol{b}]_{n-k}, [\boldsymbol{c}]_{n-k})$.

Packed triple generation – malicious security

[GPS22, BY24]

$\mathcal{F}_{prep}$

For each group of k multiplication gates,

Sample a random packed Beaver triple $([\boldsymbol{a}]_{n-k}, [\boldsymbol{b}]_{n-k}, [\boldsymbol{c}]_{n-k})$.

$\mathcal{F}_{prep-mal}$

For each group of k multiplication gates,

Sample an authenticated random packed Beaver triple $([\boldsymbol{a}]_{n-k}, [\boldsymbol{b}]_{n-k}, [\boldsymbol{c}]_{n-k}, [\boldsymbol{\gamma} * \boldsymbol{a}]_{n-k}, [\boldsymbol{\gamma} * \boldsymbol{b}]_{n-k}, [\boldsymbol{\gamma} * \boldsymbol{c}]_{n-k})$.

$\boldsymbol{\gamma} \in \mathbb{F}^k$ - MAC key

Packed triple generation – malicious security

[GPS22, BY24]

$\mathcal{F}_{prep}$

For each group of k multiplication gates,

Sample a random packed Beaver triple $([\boldsymbol{a}]_{n-k}, [\boldsymbol{b}]_{n-k}, [\boldsymbol{c}]_{n-k})$.

$\mathcal{F}_{prep-mal}$

$\boldsymbol{\gamma} \in \mathbb{F}^k$ - MAC key

For each group of k multiplication gates,

Sample an authenticated random packed Beaver triple $([\boldsymbol{a}]_{n-k}, [\boldsymbol{b}]_{n-k}, [\boldsymbol{c}]_{n-k}, [\boldsymbol{\gamma} * \boldsymbol{a}]_{n-k}, [\boldsymbol{\gamma} * \boldsymbol{b}]_{n-k}, [\boldsymbol{\gamma} * \boldsymbol{c}]_{n-k})$.

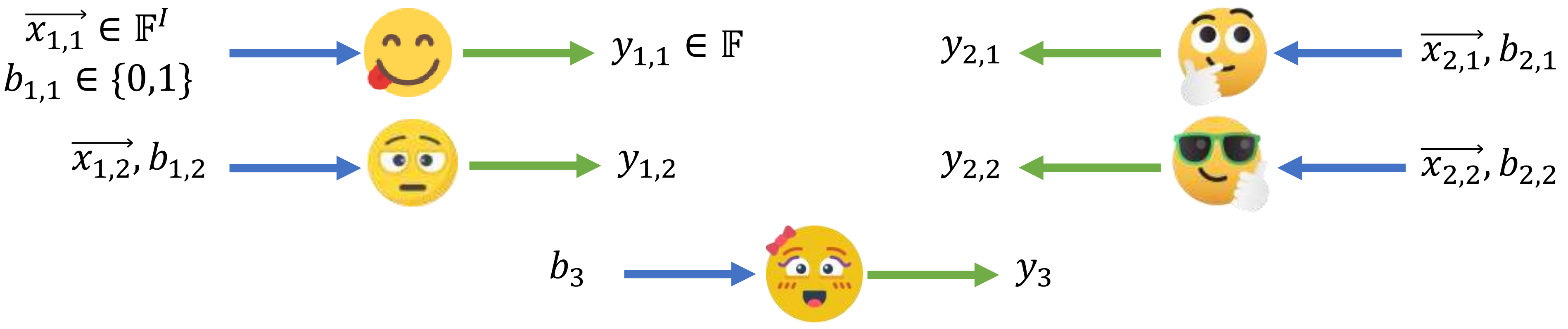
$O(1)$ Packed Beaver triples

Verification by sacrificing

Outline

- Honest majority MPC with information-theoretic security in OLE-hybrid model
- **Negative results**
  - **communication lower bound for OLE preparation in information-theoretic setting**
- Preparing OLE correlations in Minicrypt

Communication lower bound in [DLN19]

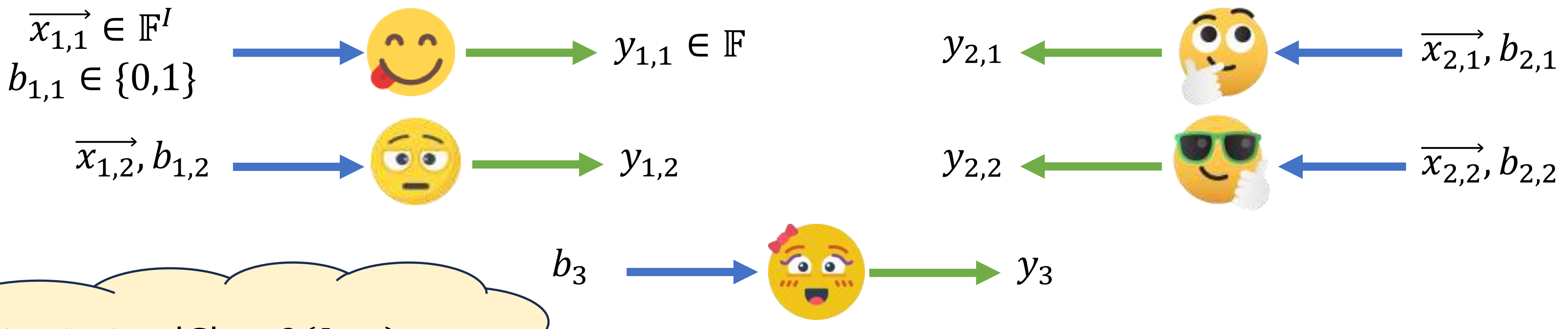


$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \dots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \dots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Theorem [DLN19] Let $n = 2t + 1$. Any statistically t-private and statistically correct protocol for $IP_{n,I}$ communicates at least $\frac{In(t-1)}{2} - negl$ elements.

Communication lower bound in [DLN19]



Circuit size: $|C| = O(I \cdot n)$

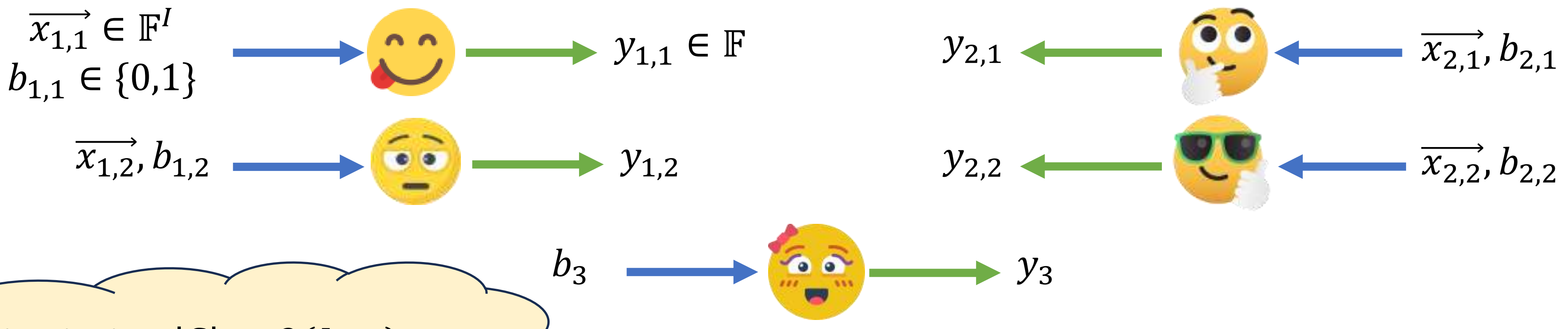
Input size: $I_C = O(I \cdot n)$

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \ldots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \ldots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Theorem [DLN19] Let $n = 2t + 1$. Any statistically t-private and statistically correct protocol for $IP_{n,I}$ communicates at least $\frac{In(t-1)}{2} - negl$ elements.

