BitGC: Garbling with 1 Bit per Gate

Xiao Wang

Two Routes to Garbling Programs

Talk

(1) Use a hash function (e.g. a random oracle)

1.5κ is unavoidable in MiniCrypt!!

(2) Take a stronger assumption (e.g. RLWE, DCR)

Later Today

"Lower Bounds for Garbled Circuits from Shannon-Type Information Inequalities", Januzelli, Rosulek, Roy. CRYPTO25

Outline

- BitGC: A garbling scheme of size **1 bit** per Boolean gate
  - Hanlin Liu, Xiao Wang, Kang Yang, Yu Yu
  - Circular-secure Ring LWE
  - Appeared in **Eurocrypt 2025**

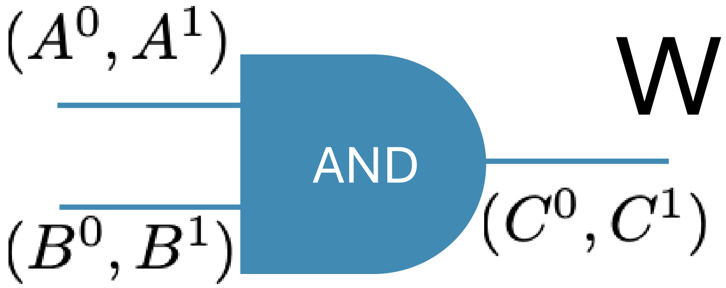
- Ongoing progress in pushing BitGC's concrete efficiency
  - LWYY + Wenhao Zhang, Wenjie Lu, and Chenkai Weng
  - 10 ms per gate for single-thread CPU (with some caveat)

Related Work

- To appear **Crypto 2025**
  - *"Authenticated BitGC for Act*
    Yang, Yu
  - *"A Unified Framework for Suc*
    Sharing" by Ishai, Li, Lin
  - *"Silent Circuit Relinearisatio*
    Garbled Circuits from DCR"

- To appear **FOCS 2025**
  - *"Succinct Homomorphic MA*
    Li, Lin,

Why Classical Garbled Circuit is So Big?



Garbler $(A^0, A^1), (B^0, B^1)$ (C^0, C^1)

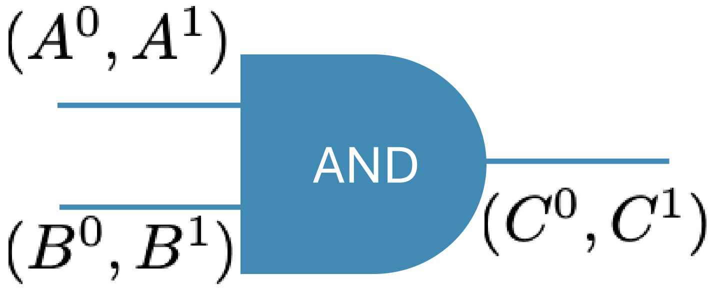
Evaluator (A^{v_a}, B^{v_b})

Encryptions of C's under (A,B)'s

Needs to contain information of C

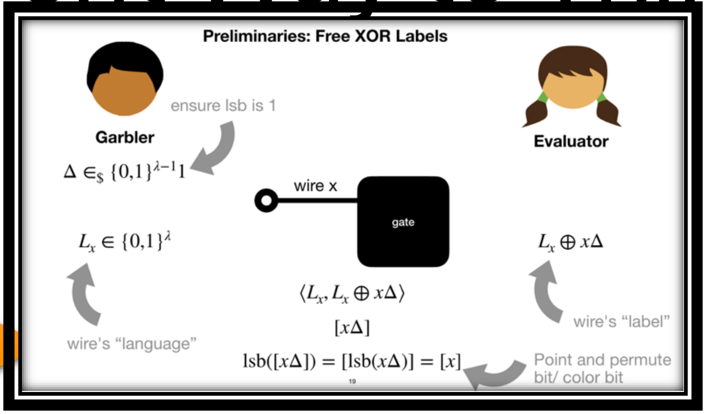
C^{v_c}

A Different Way to Think About GC



$$A^1 = A^0 \oplus \Delta$$
$$A^v = A^0 \oplus v \cdot \Delta$$

[]: secret sharing



Garbler (A^0, A^1) $[v_a\Delta]$ (B^0, B^1) $[v_b\Delta]$ (C^0, C^1)

Evaluator (A^{v_a}, B^{v_b}) $[v_a\Delta]$ $[v_b\Delta]$

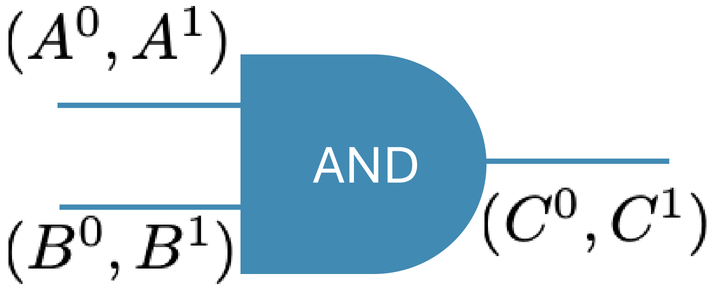
Encryptions of output shares under input shares

