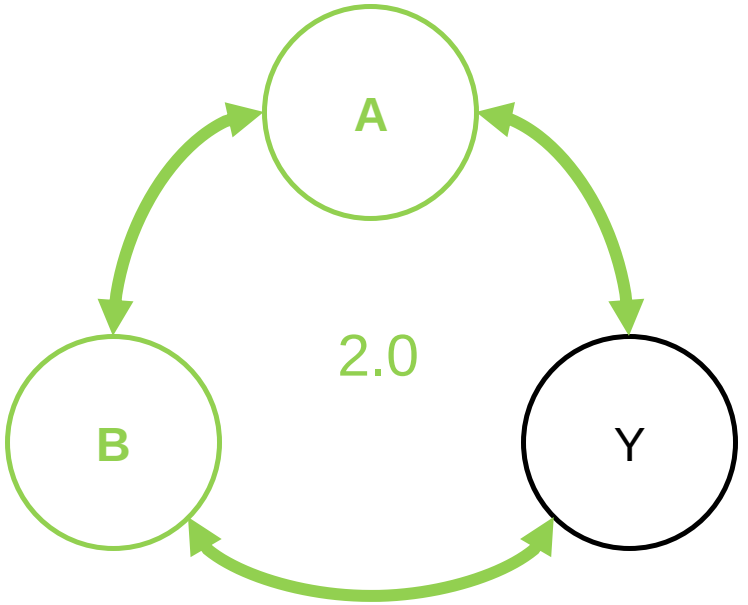
ABY2.0: Improved Mixed-Protocol Secure Two-Party Computation

Arpita Patra (IISc Bangalore), Thomas Schneider (TU Darmstadt), Ajith Suresh (IISc Bangalore), Hossein Yalame (TU Darmstadt)



Secure Two-party Computation (2PC)

Jasmine

How many diamonds do we have in common with the same color ?

Aladdin

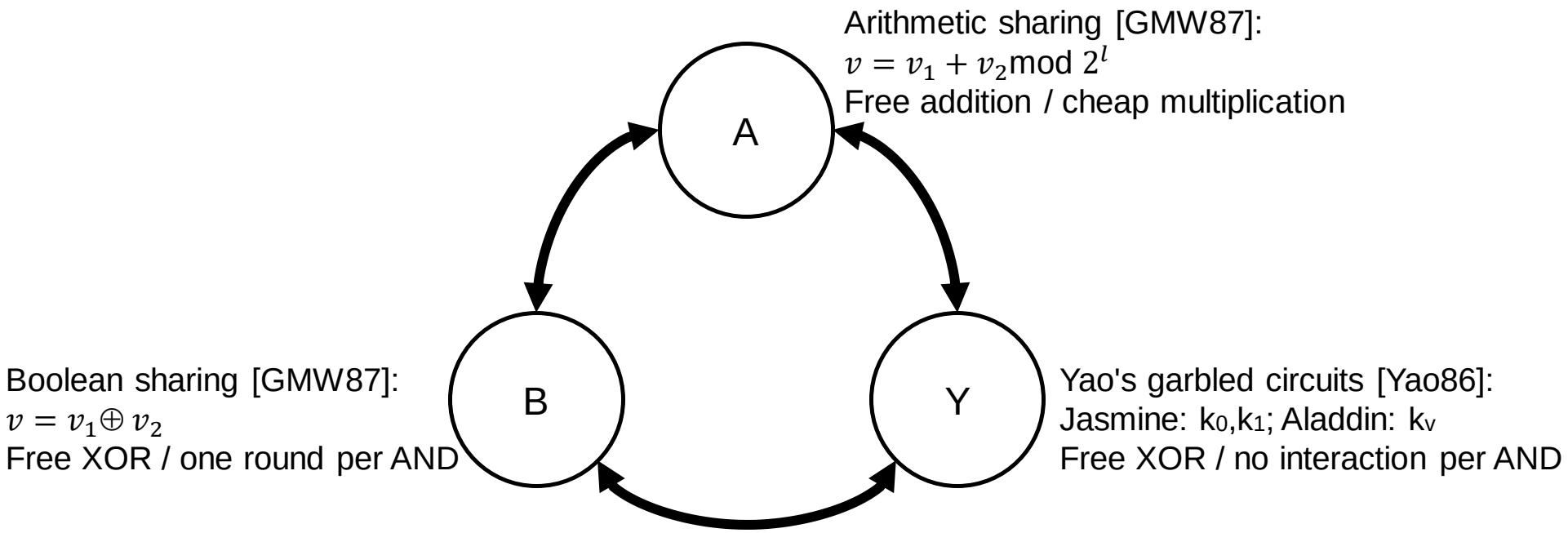
Secure Two-party Computation (2PC)