Communication lower bound in [DLN19]



Circuit size: $|C| = O(I \cdot n)$

Input size: $I_C = O(I \cdot n)$

$IP_{n,I}$

1. $y \leftarrow IP\left(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \ldots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \ldots \overrightarrow{x_{2,t}}\right)$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Theorem [DLN19] Let $n = 2t + 1$. Any statistically t-private and statistically correct protocol for $IP_{n,I}$ communicates at least $\boxed{\Omega(|C| \cdot n)}$ elements.

Lower bounds for preparing OLEs

$IP_{n,I}$

1. $y \leftarrow IP\left(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \ldots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \ldots \overrightarrow{x_{2,t}}\right)$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Π_{IP} + $O(|C|)$ OLEs

Cost: $O(|C|)$ elements

Lower bounds for preparing OLEs

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \dots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \dots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Π_{IP} + $O(|C|)$ OLEs

Cost: $O(|C|)$ elements

IT-MPCs preparing $O(|C|)$ OLEs with $O(|C|)$ elements

Lower bounds for preparing OLEs

$IP_{n,I}$

1. $y \leftarrow IP\left(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \ldots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \ldots \overrightarrow{x_{2,t}}\right)$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Π_{IP} + $O(|C|)$ OLEs

Cost: $O(|C|)$ elements

IT-MPCs for $IP_{n,I}$ with $O(|C|)$ elements

IT-MPCs preparing $O(|C|)$ OLEs with $O(|C|)$ elements

Lower bounds for preparing OLEs

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \ldots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \ldots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Π_{IP} + $O(|C|)$ OLEs

Cost: $O(|C|)$ elements

Impossibility result in [DLN19]

There is **NO**

IT-MPCs for $IP_{n,I}$ with $O(|C|)$ elements

IT-MPCs preparing $O(|C|)$ OLEs with $O(|C|)$ elements

Lower bounds for preparing OLEs

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \ldots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \ldots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Π_{IP} + $O(|C|)$ OLEs

Cost: $O(|C|)$ elements

Impossibility result in [DLN19]

There is **NO**

IT-MPCs for $IP_{n,I}$ with $O(|C|)$ elements

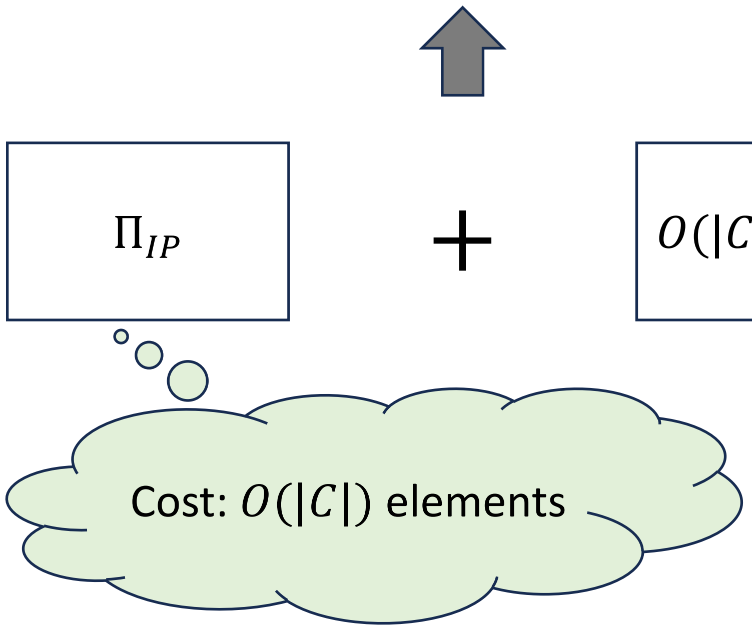
There is **NO**

IT-MPCs preparing $O(|C|)$ OLEs with $O(|C|)$ elements

Lower bounds for preparing OLEs

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \ldots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \ldots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$



There is **NO**

IT-MPCs preparing $O(|C|)$ OLEs with $O(|C|)$ elements

a specific pattern

Lower bounds for preparing OLEs

$IP_{n,I}$

1. $y \leftarrow IP\left(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \dots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \dots \overrightarrow{x_{2,t}}\right)$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Π_{IP} + $O(|C|)$ OLEs

Cost: $O(|C|)$ elements

Follow a specific (non-uniform) **pattern**.

There is **NO**

IT-MPCs preparing $\boxed{O(|C|)\text{ OLEs}}$ with $O(|C|)$ elements

a specific pattern

Lower bounds for preparing OLEs

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \dots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \dots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Π_{IP} + $O(|C|)$ OLEs

Cost: $O(|C|)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	...	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	...	$c_{3,4}$

Follow a specific (non-uniform) **pattern**.

There is **NO**

IT-MPCs preparing $\boxed{O(|C|) \text{ OLEs}}$ with $O(|C|)$ elements

Lower bounds for prepa

a specific pattern

MPC ⇐ Packed triples ⇐ OLEs with a **uniform pattern**

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

$IP_{n,I}$

1. $y \leftarrow IP\left(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \dots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \dots \overrightarrow{x_{2,t}}\right)$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Π_{IP} + $O(|C|)$ OLEs

Cost: $O(|C|)$ elements

Follow a specific (non-uniform) **pattern**.

There is **NO**

IT-MPCs preparing $\boxed{O(|C|)\text{ OLEs}}$ with $O(|C|)$ elements

a specific pattern

Lower bounds for prepa

MPC ← Packed triples ← OLEs with a **uniform pattern**

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

$IP_{n,I}$

1. $y \leftarrow IP(\overrightarrow{x_{1,1}}\ \overrightarrow{x_{1,2}} \dots \overrightarrow{x_{1,t}}, \overrightarrow{x_{2,1}}\ \overrightarrow{x_{2,2}} \dots \overrightarrow{x_{2,t}})$
2. $y_{j,i} \leftarrow b_{j,i} \cdot y$

Π_{IP} + $O(|C|)$ OLEs

Cost: $O(|C|)$ elements

An arbitrary pattern?