Only needs to contain information of v_a, v_b

$[(v_a \wedge v_b)\Delta]$

$[(v_a \wedge v_b)\Delta]$

"Blueprint"



$$A^1 = A^0 \oplus \Delta$$

Garbler $[v_a \Delta]$ $[v_b \Delta]$

Evaluator $[v_a \Delta]$ $[v_b \Delta]$

Encrypted **truth table bits**

$$[\![(v_a \wedge v_b) \cdot \Delta]\!]$$

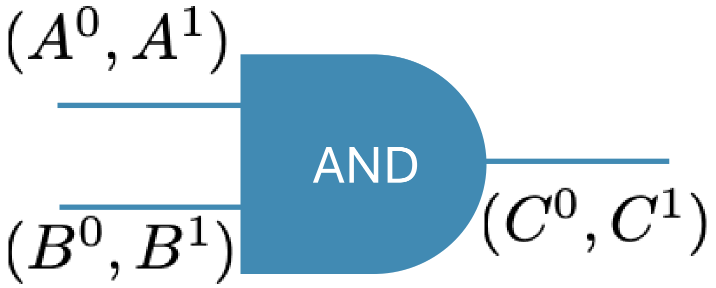
Garbler cannot obtain it non-interactively

$[(v_a \wedge v_b)\Delta]$

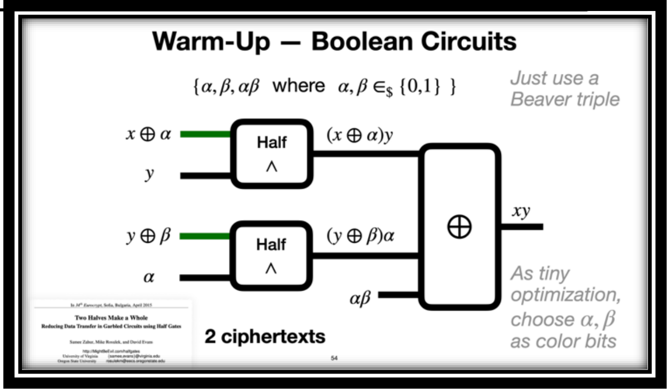
Our solution: distributively evaluate encrypted truth table directly

$[(v_a \wedge v_b)\Delta]$

Some Technical Details!



Masked bits	Input labels	Outbit bit	Output labels	Garbled table
$(0,0)$	(A^{π_a}, B^{π_b})	$z_{0,0} = \pi_a \wedge \pi_b$	$C^{z_{0,0}}$	$\tau_0 = H(A^{\pi_a}, B^{\pi_b}) \oplus C^{z_{0,0}}$
$(1,0)$	$(A^{\overline{\pi_a}}, B^{\pi_b})$	$z_{1,0} = \overline{\pi_a} \wedge \pi_b$	$C^{z_{1,0}}$	$\tau_1 = H(A^{\overline{\pi_a}}, B^{\pi_b}) \oplus C^{z_{1,0}}$
$(0,1)$	$(A^{\pi_a}, B^{\overline{\pi_b}})$	$z_{0,1} = \pi_a \wedge \overline{\pi_b}$	$C^{z_{0,1}}$	$\tau_2 = H(A^{\pi_a}, B^{\overline{\pi_b}}) \oplus C^{z_{0,1}}$
$(1,1)$	$(A^{\overline{\pi_a}}, B^{\overline{\pi_b}})$	$z_{1,1} = \overline{\pi_a} \wedge \overline{\pi_b}$	$C^{z_{1,1}}$	$\tau_3 = H(A^{\overline{\pi_a}}, B^{\overline{\pi_b}}) \oplus C^{z_{1,1}}$



Garbler $(A^0, A^1), (B^0, B^1)$

Uniformly pick (C^0, C^1)

Evaluator (A^{v_a}, B^{v_b})

$$C^{v_a \wedge v_b} = \mathsf{Eval}(A^{v_a}, B^{v_b}, \tau_{0,1,2,3})$$

Gate index omitted throughout the talk!

Masked bits	Input labels	Outbit bit	Output labels	Garbled table
$(0,0)$	(A^{π_a}, B^{π_b})	$z_{0,0} = \pi_a \wedge \pi_b$	$C^{z_{0,0}}$	$\boxed{\tau_0 = H(A^{\pi_a}, B^{\pi_b}) \oplus C^{z_{0,0}} = 0}$
$(1,0)$	$(A^{\overline{\pi_a}}, B^{\pi_b})$	$z_{1,0} = \overline{\pi_a} \wedge \pi_b$	$C^{z_{1,0}}$	$\tau_1 = H(A^{\overline{\pi_a}}, B^{\pi_b}) \oplus C^{z_{1,0}}$
$(0,1)$	$(A^{\pi_a}, B^{\overline{\pi_b}})$	$z_{0,1} = \pi_a \wedge \overline{\pi_b}$	$C^{z_{0,1}}$	$\tau_2 = H(A^{\pi_a}, B^{\overline{\pi_b}}) \oplus C^{z_{0,1}}$
$(1,1)$	$(A^{\overline{\pi_a}}, B^{\overline{\pi_b}})$	$z_{1,1} = \overline{\pi_a} \wedge \overline{\pi_b}$	$C^{z_{1,1}}$	$\tau_3 = H(A^{\overline{\pi_a}}, B^{\overline{\pi_b}}) \oplus C^{z_{1,1}}$

$$\boxed{C^{z_{i,j}} \oplus C^{z_{0,0}} = (z_{i,j} \oplus z_{0,0}) \cdot \Delta}$$

