SAND2022-8963C

Auditable, Available and Resilient Private Computation on the Blockchain via MPC

Christopher Cordi, Michael P. Frank, Kasimir Gabert, Carollan Helinski,

Ryan C. Kao, **Vladimir Kolesnikov**, Abrahim Ladha, and Nicholas Pattengale

(Sandia Labs and Georgia Tech)

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Motivation

- Blockchain:
  - reliability, availability, and verifiability (e.g., auditability)
- MPC: Garbled Circuits (GC) and Garbled FSA (GFSA)
  - Recent Advances
    - GRAM
    - Stacked Garbling
- Organization wants to run transparent and resilient private computation:
  - Sealed-bid auctions (e.g. with public interest: public contracts, airwaves, services)
  - Market-based pricing mechanisms (electricity, etc)
- How can we get privacy for BC computation
  - Formal modeling and proofs

Outline

- MPC + Blockchain
- Our architecture and trust model
- Security theorem
- Evaluation on Ethereum

Contribution

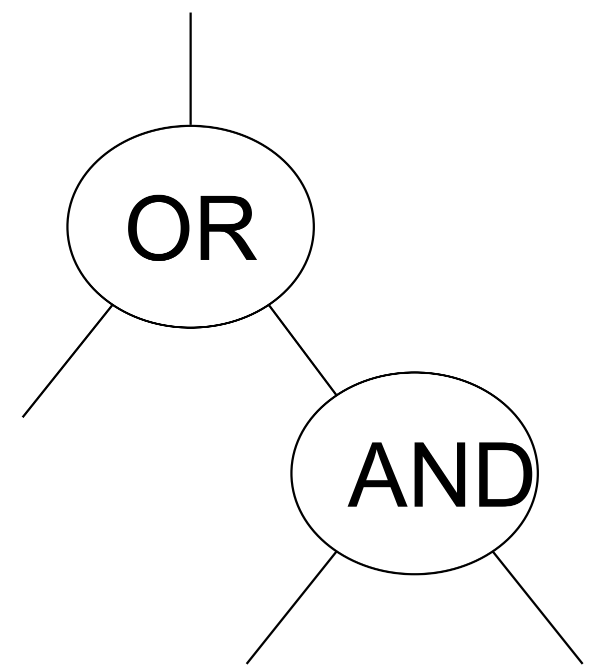
- A model for MPC on the blockchain
  - Natural roles for computation participants and trust model
  - Formal theorem statements and proofs
- Experimental evaluation for GC- and GFSA-based MPC

Related work (MPC / private computation on BC)

- P2DEX Baum et al. Publicly verifiable MPC for cross-chain exchanges
- YOSO line (BC nodes have private info; assume collusion limits)
- Hawk (use enclaves)
- MPC off chain (e.g. Enigma)
- BC for MPC fairness
- Many more

