Card-based Cryptographic Protocols with the Minimum Number of Rounds Using Private Operations

Hibiki Ono ・Yoshifumi Manabe

Kogakuin University, Japan

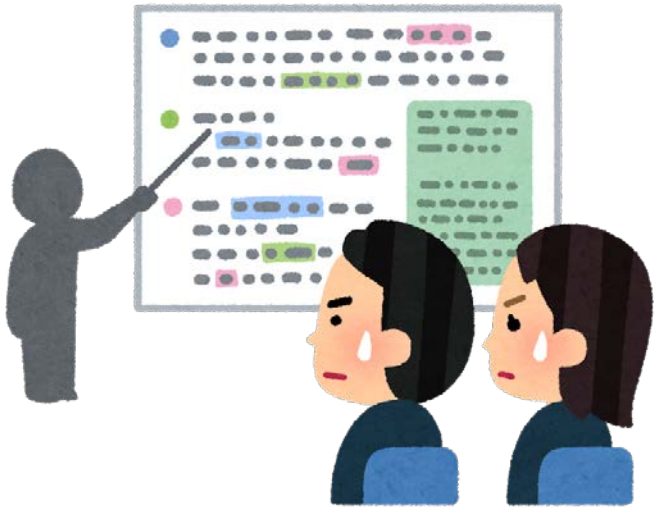
工学院大学 KOGAKUIN UNIVERSITY

Outline of Talk

- Card-based cryptographic protocols
- Private operations
- Round complexity
- Protocols with the minimum number of rounds
- Conclusion

Cryptographic protocols using computers

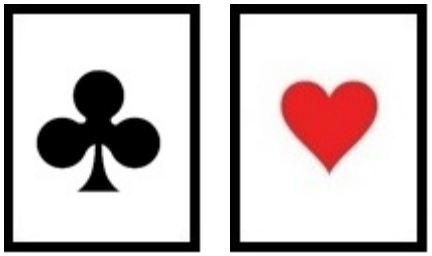
- Difficult to understand the correctness for non-experts



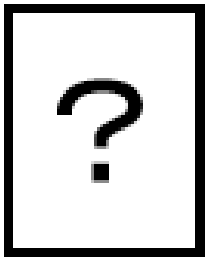
- Cannot be executed when we cannot use computers

Card-based cryptographic protocols

- Using cards



Back of the cards



- Can be executed when we cannot use computers
- Relatively easy to understand the protocol
  →can be used for teaching cryptography

History of card-based cryptographic protocols

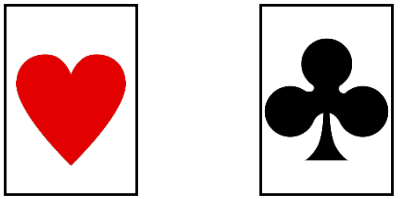
- den Boer(Eurocrypt 89) 'five card trick': logical AND of two inputs using 5 cards
- Two kinds of protocols
  - Primitives to calculate any logical functions
    - AND, XOR, copy of inputs
  - Efficient protocols to solve specific problems
    - Voting/Random permutation/Grouping/etc
- Executed by semi-honest multiple players
  - Private values must not be known to the players

Definition of security

- For $\forall$ player p, $\forall$ input x,

$VIEW_p(x = 1) \approx VIEW_p(x = 0)$

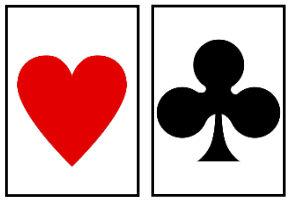
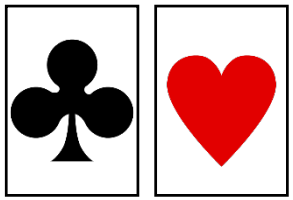
Cards used by the protocols