Garbled table
0
$\tau_1 = H(A^{\overline{\pi_a}}, B^{\pi_b}) \oplus H(A^{\pi_a}, B^{\pi_b}) \oplus (z_{1,0} \oplus z_{0,0}) \cdot \Delta$
$\tau_2 = H(A^{\pi_a}, B^{\overline{\pi_b}}) \oplus H(A^{\pi_a}, B^{\pi_b}) \oplus (z_{0,1} \oplus z_{0,0}) \cdot \Delta$
$\tau_3 = H(A^{\overline{\pi_a}}, B^{\overline{\pi_b}}) \oplus H(A^{\pi_a}, B^{\pi_b}) \oplus (z_{1,1} \oplus z_{0,0}) \cdot \Delta$

Assuming RO is homomorphic:

$$\boxed{H(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}) \oplus H(A^{\pi_a}, B^{\pi_b}) = i \cdot H(\Delta, 0) \oplus j \cdot H(0, \Delta) \oplus ij \cdot H(\Delta, \Delta)}$$

Garbled table
0
$\tau_1 = H(\Delta, 0) \oplus \boxed{(z_{1,0} \oplus z_{0,0})} \cdot \Delta$
$\tau_2 = H(0, \Delta) \oplus \boxed{(z_{0,1} \oplus z_{0,0})} \cdot \Delta$
$\tau_3 = H(\Delta, \Delta) \oplus \boxed{(z_{1,1} \oplus z_{1,0} \oplus z_{0,1} \oplus z_{0,0})} \cdot \Delta$

Encrypted Truth Table Bits

So Far...

If only there is a way to send ciphertext cheaply
and a way to instantiate homomorphic RO !

Garbler $(A^0, A^1), (B^0, B^1)$

Evaluator (A^{v_a}, B^{v_b})

$$\tau_1 = H(\Delta, 0) \oplus (z_{1,0} \oplus z_{0,0}) \cdot \Delta$$
$$\tau_2 = H(0, \Delta) \oplus (z_{0,1} \oplus z_{0,0}) \cdot \Delta$$
$$\tau_3 = H(\Delta, \Delta) \oplus (z_{1,1} \oplus z_{1,0} \oplus z_{0,1} \oplus z_{0,0}) \cdot \Delta$$

$$C^{z_{0,0}} \stackrel{\text{def}}{=} H(A^{\pi_a}, B^{\pi_b})$$

Evaluate to obtain $C^{v_a \wedge v_b}$

(C^0, C^1)

C^{v_c}

Transmitting Encrypted Bits is Cheap

To send an encryption of bit x

Bitstring labels not compatible with SWHE ciphertext!

[[seed]]

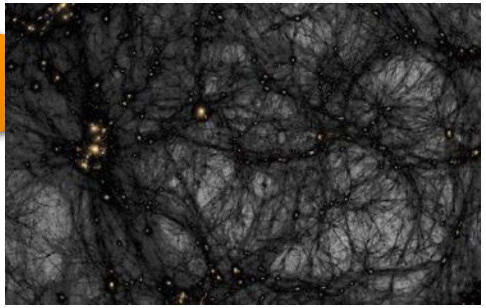
$[[r_1]]$ … $[[r_n]]$ =PRG([[seed]])

$[[r_1]]$ … $[[r_n]]$ =PRG([[seed]])

$d = x \oplus r_i$

[[x]] = [[ri]] $\oplus$ d

Need a PRG that we can evaluate in HE without bootstrapping



[TCC:BIPSW18,...]

Using Polynomial Rings as Wire Labels

$\mathbb{Z}_p[X]/(X^n+1)$

Garbling based on bit-strings

$$A^1 \stackrel{\text{def}}{=} A^0 \oplus \Delta$$

$$A^v = A^0 \oplus v \cdot \Delta$$

$$\boxed{A^v = A^{\pi_a} \oplus (\pi_a \oplus v) \cdot \Delta}$$

Garbling based on ring elements

$$A^1 \stackrel{\text{def}}{=} A^0 + (-1)^{\pi_a} \cdot \Delta$$

$$A^v = A^0 + (-1)^{\pi_a} \cdot v \cdot \Delta$$

$$\boxed{A^v = A^{\pi_a} + (\pi_a \oplus v) \cdot \Delta}$$

Garbled table
0
$\tau_1 = H(\Delta, 0) \oplus (z_{1,0} \oplus z_{0,0}) \cdot \Delta$
$\tau_2 = H(0, \Delta) \oplus (z_{0,1} \oplus z_{0,0}) \cdot \Delta$
$\tau_3 = H(\Delta, \Delta) \oplus (z_{1,1} \oplus z_{1,0} \oplus z_{0,1} \oplus z_{0,0}) \cdot \Delta$

$A^1 \stackrel{\text{def}}{=} A^0 + (-1)^{\pi_a} \cdot \Delta$
$A^v = A^0 + (-1)^{\pi_a} \cdot v \cdot \Delta$
$A^v = A^{\pi_a} + (\pi_a \oplus v) \cdot \Delta$

