Generation of Authenticated Secret-Shared Scaled Unit Vectors for Beaver Triples

Vincent Rieder Bosch Research, University of Stuttgart

08.28.2024 Selected Areas of Cryptography (SAC), Montreal

Cloud Ready Privacy Preserving Technologies (CRYPTECS), French-German Publicly funded project

University of Stuttgart

SEC

Institute of Information Security

BOSCH

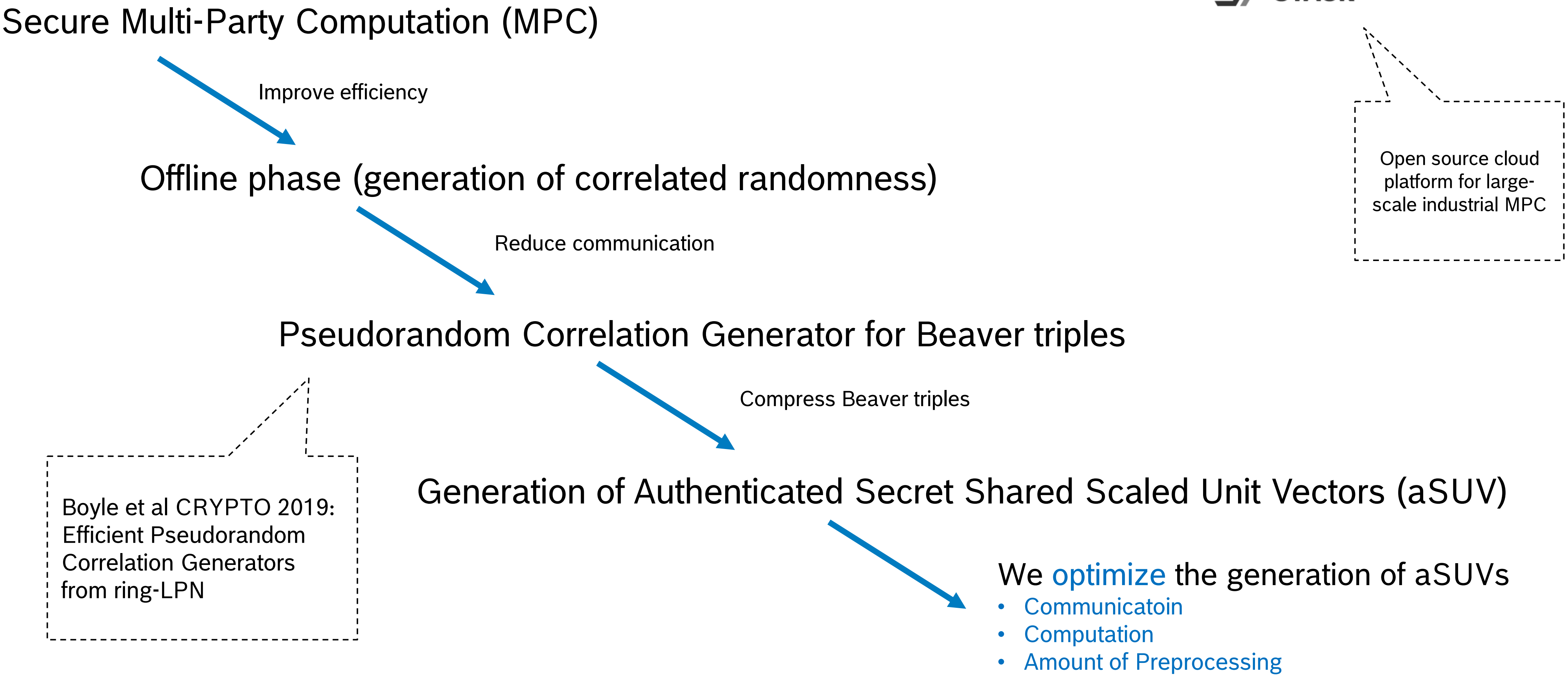
Context: Secure Multi Party Computation

Evaluate a public function on private inputs

- Two party setting
- Active malicous security

- Arithmetic Circuits
- Additive secret-sharing

MPC
Context



Secure Multi-Party Computation (MPC)

