

Estonian Theory Day, Veskisilla 2004

An Oblivious Transfer Protocol with Log-Squared Communication

Helger Lipmaa

Helsinki University of Technology

<http://www.tcs.hut.fi/~helger>

Outline

- Motivation
- Previous Work
- New Construction
- Conclusions

Outline

- Motivation
- Previous Work
- New Construction
- Conclusions

Comp.-Private Information Retrieval: Motivation

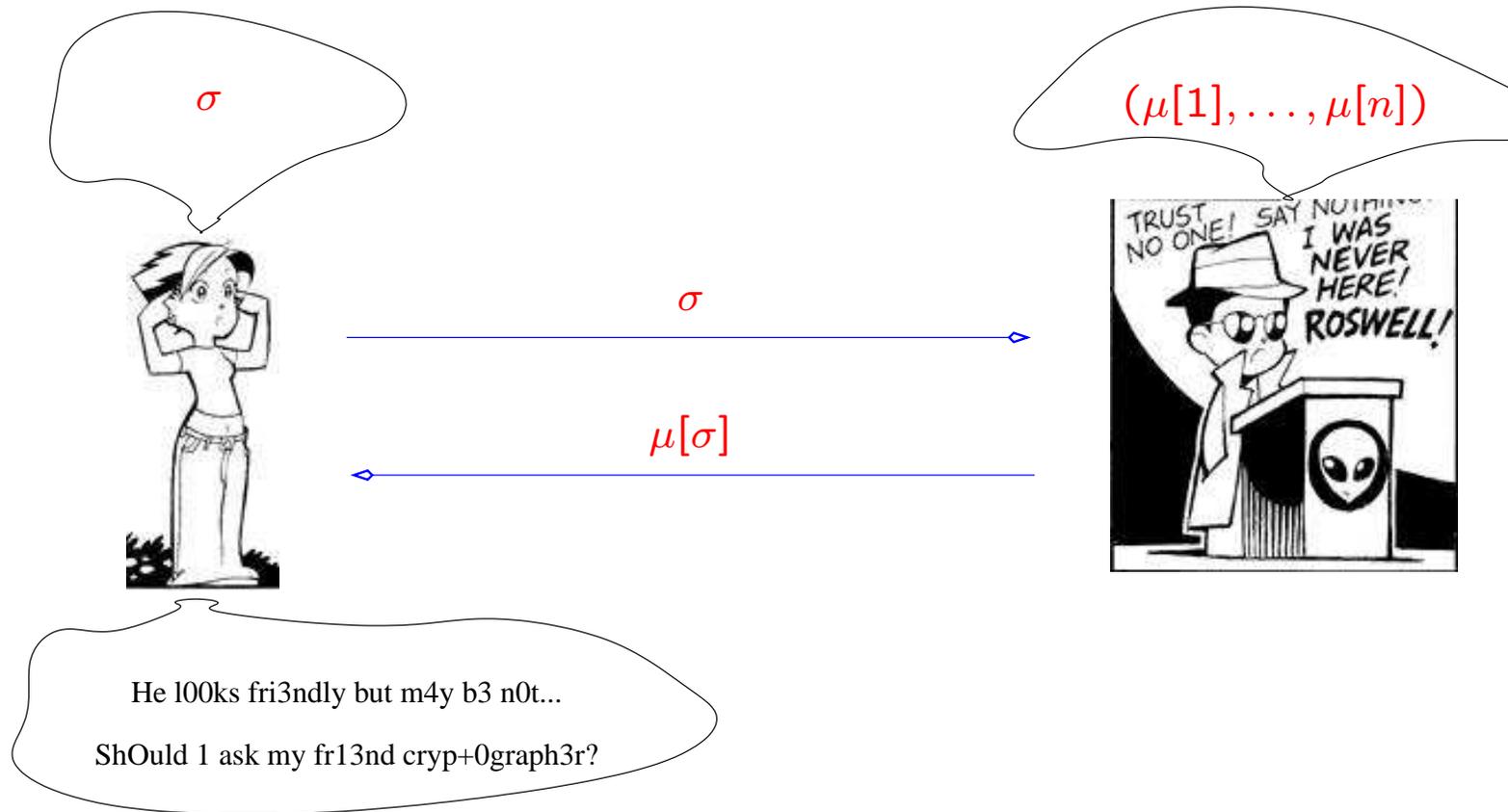
- Chooser wants to retrieve a single element from a database of size n .
- Database should not know which element was retrieved.
- Security + communication-efficiency.
 - ★ Chooser's security is computational.
 - ★ Otherwise, communication is $\Omega(n)$.
- Database has n elements.
- Every element is from \mathbb{Z}_d , ($\log d$ bits).

