Limits on the Usefulness of Random Oracles

Iftach Haitner, Eran Omri, Hila Zarosim

TCC 2013

28 May 2013

The Complexity of Cryptography

- Most cryptographic primitives cannot be achieved unconditionally (they require hardness assumptions)
- A fundamental question:
  What are the minimal assumptions for different cryptographic primitives?
    - What are one-way functions (OWF) sufficient for?
- Random Oracle model (parties have access to a truly random function) – implements strongest OWF
    - If a primitive *P* is "implied" by any OWF – then *P* should be achievable in the random oracle model [Impagliazzo-Rudich 89]

The Power of a Random Oracle

- Free randomness
- Implements many cryptographic primitives, e.g., one-way functions and cryptographic hash functions, with extremely strong security, and thus implies what ever these primitive imply
- Can even be used for constructing secure protocols for tasks that are hard to implement in the standard model, and even **completely unachievable**:
  - E.g., the Fiat-Shamir paradigm [Fiat-Shamir 87]
    - Provably secure in the random-oracle model [Pointcheval-Stern 96]
    - An instantiation of it - cannot be proven secure under any "implementation" of a

Malicious vs. Semi-Honest Settings

Malicious setting

- **Helpful** - e.g., commitment schemes, zero-knowledge proofs, coin-tossing (*limited fairness)
  - All trivially obtainable in the semi-honest setting
- **Not helpful** - key-agreement, OT, MPC,… [IR89]

<u>In this work:</u>

What is the exact power of the random-oracle model in the semi-

Our Results

- **Main thm (informal):** Any no-input, m-round semi-honest protocol π in the random oracle model has an (almost) equivalent m-round, (stateless) semi-honest, no-input protocol $\tilde{\pi}$ in the no-oracle model (i.e., information-theoretic model)

Applications:

- An alternative proof for impossibility of key agreement [IR89]
- Impossibility of accurate two-party differentially private for the inner product functionality
- Impossibility of non-trivial no-input (randomized) semi-honest secure-computation

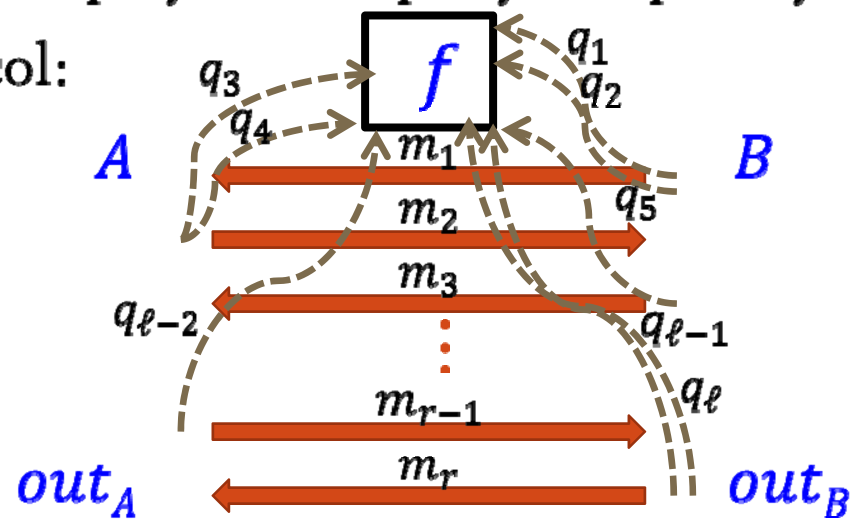
Implication: No black-box reduction to OWF for these primitives

Related Work

- [Impagliazzo-Rudich 89]– no key-agreement protocols in the random-oracle model
  - No **black-box** KA from OWFs
- [Barak-Mahmoody 09] improved query complexity of [IR89]'s attacker – match $O(n^2)$ upper bound of [Merkle 82]
- [Mahmoody-Maji-Prabhakaran 12] – No **deterministic, poly-size domain** semi-honest 2-party SFE in the RO model