There is **NO**

IT-MPCs preparing $\boxed{O(|C|) \text{ OLEs}}$ with $O(|C|)$ elements

a specific pattern

for any pair of parties

Lower bounds for preparing OLEs

IT-MPCs preparing $O(C)$ OLEs following the **specific pattern** with $O(C)$ elements

⇑

IT-MPCs preparing $O(C)$ OLEs for **any pair** of parties with $O(C)$ elements

a specific pattern

for any pair of parties

Lower bounds for preparing OLEs

There is **NO**

IT-MPCs preparing $O(C)$ OLEs following the **specific pattern** with $O(C)$ elements

IT-MPCs preparing $O(C)$ OLEs for **any pair** of parties with $O(C)$ elements

a specific pattern

for any pair of parties

Lower bounds for preparing OLEs

There is **NO**

IT-MPCs preparing $O(C)$ OLEs following the **specific pattern** with $O(C)$ elements

There is **NO**

IT-MPCs preparing $O(C)$ OLEs for **any pair** of parties with $O(C)$ elements

a specific pattern

for any pair of parties

a uniform pattern

Lower bounds for preparing OLEs [CP17]

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

IT-MPCs preparing $O(C)$ OLEs with a **uniform pattern** with $O(C)$ elements

Lower bounds for preparing OLEs [CP17]

a specific pattern

for any pair of parties

a uniform pattern

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

IT-MPCs preparing $O(C)$ OLEs with a **uniform pattern** with $O(C)$ elements

C OLEs with a uniform pattern

a specific pattern

for any pair of parties

a uniform pattern

Lower bounds for preparing OLEs [CP17]

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

Adversary only knows $\boldsymbol{\gamma = \frac{1}{4} < \frac{1}{2}}$ fraction of OLEs.

IT-MPCs preparing $O(C)$ OLEs with a **uniform pattern** with $O(C)$ elements

C OLEs with a uniform pattern

Lower bounds for preparing OLEs [CP17]

a specific pattern

for any pair of parties

a uniform pattern

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

$\left(\frac{1}{2} - \gamma\right) \cdot C$ random OLEs btw (P_0, P_1)

Adversary only knows $\boldsymbol{\gamma = \frac{1}{4} < \frac{1}{2}}$ fraction of OLEs.

OLE extraction

IT-MPCs preparing $O(C)$ OLEs with a **uniform pattern** with $O(C)$ elements

C OLEs with a uniform pattern

Lower bounds for preparing OLEs [CP17]

a specific pattern

for any pair of parties

a uniform pattern

There is **NO**

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

$\left(\frac{1}{2} - \gamma\right) \cdot C$ random OLEs btw (P_0, P_1)

Adversary only knows $\gamma = \frac{1}{4} < \frac{1}{2}$ fraction of OLEs.

OLE extraction

IT-MPCs preparing $O(C)$ OLEs with a **uniform pattern** with $O(C)$ elements

C OLEs with a uniform pattern

Lower bounds for preparing OLEs [CP17]

a specific pattern

for any pair of parties

a uniform pattern

There is **NO**

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

$\left(\frac{1}{2} - \gamma\right) \cdot C$ random OLEs btw (P_0, P_1)

Adversary only knows $\gamma = \frac{1}{4} < \frac{1}{2}$ fraction of OLEs.

OLE extraction

There is **NO**

IT-MPCs preparing $O(C)$ OLEs with a **uniform pattern** with $O(C)$ elements

C OLEs with a uniform pattern

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

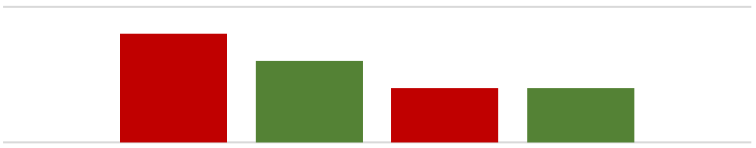
a uniform pattern

an arbitrary pattern

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_2, P_3)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

OLE pattern



Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_2, P_3)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

OLE pattern

Apply a perm

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_2, P_3)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

OLE pattern

Apply a perm

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_2, P_3)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

OLE pattern

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	$(\boldsymbol{P_1}, \boldsymbol{P_2})$	$(\boldsymbol{P_1}, \boldsymbol{P_3})$	...	$(\boldsymbol{P_3}, \boldsymbol{P_4})$
# of OLEs	$c_{1,2}$	$c_{1,3}$	...	$c_{3,4}$

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

Lower bounds for preparing OLEs

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Fix the set of corrupted parties

Pair of parties	$(\boldsymbol{P_1}, \boldsymbol{P_2})$	$(\boldsymbol{P_1}, \boldsymbol{P_3})$	...	$(\boldsymbol{P_3}, \boldsymbol{P_4})$
# of OLEs	$c_{1,2}$	$c_{1,3}$	...	$c_{3,4}$

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Fix the set of corrupted parties

Select a random perm

Pair of parties	$(\boldsymbol{P_1}, \boldsymbol{P_2})$	$(\boldsymbol{P_1}, \boldsymbol{P_3})$	…	$(\boldsymbol{P_3}, \boldsymbol{P_4})$
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

Adversary only knows $\gamma = \frac{1}{4} < \frac{1}{2}$ fraction of OLEs in expectation.

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	...	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	...	$c_{3,4}$

Fix the set of corrupted parties

Select a random perm

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	...	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	...	$c_{3,4}$

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

Adversary only knows $\gamma = \frac{1}{4} < \frac{1}{2}$ fraction of OLEs in expectation.

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

$O(n+\kappa)$ times + Chernoff's bd + union bd

Fix the set of corrupted parties

Select a random perm

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

Adversary only knows $\gamma = \frac{3}{8} < \frac{1}{2}$ fraction of OLEs w.h.p.

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

Pair of parties	$(\boldsymbol{P_1}, \boldsymbol{P_2})$	$(\boldsymbol{P_1}, \boldsymbol{P_3})$	…	$(\boldsymbol{P_3}, \boldsymbol{P_4})$
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

$O(n+\kappa)$ times + Chernoff's bd + union bd

For **any** set of corrupted parties

Select a random perm

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	$(\boldsymbol{P_1}, \boldsymbol{P_2})$	$(\boldsymbol{P_1}, \boldsymbol{P_3})$	…	$(\boldsymbol{P_3}, \boldsymbol{P_4})$
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

Adversary only knows $\gamma = \frac{3}{8} < \frac{1}{2}$ fraction of OLEs w.h.p.

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

OLE extraction

Pair of parties	(P_1, P_2)	(P_1, P_3)	...	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	...	$c_{3,4}$

$O(n+\kappa)$ times + Chernoff's bd + union bd

For **any** set of corrupted parties

Select a random perm

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	...	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	...	$c_{3,4}$

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

Adversary only knows $\gamma = \frac{3}{8} < \frac{1}{2}$

fraction of OLEs w.h.p.

There is **NO**

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

OLE extraction

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

$O(n + \kappa)$ times + Chernoff's bd + union bd

For **any** set of corrupted parties

Select a random perm

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

Lower bounds for preparing OLEs

a specific pattern

for any pair of parties

a uniform pattern

an arbitrary pattern

Adversary only knows $\gamma = \frac{3}{8} < \frac{1}{2}$ fraction of OLEs w.h.p.