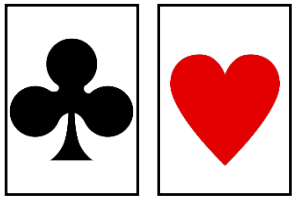
Back is 

indistinguishable

- Encoding of one bit data

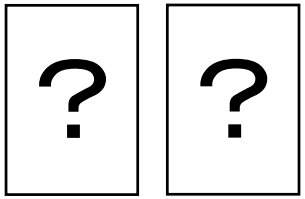
 $=1$  $=0$

- Data and its commitment



x

Face down

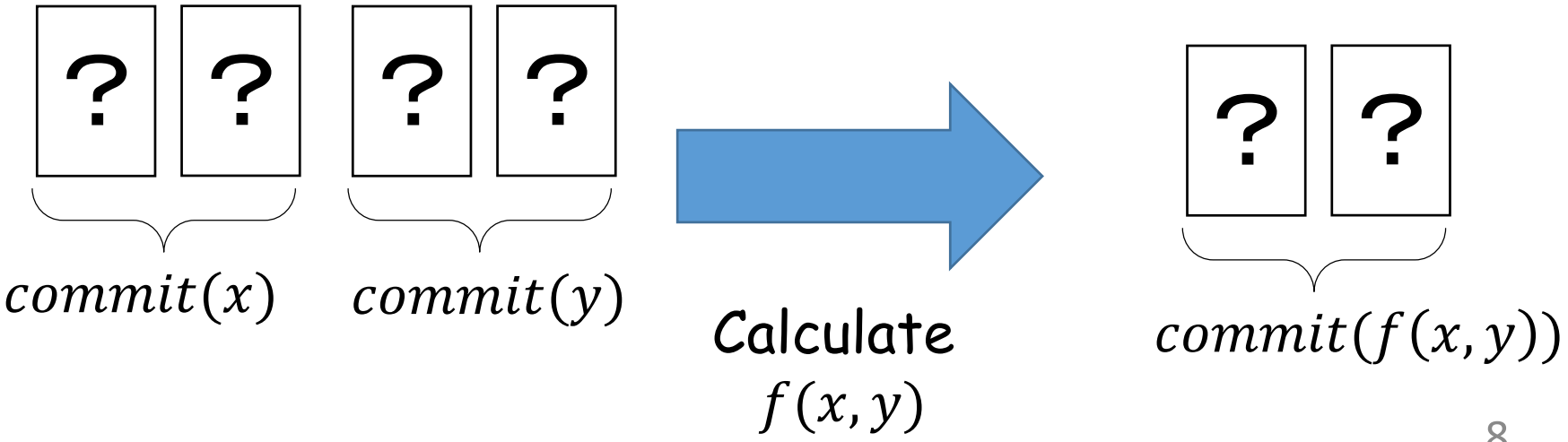


$commit(x)$

Requirements for inputs and outputs

- Input: protocols must allow committed inputs
  - Players must be able to calculate using unknown values
- Output: protocols must output committed results
  - Can be used as inputs to further computation

  (den Boer's "five card trick" is not committed-output)

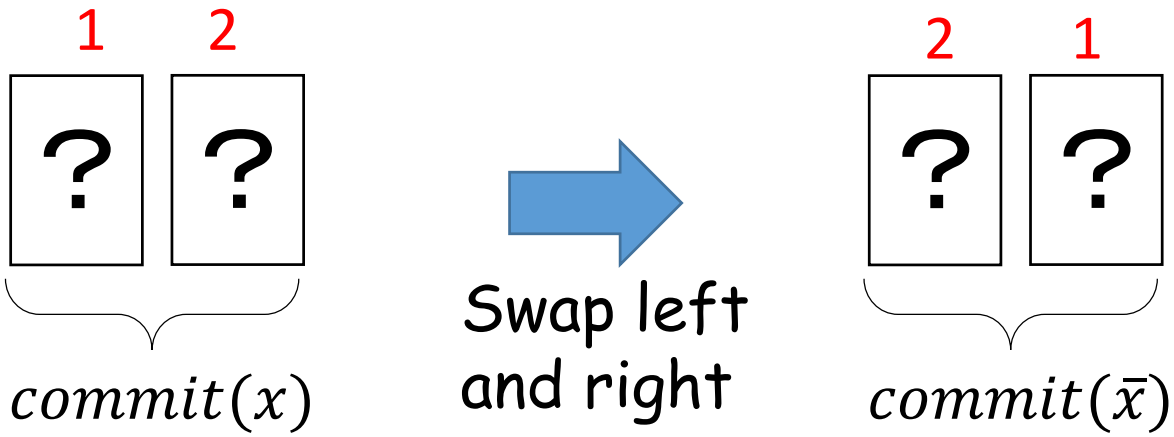


Protocols using private operations

- Operations executed in a place where the other players cannot see
  - Under the table, In the back, etc.
- [NTMIO16] used to solve millionaires' problem
- [OM18] achieved minimum number of cards(4) for AND and copy
  - Impossible with 4 cards by the standard model without private operations

Primitive executable by anyone

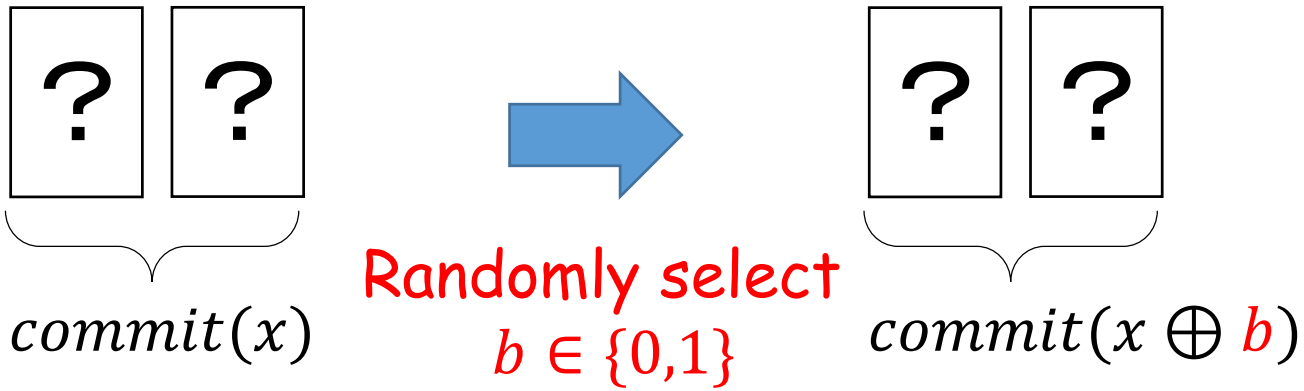
- $commit(x) \rightarrow commit(\bar{x})$



Private operation(1)