The Model

- Two parties – oracle machines
- Oracle is a random function $f: \{0,1\}^n \to \{0,1\}^n$
- Semi-honest adversaries – follow the prescribed protocol, may try to obtain additional info.
- Unbounded computation, polynomial query complexity
- An oracle-aided protocol:



Exemplifying Main Issues - Key Agreement

- No-oracle model (information-theoretic) – No key agreement
  - At each point of an execution – given the transcript so far t, the views of A and B are in a product distribution:

    $$\Pr[(v_A, v_B)|t] = \Pr[v_A|t] \cdot \Pr[v_B|t]$$

  - At the end of the protocol, attacker can sample a view for B, and the output of this view agrees with the A's key with the **same** probability as does B
- Random oracle model – If v_A, v_B have intersecting queries (not determined by t), then they are **not** independent
  - No product distribution on views (oracle answers may not agree)
  - Still – easy to attack – output $f(0^n)$

$q_2 = f(0^n)$ $\boxed{f}$ $q_1 = f(0^n)$

A $m_1 = \perp$ B

$out_A = f(0^n)$ $out_B = f(0^n)$

Eliminating Views' Dependencies

- Main idea: make all oracle queries that were asked with high probability until now – sample views conditioned on the transcript t and the query/answer pairs we obtained
  - [IR89]: attacker Eve – samples random executions to obtain queries
  - [BM09]: attacker Eve – computes actual probability of each possible query
  - [Here]: algorithm **Finder** – similar to [BM09]'s Eve

Rest of This Talk

- Formal statement of our main theorem
- Proof idea
- Applications
  - Limits to Random Oracle key agreement
  - Limits to Random Oracle differentially private computation

Oracle Model to no-oracle Model Mapping

Def: a function family $\mathcal{F}$ and an oracle-aided protocol $\pi = (A, B)$ have a (T, ϵ)-mapping if:

- Exist no-oracle protocol $\tilde{\pi} = (\tilde{A}, \tilde{B})$ and T-query algorithm Map:

Oracle-aided π		No-oracle $\tilde{\pi}$
$\mathbb{D}_{\mathcal{F}} = (\mathrm{out}_A, \quad, \mathrm{Map}^f(t))_{f \leftarrow \mathcal{F},\, \mathrm{View}_{AB} \leftarrow \pi^f}$	$\overset{\epsilon}{\approx}$	$\mathbb{D}_{\mathrm{P}} = (\mathrm{out}_{\tilde{A}}, \quad, \tilde{t})_{\mathrm{View}_{\tilde{A}\tilde{B}} \leftarrow \tilde{\pi}}$

- Furthermore, $\mathbb{D}_{\mathcal{F}}[1,3] \equiv \mathbb{D}_{\mathrm{P}}[1,3]$ and $\mathbb{D}_{\mathcal{F}}[2,3] \equiv \mathbb{D}_{\mathrm{P}}[2,3]$

- Holds for every <u>partial</u> execution
- Map should be <u>consistent</u> (with partial executions)

Main Theorem

- **Main theorem**: Any <u>no-input</u>, ℓ-query protocol π in the random oracle model has an $\left(\left(\frac{\ell}{\epsilon}\right)^2, \epsilon\right)$-mapping

Exists stateless no-oracle protocol $\widetilde{\boldsymbol{\pi}} = (\widetilde{\boldsymbol{A}}, \widetilde{\boldsymbol{B}})$ and $\left(\frac{\ell}{\epsilon}\right)^2$-query algorithm Map:

$$\mathbb{D}_{\mathcal{F}} = (\mathrm{out}_A, \mathrm{out}_B, \mathrm{Map}(t))_{f \leftarrow \mathcal{F}, \mathrm{View}_{AB} \leftarrow \pi^f}$$

is ϵ-close to

$$\mathbb{D}_{\mathrm{P}} = (\mathrm{out}_{\tilde{A}}, \mathrm{out}_{\tilde{B}}, \tilde{t})_{\mathrm{View}_{\tilde{A}\tilde{B}} \leftarrow \tilde{\pi}}$$