Comp.-Private Information Retrieval: Motivation



** Parental advisory: this is not the only application of PIR-s. Stay tuned!*

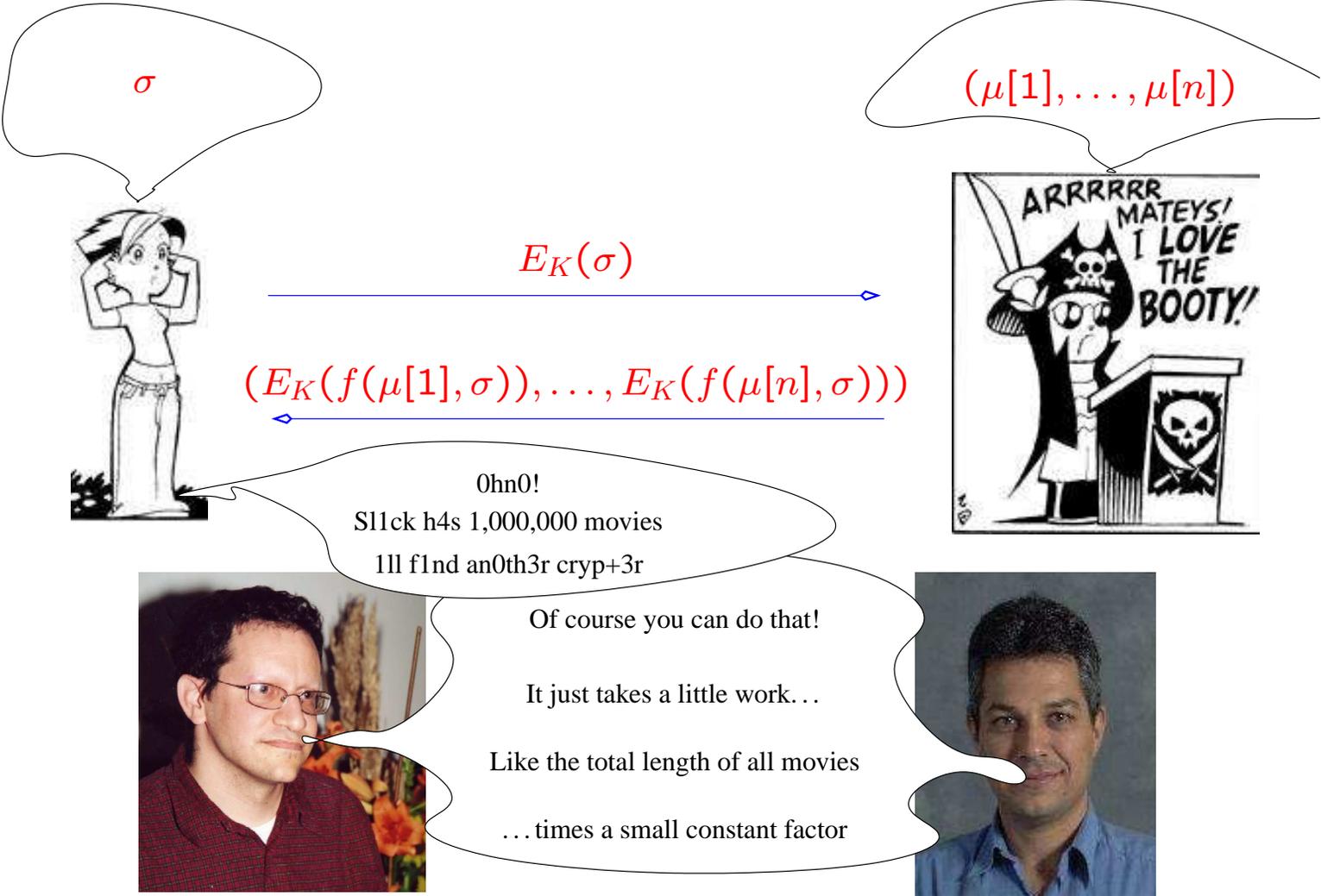
Comp.-Private Information Retrieval: Motivation



