Efficient Three-party Boolean-to-Arithmetic Share Conversion

Nan Cheng[1] Feng Zhang[2] Katerina Mitrokotsa[1]

[1]University of St. Gallen, Switzerland

[2]Nanyang Institute of Technology, China

June 14, 2024

University of St.Gallen

Outline

- Background.
- SOTA.
- Our ideas.
- Evaluation.

Secret Sharing (SS)

A **secret sharing** over x is denoted $[x]$.

$$([x]_0, [x]_1) \leftarrow \mathsf{SS.Share}(x)$$

$$x \leftarrow \mathsf{SS.Reveal}([x]_0, [x]_1)$$

Sharing types

Given a secret $x \in \{0,1\}^n$

- Boolean Secret Sharing: denoted with $[x]^{\mathsf{B}}$.

$$[x_1]^{\mathsf{B}}, \cdots, [x_n]^{\mathsf{B}}$$

- Arithmetic Secret Sharing: denoted with $[x]^{\mathsf{A}}$.

$$[x]_0^{\mathsf{A}}, [x]_1^{\mathsf{A}}$$

A concrete example

Given a secret $x = x_1 \| x_2 = 1 \| 0$, share it among two parties $\mathsf{P}_0, \mathsf{P}_1$.

- $[x]^{\mathsf{B}}$:

	$[x_1]^{\mathsf{B}}$	$[x_2]^{\mathsf{B}}$
P_0	1	1
P_1	0	1

- $[x]^{\mathsf{A}}$ in $\mathbb{Z}_{2^3}$:

	$[x]^{\mathsf{A}}$
P_0	4
P_1	6

Boolean-to-Arithmetic share conversion

Definition

- Bit2A: $\forall b \in \{0,1\}, [b]^{\mathsf{A}} \leftarrow \text{Bit2A}([b]^{\mathsf{B}})$.
- B2A: $\forall x \in \{0,1\}^n, [x]^{\mathsf{A}} \leftarrow \text{B2A}([x]^{\mathsf{B}})$.

Why Bit2A/B2A?

- Essential in mixed protocol MPC frameworks, like in ABY, ABY2, ABY3.
- Verifiable secure aggregation, *e.g.*, Prio+.

Replicated three-party Secret Sharing (RSS)

Definition

The replicated **secret sharing** over $b \in \{0,1\}$ is denoted as $[\![x]\!]$.

$$(b_0, b_1), (b_1, b_2), (b_2, b_0) \leftarrow \mathrm{RSS.Share}(b)$$

$$b \leftarrow \mathrm{RSS.Reveal}(b_0, b_1, b_2)$$

	P_0	P_1	P_2	
$[\![b]\!]^{\mathrm{B}}$	(b_0, b_1)	(b_1, b_2)	(b_2, b_0)	$b = b_0 \oplus b_1 \oplus b_2$
$[\![b]\!]^{\mathrm{A}}(\mathbb{Z}_{2^\ell})$	(b_0, b_1)	(b_1, b_2)	(b_2, b_0)	$b = \sum_{i=0}^{\ell-1} b_i \mod 2^\ell$

A recap of Bit2A in ABY3

The three-party Oblivious transfer

	P_0 (m_0, m_1)	P_1 b	P_2 b
OT	ϵ	m_b	-
3P-OT	ϵ	m_b	ϵ

- Offline: P_0 and P_2 agree upon $(\mathrm{OTP}_0, \mathrm{OTP}_1)$.
- Online:
  1. P_0 sends (c_0, c_1) to P_1 where
  $$(c_0, c_1) = (\mathrm{OTP}_0 \oplus m_0, \mathrm{OTP}_1 \oplus m_1).$$
  2. P_2 sends OTP_b to P_1.
  3. P_1 computes $m_b = c_b \oplus \mathrm{OTP}_b$.

How is Bit2A realized in ABY3?

	P_0	P_1	P_2
$[\![b]\!]^{\mathrm{B}}$ PRG seeds	(b_0, b_1) (S_0, S_1)	(b_1, b_2) (S_1, S_2)	(b_2, b_0) (S_2, S_0)
-	(r_0, r_1)	$(r_1, -)$	$(-, r_0)$
-	$\begin{Bmatrix} m_0 = (b_0 \oplus b_1) - r_0 - r_1 \\ m_1 = (1 \oplus b_0 \oplus b_1) - r_0 - r_1 \end{Bmatrix}$	b_2	b_2

- $\mathrm{P}_0, \mathrm{P}_1, \mathrm{P}_2$ invokes the 3P-OT protocol, such that P_1 obtains $m_{b_2} = b - r_0 - r_1$.
- $\mathrm{P}_0, \mathrm{P}_2, \mathrm{P}_1$ invokes the 3P-OT protocol, such that P_2 obtains $m_{b_2} = b - r_0 - r_1$.