There is **NO**

IT-MPCs preparing $O(C)$ OLEs for $(\boldsymbol{P_0}, \boldsymbol{P_1})$ with $O(C)$ elements

OLE extraction

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

$O(n+\kappa)$ times + Chernoff's bd + union bd

For **any** set of corrupted parties

Select a random perm

There is **NO**

IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Pair of parties	(P_1, P_2)	(P_1, P_3)	…	(P_3, P_4)
# of OLEs	$c_{1,2}$	$c_{1,3}$	…	$c_{3,4}$

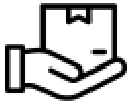
Outline

- Honest majority MPC with information-theoretic security in OLE-hybrid model
- Negative results
    - communication lower bound for OLE preparation in information-theoretic setting
- **Preparing OLE correlations in Minicrypt**

Sec par: κ
Length: ℓ

Preparing OLE correlations – base OT [GS20]

$\rightarrow$ **1** $\mathcal{F}_{OT}(\kappa, \kappa)$ $\rightarrow$ $\mathcal{F}_{ROT}(\ell, O(M \cdot \ell))$ $\rightarrow$ $\mathcal{F}_{OLE}(M)$



κ OTs

m_0	b
m_1	m_b
$\vdots$	
m_0	b
m_1	m_b

κ bits

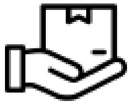
$$m_b = m_0 + b \cdot (m_1 - m_0)$$

Sec par: κ
Length: ℓ

Preparing OLE correlations – base OT [GS20]

1

$\mathcal{F}_{OT}(\kappa, \kappa)$ → $\mathcal{F}_{ROT}(\ell, O(M \cdot \ell))$ → $\mathcal{F}_{OLE}(M)$



κ OTs

m_0	b
m_1	m_b
⋮	
m_0	b
m_1	m_b

κ bits

$$m_b = m_0 + b \cdot (m_1 - m_0)$$

DN protocol with malicious security [GS20]

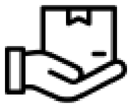
$O(\kappa^2 \cdot n + n^2 \cdot \kappa)$ bits

Preparing OLE correlation – OT extension

[IKNP03, KOS15]

Sec par: κ
Length: ℓ

$\mathcal{F}_{OT}(\kappa, \kappa)$ → **2** $\mathcal{F}_{ROT}(\ell, O(\boldsymbol{M} \cdot \ell))$ → $\mathcal{F}_{OLE}(\boldsymbol{M})$



κ OTs

m_0	b
m_1	m_b
⋮	
m_0	b
m_1	m_b

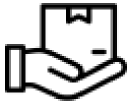
κ bits

Preparing OLE correlation – OT extension

[IKNP03, KOS15]

Sec par: κ
Length: ℓ

$\mathcal{F}_{OT}(\kappa, \kappa)$ → **2** $\mathcal{F}_{ROT}(\ell, O(\boldsymbol{M} \cdot \ell))$ → $\mathcal{F}_{OLE}(\boldsymbol{M})$



κ OTs

m_0	b
m_1	m_b
⋮	
m_0	b
m_1	m_b

κ bits

OT extension with malicious security [KOS15]

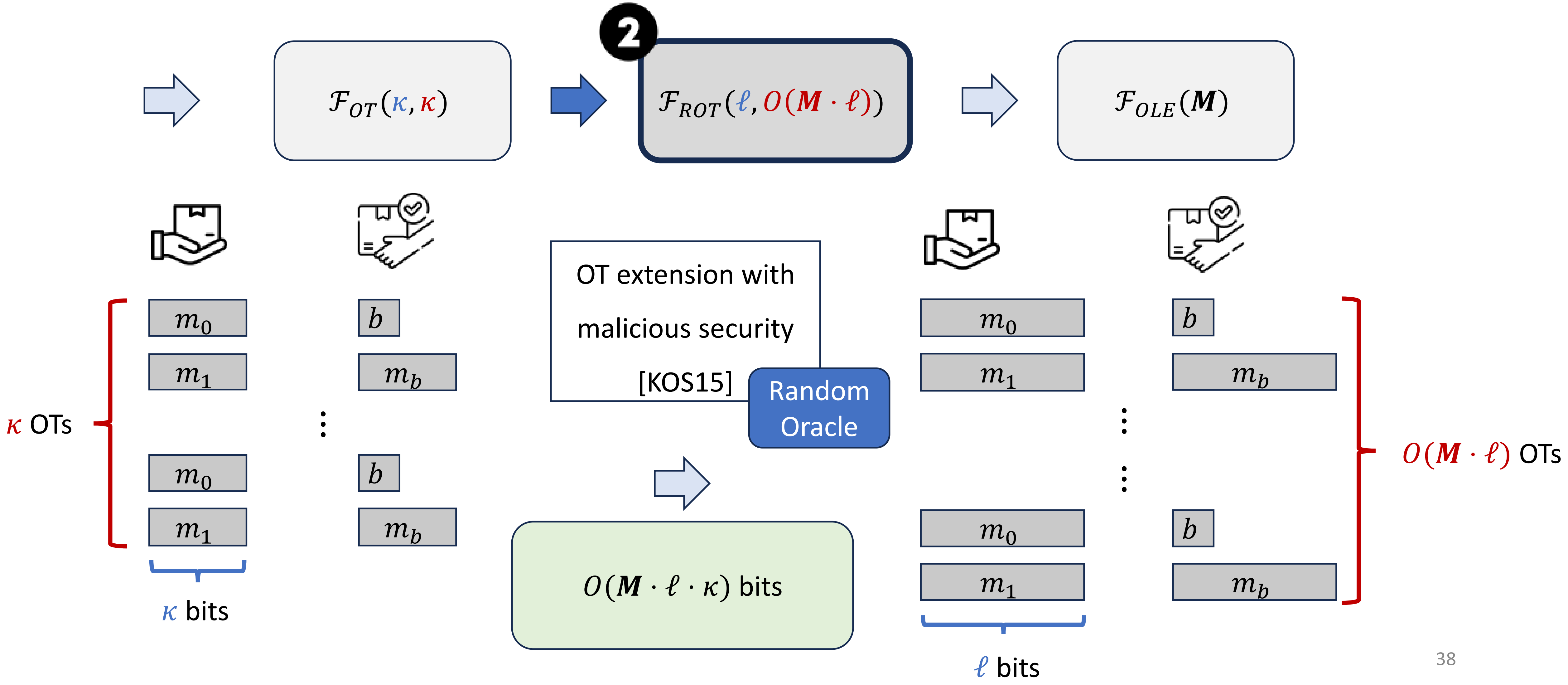
Random Oracle

$O(\boldsymbol{M} \cdot \ell \cdot \kappa)$ bits

Preparing OLE correlation – OT extension

[IKNP03, KOS15]