Improve efficiency

Offline phase (generation of correlated randomness)

Reduce communication

Pseudorandom Correlation Generator for Beaver triples

Compress Beaver triples

Generation of Authenticated Secret Shared Scaled Unit Vectors (aSUV)

We optimize the generation of aSUVs
- Communicatoin
- Computation
- Amount of Preprocessing

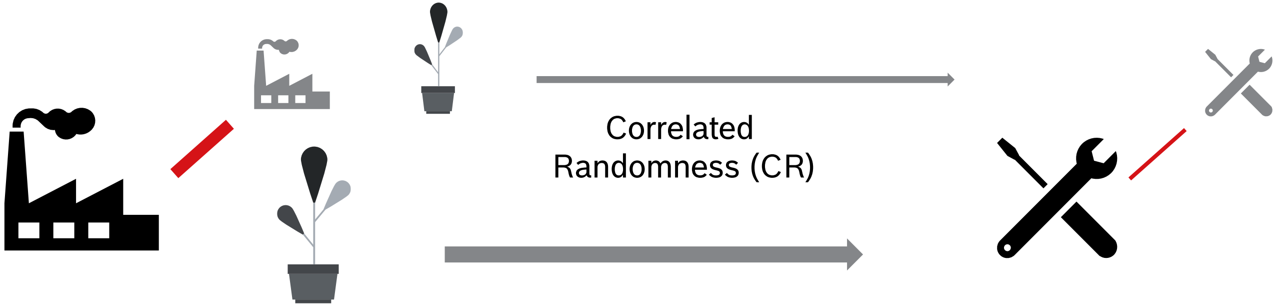
CARBYNE STACK

Open source cloud platform for large-scale industrial MPC

Boyle et al CRYPTO 2019: Efficient Pseudorandom Correlation Generators from ring-LPN

MPC
Preprocessing Model

— Communication channel



Offline Phase

» Input-independent generation of CR

» Heavyweight cryptographic protocols

» Massive communication

Online Phase

» Input-dependent secure function evaluation

» Lightweight cryptographic protocols

» Little communication

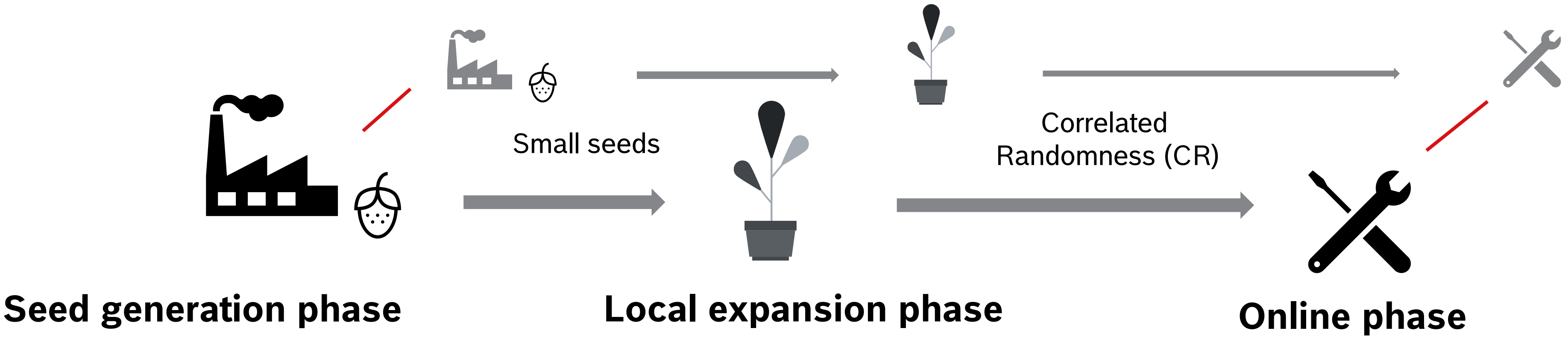
Active secure Beaver triple generation

» Tools: Homomorphic encryption, zero knowledge proofs, or oblivious transfer

» 100 MB worth Beaver triples take a few GB of communication

» Maturity: MP-SPDZ implements variety of protocols

MPC
Pseudorandom Correlation Generators (PCG)