Masked bits	Input labels	Outbit bit	Output labels	Garbled table
$(0,0)$	(A^{π_a}, B^{π_b})	$z_{0,0} = \pi_a \wedge \pi_b$	$C^{z_{0,0}}$	0
$(1,0)$	$(A^{\overline{\pi_a}}, B^{\pi_b})$	$z_{1,0} = \overline{\pi_a} \wedge \pi_b$	$C^{z_{1,0}}$	$\tau_1 = [\![(-1)^{\pi_c} \cdot (z_{1,0} - z_{0,0})]\!]$
$(0,1)$	$(A^{\pi_a}, B^{\overline{\pi_b}})$	$z_{0,1} = \pi_a \wedge \overline{\pi_b}$	$C^{z_{0,1}}$	$\tau_2 = [\![(-1)^{\pi_c} \cdot (z_{0,1} - z_{0,0})]\!]$
$(1,1)$	$(A^{\overline{\pi_a}}, B^{\overline{\pi_b}})$	$z_{1,1} = \overline{\pi_a} \wedge \overline{\pi_b}$	$C^{z_{1,1}}$	$\tau_3 = [\![(-1)^{\pi_c} \cdot (z_{1,1} - z_{0,1} - z_{1,0} + z_{0,0})]\!]$

Garbler $(A^0, A^1), (B^0, B^1)$

$$C^{z_{0,0}} \stackrel{\text{def}}{=} \text{Eval}(A^{\pi_a}, B^{\pi_b}, \tau_{1,2,3})$$

Evaluator (A^{v_a}, B^{v_b})

$$C^{v_a \wedge v_b} := \text{Eval}(A^{v_a}, B^{v_b}, \tau_{1,2,3})$$

If only there is a way to send ciphertext cheaply and a way to instantiate homomorphic RO!

Garbled table
0
$\tau_1 = [\![(-1)^{\pi_c} \cdot (z_{1,0} - z_{0,0})]\!]$
$\tau_2 = [\![(-1)^{\pi_c} \cdot (z_{0,1} - z_{0,0})]\!]$
$\tau_3 = [\![(-1)^{\pi_c} \cdot (z_{1,1} - z_{0,1} - z_{1,0} + z_{0,0})]\!]$

$$A^1 \stackrel{\mathsf{def}}{=} A^0 + (-1)^{\pi_a} \cdot \Delta$$
$$A^v = A^0 + (-1)^{\pi_a} \cdot v \cdot \Delta$$
$$A^v = A^{\pi_a} + (\pi_a \oplus v) \cdot \Delta$$

Instantiating Homomorphic RO

Distributed "Evaluation":

$$\mathsf{Eval}(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}, \tau_{1,2,3}) - \mathsf{Eval}(A^{\pi_a}, B^{\pi_b}, \tau_{1,2,3})$$
$$= i \cdot \mathsf{Dec}(\Delta, \tau_1) + j \cdot \mathsf{Dec}(\Delta, \tau_2) + i \cdot j \cdot \mathsf{Dec}(\Delta, \tau_3)$$

Assume $\mathsf{Dec}(\Delta, [\![m]\!]) = m \cdot \Delta$

Homomorphic RO:

$$H(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}) - H(A^{\pi_a}, B^{\pi_b}) = i \cdot H(\Delta, 0) + j \cdot H(0, \Delta) + i \cdot j \cdot H(\Delta, \Delta)$$

$$\mathsf{Eval}(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}, \tau_{1,2,3}) - \mathsf{Eval}(A^{\pi_a}, B^{\pi_b}, \tau_{1,2,3}) = (-1)^{\pi_c} \cdot (z_{i,j} - z_{0,0}) \cdot \Delta$$

$$A^1 \stackrel{\text{def}}{=} A^0 + (-1)^{\pi_a} \cdot \Delta$$

$$A^v = A^0 + (-1)^{\pi_a} \cdot v \cdot \Delta$$

$$A^v = A^{\pi_a} + (\pi_a \oplus v) \cdot \Delta$$

Masked bits	Input labels	Outbit bit	Output labels	Garbled table
$(0,0)$	(A^{π_a}, B^{π_b})	$z_{0,0} = \pi_a \wedge \pi_b$	$C^{z_{0,0}}$	0
$(1,0)$	$(A^{\overline{\pi_a}}, B^{\pi_b})$	$z_{1,0} = \overline{\pi_a} \wedge \pi_b$	$C^{z_{1,0}}$	$\tau_1 = [\![(-1)^{\pi_c} \cdot (z_{1,0} - z_{0,0})]\!]$
$(0,1)$	$(A^{\pi_a}, B^{\overline{\pi_b}})$	$z_{0,1} = \pi_a \wedge \overline{\pi_b}$	$C^{z_{0,1}}$	$\tau_2 = [\![(-1)^{\pi_c} \cdot (z_{0,1} - z_{0,0})]\!]$
$(1,1)$	$(A^{\overline{\pi_a}}, B^{\overline{\pi_b}})$	$z_{1,1} = \overline{\pi_a} \wedge \overline{\pi_b}$	$C^{z_{1,1}}$	$\tau_3 = [\![(-1)^{\pi_c} \cdot (z_{1,1} - z_{0,1} - z_{1,0} + z_{0,0})]\!]$

Garbler $(A^0, A^1), (B^0, B^1)$

Define

$$C^{z_{0,0}} \stackrel{\text{def}}{=} \mathsf{Eval}(A^{\pi_a}, B^{\pi_b}, \tau_{1,2,3})$$