Compare our work to ABY3

	Ref.	Comm. volume	Comm. rounds	Compu. (PRG calls, #Mod Operations)
Offline	GC	-	-	-
	LSS	-	-	-
	FHE	-	-	-
	HSS	-	-	-
Online	GC	*	*	*
	LSS	*	*	*
	FHE	*	*	*
	HSS	*	*	*

	Ref.	Comm.[bits] Setup	Online	Total	Online Rounds
Bit2A	ABY3	0	2ℓ	2ℓ	1
	Our Π_1	$4\ell/3$	1	$4\ell/3+1$	1
B2A	ABY3	-	$\ell+\ell\log\ell$	-	$1+\log\ell$
	Our Π_2	$\ell^2/3$	$5\ell/3$	$(\ell^2+5\ell)/3$	2

Less online comm.

Less comm. in total

	Ref.	Comm.[bits] Setup	Online	Total	Online Rounds
Bit2A	ABY3	0	2ℓ	2ℓ	1
	Our Π_1	$4\ell/3$	1	$4\ell/3+1$	1
B2A	ABY3	-	$\boxed{\ell+\ell\log\ell}$	-	$\boxed{1+\log\ell}$
	Our Π_2	$\ell^2/3$	$\boxed{5\ell/3}$	$(\ell^2+5\ell)/3$	$\boxed{2}$

Less online comm.

Less online rounds

Let's understand the relationship between b, γ, and θ. $\forall \theta, \gamma, b \in \{0,1\}$, define:

- $v = (-1)^{\theta} \cdot \gamma$
- $\beta = b \oplus \gamma$
- $\sigma = b \oplus \gamma \oplus \theta$

Now, it turns out:

$$b = (-1)^{\sigma} \cdot v + \beta.$$

Technical Overview

Let's understand the relationship between b, γ, and θ. $\forall \theta, \gamma, b \in \{0,1\}$, define:

- $v = (-1)^{\theta} \cdot \gamma$
- $\beta = b \oplus \gamma$
- $\sigma = b \oplus \gamma \oplus \theta$

Now, it turns out:

$$b = (-1)^{\sigma} \cdot v + \beta.$$

Breaking this down...

$$\begin{aligned}(-1)^{\sigma} \cdot v + \beta &= (-1)^{\sigma+\theta} \cdot \gamma + \beta \\ &= (-1)^{b \oplus \gamma} \cdot \gamma + (b \oplus \gamma)\end{aligned}$$

Let's understand the relationship between b, γ, and θ. $\forall \theta, \gamma, b \in \{0,1\}$, define:

- $v = (-1)^{\theta} \cdot \gamma$
- $\beta = b \oplus \gamma$
- $\sigma = b \oplus \gamma \oplus \theta$

Now, it turns out:

$$b = (-1)^{\sigma} \cdot v + \beta.$$

Breaking this down...

$$\begin{aligned}(-1)^{\sigma} \cdot v + \beta &= (-1)^{\sigma+\theta} \cdot \gamma + \beta \\ &= (-1)^{b \oplus \gamma} \cdot \gamma + (b \oplus \gamma)\end{aligned}$$

So, what does this mean?

- When b matches γ: $b \oplus \gamma = 0$, so our equation simplifies to b.
- When b is the opposite of γ: The equation still simplifies to b!

Let's understand the relationship between b, γ, and θ. $\forall \theta, \gamma, b \in \{0,1\}$, define:

- $v = (-1)^{\theta} \cdot \gamma$
- $\beta = b \oplus \gamma$
- $\sigma = b \oplus \gamma \oplus \theta$

Now, it turns out:

$$b = (-1)^{\sigma} \cdot v + \beta.$$

Which implies...

$$[\![b]\!]^{\mathsf{A}} = (-1)^{\sigma} \cdot [\![v]\!]^{\mathsf{A}} + [\![\beta]\!]^{\mathsf{A}}$$

So, what does this mean?

- Prepare $\gamma, \theta \in \{0,1\}$, compute $[\![v]\!]^{\mathsf{A}}$ in the offline phase.
- Compute β, σ, and $[\![b]\!]^{\mathsf{A}} = (-1)^{\sigma} \cdot [\![v]\!]^{\mathsf{A}} + [\![\beta]\!]^{\mathsf{A}}$ in the online phase.