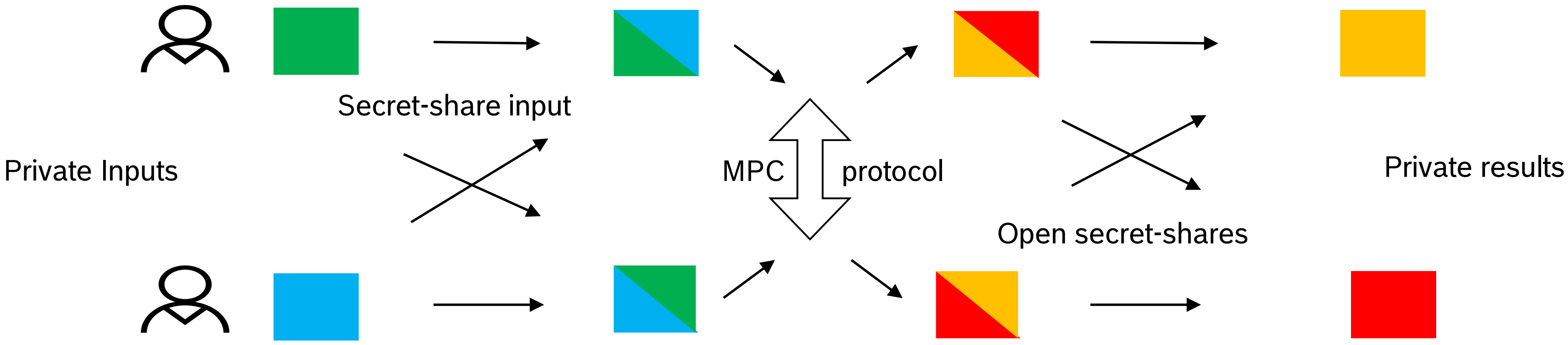
Active secure PCG for Beaver triples

- Boyle et al.: Pseudrandom Correlation Generation from ring-LPN
- Tools: PRGs, Distributed Point Functions, coding theoretic assumption
- 100 MB worth Beaver triples take a few MB of communication
- Maturity: One publication, conjectured efficiency

We optimize this protocol in preparation of an implementation

— Communication channel

MPC
Additive Secret Sharing



◣ reveals nothing about ⬜

$f($🟩, 🟦$) = ($🟨, 🟥$)$

For active security

Additive secret sharing: $[r] = r_0 + r_1$ where P_σ holds r_σ

Authenticated additive secret sharing: $[\![r]\!] = ([r], [r'])$ with MAC $r' = m \cdot r$

Beaver triple: $[\![a]\!], [\![b]\!], [\![c]\!]$ with a, b random, $c = a \cdot b$