Evaluator (A^{v_a}, B^{v_b})

$$\begin{aligned}
&\mathsf{Eval}(A^{v_a}, B^{v_b}, \tau_{1,2,3}) \\
&= \mathsf{Eval}(A^{\pi_a \oplus (\pi_a \oplus v_a)}, B^{\pi_b \oplus (\pi_b \oplus v_b)}, \tau_{1,2,3}) \\
&= C^{z_{0,0}} + (-1)^{\pi_c} \cdot (z_{\pi_a \oplus v_a, \pi_b \oplus v_b} - z_{0,0}) \cdot \Delta \\
&= C^0 + (-1)^{\pi_c} \cdot z_{\pi_a \oplus v_a, \pi_b \oplus v_b} \cdot \Delta \\
&= C^{z_{\pi_a \oplus v_a, \pi_b \oplus v_b}} \\
&= C^{v_a \wedge v_b}
\end{aligned}$$

$\mathsf{Eval}(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}, \tau_{1,2,3}) - \mathsf{Eval}(A^{\pi_a}, B^{\pi_b}, \tau_{1,2,3})$
$= (-1)^{\pi_c} \cdot (z_{i,j} - z_{0,0}) \cdot \Delta$

Recap

Garbler

$(A^0, A^1), (B^0, B^1)\ [\![\pi_a]\!], [\![\pi_b]\!]\ [\![r_c]\!]$

Evaluator

$(A^{v_a}, B^{v_b})\ [\![\pi_a]\!], [\![\pi_b]\!]\ [\![r_c]\!]$

Compute π_c

$d = r_c \oplus \pi_c$

τ_1, τ_2, τ_3 $\qquad$ τ_1, τ_2, τ_3

$C^{z_{0,0}} \stackrel{\text{def}}{=} \mathsf{Eval}(A^{\pi_a}, B^{\pi_b}, \tau_{1,2,3})$

$\mathsf{Eval}(A^{v_a}, B^{v_b}, \tau_{1,2,3}) = C^{v_a \wedge v_b}$

$(C^0, C^1)\ [\![\pi_c]\!]$

$C^{v_c}\ [\![\pi_c]\!]$

Properties that we need

Homomorphic evaluation of a PRG and 2 more levels to assemble

Encrypted
truth table bits

$$\mathsf{Dec}(\Delta, [\![m]\!]) = m \cdot \Delta$$

$$\mathsf{Eval}(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}, \tau_{1,2,3}) - \mathsf{Eval}(A^{\pi_a}, B^{\pi_b}, \tau_{1,2,3})$$
$$= i \cdot \mathsf{Dec}(\Delta, \tau_1) + j \cdot \mathsf{Dec}(\Delta, \tau_2) + i \cdot j \cdot \mathsf{Dec}(\Delta, \tau_3)$$

GSW
OR
Any regev-like scheme + send an encryption of Δ, invoking circularity

$$A^1 \stackrel{\text{def}}{=} A^0 + (-1)^{\pi_a} \cdot \Delta$$
$$A^v = A^0 + (-1)^{\pi_a} \cdot v \cdot \Delta$$
$$A^v = A^{\pi_a} + (\pi_a \oplus v) \cdot \Delta$$

$$\mathsf{Eval}(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}, \tau_{1,2,3}) - \mathsf{Eval}(A^{\pi_a}, B^{\pi_b}, \tau_{1,2,3})$$
$$= \boxed{i \cdot \mathsf{Dec}(\Delta, \tau_1)} + j \cdot \mathsf{Dec}(\Delta, \tau_2) + \boxed{i \cdot j \cdot \mathsf{Dec}(\Delta, \tau_3)}$$

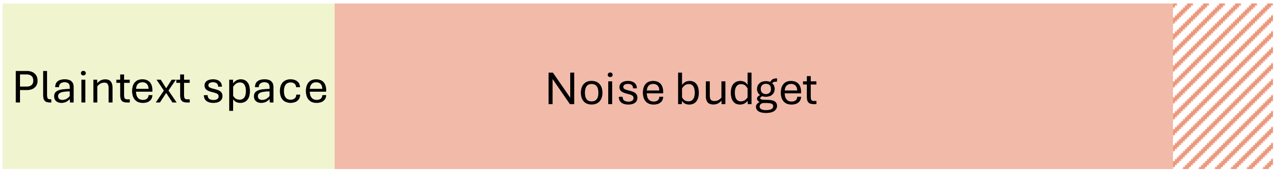
$$\boxed{\begin{aligned} &\mathsf{Dec}(A^{\pi_a \oplus i}, \tau_1) - \mathsf{Dec}(A^{\pi_a}, \tau) \\ &= \mathsf{Dec}(A^{\pi_a} + i \cdot \Delta, \tau_1) - \mathsf{Dec}(A^{\pi_a}, \tau) \\ &= i \cdot \mathsf{Dec}(\Delta, \tau_1) \end{aligned}}$$

$$\boxed{\mathsf{Eval}(A, B, \tau_{1,2,3}) = \mathsf{Dec}(A, \tau_1) + \mathsf{Dec}(B, \tau_2)} + \boxed{\mathsf{Something}(A, B, \tau_3)}$$