Jasmine

Introduced by Andrew C Yao [Yao86]
Allows two mutually distrusting parties to securely compute a joint function on their private inputs.

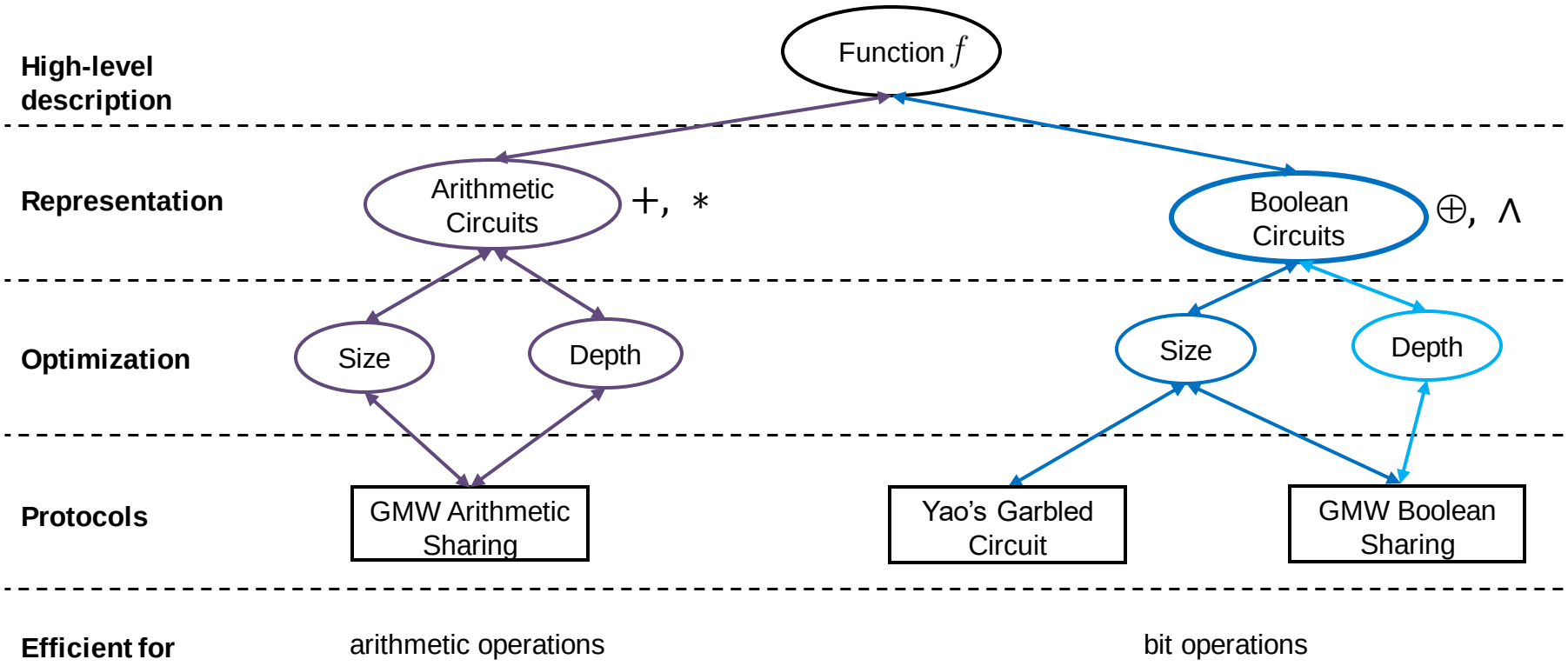
Aladdin

State-of-the-art Framework: ABY [DSZ15]

