STOC 2018

Breaking the Circuit-Size Barrier in Secret Sharing

Tianren Liu, Vinod Vaikuntanathan

MIT

Poster and slides on liutianren.com

General Secret Sharing

A secret sharing scheme over n parties is a randomized algorithm that distributes a one-bit secret among n shares

$$\mathsf{Sharing\ Algo}: s \in \{0,1\} \mapsto (share_1, \ldots, share_n).$$

The secret sharing scheme is associated to a monotone boolean function $\mathsf{F}: \{0,1\}^n \to \{0,1\}$, such that for any subset of parties $T \subseteq [n]$,

$$\mathsf{F}(T) = 1 \implies s \text{ can be recovered from } \{share_i\}_{i \in T},$$

$$\mathsf{F}(T) = 0 \implies s \text{ is independent from } \{share_i\}_{i \in T}.$$

One of the major long-standing questions in information-theoretic cryptography is to minimize the (total) size of the shares in a secret sharing scheme for arbitrary monotone functions F. [Ito-Saito-Nishizeki'89]

Previous Works

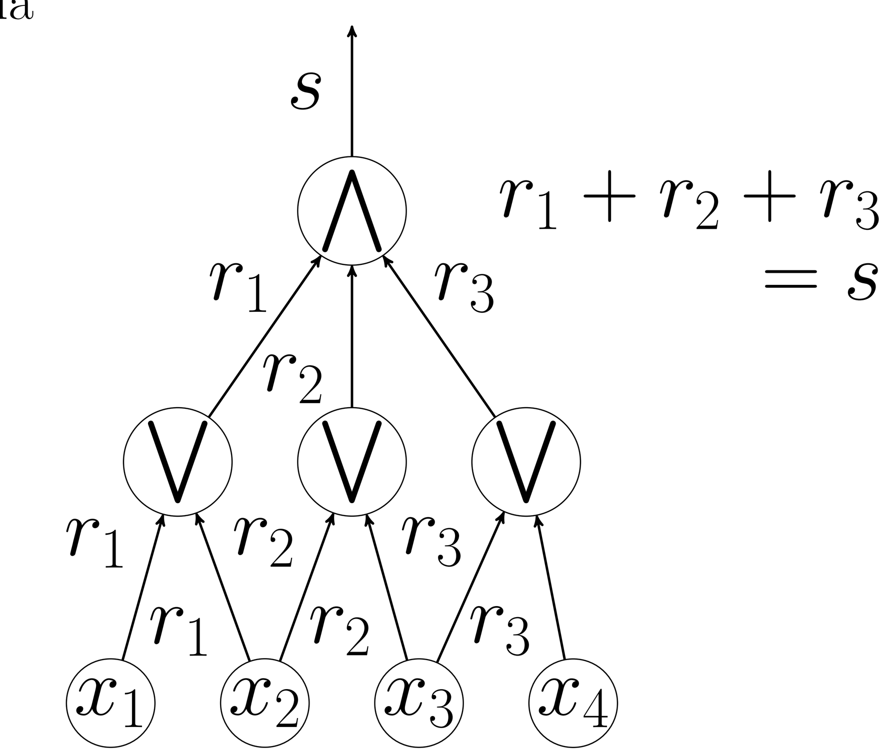
	General Secret Sharing	Linear Secret Sharing*
Upper Bounds: $\forall\mathsf{F}$, the share size is no more than	2^n (naïve solution) $O(\text{monotone formula size}) \le \frac{2^n}{\text{poly}(n)}$ [Benaloh-Leichter'88] $O(\text{monotone span program size}) \le \frac{2^n}{\text{poly}(n)}$ [Karchmer-Wigderson'93]	the same
Lower Bounds: $\exists\mathsf{F}$, the share size is no less than	$\frac{n^2}{\log n}$ [Csirmaz'97]	$\frac{2^{n/2}}{\text{poly}(n)}$

Our Results

	General Secret Sharing	Linear Secret Sharing*
New Upper Bounds: $\forall\mathsf{F}$, the share size is no more than	$2^{0.994n}$	$2^{0.999n}$

Formula-Based Secret Sharing and its Bottleneck

- Monotone function F is computed by a monotone formula
- Generate a tag for each wire
  - Output wire: the secret s
  - AND gate: additively share its output wire tag
  - OR gate: copy its output wire tag
- The i-th party's share: all tags of input wire x_i
- Total share size $\approx$ formula size of $F \le 2^n/poly(n)$ [Benaloh-Leichter'88]



Representation size barrier:

$$\text{formula size} \times \log(\#\text{ base gates}) \ge \log(\#\text{ monotone functions}) = \frac{2^n}{\text{poly}(n)}$$

Proof Outline

Every monotone function has secret sharing scheme with share size $2^{0.994n}$, which is the corollary of the following two theorems.

[Liu-Vaikuntanathan-Wee'18]

Every **slice functions** — function F s.t.

$$\|x\| > n/2 \implies \mathsf{F}(x) = 1 \text{ and}$$

$$\|x\| < n/2 \implies \mathsf{F}(x) = 0,$$

has a secret sharing scheme /w share size $2^{\tilde{O}(\sqrt{n})}$.

[This work]

Every monotone function can be computed by a monotone formula s.t.

- Formula size: $2^{0.994n}$
- Constant depth
- Base gates: AND, OR, slice functions

Open Problems

- Every monotone function is computed by a monotone formula of size $2^{o(n)}$ using slice functions as gates? (It implies every monotone function has a secret sharing scheme with $2^{o(n)}$ share size.)
- Does amortization help improve information ratio?

Secret Sharing for all Functions [This work] $\Longleftarrow$ Secret Sharing for Slice Functions $\Longleftarrow$ Multi-party Conditional Disclosure of Secret [LVW'18] $\Longleftarrow$ 2-party Conditional Disclosure of Secret [LVW'17] $\Longleftarrow$ 2-server PIR [Yek'08,Efr'09,DG'15]

LaTeX TikZposter