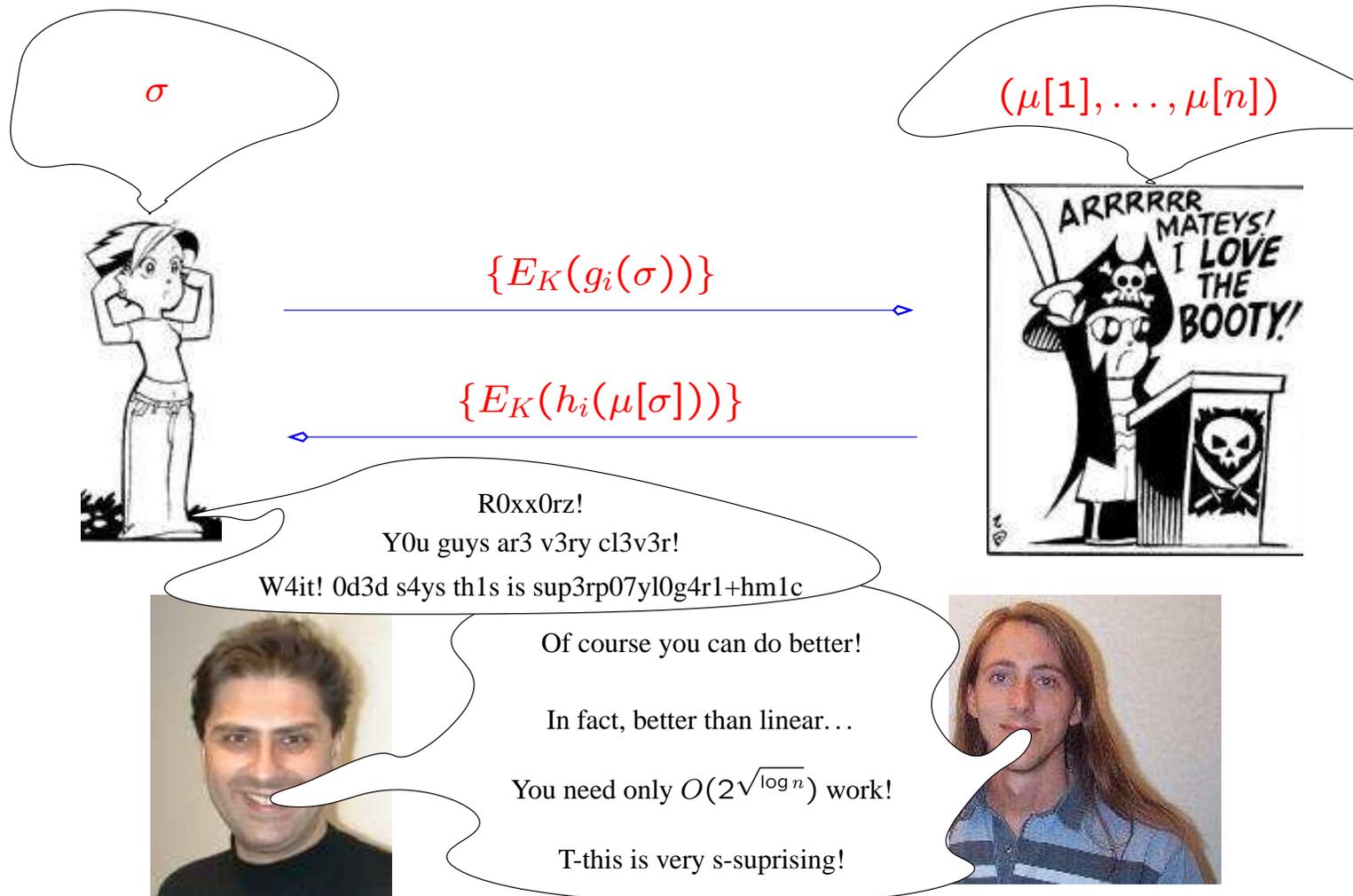
Outline

- Motivation
- Previous Work
- New Construction
- Conclusions

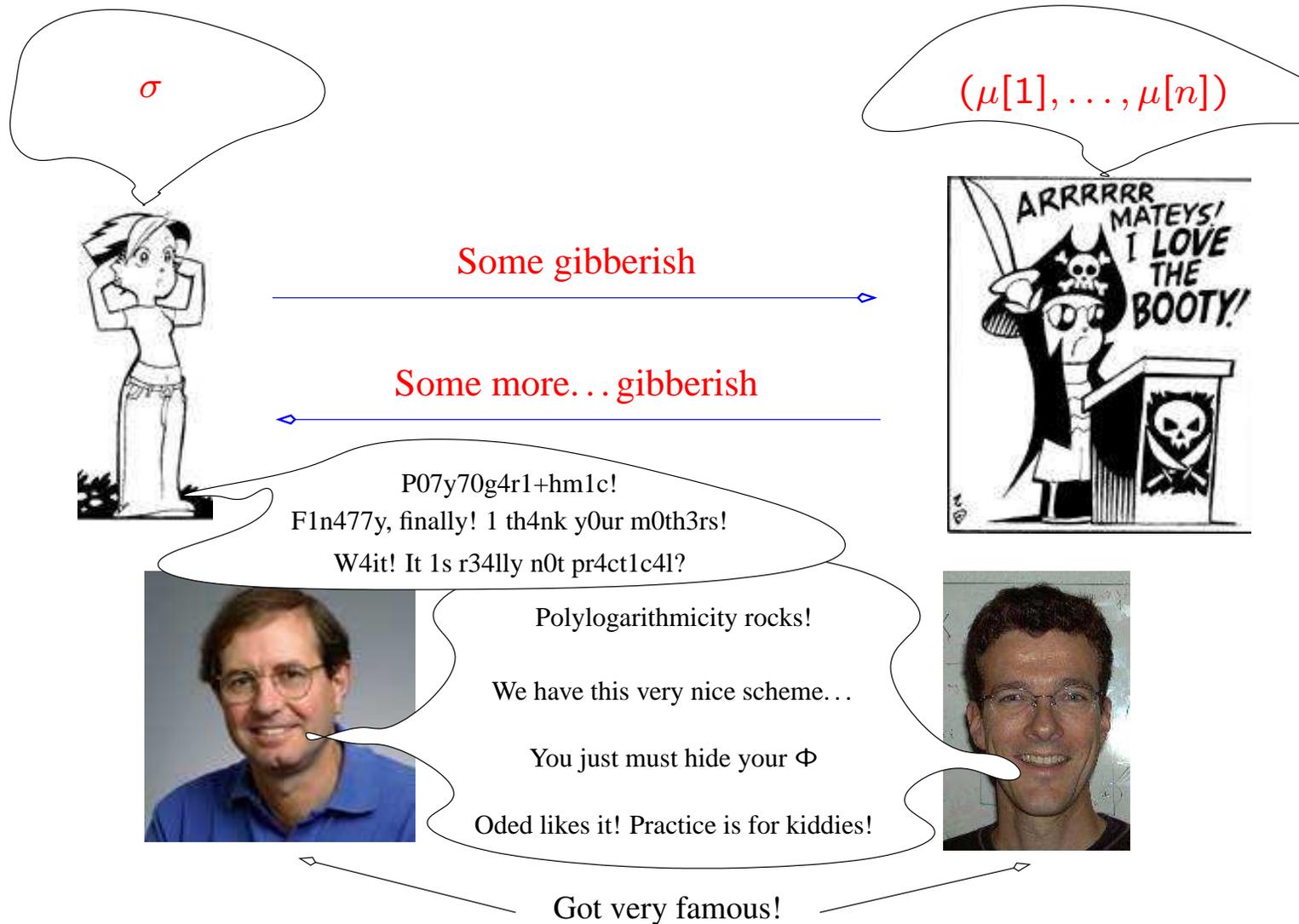
Previous Work



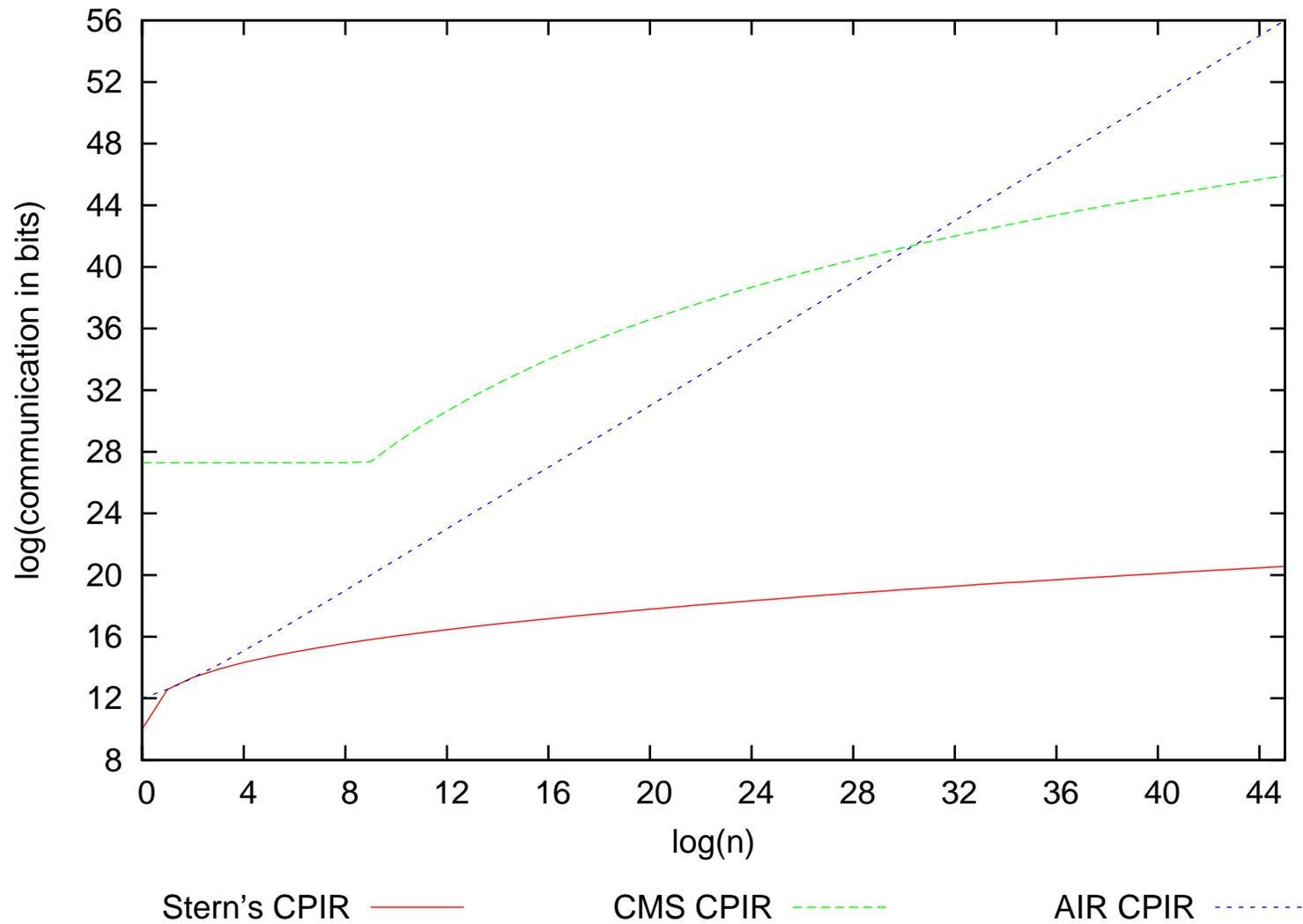
Previous Work



Previous Work



Previous Work: Overview



Previous Work: Overview

- [Aiello, Ishai, Reingold 2001][Naor, Pinkas, 2001]: 2-round CIPR, $O(n \cdot \log d)$ communication.
- [Kushilevitz, Ostrovsky, 1997][Stern, 1998][Chang, 2004]: improved communication to $O(\sqrt{\log n} \cdot 2^{\sqrt{\log n}} \cdot \log d)$.
 - ★ Not polylogarithmic, but up to now the most practical!
- [Cachin, Micali, Stadler, 1999]: can do polylogarithmic.
 - ★ $O((\log^8 n + \log^{2f} n) \cdot \log d)$, $f \geq 4$ unknown (but “constant!”).
- **Need: practical and polylogarithmic**

Outline

- Motivation
- Previous Work
- **New Construction**
- Conclusions

Generic Idea

- Consider μ as an α -dimensional database, and $\sigma = (\sigma_1, \dots, \sigma_\alpha)$ as coordinates of the requested element.
- Chooser sends encrypted coordinates to Sender.
- Server reduces recursively the dimension of the database by computing intermediate i -dimensional databases of ciphertexts.
- Final, 1-dimensional, database is an α -times encryption of requested element. Sender returns it to Chooser.