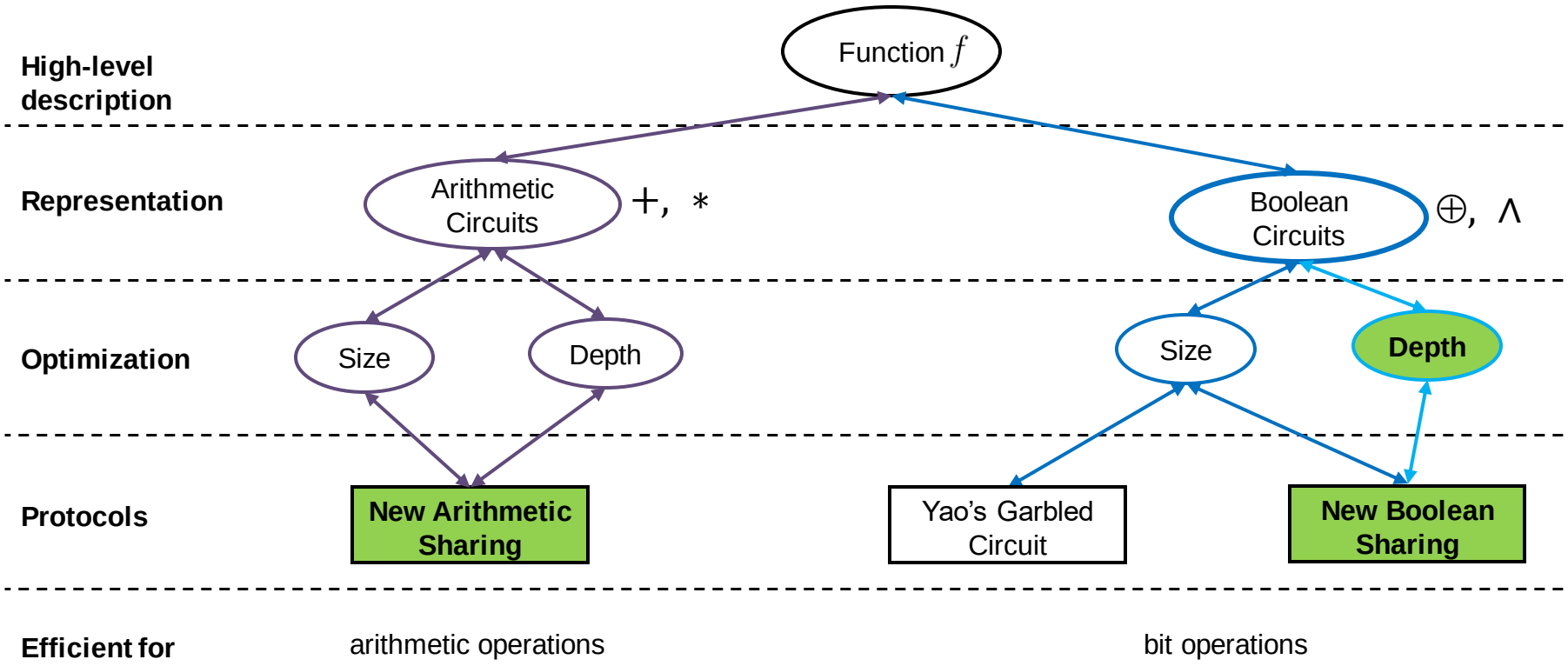


encrypto.de/code/

2PC in ABY

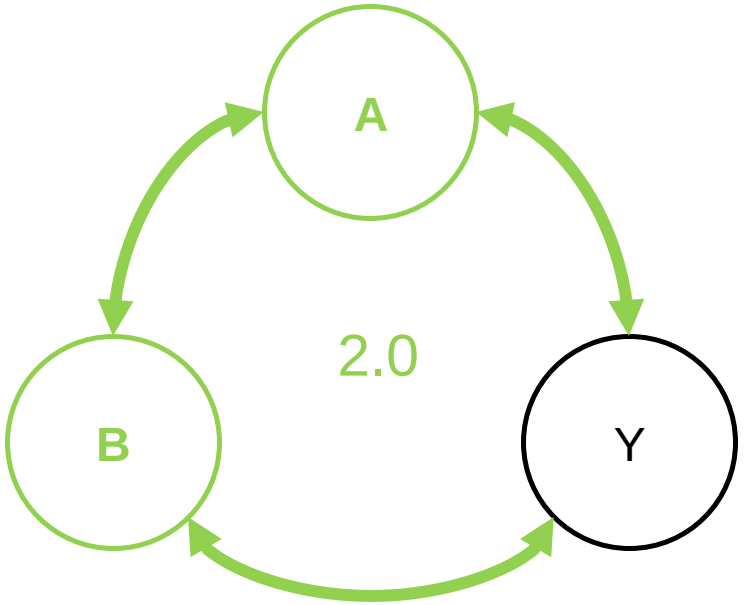

High-level description
Function f
Representation
Arithmetic Circuits
$+, *$
Boolean Circuits
$\oplus, \wedge$
Optimization
Size
Depth
Size
Depth
Protocols
GMW Arithmetic Sharing
Yao's Garbled Circuit
GMW Boolean Sharing
Efficient for
arithmetic operations
bit operations

2PC in ABY2.0



Our Contributions

- Passively secure **2PC protocol** over rings with function-dependent preprocessing



Our Contributions

- Passively secure 2PC protocol over rings with **function-dependent preprocessing**
  - Was function-independent in ABY
  - Improved online communication

Our Contributions

- Passively secure 2PC protocol over rings with **function-dependent preprocessing**
  - Was function-independent in ABY
  - Improved online communication
- Support for **Multi-input** Multiplication
  - Constant online communication / efficient building blocks

Our Contributions

- Passively secure 2PC protocol over rings with **function-dependent preprocessing**
  - Was function-independent in ABY
  - Improved online communication
- Support for **Multi-input Multiplication**
  - Constant online communication / efficient building blocks
- Often more efficient **Mixed World** Conversions

Our Contributions

- Passively secure 2PC protocol over rings with **function-dependent preprocessing**
  - Was function-independent in ABY
  - Improved online communication
- Support for **Multi-input multiplication**
  - Constant online communication / efficient building blocks
- Often more efficient **Mixed World** Conversions
- Special tools for **Privacy Preserving Machine Learning (PPML)**
  - Scalar Product with online complexity independent of the dimension

New Sharing in ABY2.0: $\Delta_v = v + \delta_v$

Sharing of a value v:

The general idea is from the 3PC ASTRA [CCPS19]

New Sharing in ABY2.0: $\Delta_v = v + \delta_v$

Sharing of a value v:
mask for value $= \delta_v$

The general idea is from the 3PC ASTRA [CCPS19]

New Sharing in ABY2.0: $\Delta_v = v + \delta_v$

Sharing of a value v:
mask for value $= \delta_v$
masked value $= \Delta_v$

$$\Delta_v = v + \delta_v$$

The general idea is from the 3PC ASTRA [CCPS19]