Offline phase

Prepare $\gamma, \theta \in \{0,1\}$, compute $[\![v]\!]^{\mathsf{A}}$.

	P_0	P_1	P_2	
Input	γ	θ	θ	$\gamma, \theta \in \{0,1\}$
Round-1	0	r_0	$r_1 = \gamma - r_0$	$r_0, r_1 \in \mathbb{Z}_{2^\ell}$
-	0	$(-1)^\theta \cdot r_0$	$(-1)^\theta \cdot r_1$	$[v]^{\mathsf{A}}$
Round-2	$(R_0, R_1 + r_0)$	$(R_1 + r_0, R_2 + r_1)$	$(R_2 + r_1, R_0)$	$[\![v]\!]^{\mathsf{A}}$

Online phase

Compute β, σ, and $[\![b]\!]^{\mathsf{A}} = (-1)^\sigma \cdot [\![v]\!]^{\mathsf{A}} + [\![\beta]\!]^{\mathsf{A}}$.

1. All parties reveal $\sigma = b \oplus \gamma \oplus \theta$.
2. Define $[\![\beta]\!]^{\mathsf{A}} = \{(0,0),(0,\beta),(\beta,0)\}$, then all parties locally compute

$$[\![b]\!]^{\mathsf{A}} = (-1)^\sigma \cdot [\![v]\!]^{\mathsf{A}} + [\![\beta]\!]^{\mathsf{A}}$$

To compute $x \in \{0,1\}^{\ell}$, instead of naively invoking

$$[\![x]\!]^{\mathsf{A}} = \sum_{i=0}^{\ell-1} \mathsf{Bit2A}([\![x_i]\!]^{\mathsf{B}}) \cdot 2^{\ell-1-i},$$

To compute $x \in \{0,1\}^{\ell}$, instead of naively invoking

$$[\![x]\!]^{\mathsf{A}} = \sum_{i=0}^{\ell-1} \mathsf{Bit2A}([\![x_i]\!]^{\mathsf{B}}) \cdot 2^{\ell-1-i},$$

we use

$$[x]^{\mathsf{A}} = \sum_{i=0}^{\ell-1} \left((-1)^{\sigma_i}[v_i]^{\mathsf{A}} + [\beta_i]^{\mathsf{A}}\right) \cdot 2^{\ell-1-i} \text{ and}$$

$$[\![x]\!]^{\mathsf{A}} \leftarrow [x]^{\mathsf{A}}.$$

Realizing B2A

To compute $x \in \{0,1\}^\ell$, instead of naively invoking

$$[\![x]\!]^{\mathsf{A}} = \sum_{i=0}^{\ell-1} \mathsf{Bit2A}([\![x_i]\!]^{\mathsf{B}}) \cdot 2^{\ell-1-i},$$

we use

$$[x]^{\mathsf{A}} = \sum_{i=0}^{\ell-1} ((-1)^{\sigma_i}[v_i]^{\mathsf{A}} + [\beta_i]^{\mathsf{A}}) \cdot 2^{\ell-1-i} \text{ and}$$

$$[\![x]\!]^{\mathsf{A}} \leftarrow [x]^{\mathsf{A}}.$$

- Pro: **Reduced communication bits** in the offline phase from $4\ell^2/3$ to $\ell^2/3$ bits per party.
- Con: **One more round** in the online phase.

Conversion	Ref.	#Parties	Comm.[bits] Setup	Online	Total	Rounds
Bit2A	ABY2.0	2	0	$(\lambda+\ell)/2$	$(\lambda+\ell)/2$	$\mathbf{1}$
	ABY3 (OT)	3	0	2ℓ	2ℓ	$\mathbf{1}$
	edabits-based Bit2A	n	-	$t^2+2\ell+3$	-	$\log(t+1)+3$
	Our Π_1	3	$4\ell/3$	$\mathbf{1}$	$\mathbf{4\ell/3+1}$	$\mathbf{1}$
B2A	ABY	2	$\lambda\ell/2$	$(\ell^2+\ell)/4$	$(2\lambda+1)\ell/4+\ell^2/4$	2
	ABY2.0	2	$(\lambda\ell+\ell^2)/2$	ℓ	$(\lambda/2+1)\ell+\ell^2/2$	1
	ABY3	3	-	$\ell+\ell\log\ell$	-	$1+\log\ell$
	Our Π_2	3	$\ell^2/3$	$5\ell/3$	$(\ell^2+5\ell)/3$	2

Thank you for your Attention!