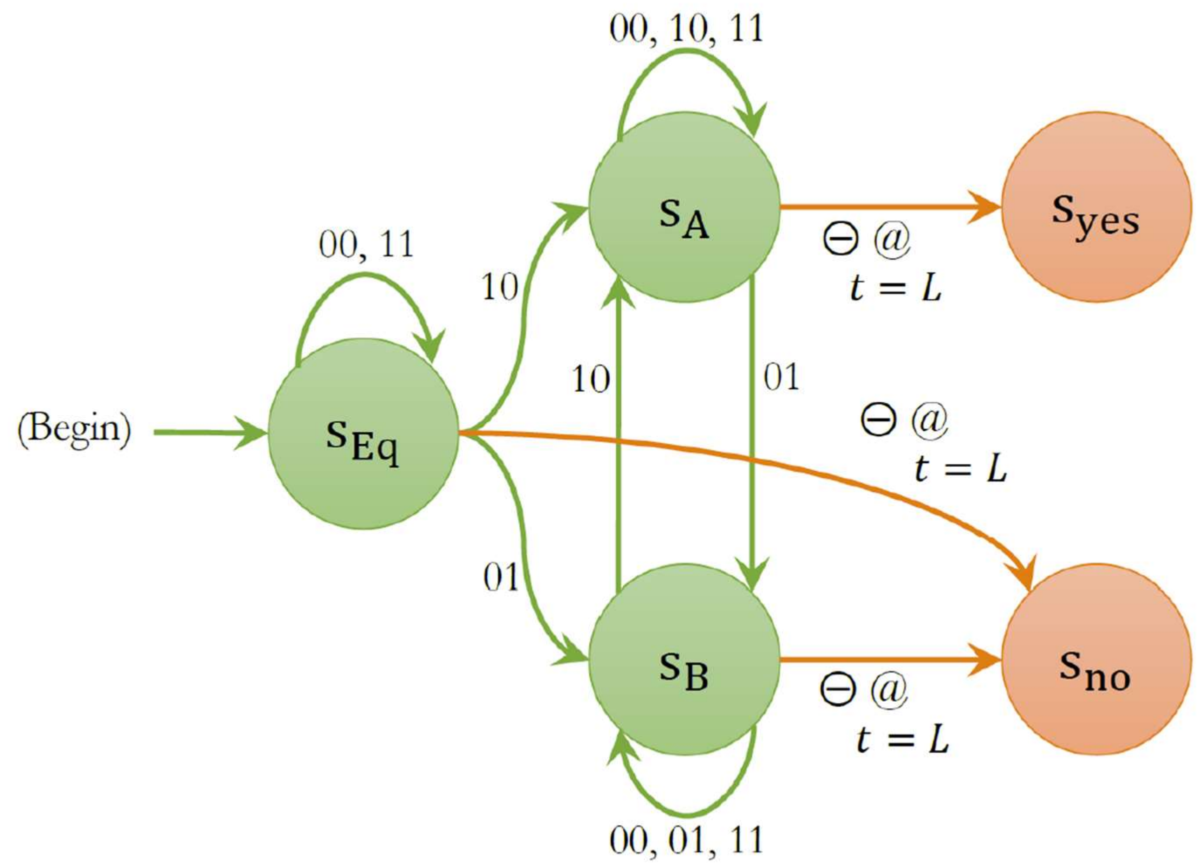
Functions are circuits

$$F(x, y) \quad \rightarrow$$

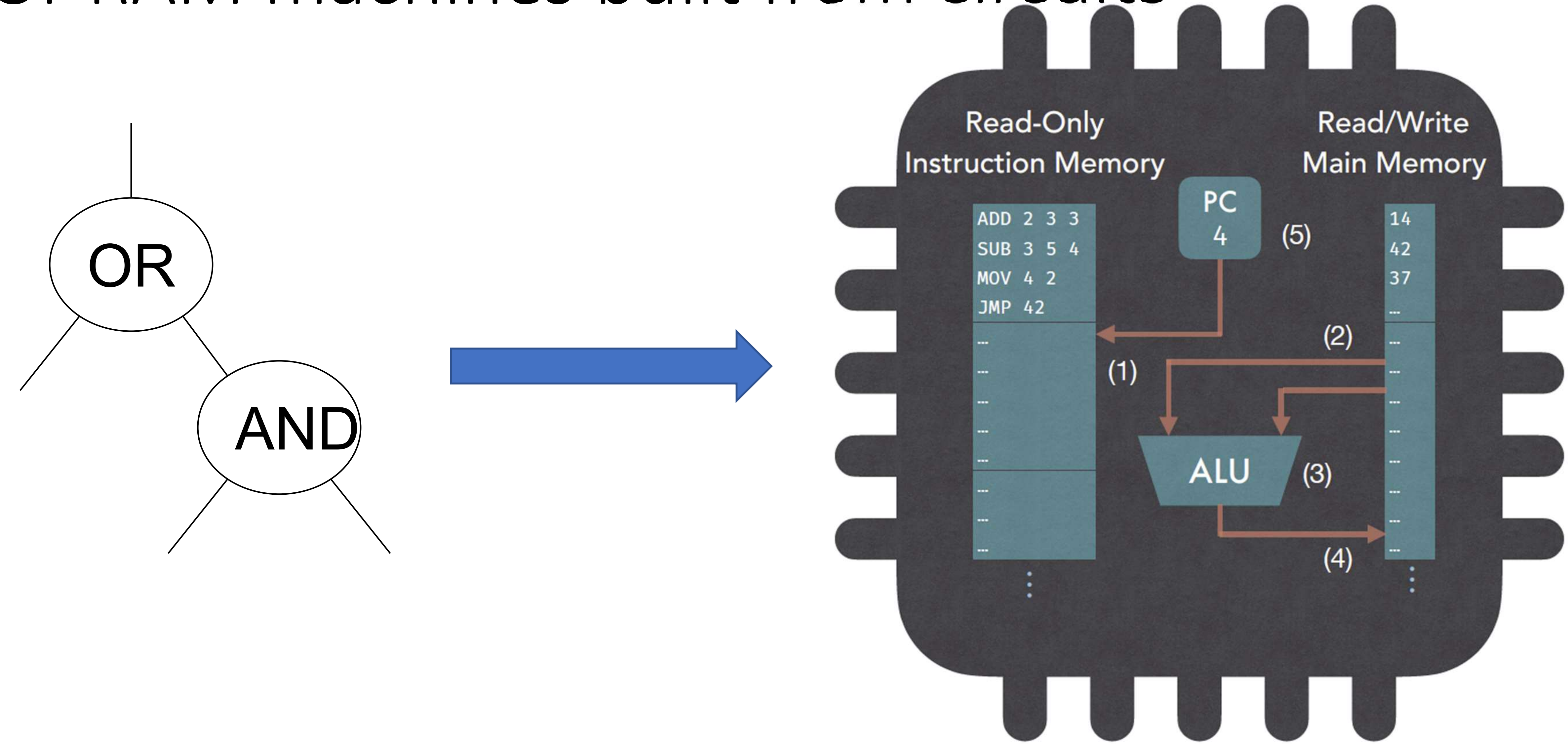


Or State Machines

$$F(x, y) \quad \rightarrow$$



Or RAM machines built from circuits

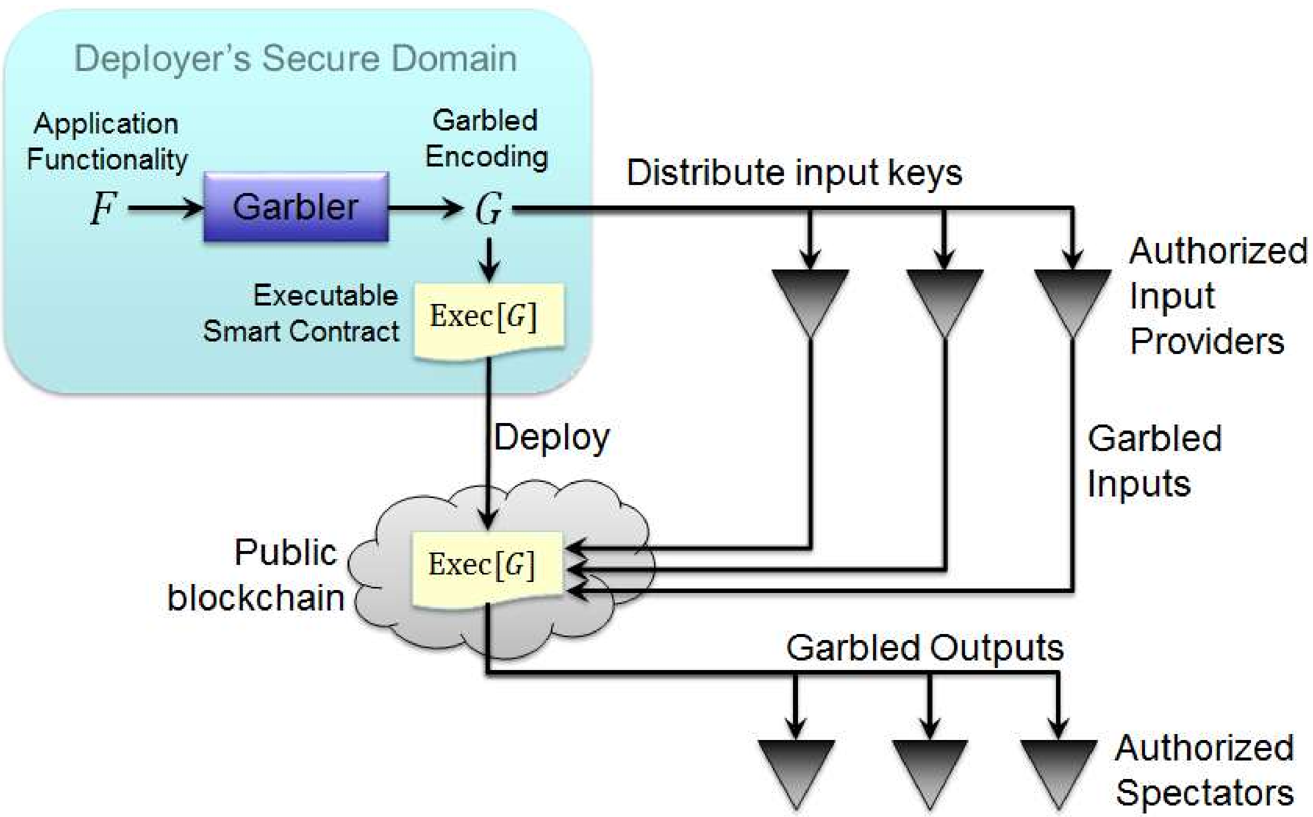


Public Blockchain

- Distributed ledger (unchangeable, but appendable record)
- Jointly maintained and managed by all
- All information on the blockchain is public
- Any computation would be reliable, but not private

- Example chain: Ethereum

A Blockchain MPC Architecture



Roles:

Garbler. May be a single trusted entity or run distributedly, e.g., by an MPC over a private chain.

Input Provider contributes (encrypted) inputs to the computation.