Generic Idea

- Use a length-flexible additively homomorphic public-key cryptosystem.
 - ★ $\forall s \geq 1$: encrypts plaintext of sk bits to a ciphertext of $(s + 1)k$ bits.
 - ★ $E_K^s(m_1)E_K^s(m_2) = E_K^s(m_1 + m_2)$, thus also

$$E_K^{s+1}(\underbrace{m_1}_{(s+1)k}) \overbrace{E_K^s(m_2)}^{(s+1)k} = E_K^{s+1}(\underbrace{m_1 E_K^s(m_2)}_{(s+1)k}) .$$

- Chooser knows the secret key, Sender knows the public key.
- Sender operates on ciphertexts, sent by Chooser.
- The length parameter s grows in the process.

Generic Idea ($\alpha = 2$)

$$\beta_{11} = \beta_{12} = \beta_{13} = \beta_{14} = E_K^s(0) \quad E_K^s(0) \quad E_K^s(1) \quad E_K^s(0)$$

| | | | | | |
|-------------|-------------|-------------|-------------|---------------|--|
| $\mu(1, 1)$ | $\mu(2, 1)$ | $\mu(3, 1)$ | $\mu(4, 1)$ | \Rightarrow | $w_{11} = \prod_i \beta_{1i}^{\mu(1,i)} = E_K^s(\mu(1, \sigma_1))$ |
| $\mu(1, 2)$ | $\mu(2, 2)$ | $\mu(3, 2)$ | $\mu(4, 2)$ | \Rightarrow | $w_{12} = \prod_i \beta_{1i}^{\mu(2,i)} = E_K^s(\mu(2, \sigma_1))$ |
| $\mu(1, 3)$ | $\mu(2, 3)$ | $\mu(3, 3)$ | $\mu(4, 3)$ | \Rightarrow | $w_{13} = \prod_i \beta_{1i}^{\mu(3,i)} = E_K^s(\mu(3, \sigma_1))$ |
| $\mu(1, 4)$ | $\mu(2, 4)$ | $\mu(3, 4)$ | $\mu(4, 4)$ | \Rightarrow | $w_{14} = \prod_i \beta_{1i}^{\mu(1,i)} = E_K^s(\mu(1, \sigma_1))$ |

Generic Idea ($\alpha = 2$)

$$\beta_{11} = \beta_{12} = \beta_{13} = \beta_{14} = E_K^s(0) \quad E_K^s(0) \quad E_K^s(1) \quad E_K^s(0)$$

Chooser sends $\{\beta_{jt} = E_K^s(\sigma_j =? t)\}$ to Sender

| | | | |
|-------------|-------------|-------------|-------------|
| $\mu(1, 1)$ | $\mu(2, 1)$ | $\mu(3, 1)$ | $\mu(4, 1)$ |
| $\mu(1, 2)$ | $\mu(2, 2)$ | $\mu(3, 2)$ | $\mu(4, 2)$ |
| $\mu(1, 3)$ | $\mu(2, 3)$ | $\mu(3, 3)$ | $\mu(4, 3)$ |
| $\mu(1, 4)$ | $\mu(2, 4)$ | $\mu(3, 4)$ | $\mu(4, 4)$ |