- Private random bisection cut: randomly

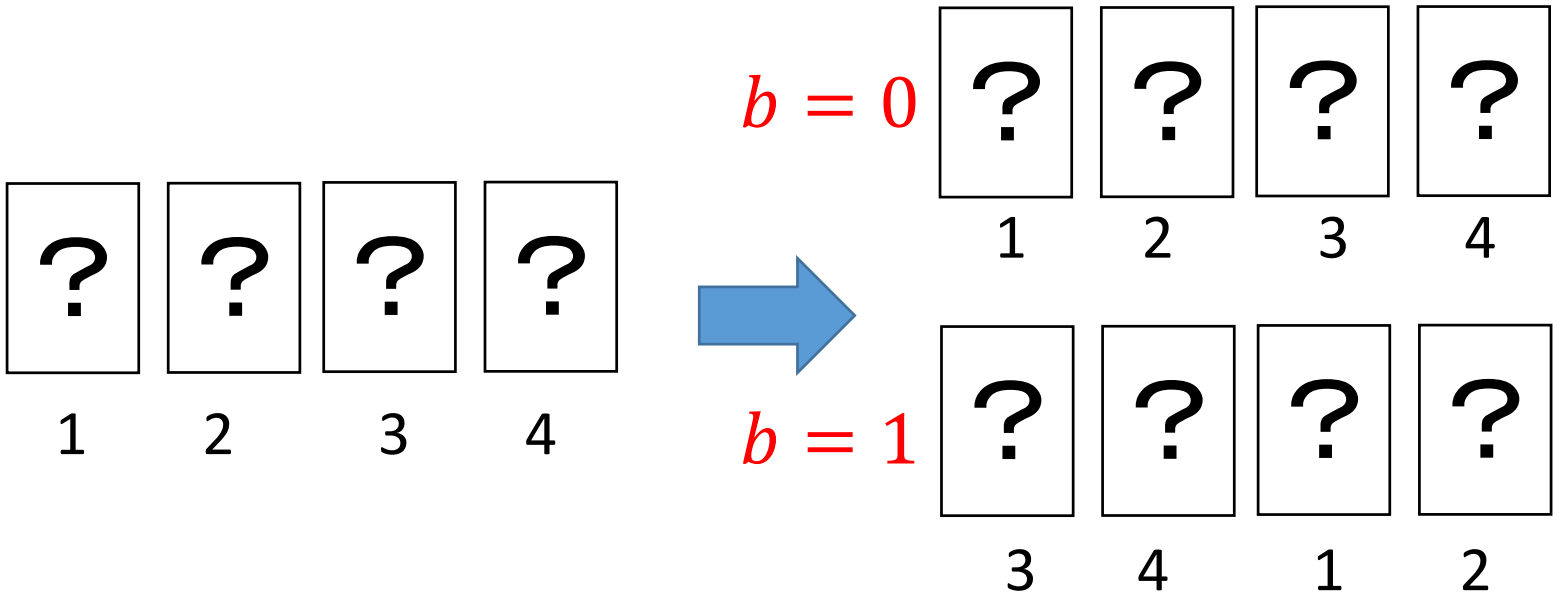
(1) Swap left and right or (2) Do nothing in a hidden place



- Remember b. Do not disclose to any other player

Private operation(2)

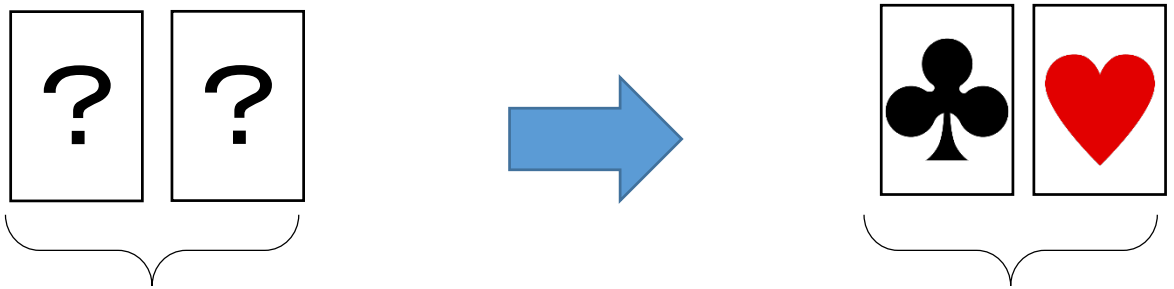
- Private reverse cut : swaps left and right of an even sequence using given b



- (Private reverse selection :
  select left if $b = 0$, right if $b = 1$)

Private operation(3)

- Private reveal: open cards and read the value

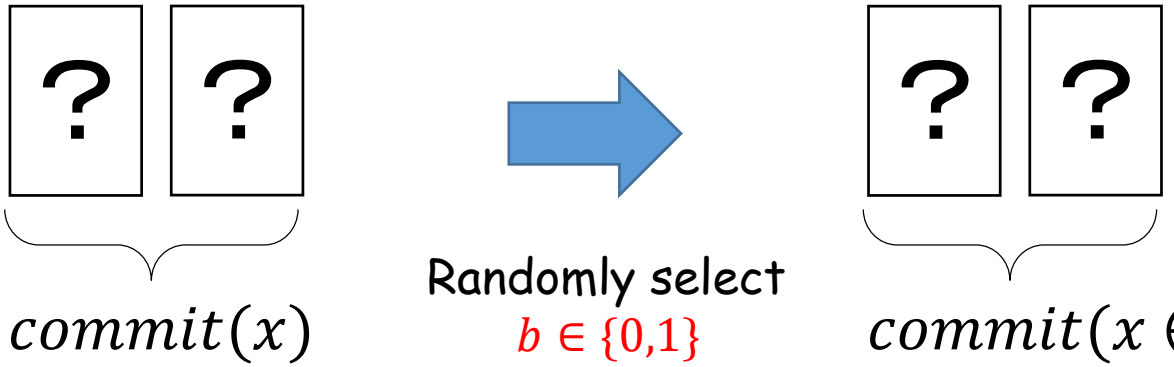


Executed by a player who does not know b
$\rightarrow$No information about x

AND protocol[OM18]