Unlocker manages encrypted inputs, preventing input providers from learning unauthorized information about the computation and adapting their input based on it.

Evaluator (the blockchain itself) evaluates GC/GFSA/etc and obtains encrypted output.

Reader (Spectator) obtains designated output

Trust Model

- Garbler is honest and erases its state after GF generation
  - Implemented by an MPC/trusted entity/HSM
- Auditability option:
  - Securely store GC secrets; if needed, "open" GC computation
  - Opening can be in plaintext (everyone learns the computation) or inside MPC (verify needed aspects)
- Input Providers do not collude with Unlockers

General security theorem

Theorem 1. *Let* $\mathcal{G} = (\mathsf{ev}, \mathsf{Gb}, \mathsf{En}, \mathsf{Ev}, \mathsf{De})$ *be a garbling scheme as above. Let* $(y_0, ..., y_p) = f(x_0, ..., x_q)$ *be the function desired to be computed, such that each bit of the function output depends on all inputs*[8]*. Let* `Gen` *be the contract generator,* $IP_1, ..., IP_n$ *be the input providers,* $U_1, ..., U_m$ *be the unlockers, and* $R_1, ..., R_\ell$ *be the output receivers. Assume* `Gen` *is honest and generates* $(F, e, d) = \mathcal{G}.\mathsf{Gb}$ *and distributes* (F, e, d) *to players as described above. Let* $I \subset \{IP_i, U_j, R_k\}$ *be the set of colluding malicious players, such that for no input wire* W_i *both its input provider and unlocker are in* I*.*