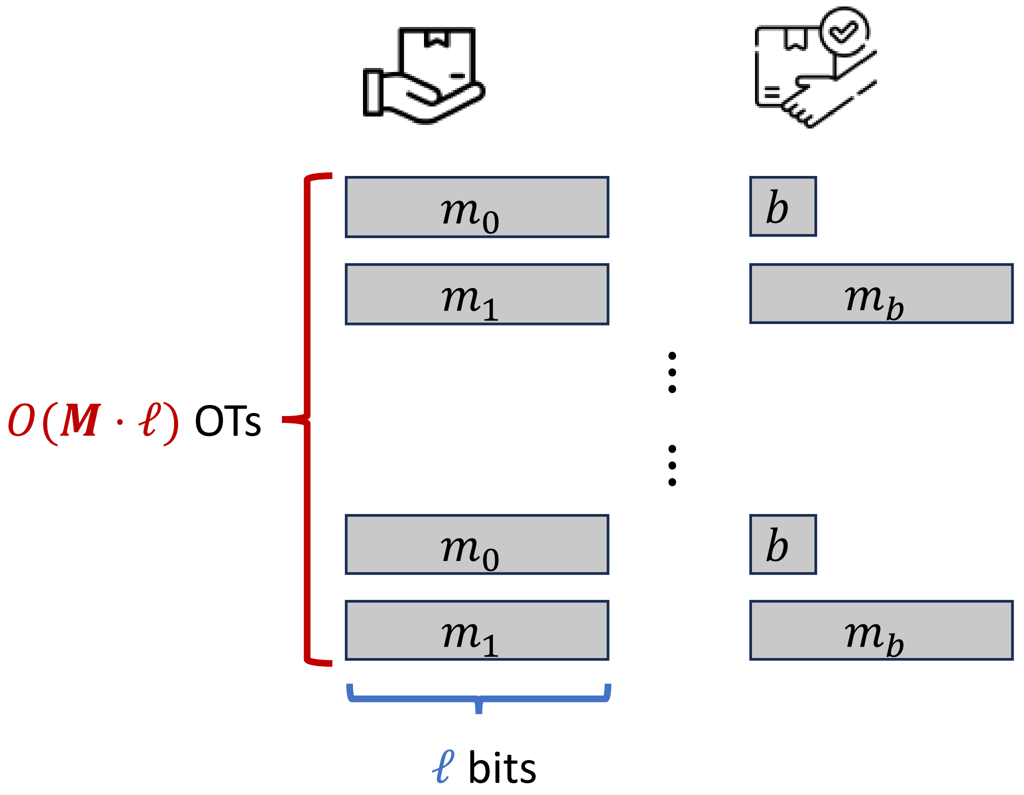
Sec par: κ
Length: ℓ



Sec par: κ
Length: ℓ

Preparing OLE correlation – OT to OLE [KOS16]

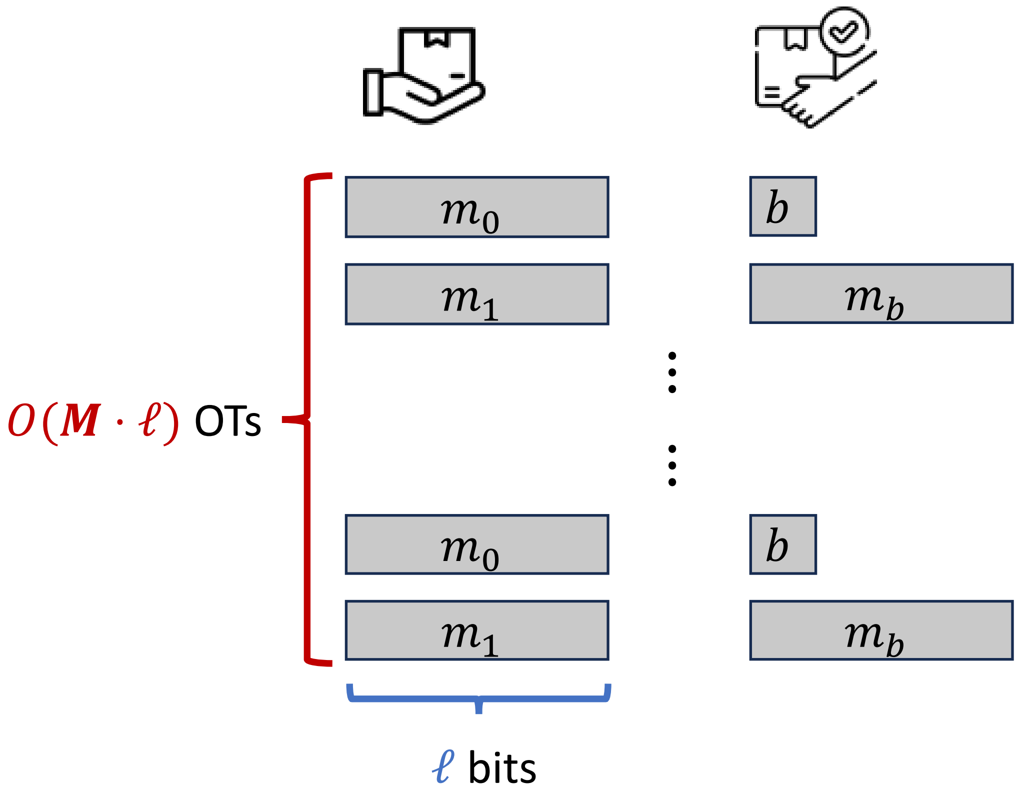
$\Rightarrow \mathcal{F}_{OT}(\kappa, \kappa) \Rightarrow \mathcal{F}_{ROT}(\ell, O(\boldsymbol{M} \cdot \ell)) \Rightarrow$ **3** $\mathcal{F}_{OLE}(\boldsymbol{M})$



Sec par: κ
Length: ℓ

Preparing OLE correlation – OT to OLE [KOS16]

$\mathcal{F}_{OT}(\kappa, \kappa)$ → $\mathcal{F}_{ROT}(\ell, O(\boldsymbol{M} \cdot \ell))$ → 3 $\mathcal{F}_{OLE}(\boldsymbol{M})$

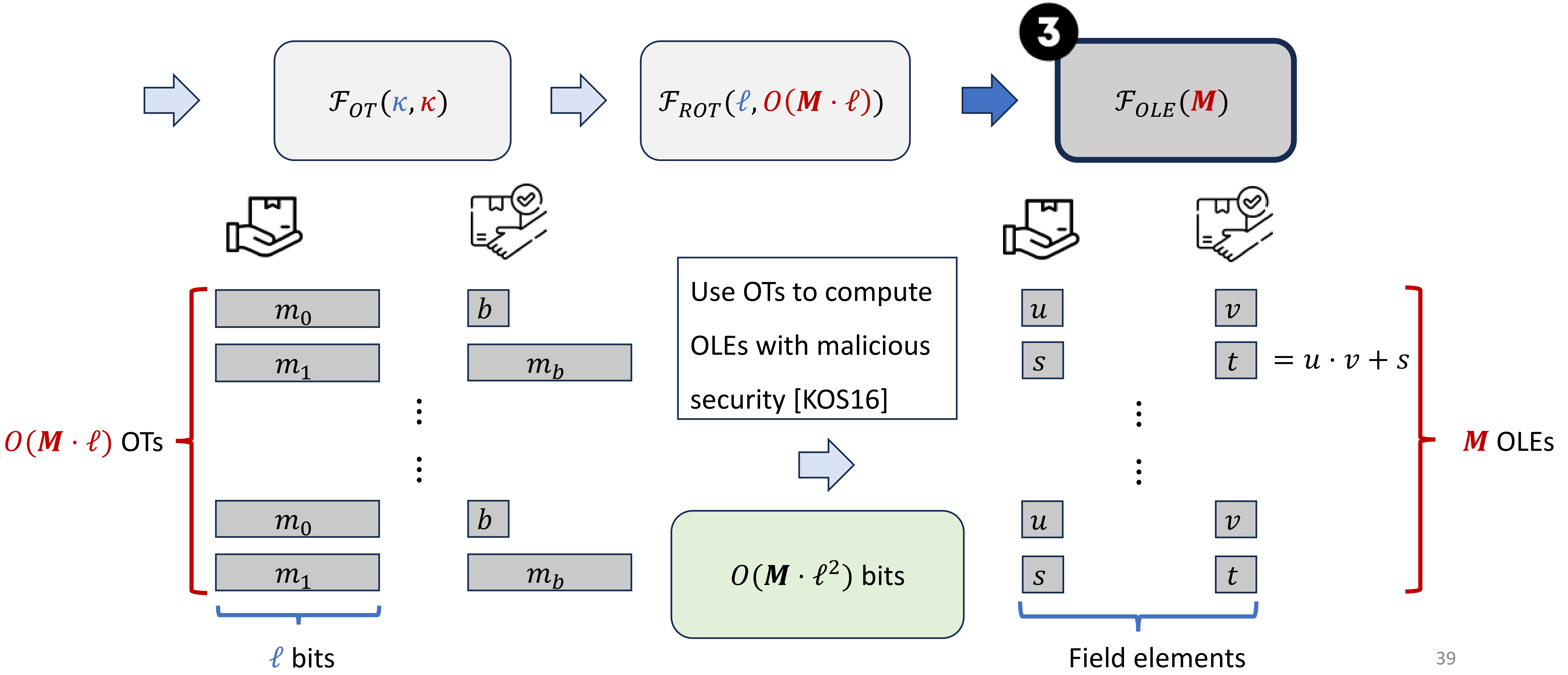


Use OTs to compute OLEs with malicious security [KOS16]