Furthermore, $\mathbb{D}_{\mathcal{F}}[1,3] \equiv \mathbb{D}_{\mathrm{P}}[1,3]$ and $\mathbb{D}_{\mathcal{F}}[2,3] \equiv \mathbb{D}_{\mathrm{P}}[2,3]$

Proving Main Theorem

- **Main thm**: Any random oracle model, no-input, ℓ-query protocol has $\left(\left(\frac{\ell}{\epsilon}\right)^2, \epsilon\right)$-mapping

Proof idea: let π=(A,B) be the random oracle protocol

- Emulate by a stateless no-oracle protocol $\tilde{\pi}$:
  - In each round – given that the transcript so far is t –
  1. active party samples a joint view (v_A, v_B) conditioned on t, and
  2. computes next message accordingly
- If views were in prod. distribution $(\Pr[(v_A, v_B)|t] = \Pr[v_A|t] \cdot \Pr[v_B|t])$: Then, we would be done, unfortunately, they are NOT
  - Solution: use Algorithm Finder to bring them close to prod. distribution

Algorithms Finder and Map

Algorithm Finder
- **Input:** partial transcript t and set of query/answer pairs I
- **Oracle:** f
- **Output:** set of query/answer pairs I' containing all queries asked with probability at least δ by either party

- **Lemma** [BM09] (reproved [Here]): the views of A and B, given $(t, \text{Finder}(t, I))$, are close to being a product distribution

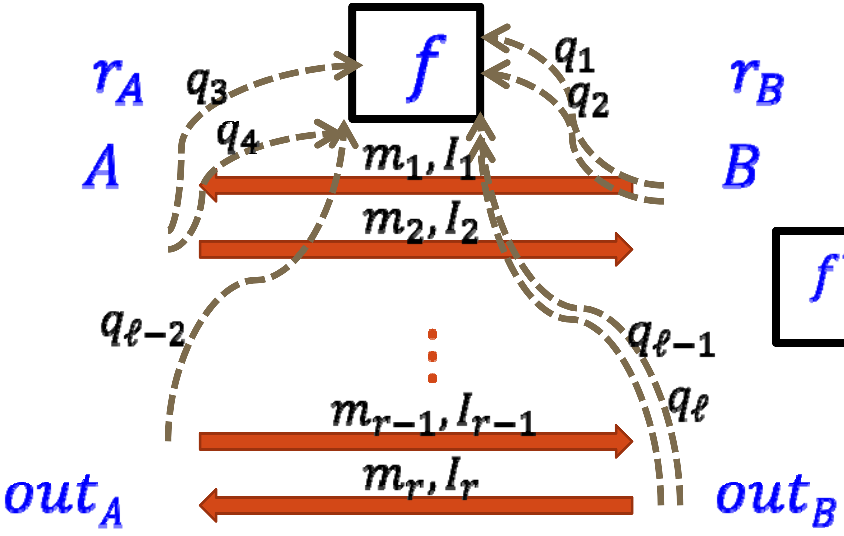
- Algorithm Map:
  - Give a transcript t – augment each partial transcript by applying Finder
  - A syntactic operation