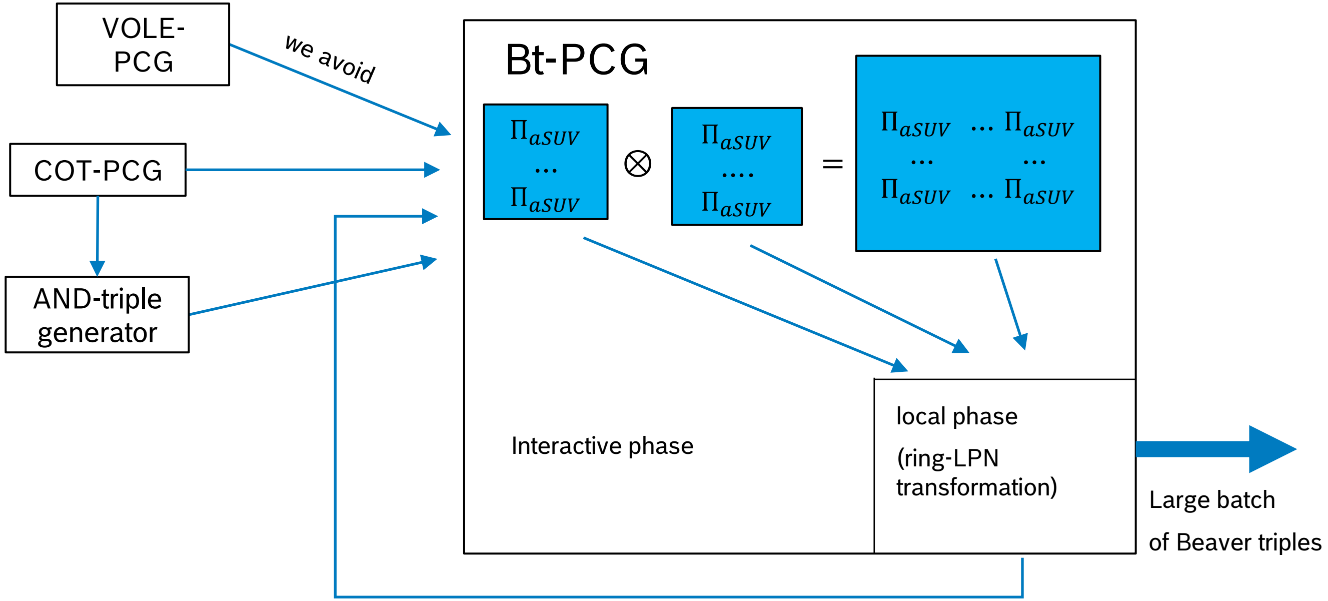
Supports one secure multiplication

Improved Generation of aSUVs

Overview
aSUV for Beaver triples

Large size but small description

A scaled unit vector is a vector $x \in \mathbb{F}^N$ which is zero except for one position and payload
→ SUV: *share* each coefficient with $[\cdot]$
→ aSUV: share each coefficient with *authentication* $[\![\cdot]\!]$