New Sharing in ABY2.0: $\Delta_v = v + \delta_v$

Δ_v, δ_{v_1}

Sharing of a value v:
mask for value $= \delta_v$
masked value $= \Delta_v$

$$\Delta_v = v + \delta_v$$
$$\delta_v = \delta_{v_1} + \delta_{v_2}$$

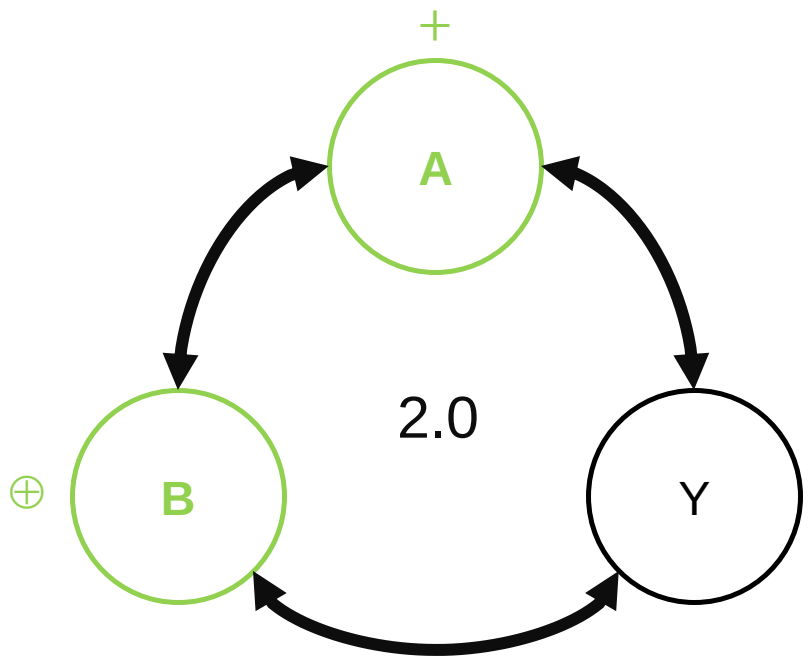
Δ_v, δ_{v_2}

The general idea is from the 3PC ASTRA [CCPS19]

Addition in ABY2.0

$(\Delta_x, \delta_{x_1}), (\Delta_y, \delta_{y_1})$

Addition: $z = x + y$



$(\Delta_x, \delta_{x_2}), (\Delta_y, \delta_{y_2})$

Addition in ABY2.0

$(\Delta_x, \delta_{x_1}), (\Delta_y, \delta_{y_1})$

$$\text{Addition: } z = x + y$$
$$= (\Delta_x - \delta_x) + (\Delta_y - \delta_y)$$

$(\Delta_x, \delta_{x_2}), (\Delta_y, \delta_{y_2})$

Addition in ABY2.0

$(\Delta_x, \delta_{x_1}), (\Delta_y, \delta_{y_1})$

$$\text{Addition: } z = x + y$$
$$= (\Delta_x - \delta_x) + (\Delta_y - \delta_y)$$
$$= (\Delta_x + \Delta_y) - (\delta_x + \delta_y)$$
$$= \Delta_z - \delta_z$$

$(\Delta_x, \delta_{x_2}), (\Delta_y, \delta_{y_2})$

Addition in ABY2.0

$(\Delta_x, \delta_{x_1}), (\Delta_y, \delta_{y_1})$

Addition: $z = x + y$

$= (\Delta_x - \delta_x) + (\Delta_y - \delta_y)$

$= (\Delta_x + \Delta_y) - (\delta_x + \delta_y)$

$= \Delta_z - \delta_z$

$(\Delta_x, \delta_{x_2}), (\Delta_y, \delta_{y_2})$

$\Delta_z = \Delta_x + \Delta_y$

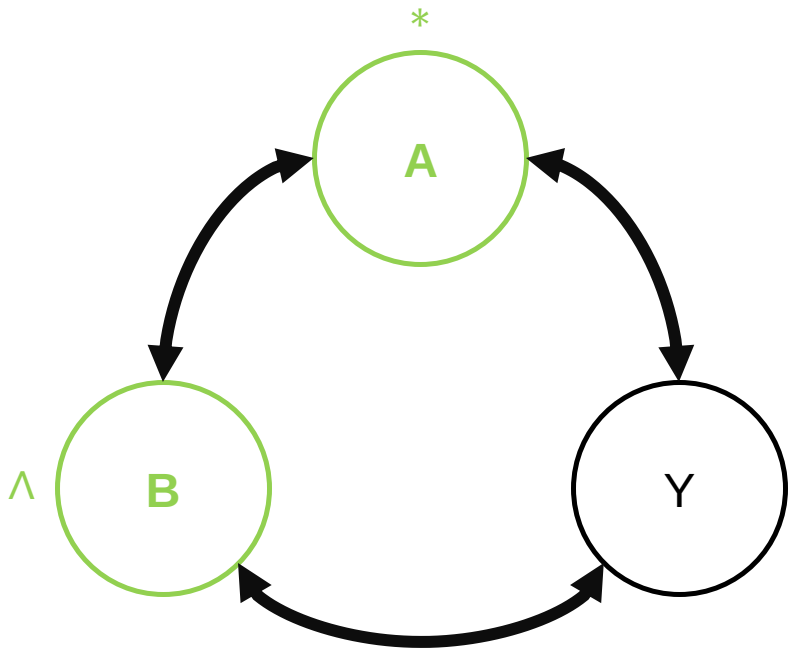
$\delta_{z_1} = \delta_{x_1} + \delta_{y_1}$