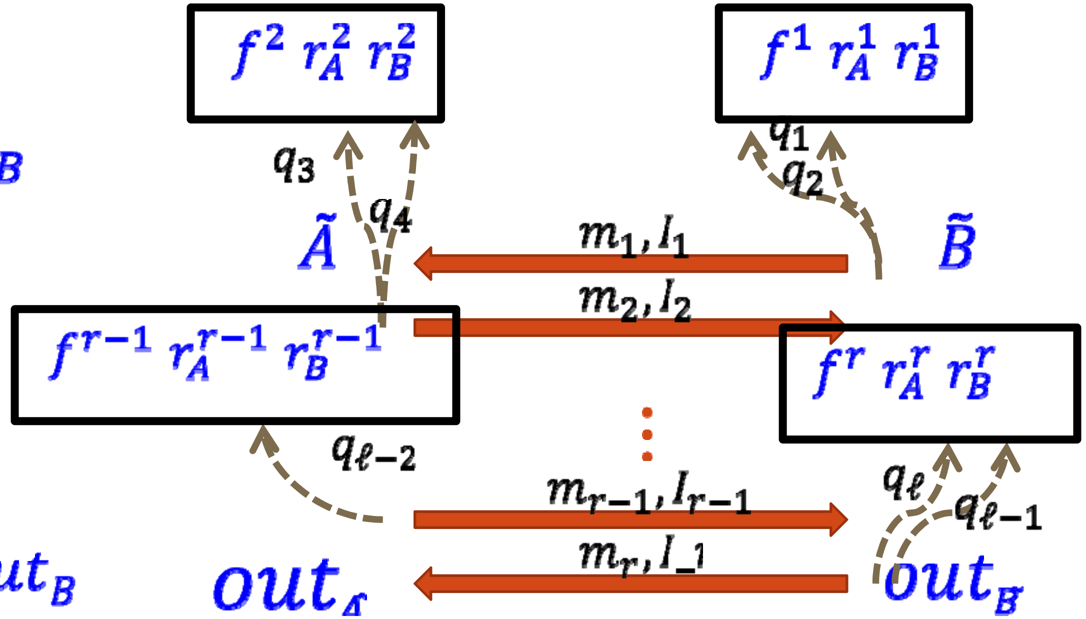
Random Oracle π to No-Oracle $\tilde{\pi}$

Random Oracle – augmented π	No-oracle $\tilde{\pi}$
$\mathbb{D}_{\mathcal{F}} = (\mathrm{out}_A, \mathrm{out}_B, \mathrm{Map}^f(t))_{f \leftarrow \mathcal{F},\, \mathrm{View}_{AB} \leftarrow \pi^f}$	$\mathbb{D}_{\mathrm{P}} = (\mathrm{out}_{\tilde{A}}, \mathrm{out}_{\tilde{B}}, \tilde{t})_{\mathrm{View}_{\tilde{A}\tilde{B}} \leftarrow \tilde{\pi}}$

- $\mathbb{D}_{\mathrm{P}}$ is Close to $\mathbb{D}_{\mathcal{F}}$ – since joint views in $\tilde{\pi}$ conditioned on $\tilde{t}$ are (close to being) a <u>product distribution</u> – follows by [BM09]

(r_A, r_B, f) sampled conditioned on (t, I)





Applications

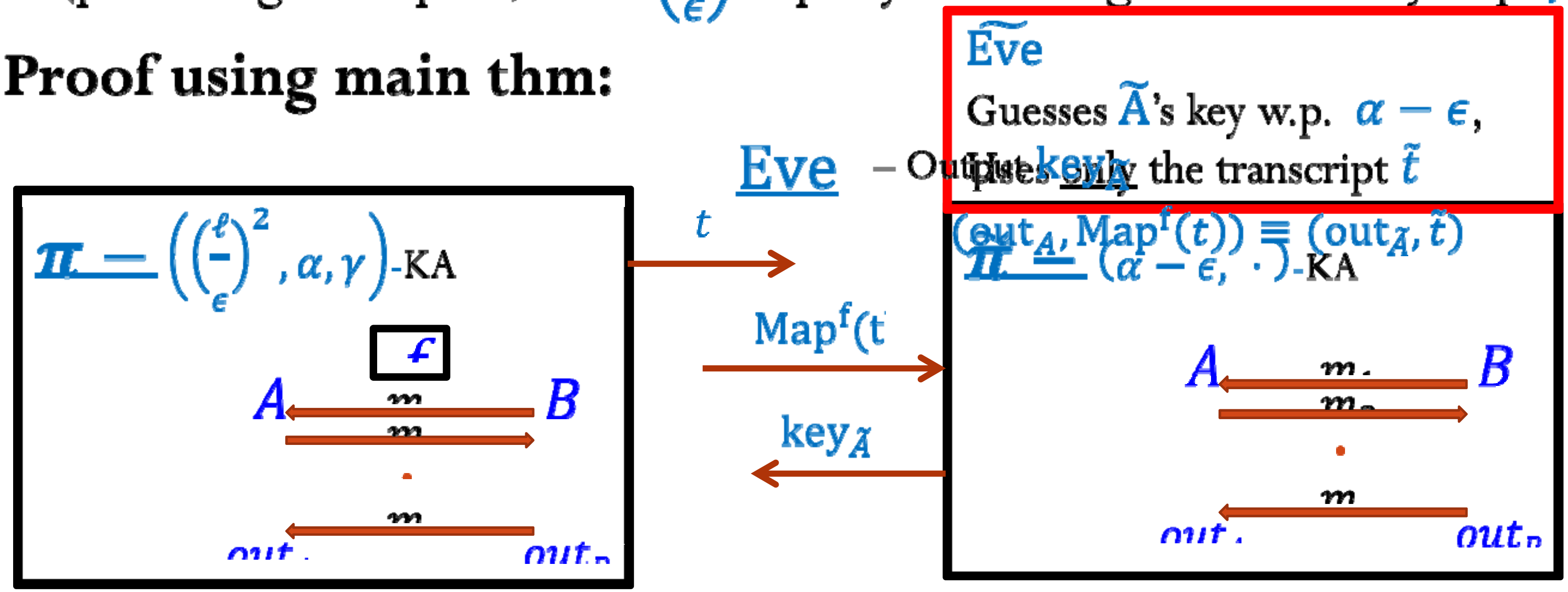
- Limits to Random Oracle key agreement
  - Reproving [Imagliazzo, Rudich 89]
- Limits to Random Oracle differentially private computation
  - Limits to with-input (randomized) protocols