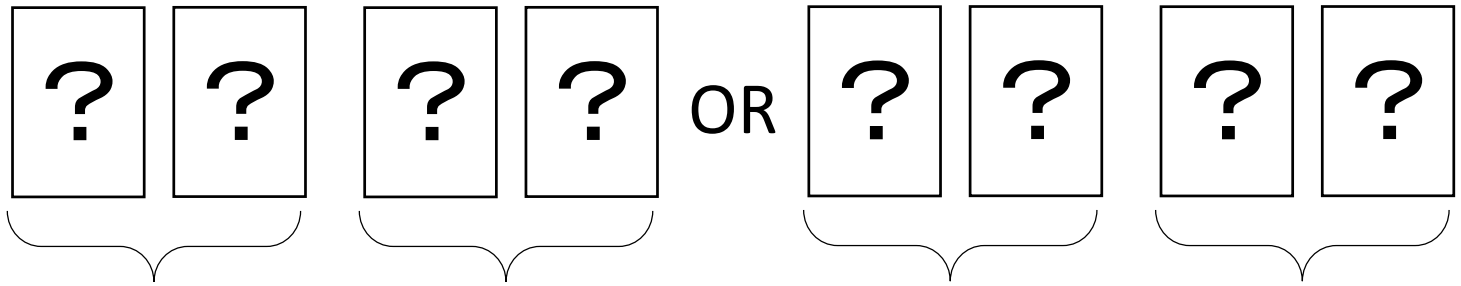
- Input: $commit(x), commit(y)$, Output: $commit(x \wedge y)$

(1)Alice:
Private random bisection cut on x



$commit(x)$ $commit(x \oplus b)$

(2)Bob:
Private reveal $x \oplus b$
→Set cards



$commit(y)$ $commit(0)$ $commit(0)$ $commit(y)$

$x \oplus b = 1$ $x \oplus b = 0$

(3)Alice:
Private reverse selection
left pair if $b = 0$
right pair if $b = 1$

# of cards:4 re-use $commit(x \oplus b)$ to set $commit(0)$

Complexity

- Number of cards : Space complexity
- Number of rounds: Time complexity of the protocols with private operations

- What is the number of rounds?

Number of rounds

- (Roughly speaking) The number of times of handing cards between players
- Used for distributed algorithms

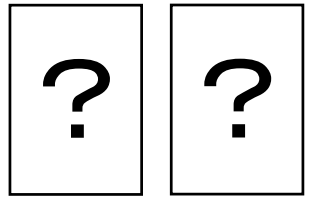
(handing cards : message passing)

- Why number of rounds?
  - Each operation is simple →Dominating time is handing cards between players

AND protocol[OM18]

- Input: $commit(x), commit(y)$, Output: $commit(x \wedge y)$

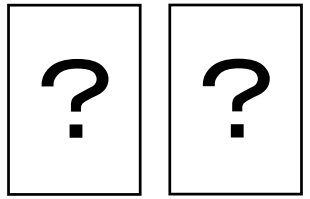
(1)Alice:
Private random
bisection cut on x