$\Delta_z = \Delta_x + \Delta_y$

$\delta_{z_2} = \delta_{x_2} + \delta_{y_2}$

Multiplication in ABY2.0

Multiplication: z = xy



Multiplication in ABY2.0

Multiplication: $z = xy$

$= (\Delta_x - \delta_x) + (\Delta_y - \delta_y)$

Multiplication in ABY2.0

Multiplication: z = xy

$$= (\Delta_x - \delta_x) + (\Delta_y - \delta_y)$$

$$= \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y$$

Multiplication in ABY2.0

Multiplication: $z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y$

$$\Delta_z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y + \delta_z$$

Multiplication in ABY2.0

Multiplication: $z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y$

$$\Delta_z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y + \delta_z$$

$(\Delta_x, \delta_{x_1}), (\Delta_y, \delta_{y_1}), \delta_{z_1}$

$(\Delta_x, \delta_{x_2}), (\Delta_y, \delta_{y_2}), \delta_{z_2}$

Multiplication in ABY2.0

Multiplication: $z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y$

$$\Delta_z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y + \delta_z$$

$(\Delta_x, \delta_{x_1}), (\Delta_y, \delta_{y_1}), \delta_{z_1}$

$(\Delta_x, \delta_{x_2}), (\Delta_y, \delta_{y_2}), \delta_{z_2}$

Beaver Triple

$(\delta_x \delta_y)_1 + (\delta_x \delta_y)_2 = (\delta_{x_1} + \delta_{x_2})(\delta_{y_1} + \delta_{y_2})$

$(\delta_x \delta_y)_1$

$(\delta_x \delta_y)_2$

Multiplication in ABY2.0

Multiplication: $z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y$

$$\Delta_z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y + \delta_z$$

$(\Delta_x, \delta_{x_1}), (\Delta_y, \delta_{y_1}), \delta_{z_1}$

$(\Delta_x, \delta_{x_2}), (\Delta_y, \delta_{y_2}), \delta_{z_2}$

Beaver Triple

$(\delta_x \delta_y)_1 + (\delta_x \delta_y)_2 = (\delta_{x_1} + \delta_{x_2})(\delta_{y_1} + \delta_{y_2})$

$(\delta_x \delta_y)_1$

$(\delta_x \delta_y)_2$

$\Delta_{z_1} = \Delta_x \Delta_y - \Delta_x \delta_{y_1} - \Delta_y \delta_{x_1} + (\delta_x \delta_y)_1$

$\Delta_{z_2} = -\Delta_x \delta_{y_2} - \Delta_y \delta_{x_2} + (\delta_x \delta_y)_2$

Δ_{z_1} →

← Δ_{z_2}

Multiplication in ABY2.0

Multiplication: $z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y$

$$\Delta_z = \Delta_x \Delta_y - \Delta_x \delta_y - \Delta_y \delta_x + \delta_x \delta_y + \delta_z$$

$(\Delta_x, \delta_{x_1}), (\Delta_y, \delta_{y_1}), \delta_{z_1}$

$(\Delta_x, \delta_{x_2}), (\Delta_y, \delta_{y_2}), \delta_{z_2}$

Beaver Triple

$$(\delta_x \delta_y)_1 + (\delta_x \delta_y)_2 = (\delta_{x_1} + \delta_{x_2})(\delta_{y_1} + \delta_{y_2})$$

$(\delta_x \delta_y)_1$

$(\delta_x \delta_y)_2$

$\Delta_{z_1} = \Delta_x \Delta_y - \Delta_x \delta_{y_1} - \Delta_y \delta_{x_1} + (\delta_x \delta_y)_1$

