Games and the Impossibility of Realizable Ideal Functionality

Anupam Datta Ante Derek John C. MitchelL
Ajith Ramanathan Andre Scedrov

Background

- Games [GM84]:
  - Defines specific moves for each player and properties that need to hold
  - Not composable
  - Examples: IND-CPA, IND-CCA for encryption
- Functionalities [Can01, PW01]:
  - Simulation relation between real protocol and ideal functionality, which is “secure by construction”
  - Composable (main advantage)
  - Example: Secure channel using trusted party
- Goal: Investigate relationships between the two specification methods

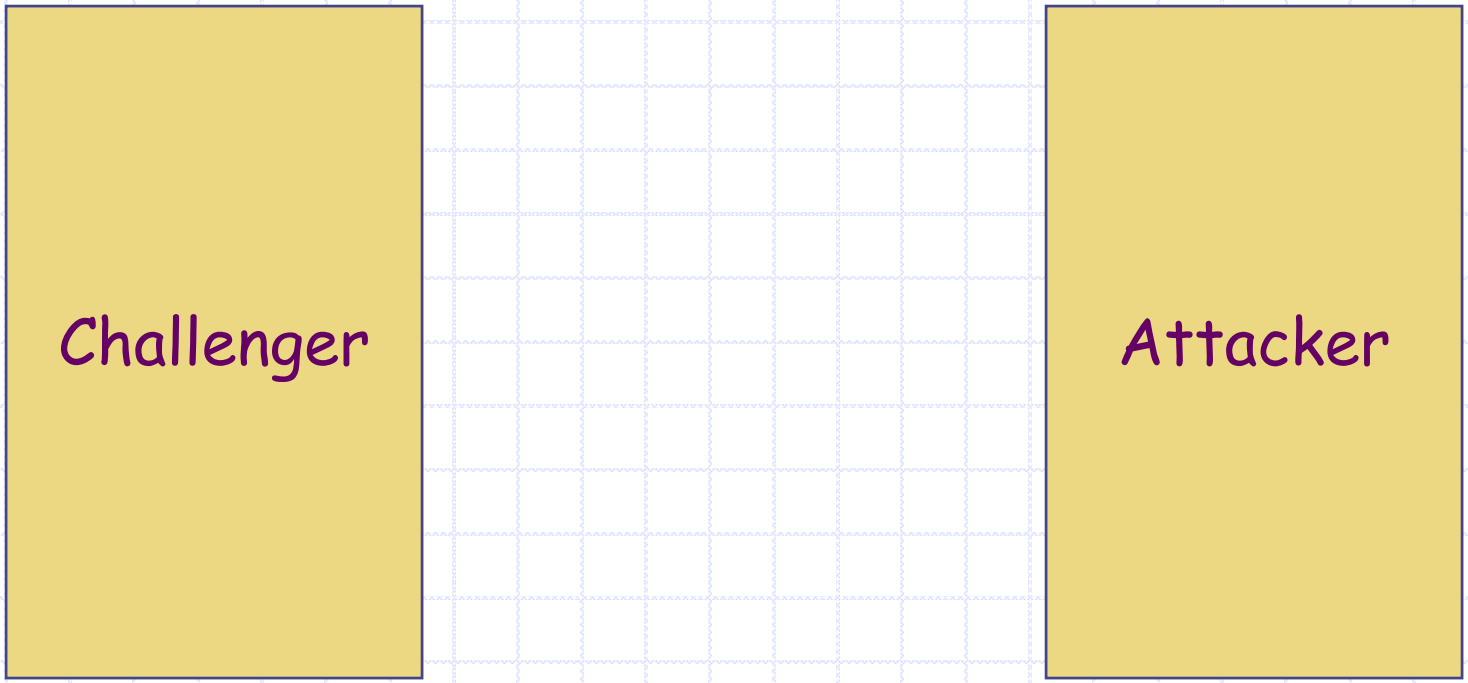
Contributions

- Formalize the connection between two notions
  - For a primitive P specified by games we propose a definition of an ideal functionality *for P*
- Impossibility theorem for bit-commitment
  - Motivated by [CF2001]
  - No ideal functionality for bit-commitment can be realizable (plain model)
- Generalizations
  - Variants of symmetric encryption and group signatures
  - Handle setup assumptions (work in progress)

