Order-C Secure Multiparty Computation for Highly Repetitive Circuits

Gabrielle Beck Aarushi Goel Abhishek Jain Gabriel Kaptchuk

JOHNS HOPKINS UNIVERSITY

BOSTON UNIVERSITY

EUROCRYPT 2021

Existing Efficient and Implemented Protocols

[HN06, DN07, LN17, CGHIKLN18, NV18, FL19, GSZ20]

No. of parties

Size of circuit

$O(n|C|)$: Total computation/communication complexity

- Per-party work: $O(|C|)$
- For large computations, parties need to have large computing resources.
- Limits the kind of parties who can participate.

Better than $O(n|C|)$?

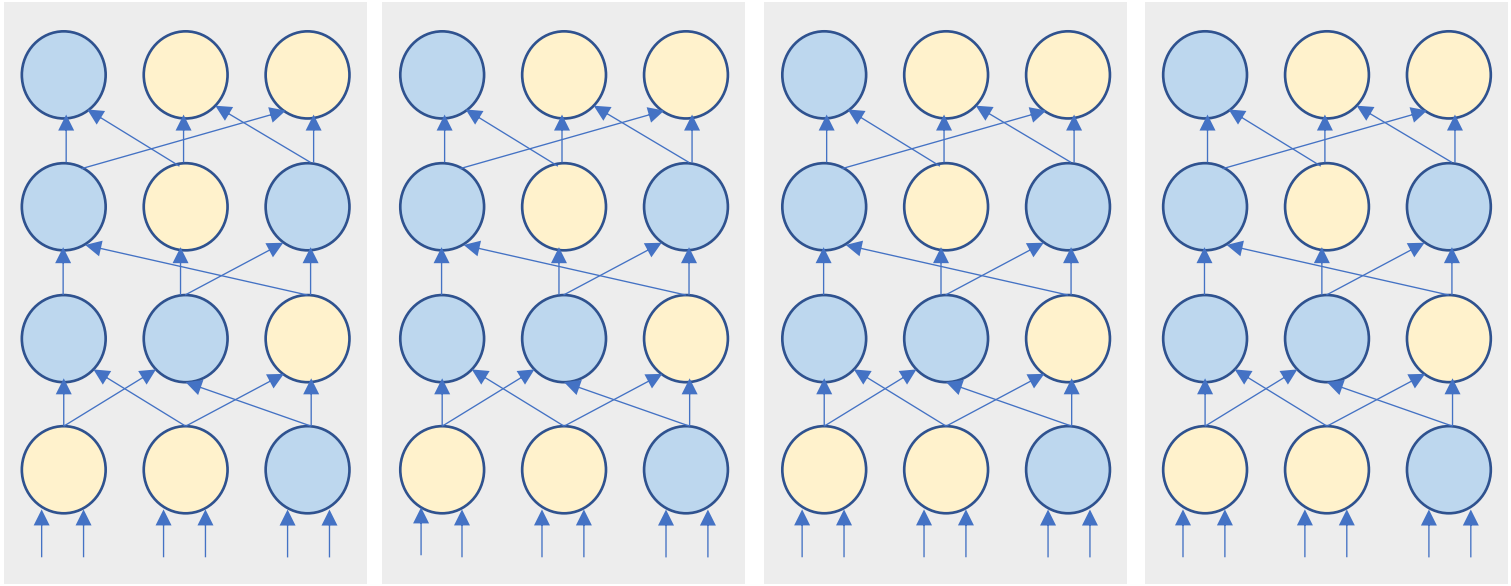
$\tilde{O}(|C|)$: Total computation and communication [DIKNS08, DIK10, GIP15]

- $\tilde{O}$ hides polynomial factors in $\log|C|$ and security parameter κ
- Not concretely efficient

$O(|C|)$: Total computation and communication [DIK10, GIP15]

- Only for SIMD circuits
- No known implementations

Single Instruction Multiple Data Circuits



Circuits that comprise of multiple parallel copies of the same sub circuit

Better than $O(n|C|)$?

$\tilde{O}(|C|)$: Total computation and communication [DIKNS08, DIK10, GIP15]

- $\tilde{O}$ hides polynomial factors in $\log|C|$ and security parameter κ
- Not concretely efficient

$O(|C|)$: Total computation and communication [DIK10, GIP15]

- Only for SIMD circuits
- No known implementations

Main Question:

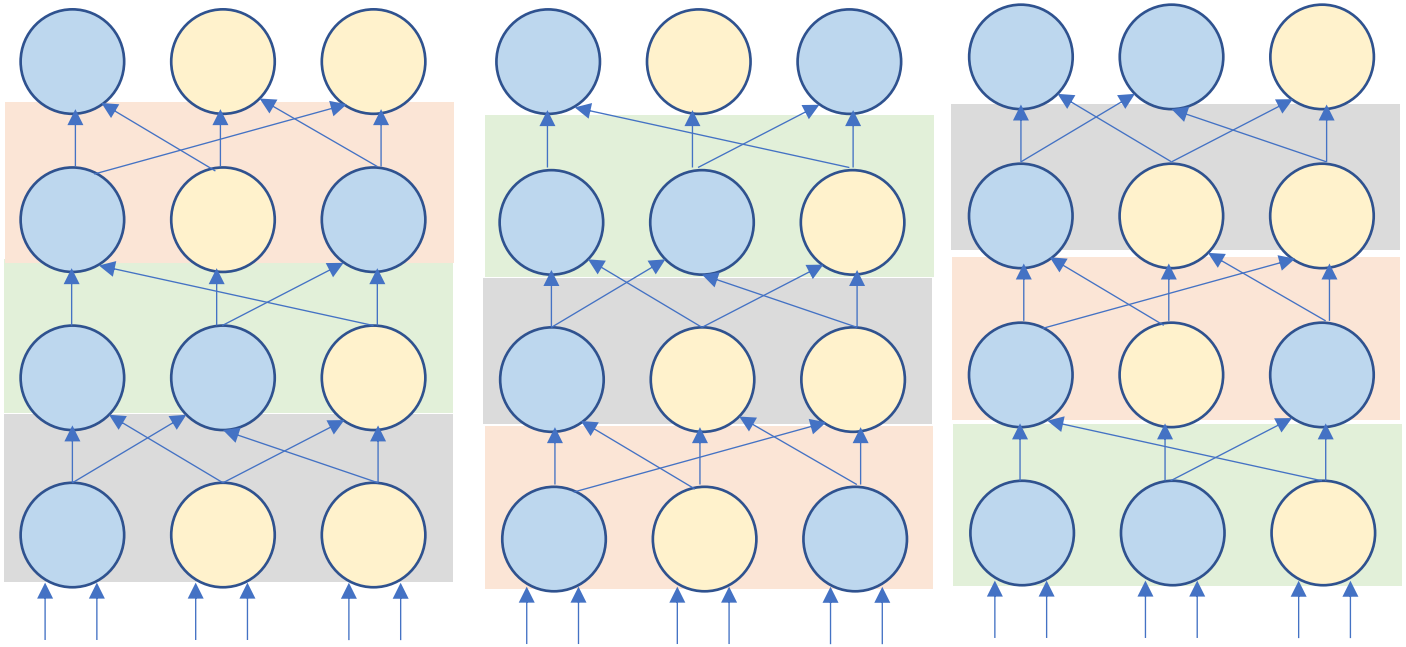
Can we design an $O(|C|)$ MPC protocol for a larger class of circuits?

Our Contributions

$O(|C|)$ MPC protocol for Highly Repetitive Circuits

- Semi-honest and maliciously secure protocols
- $t < n\left(\frac{1}{2} - \frac{2}{\varepsilon}\right)$ static corruptions
- Information theoretic
- No setup assumptions
- Security with Abort
- Supports “Division of labor”
- Provide Implementation - first implementation of MPC that uses packed secret sharing

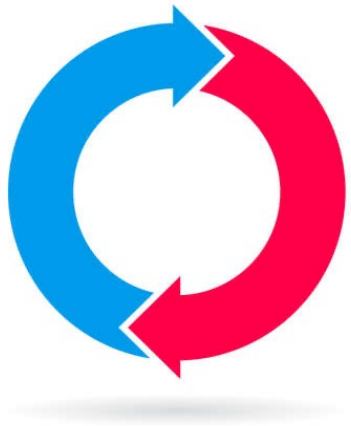
Defining Highly Repetitive Circuits



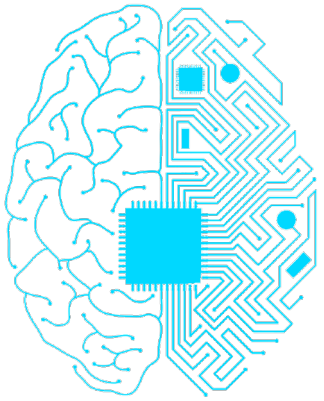
Example of (3,3)-repetitive circuit

(A, B)-Repetitive Circuits:	Composed of an arbitrary number of blocks of width at least A, that recur at least B times.
Highly Repetitive Circuits:	(A, B)- repetitive circuit is highly repetitive w.r.t. n parties, if $A \in \Omega(n)$ and $B \in \Omega(n)$.

Examples of Highly Repetitive Circuits

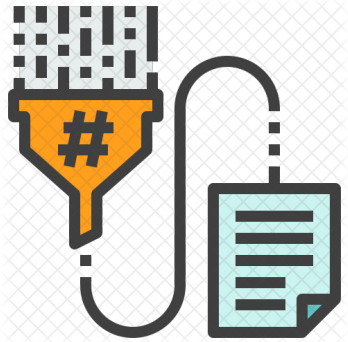


For/While Loops



Machine Learning

Block Ciphers



Cryptographic Hash Functions

$\tilde{O}(|C|)$ and $O(|C|)$ template

Existing Efficient $O(n|C|)$
Multiplication Protocols (eg. [DN07])

+

Packed Secret Sharing

=

Existing $\tilde{O}(|C|)$ and $O(|C|)$ Protocols

→ Correlated Randomness of form $[r]_t$, $[r]_{2t}$
Multiplication: mask, reconstruct, unmask

→ Vectorized/SIMD version of Shamir SS
over $O(|n|)$ sized vectors

Highly Repetitive Circuits

Existing Efficient $O(n|C|)$
Multiplication Protocols (eg. [DN07])

+

"Modified" Packed Secret Sharing
- Differing-operations PSS
- Realignment PSS

=

$O(|C|)$ MPC Protocol
For Highly Repetitive Circuits

Differing Operations PSS
- Parties choose operation $(+, \times)$ for each entry in the vector
- Realized by performing *both* operations and allowing the leader to *select* desired values
- Requires "special" correlated randomness

Realignment PSS
- Parties can efficiently swap elements within/between vectors
- Realized by having the leader permute entry order during before resharing and unmasking
- Requires "special" correlated randomness

Highly Repetitive Circuits required so special randomness generation is *amortized* $O(|C|)$

Conclusion

$O(|C|)$ MPC protocols for Highly Repetitive Circuits

ia.cr/2021/500