$\Delta_{z_2} = -\Delta_x \delta_{y_2} - \Delta_y \delta_{x_2} + (\delta_x \delta_y)_2$

Δ_{z_1} →

← Δ_{z_2}

$\Delta_z = \Delta_{z_1} + \Delta_{z_2}$

$\Delta_z = \Delta_{z_1} + \Delta_{z_2}$

Multiplication in ABY2.0

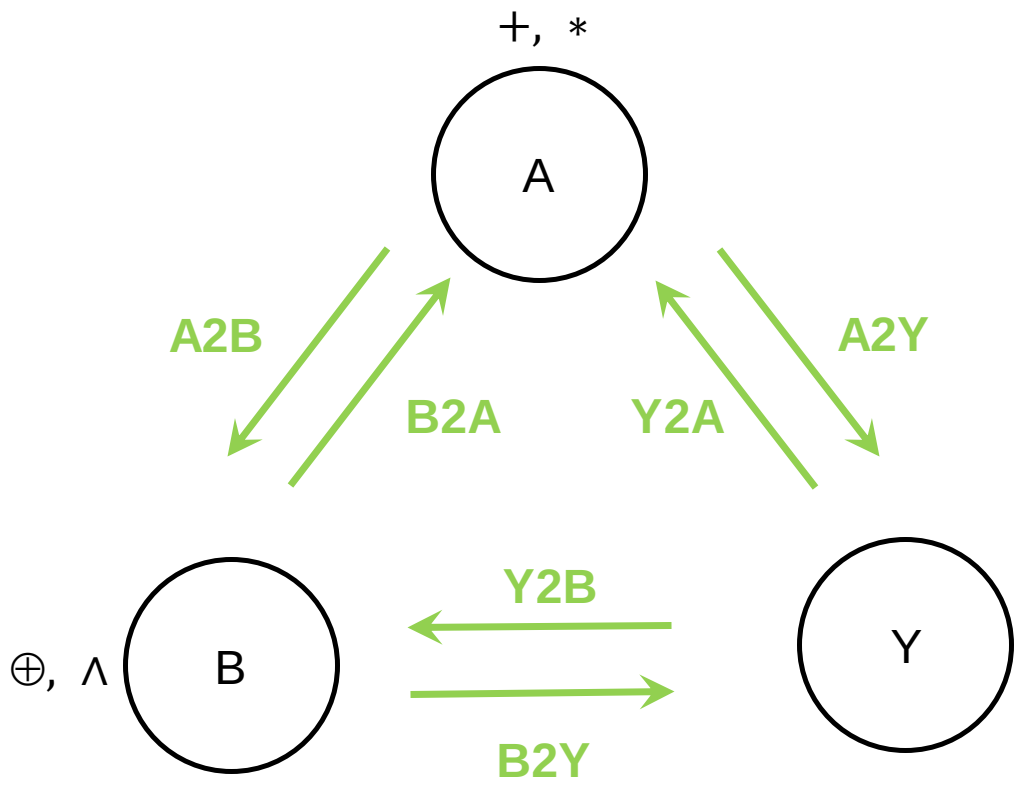
	Pre-processing Communication		Online Communication (#elements)
ABY [DSZ15]	|Triples|	Function-independent	4
ABY2.0 (This Work)	|Triples|	Function-dependent	**2**

|Triples|: cost for generating one Beaver triple via OT or HE

Multi-Input Multiplication in ABY2.0

		Pre-processing Communication	Online Communication (#elements)	
3-input Multiplication	ABY [DSZ15]	2·|Triples|	8	
	ABY2.0 (This Work)	4·|Triples|	**2**	⟵Independent of fan-in!
4-input Multiplication	ABY [DSZ15]	3·|Triples|	12	
	ABY2.0 (This Work)	11·|Triples|	**2**	⟵Independent of fan-in!