Game examples: encryption

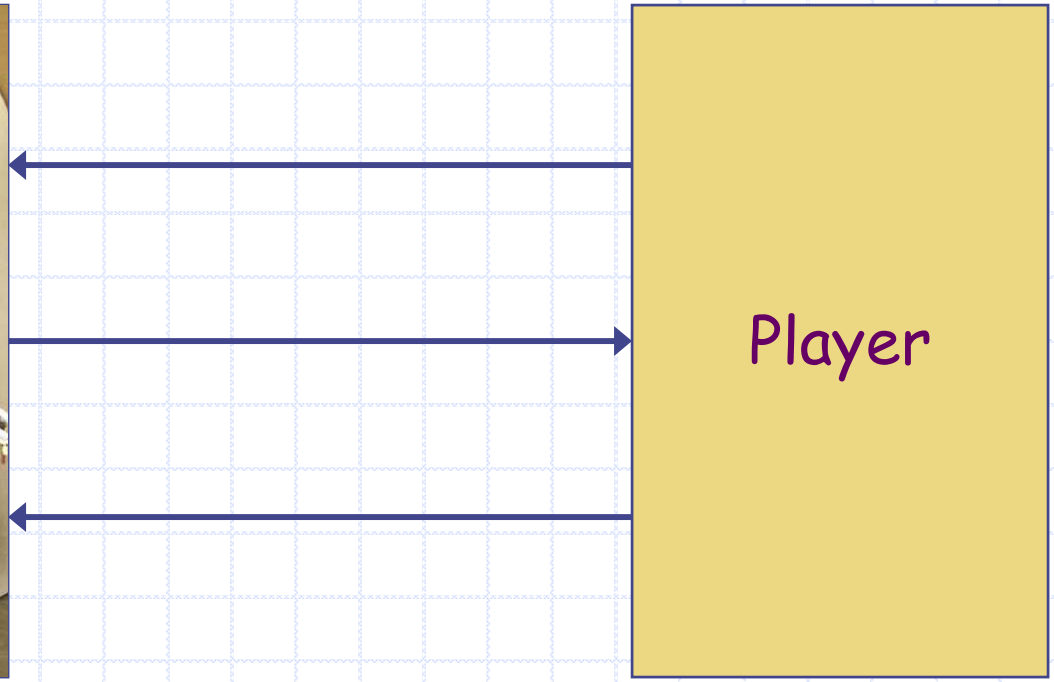
- Passive adversary
  - Semantic security
- Chosen ciphertext attacks (CCA1)
  - Adversary can experiment with decryption before receiving a challenge ciphertext
- Chosen ciphertext attacks (CCA2)
  - Adversary can experiment with decryption before *and after* receiving a challenge ciphertext

Game Format



Game Format


1 COIN WINNERS
2 COIN WINNERS
800
1600
BAR
BAR
BAR
INSERT BILLS BELOW
THIS MACHINE ACCEPTS
$1 $5 $10 $20 $50 $100 BILLS
DOUBLE DOLLARS


Player

Passive Adversary

m_0, m_1

$E(m_i)$

guess 0 or 1

Player

Chosen ciphertext CCA1

c

D(c)

m_0, m_1

$E(m_i)$

guess 0 or 1

Player

Chosen ciphertext CCA2

c

D(c)

m_0, m_1

$c = E(m_i)$

Player

$c' \neq c$

D(c')

guess 0 or 1

Games

- Defines security properties
  - Specific moves for each player
  - Properties that need to hold
- Very flexible
- Some disadvantages
  - Not composable