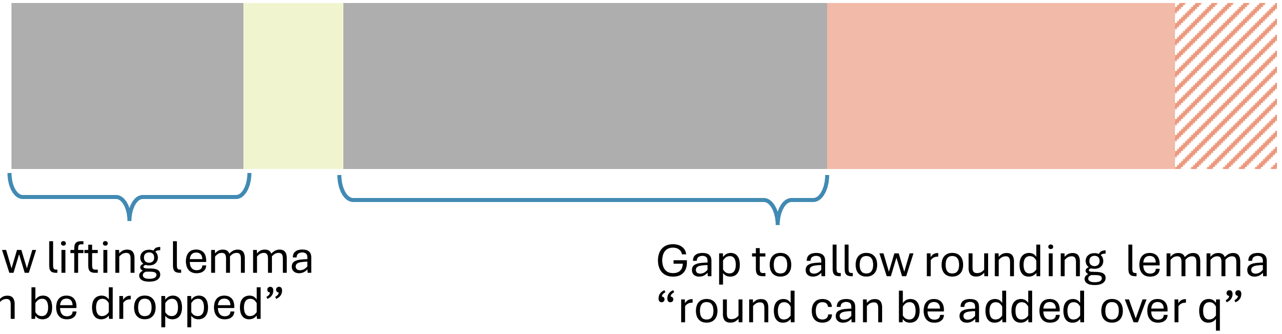
Near Linear Decryption [BoyleKohlScholl19]

$$\mathsf{Dec}(X + \mathsf{sk}, [\![m]\!]) = \mathsf{Dec}(X, [\![m]\!]) + \mathsf{Dec}(\mathsf{sk}, [\![m]\!])$$

$$[\![m]\!] = (a, a \cdot \mathsf{sk} + e + m \cdot t)$$



Requirement:



$$A^1 \overset{\text{def}}{=} A^0 + (-1)^{\pi_a} \cdot \Delta$$
$$A^v = A^0 + (-1)^{\pi_a} \cdot v \cdot \Delta$$
$$A^v = A^{\pi_a} + (\pi_a \oplus v) \cdot \Delta$$

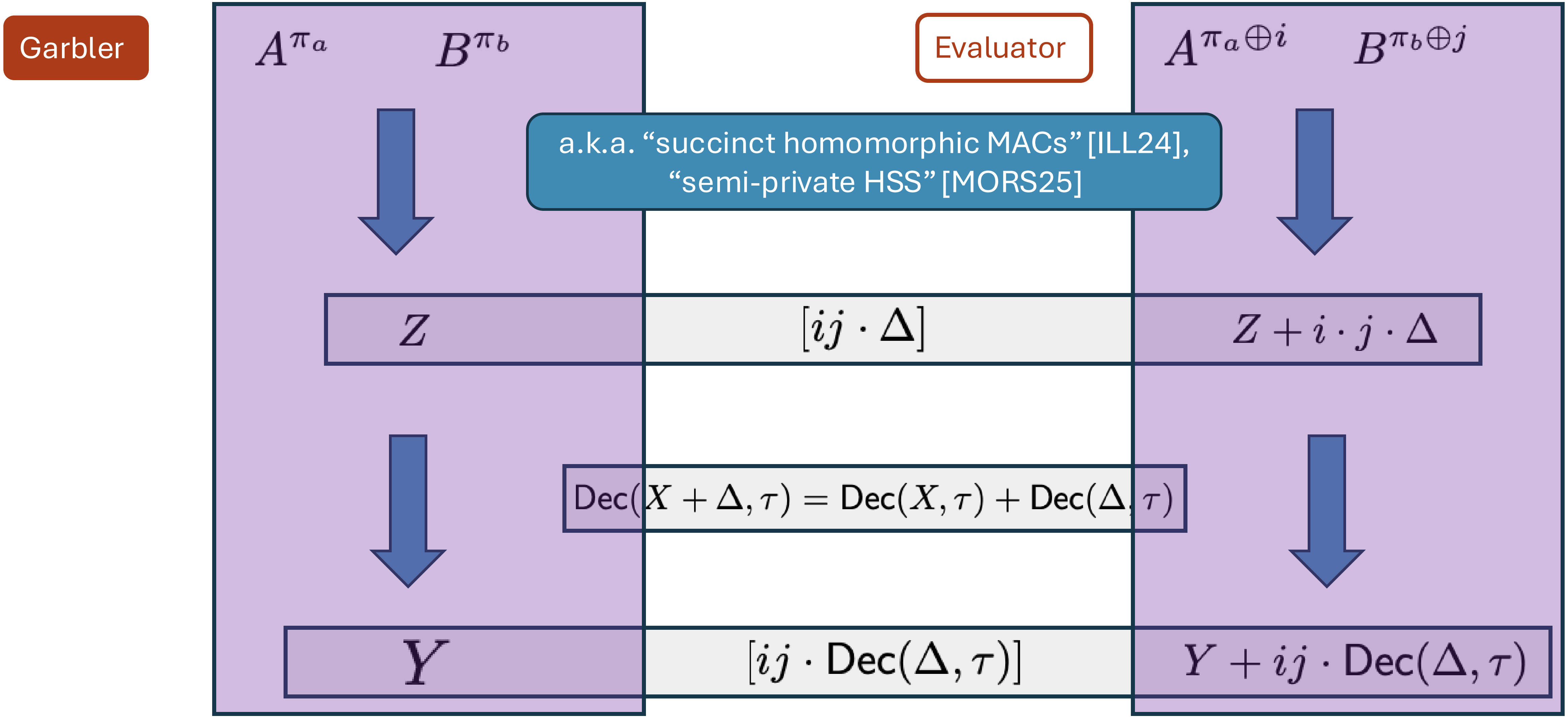
$$\mathsf{Eval}(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}, \tau_{1,2,3}) - \mathsf{Eval}(A^{\pi_a}, B^{\pi_b}, \tau_{1,2,3})$$
$$= \boxed{i \cdot \mathsf{Dec}(\Delta, \tau_1)} + j \cdot \mathsf{Dec}(\Delta, \tau_2) + \boxed{i \cdot j \cdot \mathsf{Dec}(\Delta, \tau_3)}$$

