MPC with Silent Preprocessing
or: Two-Round OT Extension from LPN

Elette Boyle, Geoffroy Couteau, Niv Gilboa, Yuval Ishai,
Lisa Kohl, Peter Rindal, Peter Scholl

TPMPC 2019, Bar-Ilan University
Secure Computation with Preprocessing

[Beaver ’91]

Correlated randomness

\[ \text{Preprocessing} \]

Online phase

\[ f(x, y) \]

- Information-theoretic
- Constant comp. overhead
- Comm: e.g. 4 bits per AND gate (2-PC)
Secure Computation with Preprocessing

[Beaver ’91]

Dominates overall cost

Correlated randomness

Interactive protocol

Online phase

\[ f(x, y) \]

• Information-theoretic
• Constant comp. overhead
• Comm: e.g. 4 bits per AND gate (2-PC)

Peter Scholl
Secure Computation with **Silent** Preprocessing

[BCGI 18, BCGIKS 19]

Correlated, short seeds

Silent expansion

Correlated pseudorandomness

Setup functionality

Online phase

\[ f(x, y) \]

Peter Scholl
Secure Computation with Silent Preprocessing

[BCGI 18, BCGIKS 19]

Silent expansion
Correlated, short seeds

“Small” setup protocol
• Less communication
• Ad-hoc future interactions

Correlated pseudorandomness

Online phase

\[ f(x, y) \]

Peter Scholl
Pseudorandom Correlation Generators

[BCGI 18, BCGIKS 19]

• Target correlation: \((R_0, R_1)\)
  ➢ E.g. random OT \(((b, m_b), (m_0, m_1))\)

• Algorithms Gen, Expand:

\[
(k_0, k_1) \leftarrow \text{Gen}(1^\lambda)
\]

\[\tilde{R}_0 \leftarrow \text{Expand}(k_0)\]
\[\tilde{R}_1 \leftarrow \text{Expand}(k_1)\]

Security: \(\approx (R_0, R_1)\) (in some sense)
Constructions of PCGs

- Multi-party linear correlations (PRG)
  - [GI 99], [CDI 05]

- Vector-OLE (LPN)
  - [BCGI 18]

- Constant-degree poly (LPN)
  - [BCGIK 19]

- Truth tables (PRG)
  - Multi-party bilinear (LPN, LWE)
  - Low-degree via HSS (LWE, pairings)

- OT (LPN)
  - [BCGIKRS 19]

Also: [BCGIO 17], [$ 18$] (less practical)
Silent OT Extension: PCG for random OT from LPN

[BCGIKS 19, BCGIKRS 19]

• Two-round OT extension
  ➢ Useful for non-interactive secure computation [IKOPS 11, AMPR 15, MR 17]
  ➢ Previously only w/ [Beaver 96]
  ➢ Inherently non-black box [GMMM 18]

• Malicious security
  ➢ 2 rounds with Fiat-Shamir

• Faster than [IKNP 03] in WAN:
  ➢ Almost no communication: only pay |input|
  ➢ 3.6 million OT/s

E.g. Semi-honest 2-PC w/ 4.2 bits per AND gate, 30x less than [DKSSZZ17]
Silent OT Extension: Overview

Seed $\in \mathbb{F}_{2^\lambda}$

1. Sparse $\in \{0,1\}^m$
2. Expand
3. Randomize
4. Uniform
5. Hash
6. Random OTs

Setup

Receiver

Sender

Local comp.

Random OTs

Peter Scholl
Main tool: puncturable PRF

- PRF $F : \{0,1\}^\lambda \times \{1, \ldots, N\} \rightarrow \{0,1\}^\lambda$

- $k \leftarrow \text{Gen}(1^\lambda)$
  - Master key: allows evaluating $F(k, x)$ for all $x$

- $k^* \leftarrow \text{Punc}(k, \alpha)$
  - Punctured key: can evaluate at all points except for $x = \alpha$

- Security: $F(k, \alpha)$ is pseudorandom, given $k^*$

Simple tree-based construction from a PRG: $|k| = \lambda, \quad |k^*| = \lambda \cdot \log N$

[BW13], [BGI 13], [KPTZ 13]
Key observation: puncturable PRF compresses unit vector products

\[ F(k, \alpha) = \begin{bmatrix} 0 \cdots 0 \parallel 1 \cdots 0 \end{bmatrix} \cdot 0 \cdots 0 \]

- Shares compressed from \( \lambda \cdot N \) to \( \approx \lambda \cdot \log N \) bits
Key observation: puncturable PRF compresses unit vector products

\[ z = F(k, \alpha) + y \]

- **Setup**
  - \[ \alpha \leftarrow \{1, \ldots, N\} \]
  - \[ k \leftarrow \text{Gen}(1^\lambda) \]
  - \[ k^* \leftarrow \text{Punc}(k, \alpha) \]

- **Receiver**
  - \[ \alpha, k^* \]
  - Eval at all \( x \neq \alpha \)
  - \[ z \text{ at pos. } \alpha \]