Ideal Functionalities

- Based on indistinguishability
  - Simulation relation between real protocol and ideal functionality
- Some advantages
  - Composable

Protocol security

Protocol execution

input

P_1 P_2 P_3 P_4

π

A

attacker

output

input

P_1 P_2 P_3 P_4

S

simulator

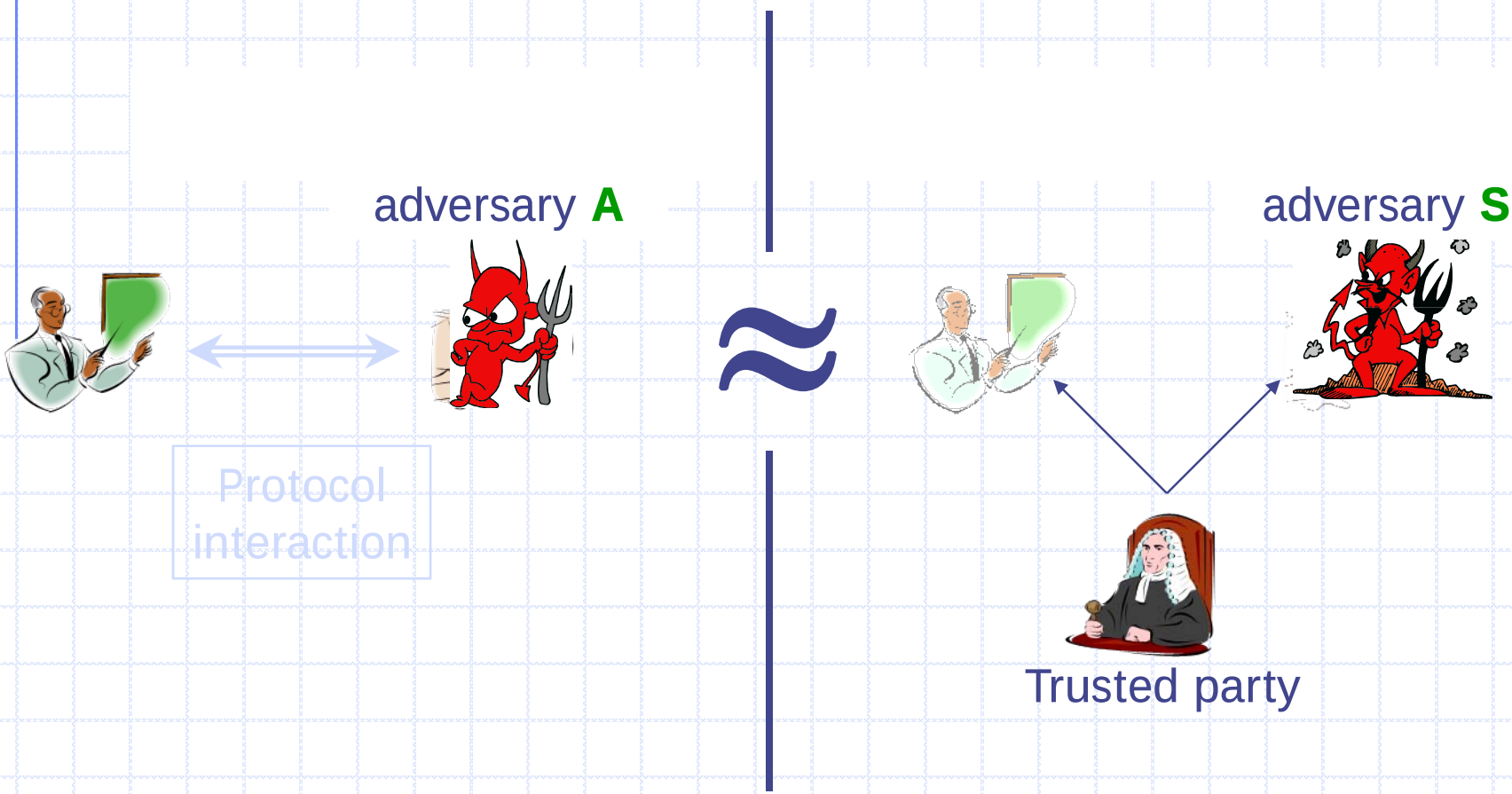
F

Ideal functionality

output

Universal composability

also “reactive simulatability” [BPW], ... see [DKMRS]