Then blockchain evaluation of f *which computes* $\mathcal{G}.\mathsf{ev}$ *as described above, is secure against a malicous adversary corrupting* I*.*

Note: Use programmable RO for simulation (to correctly decode the function output)

Evaluation (function-hiding auction GC/GFSA)

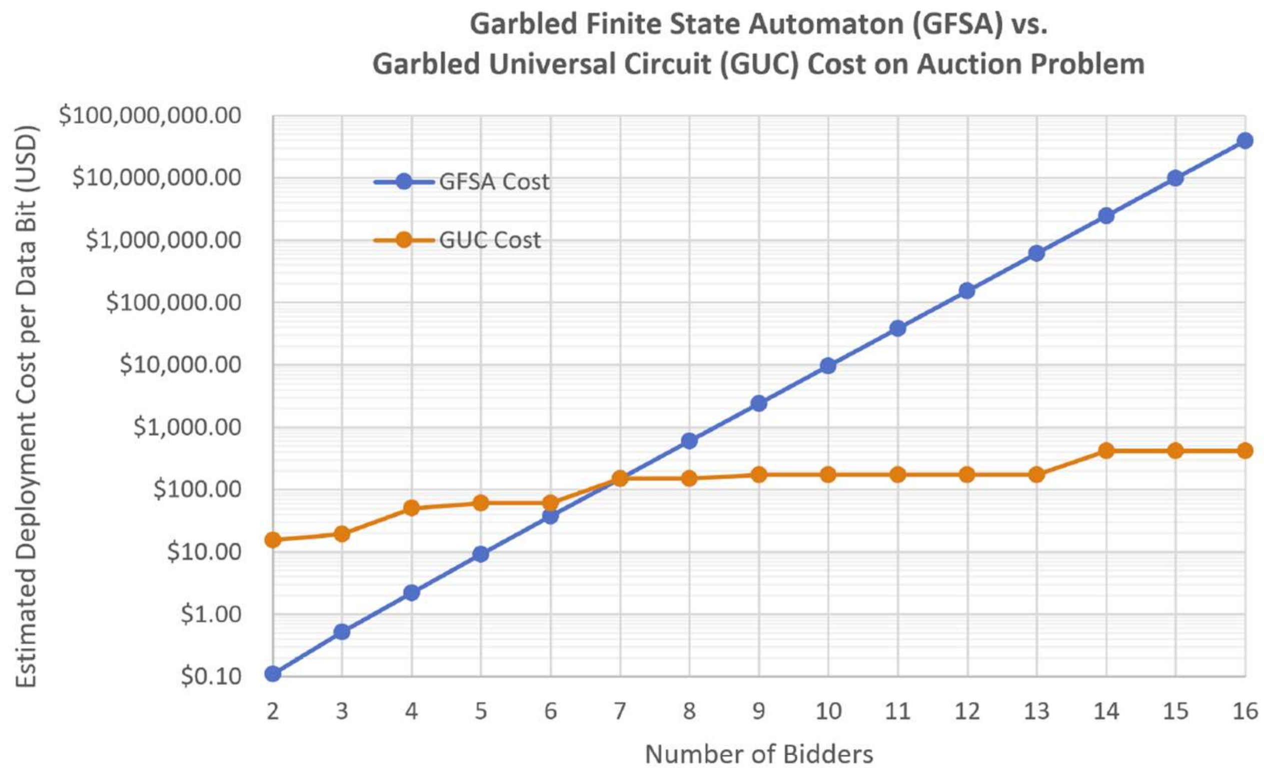


Fig. 4. ***Cost comparison for GFSA vs. configurable universal GCs for multi-party auctions.*** The break-even point occurs for $B = 7$ bidders, where the cost of both techniques is about 800 million gas per bit of input length. Prices here assumed optimistically that we are paying only 1 mETH (or in the ballpark of \$0.20) per million gas; however, after our tests, the average gas price grew substantially higher.