New Mixed World Conversions



Mixed World Conversions

Conversion		Online Communication [bits]	Rounds
Y2B	ABY	**0**	**0**
	ABY2.0	l	1
B2Y	ABY	$lk + l$	2
	ABY2.0	$\boldsymbol{lk}$	**1**
A2Y	ABY	$2lk + l$	2
	ABY2.0	$\boldsymbol{lk}$	**1**
Y2A	ABY	$(l^2 + 3l)/2$	2
	ABY2.0	$\boldsymbol{l}$	**1**
A2B	ABY	$2lk + l$	2
	ABY2.0	$\boldsymbol{lk + l}$	**2**
B2A	ABY	$(l^2 + l)/2$	2
	ABY2.0	$\boldsymbol{2l}$	**1**

l: bitlength of numbers
k: Symmetric security parameter

Dot Product

$$X \cdot Y = \sum_{i=1}^{d} x_i \cdot y_i$$

	Pre-processing Communication	Online Communication (#elements)
ABY	d·|Triples|	4d
ABY2.0	d·|Triples|	**2** ← Independent of dimension!

d: dimension of vector
|Triples|: cost for generating one Beaver triple

Benchmarking and Applications

- Implemented ABY2.0 using the ENCRYPTO library
- Protocols benchmarked over LAN (25 Gbps) and WAN (75 Mbps) with Google Cloud Platform
- Servers located in East Australia and South East Asia
- Average RTT is 0.056 ms for LAN and 60.19 ms for WAN

- Maxpool
- Improved AES S-Box
- Circuit-based Private Set Intersection
- Minimum Euclidean Distance
- Biometric Matching
- Privacy-preserving Machine Learning

PPML (LR Inference)

	Runtime (ms)	
	LAN	WAN
SecureML [MZ17]	1.69	504.96
ABY2.0	**0.29**	**308.16**
Improvement	**5.6x**	**1.6x**

	Throughput (queries/min)	
	LAN	WAN
SecureML [MZ17]	1193	3.58
ABY2.0	**42371**	**39.88**
Improvement	**35.5x**	**11.1x**

Over Gisette dataset with 5000 features and up to 1,000,000 samples

PPML (NN Inference)

	Runtime (ms)	
	LAN	WAN
SecureML [MZ17]	8.77	1760
ABY2.0	**2.66**	**744**
Improvement	**3.3x**	**2.4x**

	Throughput (queries/min)	
	LAN	WAN
SecureML [MZ17]	40.89	0.12
ABY2.0	**30,795.17**	**91.57**
Improvement	**753x**	**763x**

Dot Product in ABY2.0

Two hidden layers with 128 neurons each, and output of a vector with 10 elements on the MINIST dataset

PPML (NN Training)

	Throughput (it/min)	
	LAN	WAN
SecureML	3323	15
ABY2.0	**9146**	**42**
Improvement	**2.8x**	**2.8x**

Two hidden layers with 128 neurons in each layer on the MNIST dataset

Conclusion

- ✓ New 2PC protocol for securely evaluating a circuit over a ring
- ✓ Better mixed protocol conversions w.r.t. both rounds and online communication
- ✓ Constant online cost of 2 ring elements for N-input AND gates
- ✓ Set of efficient building blocks
  - ✓ Scalar product: online communication independent of the vector dimension
  - ✓ Round efficient adder using a combination of {2,3,4}-input AND gates
  - ✓ Matrix multiplication, equality test, comparison, bit extraction…

Thank You!

encrypto.de/yalame

Bibliography

[CCPS19] Harsh Chaudhari, Ashish Choudhury, Arpita Patra, and Ajith Suresh. **ASTRA: High Throughput 3PC over Rings with Application to Secure Prediction.** In CCSW, 2019

[DSZ15] Daniel Demmler, Thomas Schneider, and Michael Zohner. **ABY– A Framework for Efficient Mixed Protocol Secure Two-party Computation.** In NDSS, 2015

[GMW87] Oded Goldreich, Silvio Micali, and Avi Wigderson. **How to Play Any Mental Game.** In STOC, 1987

[MZ17] Payman Mohassel and Yupeng Zhang. **SecureML: A System for Scalable Privacy Preserving Machine Learning.** In IEEE S&P, 2017

[Yao86] Andrew Chi-Chih Yao. **How to Generate and Exchange Secrets**. In FOCS, 1986