Example: Secrecy

- Challenge-response protocol

  $A \rightarrow B \quad \{i\}_k$

  $B \rightarrow A \quad \{i+1\}_k$

- This protocol provides secrecy if indistinguishable from “ideal” protocol

  $A \rightarrow B \quad \{random_1\}_k$

  $B \rightarrow A \quad \{random_2\}_k$

Example: Authentication

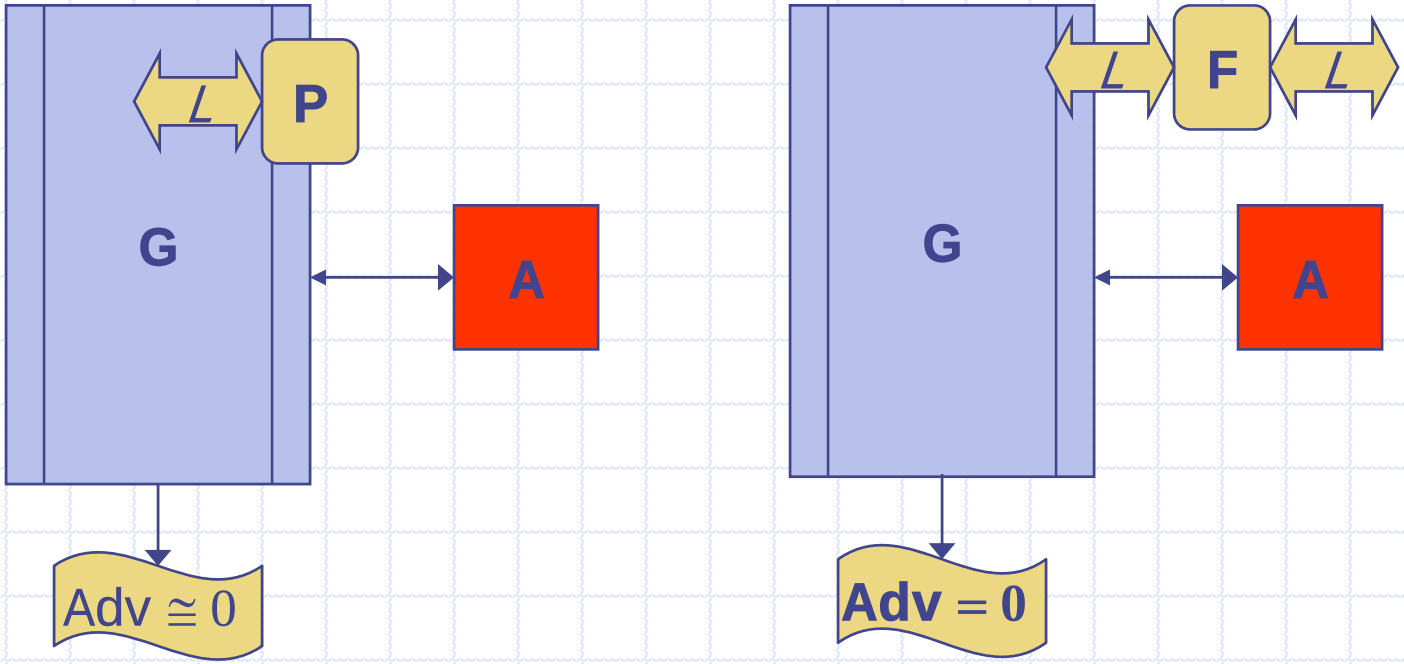
- Authentication protocol
  - $A \rightarrow B$ $\{i\}_k$
  - $B \rightarrow A$ $\{i+1\}_k$
  - $A \rightarrow B$ "Ok" if expected number received from Bob
- Secure if indistinguishable from "ideal" protocol
  - $A \rightarrow B$ $\{random_1\}_k$
  - $B \rightarrow A$ $\{random_2\}_k$
  - $B \rightarrow A$ $random_1$, $random_2$ on a magic secure channel
  - $A \rightarrow B$ "Ok" if numbers on real & magic channels match

What did we do?