$O(\boldsymbol{M} \cdot \ell^2)$ bits

Sec par: κ
Length: ℓ

Preparing OLE correlation – OT to OLE [KOS16]



Preparing OLE correlation – OT to OLE [KOS16]

Sec par: κ
Length: ℓ

$\Rightarrow$ $\mathcal{F}_{OT}(\kappa, \kappa)$ $\Rightarrow$ $\mathcal{F}_{ROT}(\ell, O(\boldsymbol{M} \cdot \ell))$ $\Rightarrow$ $\mathcal{F}_{OLE}(\boldsymbol{M})$

$O(\kappa^2 \cdot n + n^2 \cdot \kappa)$ bits $+$ $O(\boldsymbol{M} \cdot \ell \cdot \kappa)$ bits $+$ $O(\boldsymbol{M} \cdot \ell^2)$ bits

$=$ $O(\boldsymbol{M} \cdot \ell^2 + \boldsymbol{M} \cdot \ell \cdot \kappa + \kappa^2 \cdot n + n^2 \cdot \kappa)$ bits

for M OLEs

Preparing OLE correlation – OT to OLE [KOS16]

Sec par: κ
Length: ℓ

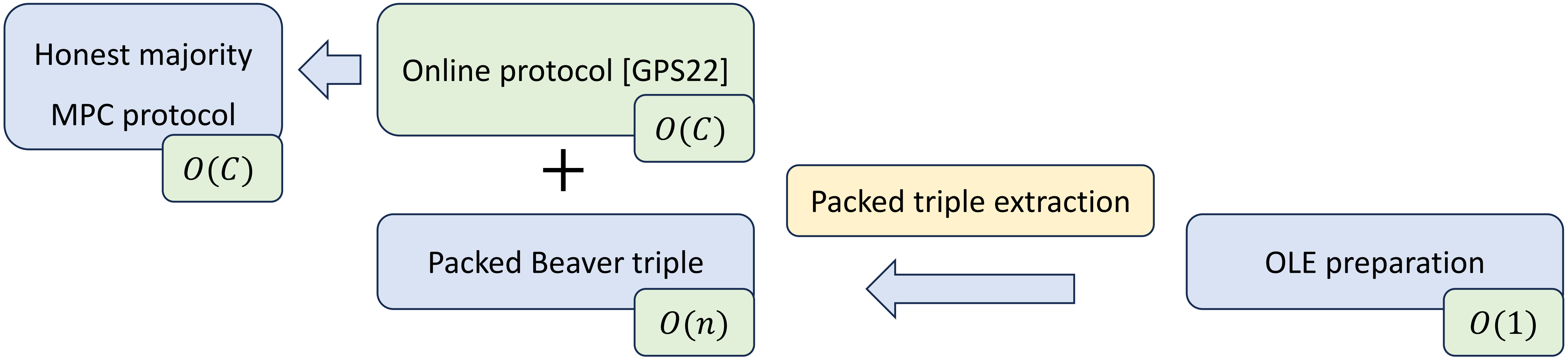
$\mathcal{F}_{OT}(\kappa, \kappa)$ → $\mathcal{F}_{ROT}(\ell, O(\boldsymbol{M} \cdot \ell))$ → $\mathcal{F}_{OLE}(\boldsymbol{M})$

$O(\kappa^2 \cdot n + n^2 \cdot \kappa)$ bits + $O(\boldsymbol{M} \cdot \ell \cdot \kappa)$ bits + $O(\boldsymbol{M} \cdot \ell^2)$ bits

= $O(\boldsymbol{M} \cdot \ell^2 + \boldsymbol{M} \cdot \ell \cdot \kappa + \kappa^2 \cdot n + n^2 \cdot \kappa)$ bits for M OLEs

$O(\ell + \kappa)$ elements per OLE

Summary



Honest majority
MPC protocol
$O(C)$
Online protocol [GPS22]
$O(C)$
+
Packed Beaver triple
$O(n)$
Packed triple extraction
OLE preparation
$O(1)$

