Randomized Encoding of Functions & MPC with Low Round Complexity

Benny Applebaum
Tel Aviv University
Back to Yuval’s Talk
Completeness Theorems [1988]

[Ben-Or, Goldwasser, Wigderson, Chaum, Crépeau, Damgård]

Every function $f$ can be perfectly computed

- Passive adversary $T < N/2$ (honest majority)
- Active adversary $T < N/3$ (strong majority)

Tight: Bounds on $T$ are optimal

Complexity: $\text{poly}(\text{circuit-size}(f))$

Rounds: Multiplicative depth of $f$

\[ \sim \log \text{degree} + 1 \]
Interaction is Expensive

Can we get constant-round protocol?

What is the best achievable round complexity $R_{\text{MPC}}$?
- $R_{\text{MPC}}>1$ even for weakest security notion

Non-Interactive IT MPC in some model?

Questions valid even with LARGE communication
Non-Interactive MPC
Private Simultaneous Message Protocols

“minimal model for secure computation” [Fei-Kil-Nao94, Ish-Kus98]

Should learn only $f(x_A, x_B)$
Private Simultaneous Message Protocols

“minimal model for secure computation” [Fei-Kil-Nao94, Ish-Kus98]

Should learn only correctness

f(x_A, x_B)
Private Simultaneous Message Protocols

“minimal model for secure computation” [Fei-Kil-Nao94, Ish-Kus98]

Should learn only $f(x_A, x_B)$
PSM vs CDS

CDS(f) ≤ PSM (g)

where

g(x, (y, s)) = s iff f(x, y) = 1
PSM vs CDS

3n \leq \text{PSM}(f) \leq 2^{n/2} 
[FKN94, A-HMS17] 

n \leq \text{CDS}(f) \leq 2^{\tilde{\Theta}(\sqrt{n})} 
[BIKK14]

\begin{align*}
\text{Alice} & \quad x_A & \quad g_A(x_A, r) \\
\text{Bob} & \quad x_B & \quad g_B(x_B, r) \\
\text{Carol} & \quad f(x_A, x_B) \\
\text{Alice} & \quad x_A & \quad g_A(x_A, r) \\
\text{Bob} & \quad x_B & \quad g_B(x_B, r) \\
\text{Carol} & \quad f(x_A, x_B) \
\end{align*}

\text{iff } f(x_A, x_B) = 1
Example: XOR

- \( f(x_A, x_B) = x_A \oplus x_B \)  \((x_A, x_B \in \{0, 1\})\)

\[ m_A \oplus m_B \]
\[ f(x_A, x_B) = x_A \cdot x_B \]
AND: Intermediate protocol

- \( f(x_A, x_B) = x_A \cdot x_B \)

\[
\begin{align*}
A &= x_A + a \\
B &= x_B + b \\
C &= x_A b + x_B a + ab \\
\end{align*}
\]

\[ x_A x_B + x_A b + x_B a + ab \]
AND: Second Step

- \( f(x_A, x_B) = x_A \cdot x_B \)

\[ r \in \{0,1\} \]

- \( w = x_A b \)
- \( z = x_B a + ab \)

Alice

Bob

\[ w + r \]
\[ z - r \]

Carol

\[ C = x_A b + x_B a + ab \]
\[ f(x_A, x_B) = x_A \cdot x_B \]

\[ a, b, r \in \{0, 1\} \]

- Alice: \( A = x_A + a \), \( C = x_A b + r \)
- Bob: \( B = x_B + b \), \( D = x_B a + ab - r \)
- Carol: \( A \cdot B \), \( -(C + D) \)
Introspection

• Gradually constructed the protocol

• Intermediate construction didn’t satisfy syntax but preserved information

• Used Simple Maneuvers
Test your intuition:

• Q: Combine PSM(f), PSM(g) to PSM(f AND g)?

• Ex: PSM(f) based on truth-table randomization

Truth table

\[ f \]

\[ y \rightarrow 11110000111101 \]

\[ x \]

\[ N = 2^n \]
Multiparty Version [IK98]

Alice1 \( x_1 \)

Alice2 \( x_2 \)

\[ f(x_1, \ldots, x_n) \]

Carol

\[ g_1(x_1, r) \]

\[ g_n(x_n, r) \]

Alice-n \( x_n \)
Example: iterated group product

Abelian Group: \( f(x_1, \ldots, x_n) = x_1 + x_2 + \ldots + x_n \)
Example: iterated group product

Non-abelian Group: \( f(x_1, \ldots, x_n) = x_1 x_2 \ldots x_n \)