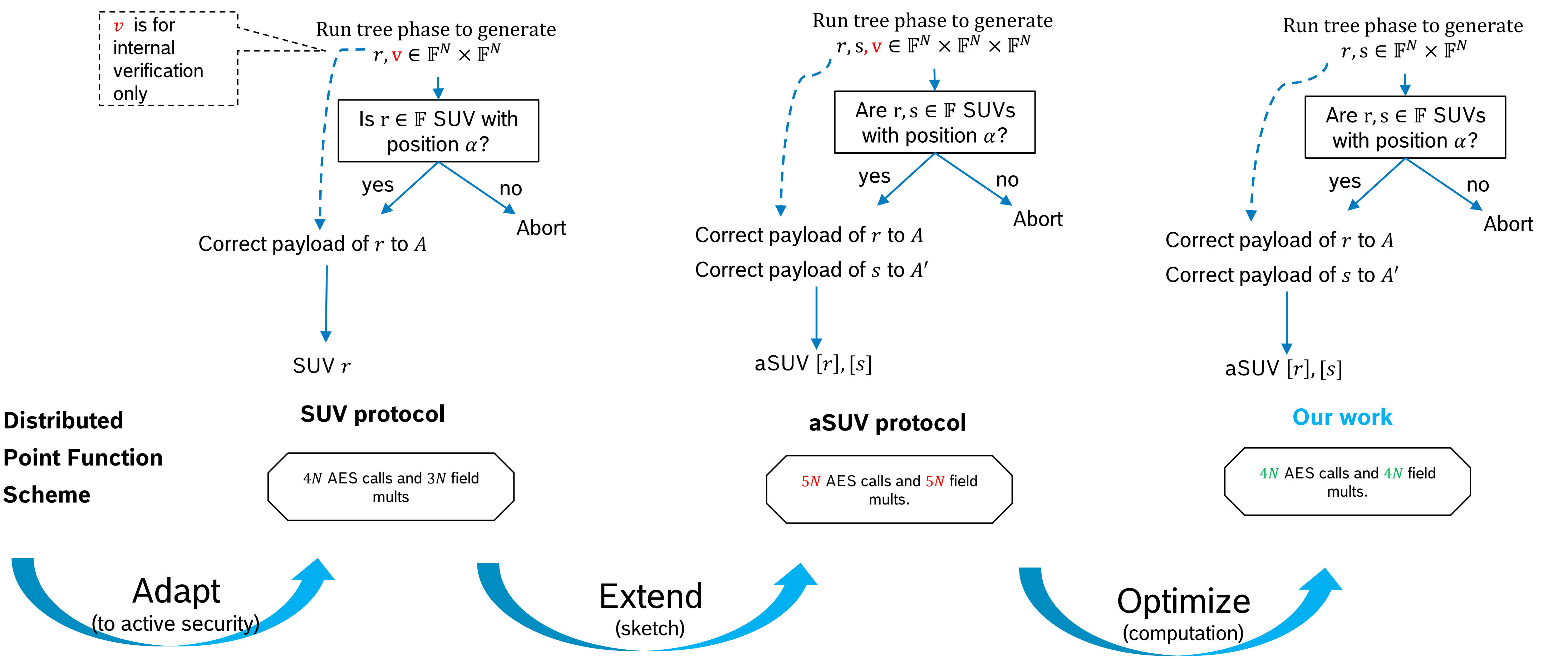

VOLE-PCG
we avoid
COT-PCG
AND-triple generator
Bt-PCG
Π_{aSUV} … Π_{aSUV}
$\otimes$
Π_{aSUV} …. Π_{aSUV}
=
Π_{aSUV} … Π_{aSUV} … … Π_{aSUV} … Π_{aSUV}
Interactive phase
local phase
(ring-LPN transformation)
Large batch
of Beaver triples

Optimizations of protocol Π_{aSUV}

1. aSUV at the computational costs of SUVs
2. Internal MPC with less interaction

Optimization 1.
Symmetric aSUV generation

Symmetry removes v



v is for internal verification only

Run tree phase to generate $r, v \in \mathbb{F}^N \times \mathbb{F}^N$

Is $\mathrm{r} \in \mathbb{F}$ SUV with position α?

yes

no

Abort

Correct payload of r to A

SUV r

Run tree phase to generate $r, \mathrm{s}, \mathrm{v} \in \mathbb{F}^N \times \mathbb{F}^N \times \mathbb{F}^N$

Are $\mathrm{r}, s \in \mathbb{F}$ SUVs with position α?

yes

no

Abort

Correct payload of r to A

Correct payload of s to A'

aSUV $[r], [s]$

Run tree phase to generate $r, s \in \mathbb{F}^N \times \mathbb{F}^N$

Are $\mathrm{r}, s \in \mathbb{F}$ SUVs with position α?

yes

no

Abort

Correct payload of r to A

Correct payload of s to A'

aSUV $[r], [s]$

Distributed Point Function Scheme

SUV protocol

$4N$ AES calls and $3N$ field mults

aSUV protocol

$5N$ AES calls and $5N$ field mults.

Our work

$4N$ AES calls and $4N$ field mults.