Summary

Boost to **strong honest majority** using idea of **party virtualization**

Honest majority MPC protocol $O(C)$

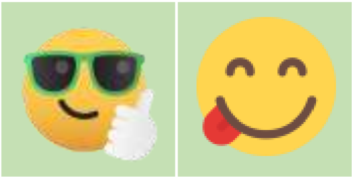
Online protocol [GPS22] $O(C)$

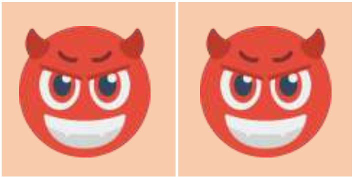
+

Packed Beaver triple $O(n)$

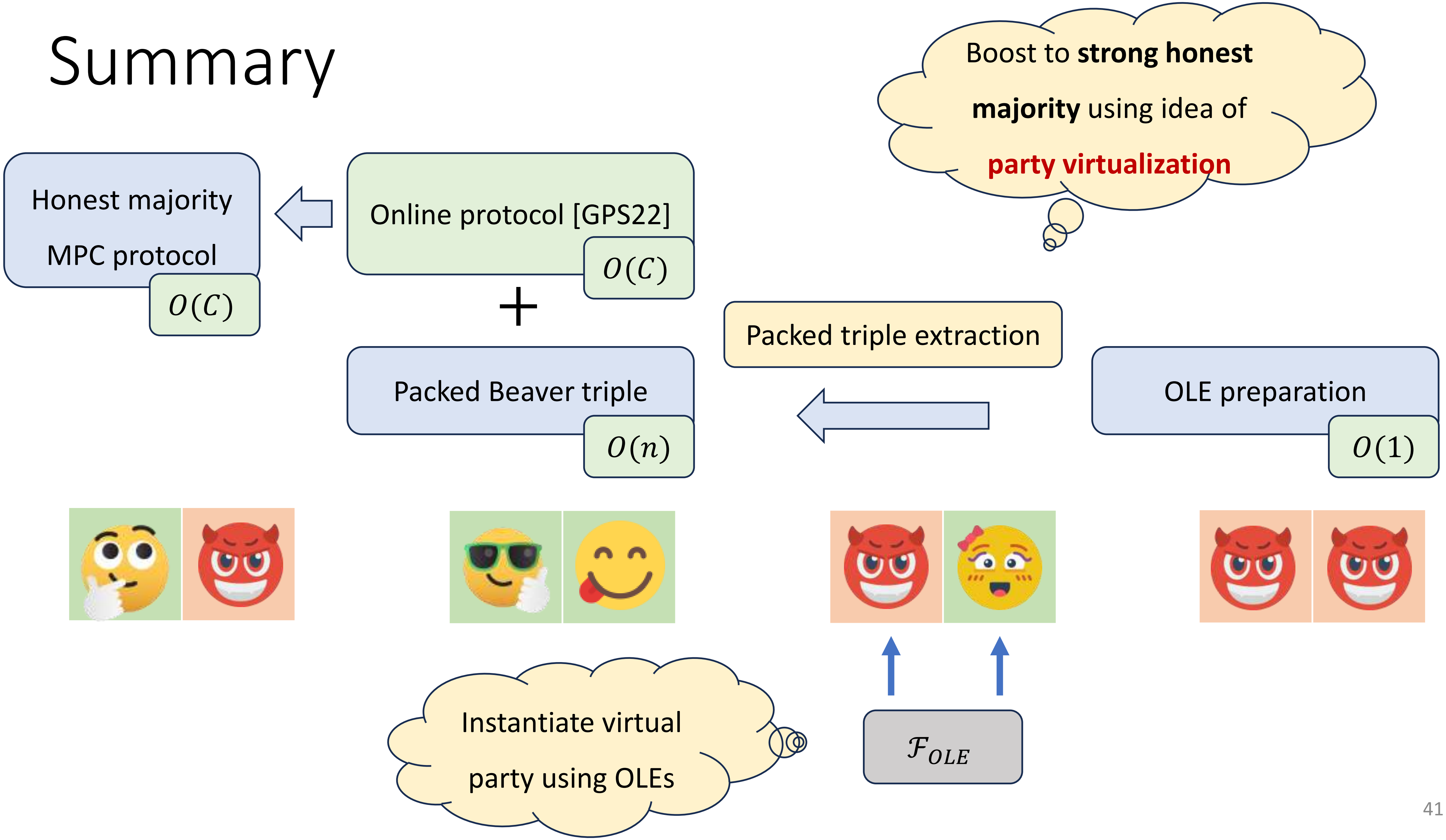
Packed triple extraction

OLE preparation $O(1)$

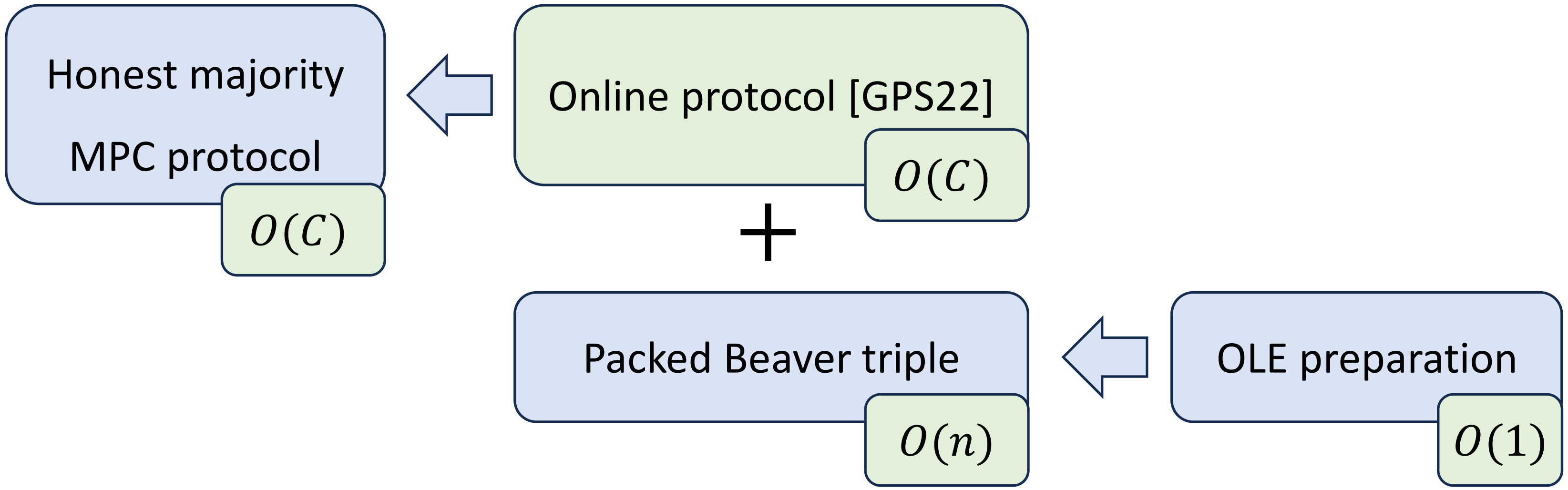




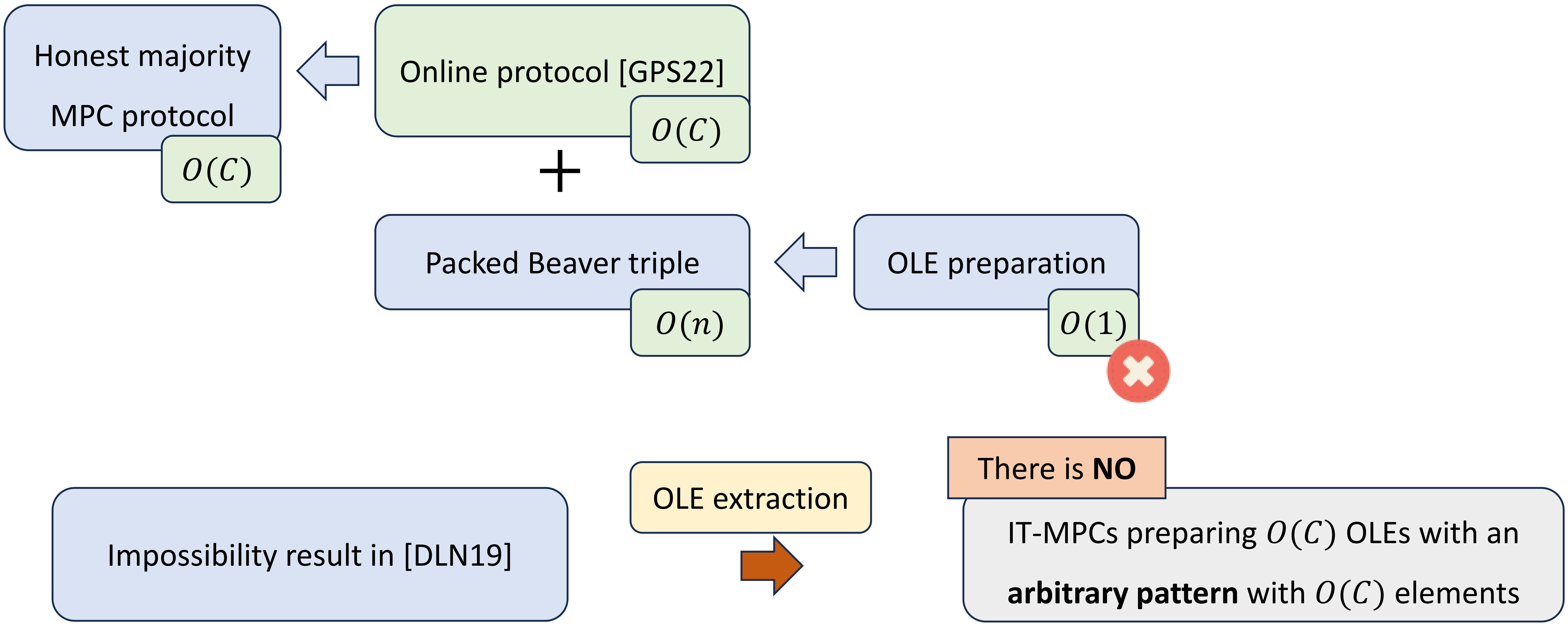
Summary


Boost to **strong honest majority** using idea of **party virtualization**
Honest majority MPC protocol
$O(C)$
Online protocol [GPS22]
$O(C)$
+
Packed triple extraction
Packed Beaver triple
$O(n)$
OLE preparation
$O(1)$
Instantiate virtual party using OLEs
$\mathcal{F}_{OLE}$