[Kilian 88]

Alice 1 \( x_1 \)

Alice 2 \( x_2 \)

\( \ldots \)

Alice -n \( x_n \)

\( x_1 r_1 \) \( r_1^{-1} x_2 r_2 \) \( r_2^{-1} x_2 r_3 \) \( \ldots \) \( r_{n-2}^{-1} x_{n-1} r_{n-1} \) \( r_{n-1}^{-1} x_n \)

Carol
Handling General Functions

**Theorem [Barrington 86]**
Every boolean \( f \in \text{NC}^1 \) can be written as iterated group product

**Corollary**
Every \( f \in \text{NC}^1 \) has multiparty PSM with poly-communication
From Multiparty PSM to OT-based MPC

Application: Basing SFE on OT [Yao, Kilian 88, ...]
From Multiparty PSM to OT-based MPC

Application: Basing SFE on OT [Yao, Kilian 88, ...]

\[ f(x_A, x_B) \]

\[ g_n(0, r) \quad g_n(1, r) \]

\[ x_n \quad g_n(x_n, r) \]
Can we leverage these ideas in more general settings?

\[ f(x_A, x_B) \]

Alice

Bob

\[ g_A(x_A, r) \]

\[ g_B(x_B, r) \]

Carol

correctness

privacy
Can we leverage these ideas in more general settings?

\[
f(x_A, x_B) = g_A(x_A, r) \quad g_B(x_B, r)
\]

**correctness**

**privacy**
Randomized Encoding of Functions
[Ish-Kus00, A-Ish-Kus04]

- $g$ is a “randomized encoding” of $f$
  - Nontrivial relaxation of computing $f$

$$\text{Dec}(g(x,r)) = f(x)$$

$g(x,r)$

Correctness: $\text{Dec}(g(x,r)) = f(x)$

Privacy: $\text{Sim}(f(x)) \equiv g(x,r)$
Randomized Encoding of Functions
[Ish-Kus00, A-Ish-Kus04]

• $g$ is a “randomized encoding” of $f$
  – Nontrivial relaxation of computing $f$

$$x_1 + r_1, \ldots, x_{n-1} + r_{n-1}, \quad x_n - \sum r_i$$

$\text{Dec}(g(x,r)) = f(x)$

Correctness

Privacy

$\text{Sim}(f(x)) \equiv g(x,r)$
Randomized Encoding of Functions
[Ish-Kus00, A-Ish-Kus04]

• Securely computing $g$ => securely computing $f$

• If $g$ is realizable in MPC-model so is $f$

$$\text{Dec}(g(x,r)) = f(x)$$

$$\text{Sim}(f(x)) \equiv g(x,r)$$
Randomized Encoding of Functions
[Ish-Kus00, A-Ish-Kus04]

General paradigm:
• $g$ should be “simpler” than $f$
  (meaning of “simpler” determined by application)

• $g$ can be used as a substitute for $f$
Applications at a Glance

Randomized encodings

Secure computation
[Yao, Kilian, FKN94, IK00, ...]

Parallel crypto

Delegation

Circuit LB’s
[Chen-Ren20]

Hardness of Approximation

Coding Theory

KDM-security

See surveys:
Ishai: Randomization Techniques for Secure Computation
Applebaum: Garbled Circuits as Randomized Encodings of Functions: a Primer
Randomized Encoding in Cryptography

Obfustopia
Secure Computation
Public-Key
Symmetric
Information Theoretic

Compact Functional Encryption/Obfuscation
Reusable Garbled Circuit
Functional Encryption
Garbled Circuits
PSM
Useful Properties

Composition: $\text{Enc} (\text{Enc}(f))$ is an encoding of $f$.

Concatenation: $(\text{Enc}(f_1), \text{Enc}(f_2))$ is an encoding of $f = (f_1, f_2)$.

Substitution: $\text{Enc}(f) \circ h$ encodes $f \circ h$.
Composition: $\text{Enc}(\text{Enc}(f))$ is an encoding of $f$.

Concatenation: $(\text{Enc}(f_1), \text{Enc}(f_2))$ is an encoding of $f=(f_1,f_2)$.