$commit(x)$

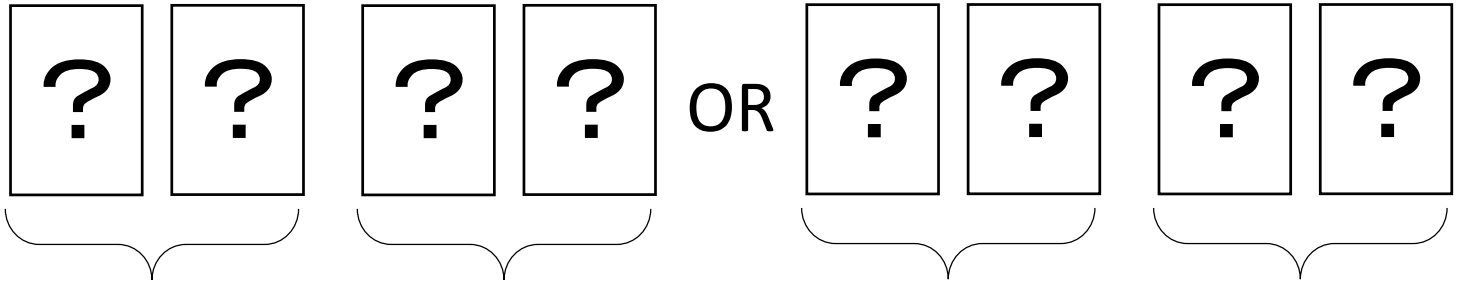
Hand from Alice to Bob

Randomly select
$b \in \{0,1\}$



$commit(x \oplus b)$

(2)Bob:
Private reveal
$x \oplus b$
$\rightarrow$Set cards



$commit(y)$ $commit(0)$ $commit(0)$ $commit(y)$

Hand from Bob to Alice

$x \oplus b = 1$ $x \oplus b = 0$

(3)Alice:
Private reverse selection
left pair if $b = 0$
right pair if $b = 1$

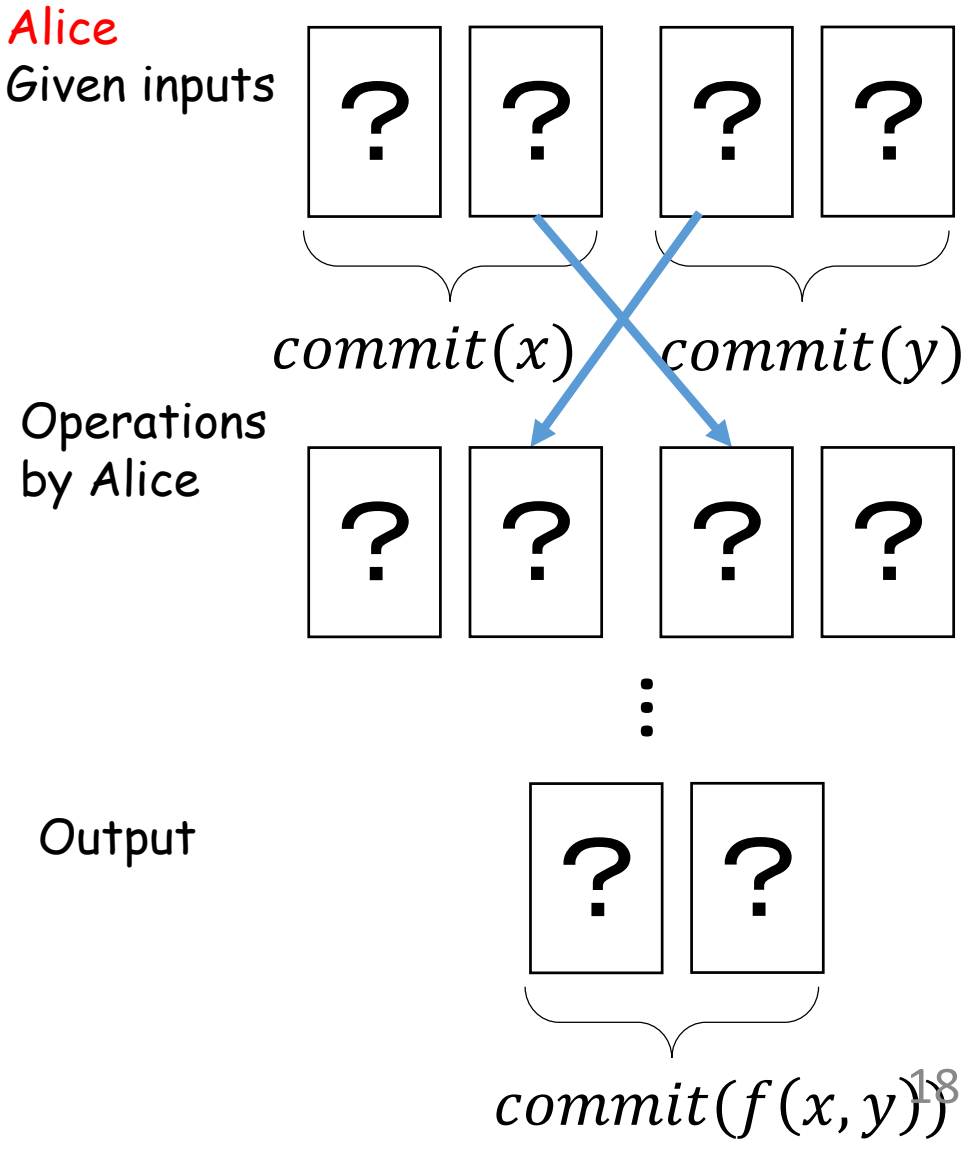
# of rounds : 3

Minimum number of rounds=2

- If one round protocol exists:

All operations are done by Alice
→Alice knows the relation between inputs and outputs

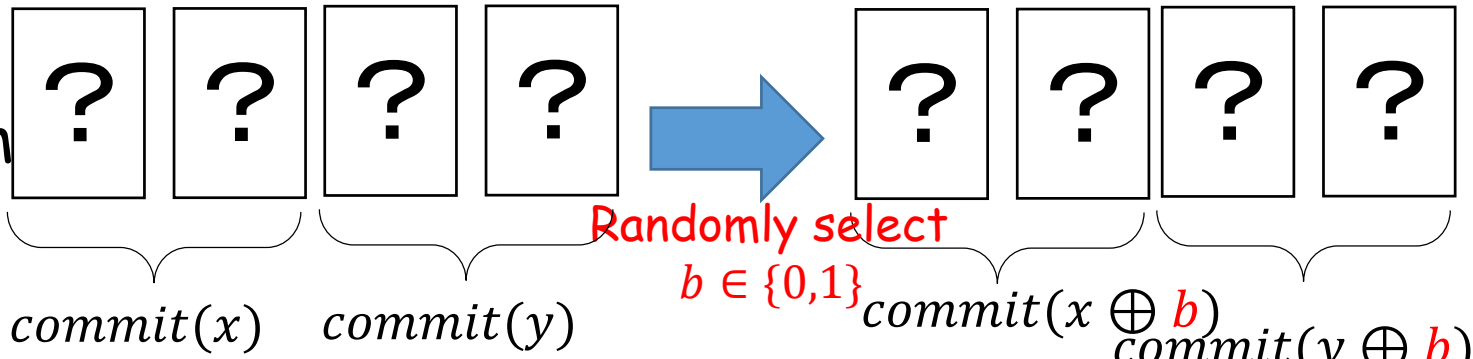
When the output is opened, Alice knows the secret inputs