- Formalize the connection between two notions
  - For a primitive P specified by games we propose a definition of an ideal functionality *for P*
- Impossibility theorem for bit-commitment
  - Motivated by [CF2001]
  - No ideal functionality for bit-commitment can be realizable (plain model)
- Generalizations
  - Variants of symmetric encryption and group signatures
  - Handle setup assumptions (work in progress)

Intuition: What is Ideal about a Functionality?

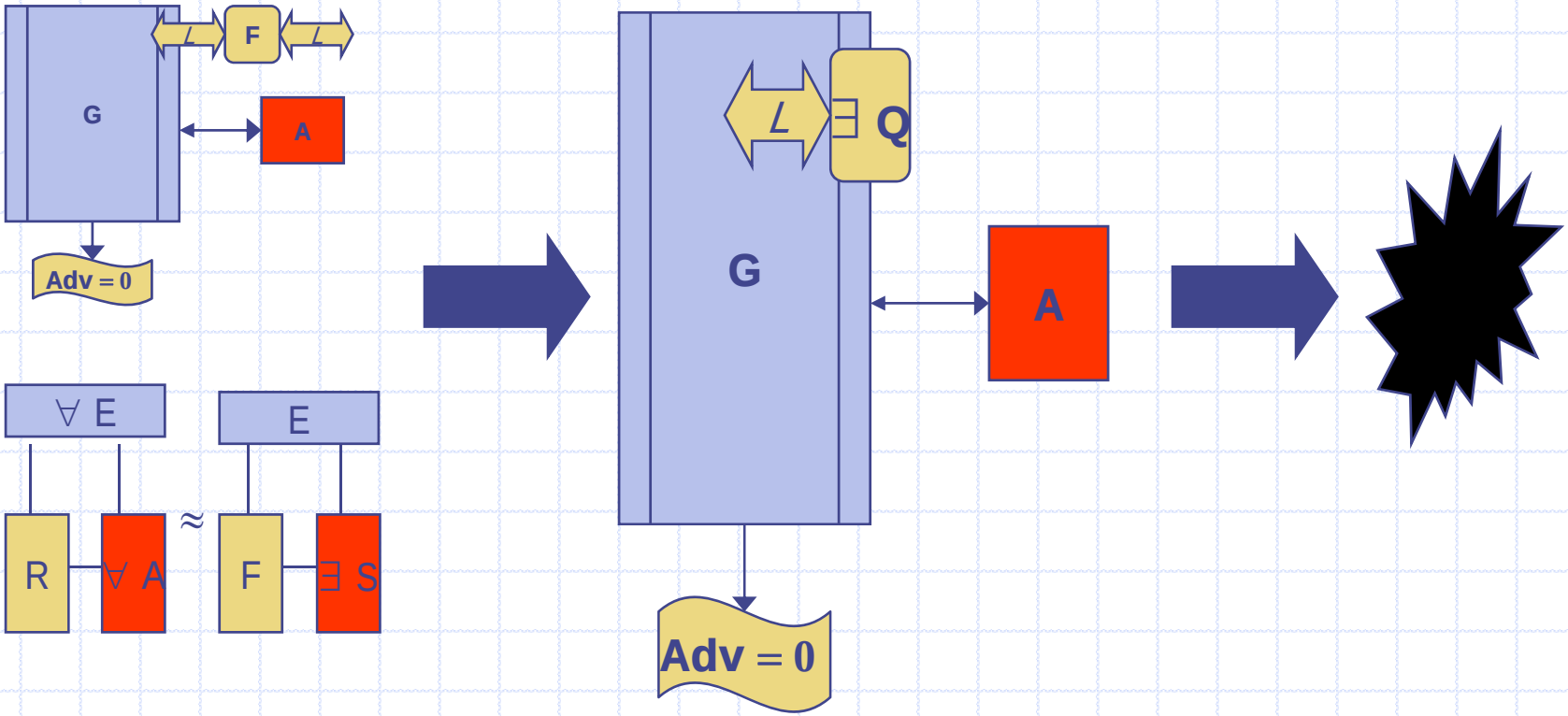
- **P** a primitive, security defined by games