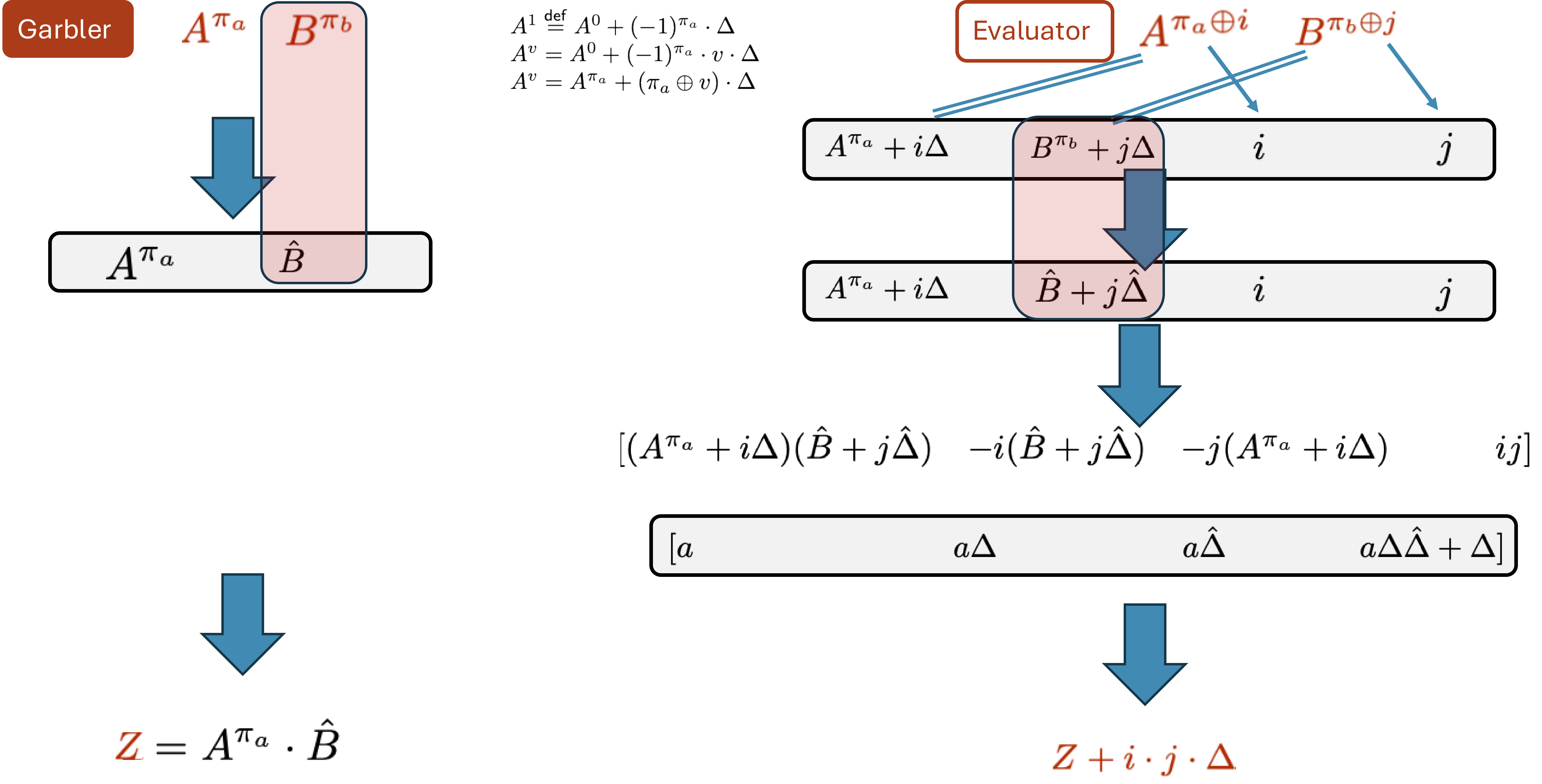
$$\boxed{\begin{aligned} &\mathsf{Dec}(A^{\pi_a \oplus i}, \tau_1) - \mathsf{Dec}(A^{\pi_a}, \tau) \\ &= \mathsf{Dec}(A^{\pi_a} + i \cdot \Delta, \tau_1) - \mathsf{Dec}(A^{\pi_a}, \tau) \\ &= i \cdot \mathsf{Dec}(\Delta, \tau_1) \end{aligned}}$$

$$\boxed{\mathsf{Eval}(A, B, \tau_{1,2,3}) = \mathsf{Dec}(A, \tau_1) + \mathsf{Dec}(B, \tau_2)} + \boxed{\mathsf{Something}(A, B, \tau_3)}$$

$$\boxed{\widetilde{\mathsf{Dec}}(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}, \tau) - \widetilde{\mathsf{Dec}}(A^{\pi_a}, B^{\pi_b}, \tau) = i \cdot j \cdot \mathsf{Dec}(\Delta, \tau)}$$

$$\widetilde{\mathsf{Dec}}(A^{\pi_a \oplus i}, B^{\pi_b \oplus j}, \tau) - \widetilde{\mathsf{Dec}}(A^{\pi_a}, B^{\pi_b}, \tau) = i \cdot j \cdot \mathsf{Dec}(\Delta, \tau)$$