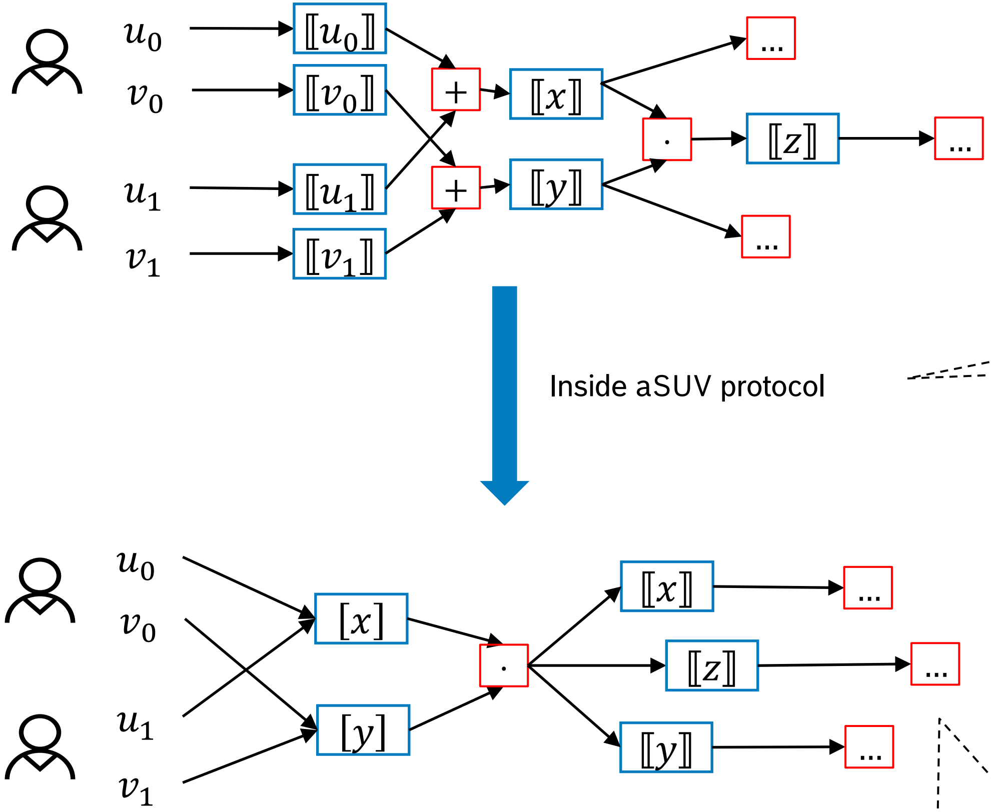
Adapt (to active security)

Extend (sketch)

Optimize (computation)

Optimization 2.

Special purpose MPC circuit



·

1. Beaver triple $[\![a]\!], [\![b]\!], [\![c]\!]$
2. Reveal $\epsilon = ([\![x]\!] - [\![a]\!]),\ \delta = ([\![y]\!] - [\![b]\!])$
3. Locally compute $[\![z]\!] = \delta \cdot [\![x]\!] + \epsilon \cdot [\![y]\!] + [\![c]\!] - \epsilon \cdot \delta$

... Requires previous optimization
... Avoids *Input* steps (communication + correlated randomness)
... Requires careful security proof

·

1. Beaver triple $[\![a]\!], [\![b]\!], [\![c]\!]$
2. Exchange $\epsilon = (x - a),\ \delta = (y - b)$
3. Locally compute $[\![x]\!] = \epsilon + [\![a]\!],\ [\![y]\!] = \delta + [\![b]\!]$
4. Locally compute $[\![z]\!] = \delta \cdot [\![x]\!] + \epsilon \cdot [\![y]\!] + [\![c]\!] - \epsilon \cdot \delta$

Z is a check value that either gives an accept or causes an abortion

Evaluation

Interactive Phase of Bt-PCG

	AES calls in million	Field mult in million	Amount of CR in KB	Communication in KB
Π_{aSUV} Boyle	1,3	1,3	1,0	1,0
Π_{aSUV} improved	1,0	1,0	0,7	0,9
	20%	20%	30%	12%
Π_{Bt} Boyle	4299	4299	4542	1730
Π_{Bt} improved	3439	3418	4043	1537
	20%	20%	11%	11%

- $2^{20} \approx$ 1Mio Beaver triples (100 MB)
- 128-bit field / security

SPDZ style protocols require a few GB of communication

Support online phase for a few seconds

Future work

- Parameter selection
- Local phase and preprocessing
- Implementation
- Generalizations

Results will be published soon