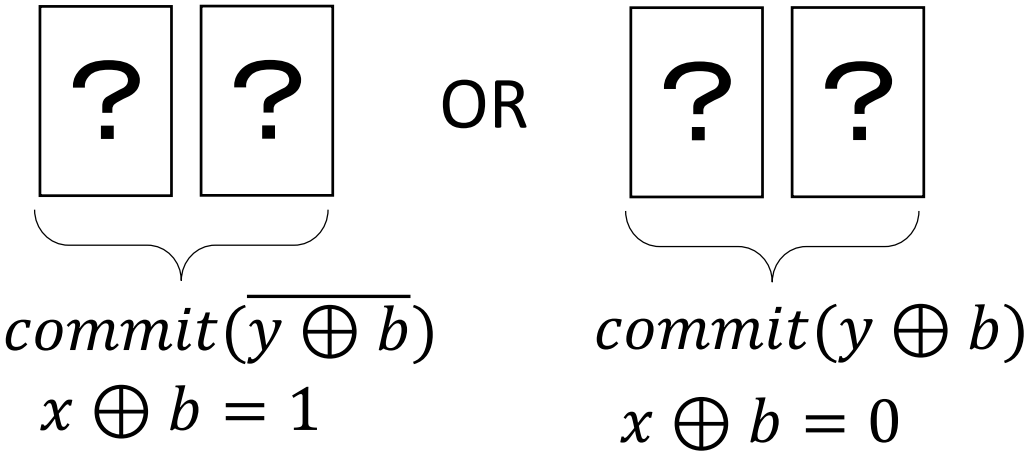
2 round XOR protocol

- Input : $commit(x), commit(y)$, Output : $commit(x \oplus y)$

(1) Alice:
Private random bisection cut on x, y



(2) Bob:
Private reveal $x \oplus b$
Swap $y \oplus b$ according to the value



Correctness:
Outut is $(y \oplus b) \oplus (x \oplus b) = y \oplus x$

Comparison of XOR protocols

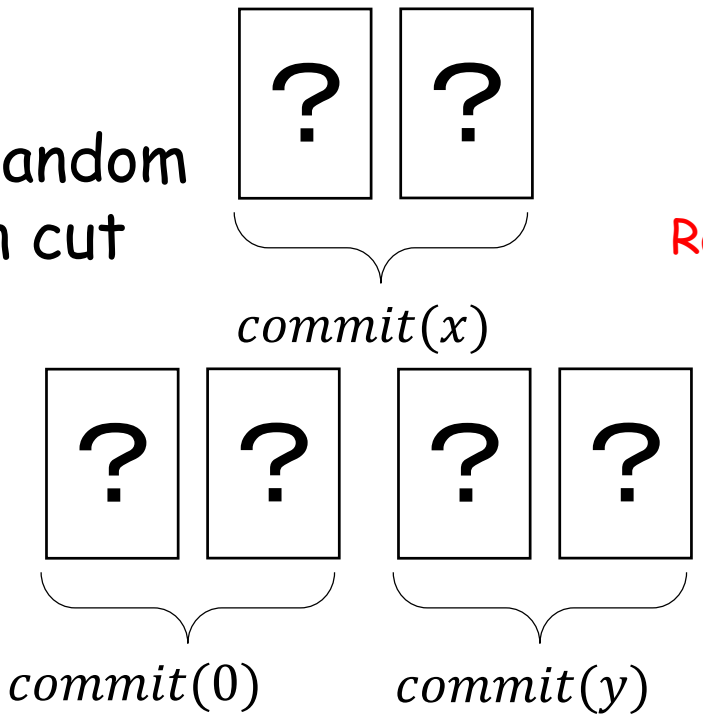
	# of rounds	# of cards
[OM18]	3	4
This paper	2	4
Modified [MS09]	2	4

[MS09] uses one public shuffle and does not use private operations
One public shuffle can be realized by two private random bisection cuts

2 round AND protocol

- Input: $commit(x), commit(y)$, Output: $commit(x \wedge y)$

(1)Alice:
Private random bisection cut on $x, 0||y$



(2)Bob:
Private reveal $x \oplus b$
$0 \rightarrow$ left two cards
$1 \rightarrow$ right two cards

$commit(y)$ If $b = 0$

$commit(0)$ If $b = 1$

$x \oplus b = 1$

OR

$commit(0)$ If $b = 0$

$commit(y)$ If $b = 1$

$x \oplus b = 0$

Correctness of AND protocol

- Input: $commit(x), commit(y)$, Output: $commit(x \wedge y)$

$$x \wedge y = \begin{cases} y & if \quad x = 1 \\ 0 & if \quad x = 0 \end{cases}$$

Output $commit(y)$: $(x \oplus b = 1$ and $b = 0)$ or $(x \oplus b = 0$ and $b = 1)$ $\Leftrightarrow$ $x = 1$

Output $commit(0)$: $(x \oplus b = 1$ and $b = 1)$ or $(x \oplus b = 0$ and $b = 0)$ $\Leftrightarrow$ $x = 0$

Bob:

Private reveal $x \oplus b$

0 → left two cards

1→ right two cards

$commit(y)$ If $b = 0$

$commit(0)$ If $b = 1$

$x \oplus b = 1$

OR

$commit(0)$ If $b = 0$

$commit(y)$ If $b = 1$

$x \oplus b = 0$

Comparison of AND protocols

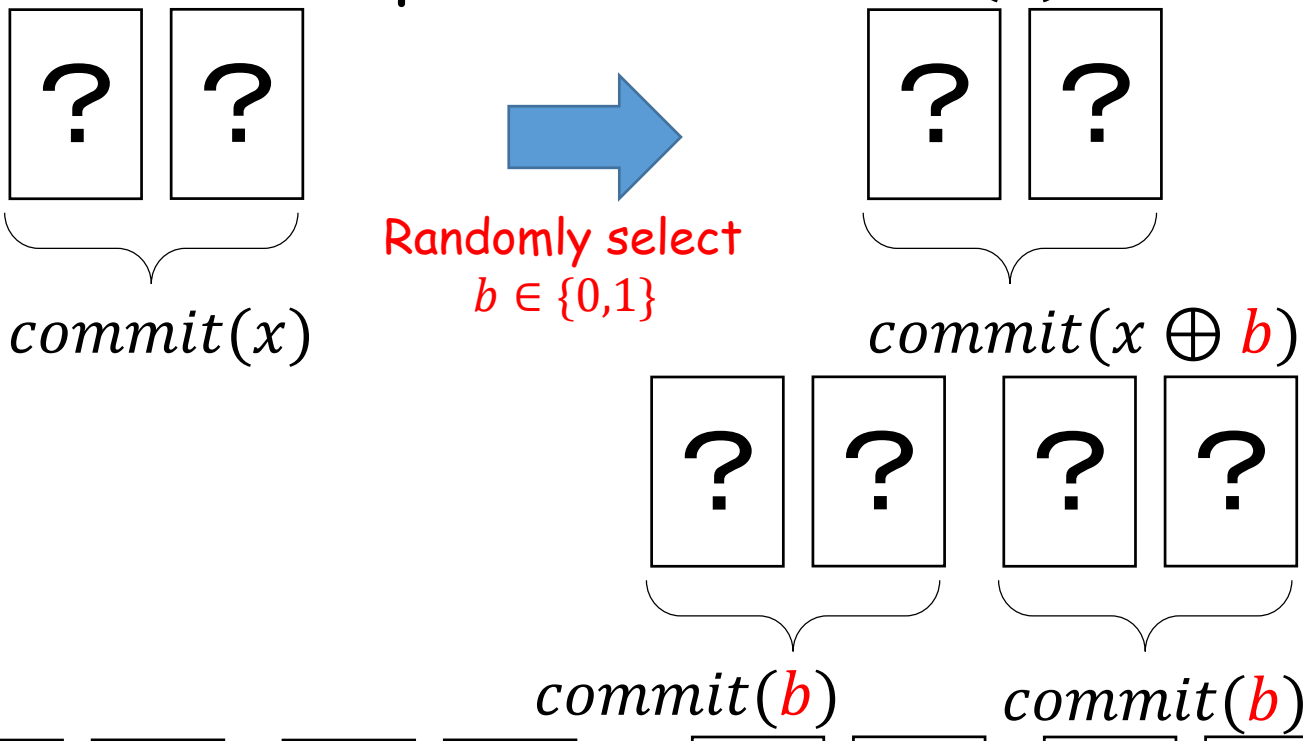
	# of rounds	# of cards
[OM18]	3	4
This paper	2	6
Modified [MS09]	2	6

[MS09] uses one public shuffle and does not use private operations
One public shuffle can be realized by two private random bisection cuts