- **F** speaks the same language
- **F** satisfies security requirements perfectly

Intuition: Impossibility results

- For a certain **P** no corresponding **F** is realizable



Bit Commitment

- Commit phase
  - Choose a random bit b
  - Announce some value f(k,b)
    - where k may be random key, etc
- Open the commitment
  - Reveal b and k
  - Since f is publicly known, can verify b
- Analogy
  - Put message in sealed envelope to open later

Example: distributed coin flipping

- Alice
  - Choose random bit a
  - Announces commitment to a
- Bob
  - Choose random bit b
  - Announces commitment to b
- Communication
  - Exchange their bits, compute $a \oplus b$
- Reveal commitment
  - Alice knows that Bob did not change his bit after seeing hers

Subtle issue: what if Bob stops before completing protocol?

Impossibility Theorem

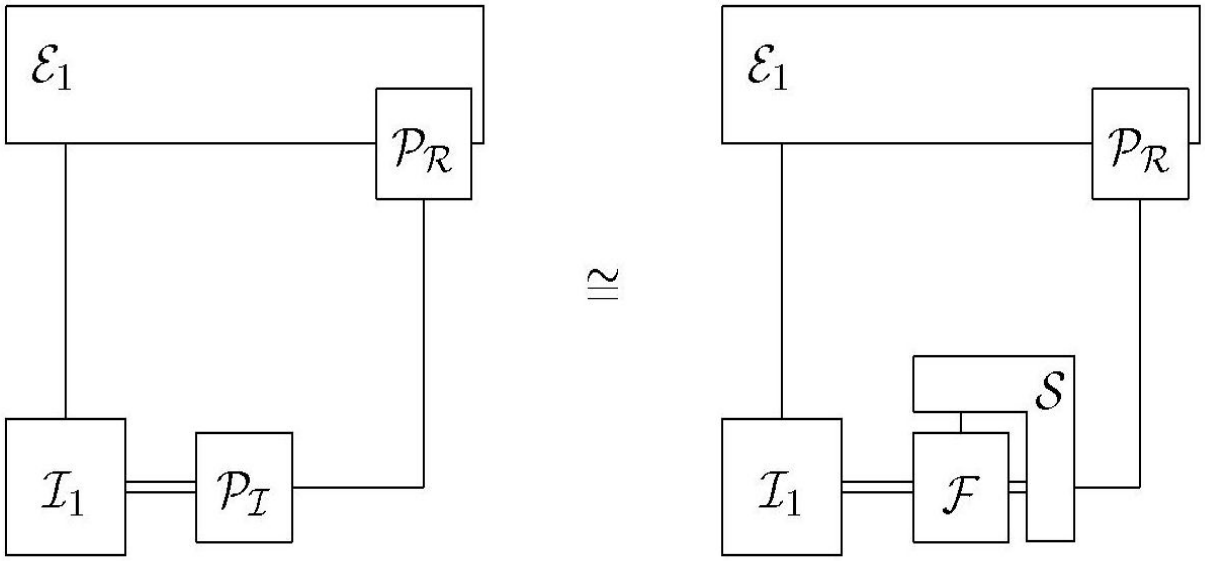
- If F is *any* ideal functionality for bit-commitment, then no real protocol securely realizes F

- Proof idea: Can construct information-theoretically hiding and binding protocol for BC that does not use TTP