Revisiting [IR89] - Key agreement (warm-up)

Cor 1 (reproving [IR89], [BM09]): No ℓ-query random-oracle $\left(O\left(\frac{\ell}{\epsilon}\right)^2, \alpha, \gamma\right)$-KA protocol with $\alpha > \gamma$

- (parties agree w.p. α, no $O\left(\frac{\ell}{\epsilon}\right)^2$-query attacker guesses A's key w.p. γ)

Proof using main thm:



Differentially Private 2-Party Inner-Product

- **Def**: A protocol (A,B) is (k, α)-DP, if for any $x, y, y' \in \{0,1\}^n$ with $H_d(y, y') = 1$, and any k-query distinguisher D, it holds that

$$\frac{\Pr[D(\mathrm{View}^A(x,y)) = 1]}{\Pr[D(\mathrm{View}^A(x,y')) = 1]} \leq e^{\alpha}$$

  - All parties (including D) are equipped with (the same) random function
  - Similarly defined for View^B

Thm [McGregor, Mironov, Pitassi, Reingold, Talwar, Vadhan 09]:

Any no-oracle protocol for the inner product that is α-DP, errs by $\Omega(\frac{\sqrt{n}}{\log n})$, where n is the input length and α in $(0,1)$

Limits to DP 2-Party Inner-Product

- **Cor. 2**: Any ℓ-query random-oracle model protocol for the inner product that is $\left(\left(\frac{\ell}{\epsilon}\right)^2, \alpha\right)$-DP, errs with $\Omega(\frac{\sqrt{n}}{\log n})$, where n is the input length and α in $(0,1)$

Proof: in the paper

- <u>Remark</u>: Lower-bound for with-input protocols – is obtained from the result on no-input protocols

Summary

- A mapping from semi-honest protocols (without inputs) in the RO model to no-oracle model
  - Semi-honest secure computation (without inputs) cannot be black-box reduced to one-way functions
- Simplification of previous treatment of these questions
- Applications: lower bound on random oracle protocols.
  - Two-party differential privacy [here]
- Main Open questions
  - Handling with-input **randomized**


תודה
Dankie
Gracias
Спасибо
شكراً
Merci
Takk
Köszönjük
Terima kasih
Grazie
Dziękujemy
Dėkojame
Ďakujeme
Vielen Dank
Paldies
Kiitos
Täname teid
谢谢
Thank You
Tak
感謝您
Obrigado
Teşekkür Ederiz
감사합니다
Σας Ευχαριστούμ
ขอบคุณ
Bedankt
Děkujeme vám
ありがとうございます
Tack