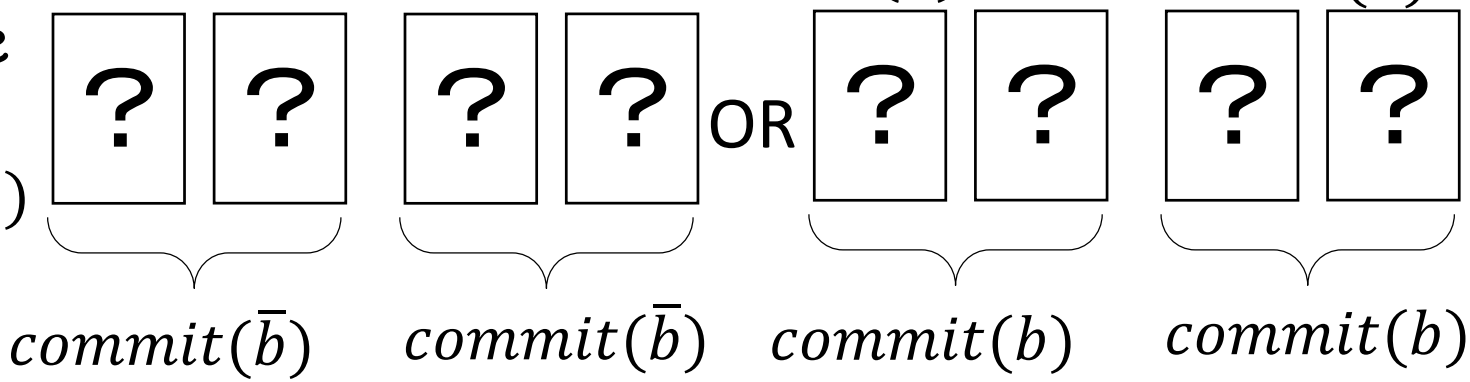
2 round COPY protocol

- Input: $commit(x)$, Output: two $commit(x)$

(1)Alice:
Private random bisection cut , two $commit(b)$



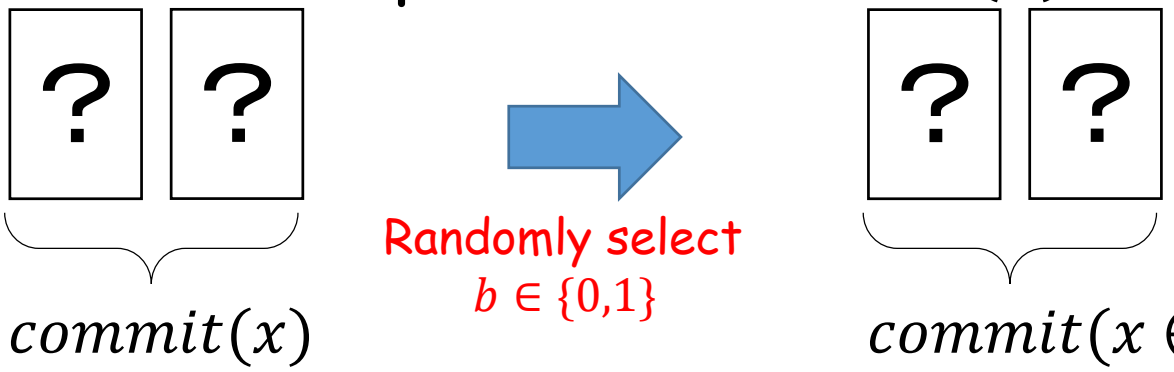
(2)Bob: Private reveal $x \oplus b$, swap $commit(b)$ if the value=1



2 round COPY protocol: correctness

- Input: $commit(x)$, Output: two $commit(x)$

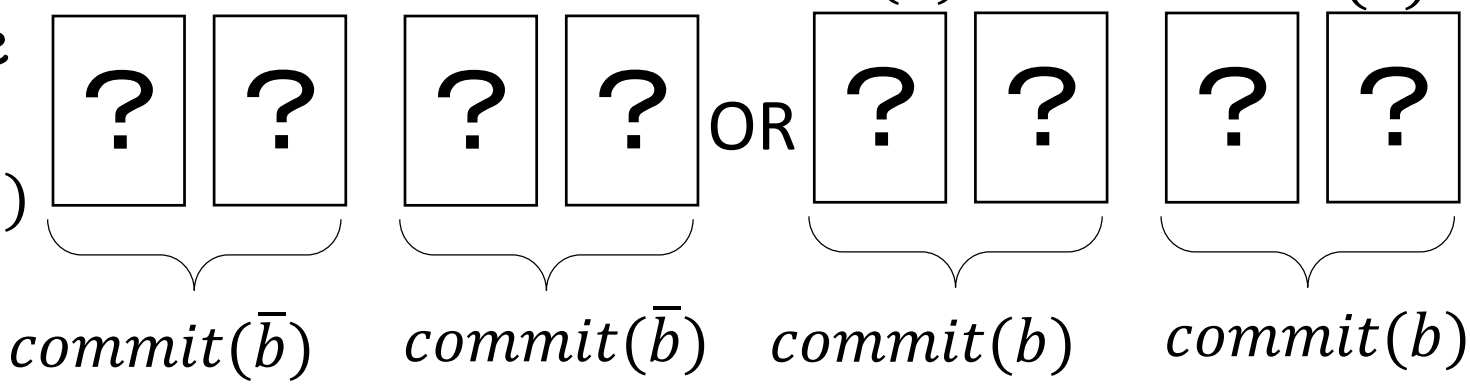
(1)Alice:
Private random bisection cut , two $commit(b)$



Output is:

$$b \oplus (x \oplus b) = x$$

(2)Bob: Private reveal $x \oplus b$, swap $commit(b)$ if the value=1



$x \oplus b = 1$ $x \oplus b = 0$

Comparison of copy protocols

	# of rounds	# of cards
[OM18]	3	4
This paper	2	6
Modified [MS09]	2	6

[MS09] uses one public shuffle and does not use private operations
One public shuffle can be realized by two private random bisection cuts

Question: protocol with the other set of private primitives?

- Private random bisection cut: swap using randomly selected bit b
- Private reverse cut: swap using given bit b
- Private reveal: open cards

- Private reveal has a problem
  - Mistake/cheat in opening cards → private value is known
- No private reveal →protection (e.g. put cards in envelops) is possible

Comparison of AND protocols

	# of rounds	# of cards
[OM18]	3	4
This paper	2	6
Modified [MS09]	2	6
w/o private reveal	~~3~~2	~~4~~6

Comparison of XOR protocols

	# of rounds	# of cards
[OM18]	3	4
This paper	2	4
Modified [MS09]	2	4
Without private reveal	2	4

Comparison of copy protocols

	# of rounds	# of cards
[OM18]	3	4
This paper	2	6
Modified [MS09]	2	6
w/o private reveal	2	6

Other results (shown in the proceedings)

- Any two-variable logical function
  - Same idea as AND protocol
- Any n-variable logical function
  - $O(2^n)$ cards, 2 rounds
- Protocol that preserves an input
  - Output $commit(x \wedge y)$ and $commit(y)$
  - 6 cards, 3 rounds
  - 4 cards, 5 rounds w/o private reveal

Conclusion

- Time complexity of card-based cryptographic protocols using private operations
- # of rounds is the measure of time complexity

Further study

- Trade-off between # of cards and # of rounds for AND and copy?
- # of rounds for the other problems