Garbler
A^{π_a} B^{π_b}
A^{π_a} $\hat{B}$
$Z = A^{\pi_a} \cdot \hat{B}$
$A^1 \stackrel{\text{def}}{=} A^0 + (-1)^{\pi_a} \cdot \Delta$
$A^v = A^0 + (-1)^{\pi_a} \cdot v \cdot \Delta$
$A^v = A^{\pi_a} + (\pi_a \oplus v) \cdot \Delta$
Evaluator
$A^{\pi_a \oplus i}$ $B^{\pi_b \oplus j}$
$A^{\pi_a} + i\Delta$ $B^{\pi_b} + j\Delta$ i j
$A^{\pi_a} + i\Delta$ $\hat{B} + j\hat{\Delta}$ i j
$[(A^{\pi_a} + i\Delta)(\hat{B} + j\hat{\Delta})$ $-i(\hat{B} + j\hat{\Delta})$ $-j(A^{\pi_a} + i\Delta)$ $ij]$
$[a$ $a\Delta$ $a\hat{\Delta}$ $a\Delta\hat{\Delta} + \Delta]$
$Z + i \cdot j \cdot \Delta$

BitGC (Being) Made Concretely Efficient

- Offline homomorphic PRG expansion:
  - ↑ Depth-5 BGV
  - ↑ SIMD-accelerated evaluation
  - ↓ Ciphertext unpacking to RLWE ciphertext

- Gate assembling:
  - ↑ 6 ring multiplications per gate
  - ↑ Additive homomorphism sufficient to assemble the gate
  - ↓ Increase size of AND to 6 bits and XOR to 2 bits

BitGC (Being) Made Concretely Efficient

CPU, single-thread

- Offline preprocessing: 1 – 7 ms per gate
- Garbling: 3 ms per gate

FHE and garbling are for different purposes but speed comparable to TFHE evaluation.

F1: A Fast and Programmable Accelerator for Fully Homomorphic Encryption (Extended Version)

Axel Feldmann[1*], Nikola Samardzic[1*], Aleksandar Krastev[1], Srini Devadas[1], Ron Dreslinski[2], Karim Eldefrawy[3], Nicholas Genise[3], Chris Peikert[2], Daniel Sanchez[1]

[1] Massachusetts Institute of Technology {axelf, nsamar, alexalex, devadas, sanchez}@csail.mit.edu
[2] University of Michigan {dreslin, cpeikert}@umich.edu
[3] SRI International {karim.eldefrawy, nicholas.genise}@sri.com

FPGA (estimated)

- Offline preprocessing: 0.1 – 0.4 us per gate
- Garbling: 0.2 us per gate

What 10 years can do?

Fairplay — A Secure Two-Party Computation System

Dahlia Malkhi[1], Noam Nisan[1], Benny Pinkas[2], and Yaron Sella[1]

~0.1 s/gate USENIX 2004

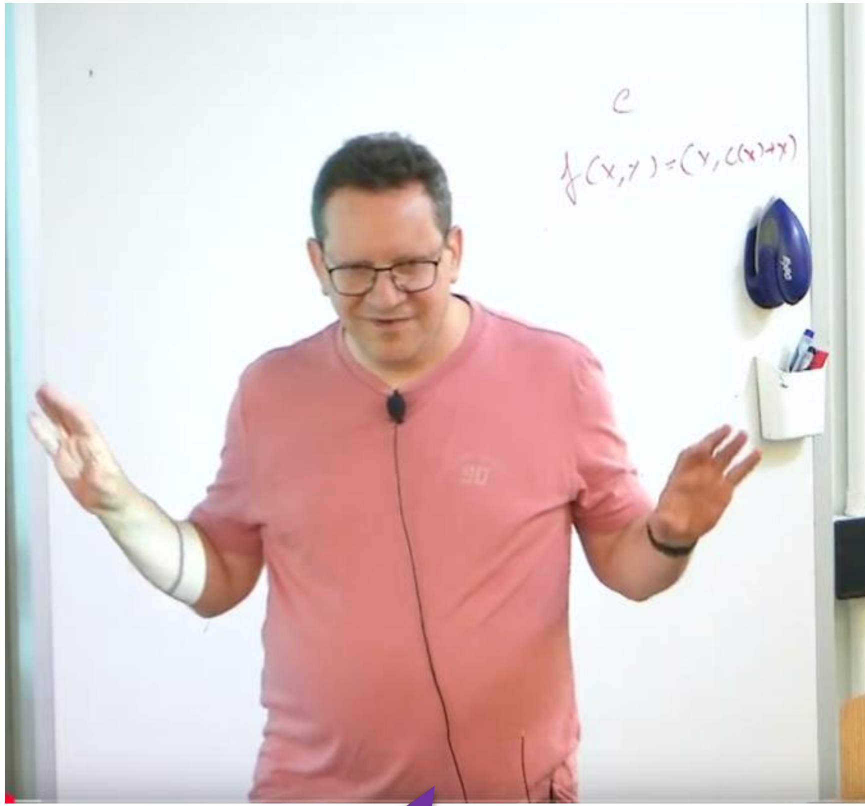
Efficient Garbling from a Fixed-Key Blockcipher

Mihir Bellare[1] Viet Tung Hoang[2] Sriram Keelveedhi[1] Phillip Rogaway[2]

~100 ns/gate SP 2013

Cryptography Boot Camp 2015

Cryptography 10 Years Later



BitGC 2025
0.01 s/gate

BitGCrazy 2035
?? s/gate