- **Sender**
  - \[ k \quad y \in \mathbb{F}_{2^\lambda} \]
  - Eval at \( \{1, \ldots, N\} \)

- **Shares compressed from** \( \lambda \cdot N \) to \( \approx \lambda \cdot \log N \) bits
- **Can tweak to multiply by arbitrary** \( y \in \mathbb{F}_{2^\lambda} \)
From weight-1 vectors to weight-\( t \) vectors

Approach 1: addition

\[
\begin{align*}
\text{Weight e.g. } t &= 4 \\
\text{Expansion cost: } &O(t \cdot N) \text{ (naïve) } \\
&O(N) \text{ possible}
\end{align*}
\]

Approach 2: concatenation

\[
O\left(t \cdot \frac{N}{t}\right) = O(N)
\]

Note: regular error pattern
From sparse to pseudorandom products

• Recall: have shares
• Want: uniform vector

Public linear $H$

$\cdot$ = weight $t$

Pseudorandom under LPN!
Our LPN setting

- “Dual” version of LPN over $\mathbb{F}_2$
  - Equivalent to LPN for dual code $H^*$
- Main challenge: $H$ is big (up to 10 million)
  - Use quasi-cyclic codes
    - $\tilde{O}(N)$ (polynomial multiplication over $\mathbb{F}_2$)
- Security:
  - Similar to PQ cryptosystems BIKE, HQC [ABB+19, AAB+19]
  - No better with regular errors [AFS 03, BLPS 11, HOSS 18, BLMZ 19]
  - Noise rate need not imply PKE or OT

$H$ is pseudorandom
From secret-shared products to random OT

\[ s_i \in \mathbb{F}_{2^\lambda}^m \quad \text{and} \quad b_i \in \{0,1\}^m \]

\[ y \in \mathbb{F}_{2^\lambda} \]

\[ b_i = 1: \quad \Rightarrow s_i + t_i = y \]
\[ b_i = 0: \quad \Rightarrow s_i + t_i = 0 \]

\[ \Rightarrow \text{OTs on correlated sender strings } (t_i, t_i + y) \]

Break correlation with hash function [IKNP 03]

Useful for e.g. TinyOT, garbled circuits
2-round setup and 2-round OT extension

• 2-round punctured PRF setup from any 2-round OT
  ➢ For tree-based PPRF (à la black-box [Ds 17] for DPF)
  ➢ $\log N$ parallel OTs (unlike [Ds 17])

• Two-round OT extension on chosen inputs
  ➢ Still need to convert random $\rightarrow$ chosen messages
  ➢ Can run conversion in parallel with setup

• Malicious security at 10-20% overhead
  ➢ LPN with 1-bit leakage
Concrete parameters for silent OT extension

128-bit security

<table>
<thead>
<tr>
<th># output OTs</th>
<th>LPN noise weight</th>
<th>Seed size</th>
<th>Setup comm.</th>
<th>Bits sent per OT</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^4$</td>
<td>126</td>
<td>18 kB</td>
<td>58 kB</td>
<td>46.2</td>
</tr>
<tr>
<td>$10^5$</td>
<td>120</td>
<td>23 kB</td>
<td>83 kB</td>
<td>6.7</td>
</tr>
<tr>
<td>$10^6$</td>
<td>118</td>
<td>30 kB</td>
<td>99 kB</td>
<td>0.3</td>
</tr>
<tr>
<td>$10^7$</td>
<td>116</td>
<td>35 kB</td>
<td>127 kB</td>
<td>0.1</td>
</tr>
</tbody>
</table>
Runtimes (ms) for 10 million random OTs

**LAN (10 Gbps)**
- IKNP: 268 ms
- 2-round silent: 3373 ms
- 3-round silent: 2441 ms

**WAN (100 MBps)**
- IKNP: 13728 ms
- 2-round silent: 3675 ms
- 3-round silent: 2726 ms

**WAN (10 MBps)**
- IKNP: 128854 ms
- 2-round silent: 3603 ms
- 3-round silent: 2756 ms

**IKNP vs 2-round silent vs 3-round silent**

Hybrid: IKNP for \(\approx 1500\) base OTs

**Total comm:** 160 MB vs 145 kB vs 127 kB
Conclusion

• 2-round, silent OT extension
  ➢ Comm: $|\text{input}|$ bits
  (previously: $\lambda + |\text{input}|$ bits per OT)

• Broader perspective:
  ➢ Pseudorandom Correlation Generators
  ➢ Silent preprocessing for MPC

• Many future directions:
  ➢ OT: faster codes, incremental output
  ➢ PCGs + setup for complex correlations:
    o OLE, Beaver triples, truth tables, random bits ($\mathbb{Z}_p$), garbled circuits...
Efficient Pseudorandom Correlation Generators: Silent OT Extension and More

Boyle, Couteau, Gilboa, Ishai, Kohl, Scholl

https://ia.cr/2019/129

Two-Round OT Extension and Silent Non-Interactive Secure Computation

BCGIKS + Rindal

Paper: coming soon!
Code: https://github.com/osu-crypto/libOTe (prototype, not yet in master branch)