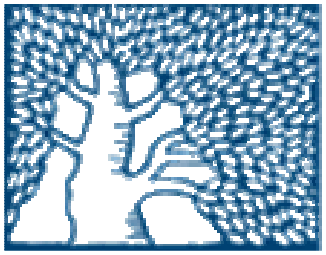
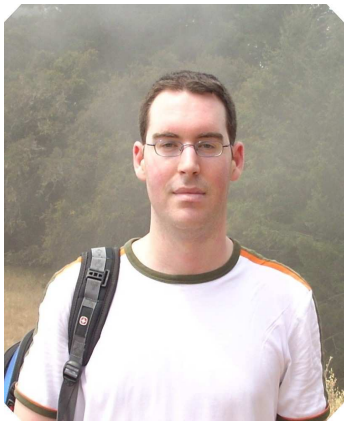
An Optimally Fair Coin Toss

Tal Moran **Moni Naor** **Gil Segev**





Weizmann Institute of Science

Coin Flipping

- **Mutually distrustful parties want to flip a fair (binary) coin**
- **One party may be malicious**
  - Can arbitrarily deviate from the protocol
  - In particular -- can **abort prematurely**
- **Output of the honest party should not be significantly biased**

Coin Flipping

- **When the parties are computationally unbounded, one of them can control the binary coin**

Lots of work: with more than two parties and **honest majority**

- **Coin flipping implies one-way functions**

  [Impagliazzo-Luby '89]

- **Blum '81: Coin flipping using bit commitment**

This Work

- **Cleve ’86:**
  Any r-round protocol is $\Omega(1/r)$ biased

- **Best previously known protocol: bias $O(1/\sqrt{r})$**
  - Cleve-Impagliazzo ‘93: An $\Omega(1/\sqrt{r})$ lower bound in a tightly related model

Our result:

Cleve’s bound is tight!

- **Construct an optimally fair protocol - bias $O(1/r)$**
  - Based on standard cryptographic assumptions (Oblivious Transfer)
  - Builds upon recent progress in **fair secure computation** [**Gordon-Hazay-Katz-Lindell** ’08]

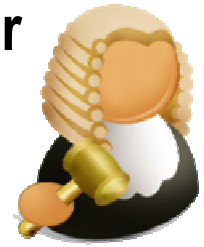
Exact constants: between 1/4 and 1/8

The Main Idea

Threshold round

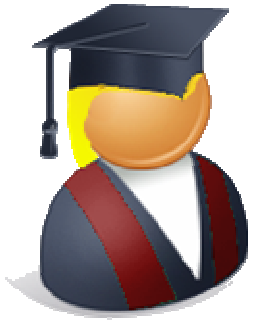
Add a new character: the dealer

At this point: coin should still be random to both parties



1. Chooses a secret $i^* \in [r]$
2. Distributes shares

Let's start



a_1 ⟷ b_1

a_{i^*} ⟷ b_{i^*}

a_r ⟷ b_r

- Until round i^* the bits a_i and b_i are random & independent
- $a_{i^*} = b_{i^*}$ is the output ($a_{i^*+1} = b_{i^*+1}$ = "Halt")
- If Alice halts at round $i \leq i^*+1$, Bob outputs b_{i-1}

Open Problems

- **Minimal assumptions for achieving the optimal $O(1/r)$ bias**
  - Blum's protocol relies on **any** one-way function
  - Our protocol relies on Oblivious Transfer

  } They are black-box separable

- **Efficient implementations**
  - Pre-processing phase: relies on **general secure computation**
  - The dealer's functionality is rather simple
  - Is there a **specific** and more efficient implementation?

- **Optimal bias in the multiparty setting**
  - Several straightforward extensions of our protocol fail
    - The adversary may increase the probability of guessing the crucial round
    - Can get bias $O(k/\sqrt{r})$

- **Fair protocols for other functionalities - GK2008**

Advertisements

- Games For Extracting Randomness
- http://math166-pc.weizmann.ac.il/
- http://www.wisdom.weizmann.ac.il/~neko/
  - Play a game to help science

- Weizmann Winter School on Foundations of Computer Science: **February 15-19th 2009**
  - **Tentative!**

תודה רבה

Thank you