\Rightarrow

$$w_{11} = \prod_i \beta_{1i}^{\mu(1,i)} = E_K^s(\mu(1, \sigma_1))$$

$$w_{12} = \prod_i \beta_{1i}^{\mu(2,i)} = E_K^s(\mu(2, \sigma_1))$$

$$w_{13} = \prod_i \beta_{1i}^{\mu(3,i)} = E_K^s(\mu(3, \sigma_1))$$

$$w_{14} = \prod_i \beta_{1i}^{\mu(4,i)} = E_K^s(\mu(1, \sigma_1))$$

$$\beta_{21} = E_K^{s+1}(0)$$

$$\beta_{22} = E_K^{s+1}(0)$$

$$\beta_{23} = E_K^{s+1}(1)$$

$$\beta_{24} = E_K^{s+1}(0)$$

Chooser sends: $\sum_{j=1}^{\alpha} \sum_{t=1}^{n^{1/\alpha}} (s + j)k$ bits

Sender sends $(s + \alpha)k$ bits

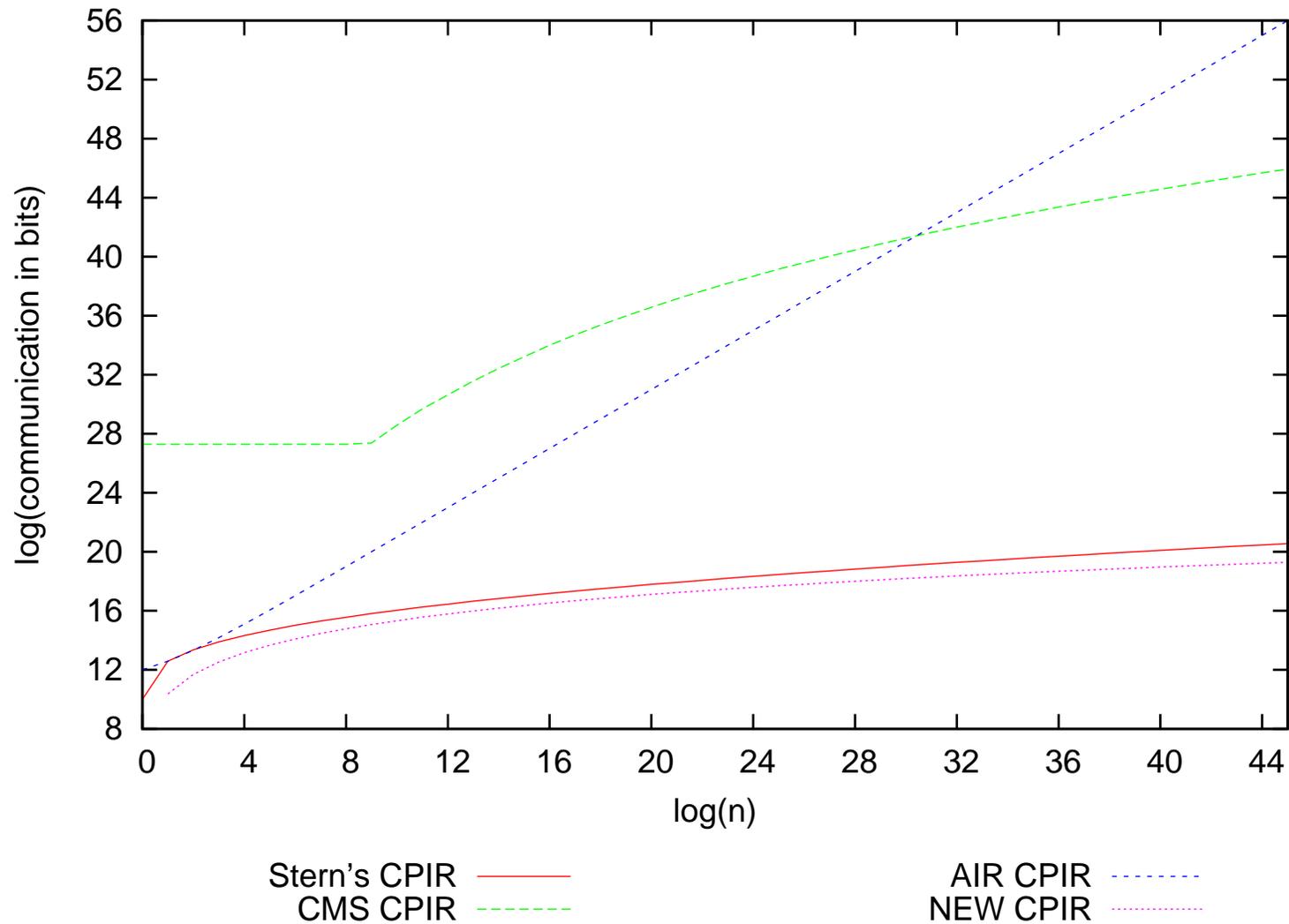


$$w_2 = \prod_i \beta_{2i}^{w_{1i}} = E_K^{s+2}(E_K^{s+1}(\mu(\sigma_1, \sigma_2)))$$

Communication

- Suitable for sending integers from \mathbb{Z}_d
- Chooser sends $\alpha(s + \frac{\alpha+1}{2})n^{1/\alpha}k$ bits.
 - ★ $sk \approx \log d$, thus $(\alpha \log d + \alpha \cdot \frac{\alpha+1}{2}k)n^{1/\alpha}$ bits.
- Optimal if $\alpha = \Theta(\log n)$: $\Theta(\log^2 n \cdot k + \log n \cdot \log d)$ bits.
- Paper discusses various optimisations
 - ★ For small d , pack several database elements into one plaintext, and assume μ is a lopsided hyperrectangle.