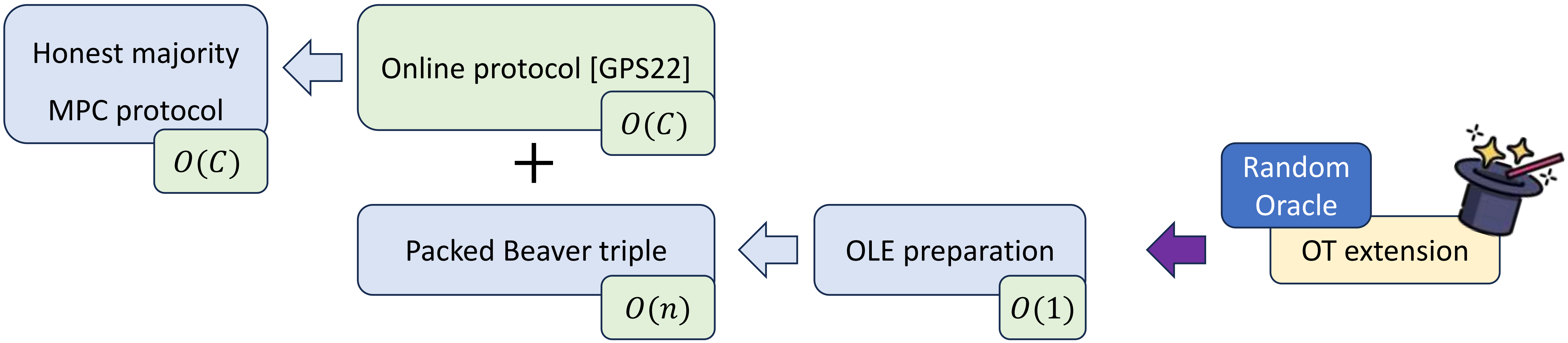
Summary


Honest majority
MPC protocol
$O(C)$
Online protocol [GPS22]
$O(C)$
+
Packed Beaver triple
$O(n)$
OLE preparation
$O(1)$

Summary

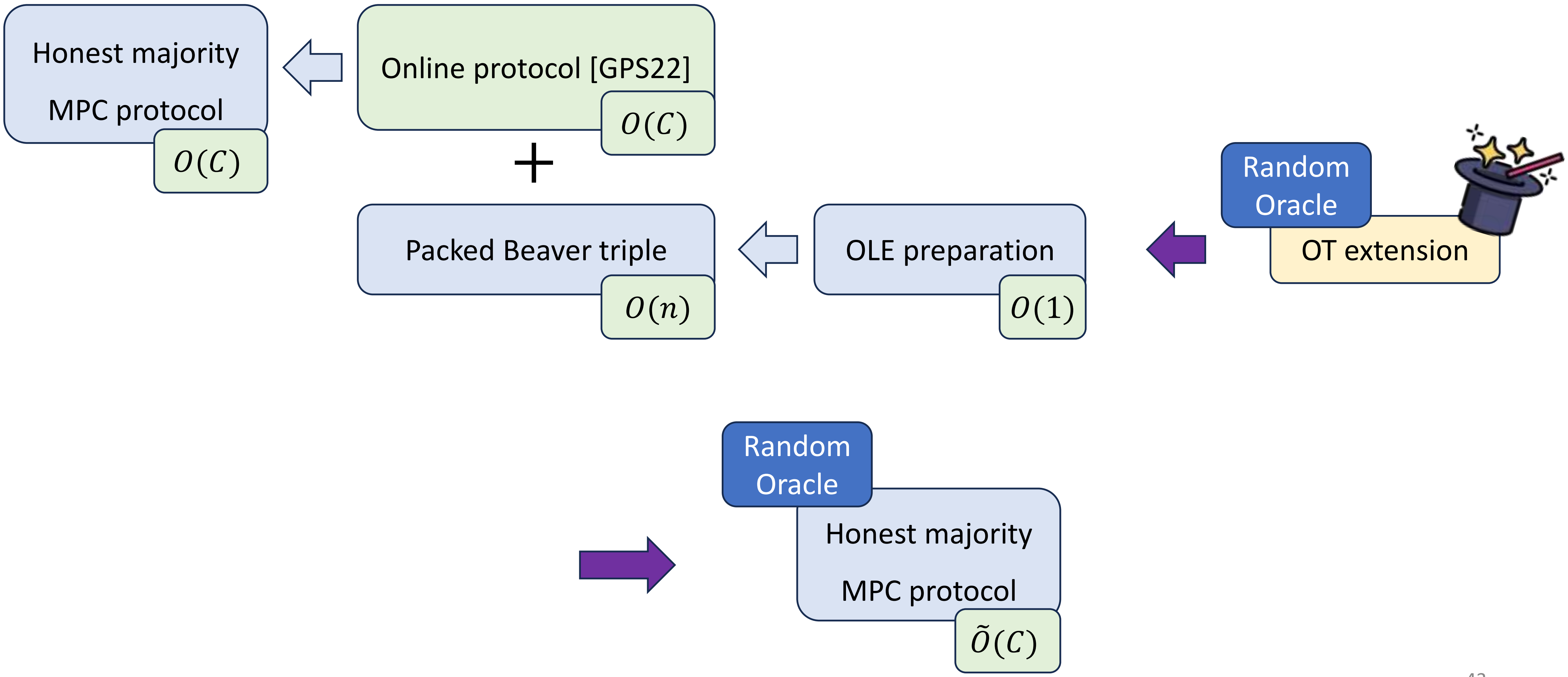

Honest majority MPC protocol
$O(C)$
Online protocol [GPS22]
$O(C)$
+
Packed Beaver triple
$O(n)$
OLE preparation
$O(1)$
Impossibility result in [DLN19]
OLE extraction
There is **NO**
IT-MPCs preparing $O(C)$ OLEs with an **arbitrary pattern** with $O(C)$ elements

Summary



Honest majority MPC protocol
$O(C)$
Online protocol [GPS22]
$O(C)$
+
Packed Beaver triple
$O(n)$
OLE preparation
$O(1)$
Random Oracle
OT extension

Summary


Honest majority MPC protocol
$O(C)$
Online protocol [GPS22]
$O(C)$
+
Packed Beaver triple
$O(n)$
OLE preparation
$O(1)$
Random Oracle
OT extension
Random Oracle
Honest majority MPC protocol
$\tilde{O}(C)$

Thank you!

Credit:
Icons: https://www.flaticon.com/