Substitution: $\text{Enc}(f) \circ h$ encodes $f \circ h$.
Back to Constant-Round MPC
Randomizing Polynomials [IK00]
Randomizing Polynomials [IK00]
Randomizing Polynomials [IK00]

MPC for $g \Rightarrow$ MPC for $f$

distribution $Y$

degree-$d$ $g$

Decoder

Simulator

$y$

$x$

$f$

$x$
Randomizing Polynomials [IK00]

g has d-round protocol $\Rightarrow$ f has d-round protocol!
Thm [IK02] Every $f$ has perfect RP of degree 3
Perfect-MPC with Constant Round

**Thm [IK02]** Every f has perfect RP of degree 3
- Efficient for NC1, log-space

**Constant-round perfect** protocol for all functions
- Passive $T < \frac{N}{2}$: 3 rounds $\geq ? > 1$
- Active $T < \frac{N}{3}$: large const. $\geq ? > 2$

**Q:** What’s the optimal round complexity?
Problem:
For most functions, **NO** degree-2 perfect RE’s

Sol: Compromise!
Aim for a **weaker** notion
Multiparty Randomized Encoding (MPRE)  
[A-Bra-Tsa18]

Relaxed correctness: Each party has a decoder

"simple" $g$ → $f$
Multiparty Randomized Encoding (MPRE) [A-Bra-Tsa18]

Relaxed correctness: Each party has a decoder

“simple” $g$

Decoder

$f$
Multiparty Randomized Encoding (MPRE)

**Relaxed privacy:** Every minority has a simulator
Multiparty Randomized Encoding (MPRE)

Relaxed privacy: Every minority has a simulator

“simple” $g$

$\simulator$

$f$
**MPRE relaxes Randomized Encoding**

- Encodes functionality
- RE is a special case of MPRE
- Protocol for g $\Rightarrow$ Protocol for f

MPRE = Distributed-GC?
Degree-2 ?

f
**Thm** [A-Bra-Tsa18]: every $f$ has MPRE of “effective” deg-2
- Efficient for log-space
- Efficient computational-MPRE for general circuits
⇒ 2-round passively-secure honest-majority protocol

Also [Gar-Ish-Sri-18]
inspired by [GS-16-17]
Back to [IK00]:
Degree-3 Randomizing Polynomials from Information-Theoretic Garbled Circuits
Correctness

Privacy?
Overall degree 3
• \( \text{deg(gate)} + 1 \)
Degree of Garbled Gate

per gate:
• 4 ciphertexts
• \( \text{deg-3} = \text{deg(gate)} + 1 \)
Overall degree 3
• $\text{deg}(\text{gate}) + 1$

Can we reduce the degree to 2?
What if...?
After local preprocessing, MPRE with degree = 2!

What if…?

Can we enforce such structure?
MPRE with degree = 2

nice MPRE
Protocol Induced MPRE

\[ f \]

Inputs

Outputs
Protocol Induced MPRE
Protocol Induced MPRE
Putting It All Together

MPRE with degree = 2

nice MPRE

Protocol $\pi$

$f$
Round Complexity of MPC

Assume a protocol with T-security for \( f \)
Then \( f \) reduces to degree-2 computation with T-security

Assuming honest majority and passive adversary:

Every function has perfect 2-round protocol
  – Efficient for \( \text{NC1} \), log-space
  – Computational variant for poly-size circuits using OWFs
Two-Round Protocol

- Practical relevance?
  - 2-round protocols easily transfer to client-server model
    [Ishai-Damgard ‘05]
Two-Round Protocol

• Practical relevance?
  – 2-round protocols easily transfer to client-server model [Ishai-Damgard ‘05]

Private as long as majority of the servers & clients are honest
Active Adversary

Assume a protocol with T-security for $f$
Then $f$ reduces to degree-2 computation with T-security

Extensions:
• Active adversaries [ABT19]
Active Adversary

Assume a protocol with T-security for f
Then f reduces to degree-2 computation with T-security

Extensions:

- Active adversaries [ABT19]
- Larger fields [AKP19]

**Perfect protocols for all functions**

- Passive $T < N/2$: 2 rounds [ABT18, GIS18]
- Active $T < N/4$: 3 rounds [ABT19]
- Active $T < N/3$: 4 rounds [AKP19]
Active Adversary

Assume a protocol with $T$-security for $f$
Then $f$ reduces to degree-2 computation with $T$-security

Extensions:
- Active adversaries [ABT19]
- Larger fields [AKP19]

**Perfect protocols** for all functions
- Passive $T < N/2$: 2 rounds [ABT18, GIS18]
- Active $T < N/4$: 3 rounds [AKP20]
- Active $T < N/3$: 4 rounds [AKP20]
Conclusion

Randomized Encoding $f$

Multiparty Randomized Encoding

Protocol for $f$
Take Home Message

Abstraction is powerful but tight results may require refined tools

Thank You