Polylogarithmic yet practical



Security

- We use a *length-flexible* additively homomorphic pkc.
- Standard security requirement for homomorphic pkc's: IND-CPA security
- [Damgård-Jurik 2001, 2003]: There exist IND-CPA secure length-flexible additively homomorphic pkc's.
- **Not sufficient here** (in some sense).
- Length-flexible cryptosystems have been used before to improve the efficiency of e-voting and e-auction schemes.
- There, IND-CPA is sufficient. Why not here?

Security

- In e-voting/e-auction schemes, the participants send out ciphertexts only with one, fixed, although large, s .
- In our protocol, Chooser sends ciphertexts that correspond to different s 's:
$$\beta_{jt} = E_K^{s+j-1}(\sigma_j =? t).$$
- This needs that the cryptosystem is secure against attacks where the attacker legally sees ciphertexts of related but unknown plaintexts with different values of s .
- We define a new security notion: IND-LFCPA security.

IND-LFCPA Security

Definition A pkc is α -IND-LFCPA secure, if every “fast” attacker has “small” success in the next game:

- A random key is chosen, attacker gets the public key.
- Attacker chooses $(m_0, m_1, s_1, \dots, s_\alpha)$.
- A random $b \leftarrow \{0, 1\}$ is chosen.
- Attacker obtains random ciphertexts $(E_K^{s_1}(m_b), \dots, E_K^{s_\alpha}(m_b))$.
- Attacker outputs a bit b' .
- Attacker wins if $b = b'$.

IND-LFCPA Security

- All IND-CPA secure length-flexible cryptosystems are IND-LFCPA secure
 - ★ ... with α -times security degradation.
- IND-LFCPA security is such a basic notion that it should be considered standard for length-flexible pkc's.
- [Damgård-Jurik, 2001, 2003] pkc's are IND-LFCPA secure with *tight reduction* (no security degradation).
- If the underlying pkc is IND-LFCPA secure, our CPIR is secure.
 - ★ Tight reduction.

Log-Squared Oblivious Transfer

- In CIPR, we care only about Chooser's privacy.
- OT: also Sender's privacy is important .
 - ★ Chooser obtains no information about $\mu[i]$ for $i \neq \sigma$.
- [Naor-Pinkas 1999] transformation: with log. overhead in communication, transforms our CIPR to OT. Bad: computational server-privacy.
- Zero-knowledge proofs: Chooser proves in ZK her inputs are correct. Information-theoretical server-privacy. Bad: four rounds or two-rounds but security only in random-oracle model (NIZK).

Log² OT w/ Conditional Disclosure of Secrets

- CDS — a relatively old but little known technique. Chooser obtains right answer iff her inputs were in a valid range.
- [Aiello-Ishai-Reingold]: for pkc with a plaintext space of prime order.
No such length-flexible cryptosystems are known.
- [Laur, Lipmaa, manuscript]: Additive CDS.
 - ★ Can be applied in conjunction with length-flexible pkc's.
- Result: two-round i-t server-private OT protocol with log-squared communication, secure in the standard model.
- Additive CDS is *less* efficient in conjunction with Stern's CPIR.

Conclusions

- CIPR/OT with log-squared communication: better than “impractical” poly-logarithmic CMS CIPR and “practical” superpolylogarithmic CIPR by Stern.
- Inspired by Stern’s CIPR, but uses length-flexible cryptosystems.
- Security: not very obvious. Requires new notion. Purely by luck(?), existing length-flexible pkc’s are secure.
- Communication: $\Theta(\log^2 n \cdot k + \log n \cdot \log d)$ — note that for large documents, this is $\approx \Theta(\log n \cdot \log d)$.
 - ★ Non-private information retrieval: $\log n + \log d$ bits — close to optimal.
- Polylogarithmicity is not everything! Exact communication matters.

Any questions?



Caveat: This presentation is based on a draft version of the paper!