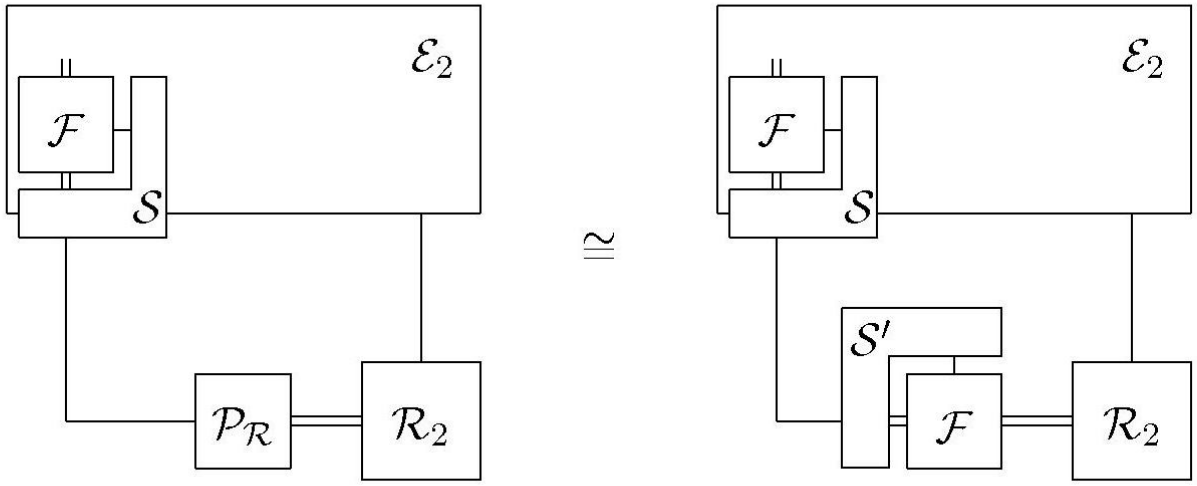
Very simple idea

- Commitment depends on chosen bit
  - It is not possible to do this *perfectly,* i.e. in a way that is indistinguishable to a computationally unbounded attacker
- This is not the proof ...
  - but perhaps this helps

Actual proof: Phase 1



Actual proof: Phase 2



More of the Proof:

- Systems F|S and F|S' together constitute a real implementation for BC that is
  - Info-theoretically binding
  - Info-theoretically hiding
  - Correct
- A contradiction

Other results

- Any property that gives BC cannot be realized
  - Composition theorem
- Variant of Symmetric encryption
  - Semantic security and Ciphertext integrity
- Variant of Group signatures
  - Anonymity and Traceability (strong variant)

Generalizations

- Handle setup assumptions (PKI, Random oracle, CRS)
  - Model setup assumption as a functionality in the hybrid model that only work in the initial phase
  - Similar impossibility results if these functionalities are global
- Proof not specific to bit-commitment
  - Intuition: contradicting game requirements lead to unrealizable functionalities
  - Like to have: a result connecting information-theoretic impossibility of satisfying games with impossibility of a realizable ideal functionality

Related Work

- Bit-commitment
  - [CF2001] Impossibility result in the plain model, constructions using CRS
  - [DN2002] More constructions using CRS
- Impossibility results
  - [Can2001] Coin-tossing, zero knowledge
  - [CKL2003] Multi-party computation
- Models
  - [PS2004] Achieves bit-commitment in plain model
- Other notions of composable security
  - [DDMP2004] Conditional security

Summary

- Formalize the notion of an ideal functionality for a primitive
  - Information theoretic security
- Impossibility theorem for bit-commitment
  - No ideal functionality for bit-commitment can be realizable (plain model)
  - Variants of symmetric encryption and group signatures
- Work in progress
  - Handle setup assumptions
  - Generalizations
- May need an alternative approach to universally compositional security in practice
  - Conditional composability instead of universal composability

Questions?