Bounding Threshold Formulas using Multiparty Communication Complexity

Shiv Kampani

Columbia University

Spring 2024

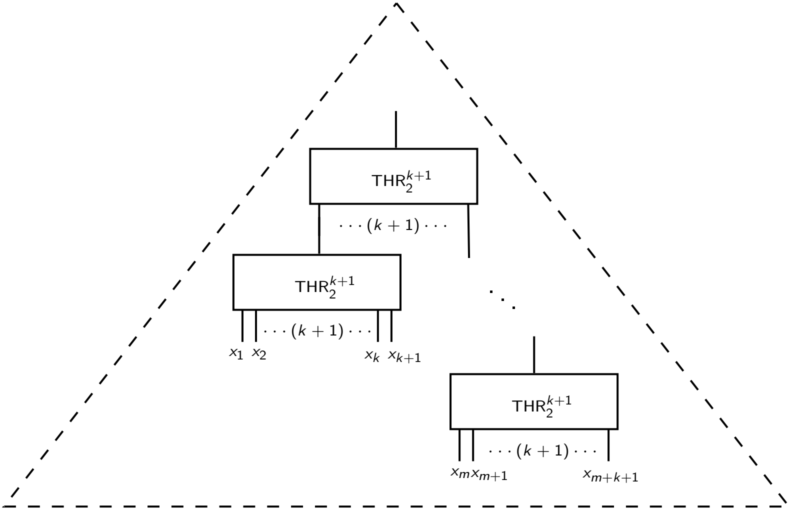
Threshold Gates

Definition (Threshold)

$$\mathrm{THR}_a^b : \{0,1\}^b \to \{0,1\} \qquad \mathrm{THR}_a^b(x) = \begin{cases} 1 & x \text{ contains} \geq a \text{ ones} \\ 0 & \text{otherwise} \end{cases}$$

We denote THR_2^3 as MAJ_3 gates. In general, we will focus on THR_2^{k+1} gates for values of $k \geq 2$. Threshold formulas consist of variables and THR_2^{k+1} gates **only**.

Threshold Formulas



Motivation

How do we design **secure multiparty computation** protocols?

Let $\mathcal{N}$ be a set of n players who wish to compute a function f on their private inputs securely with respect to an adversary.

Definition (Adversaries and Structures)

An adversary $A \subseteq \mathcal{N}$ "corrupts" a subset A of players. A structure $\mathcal{S}$ is a set of adversaries.

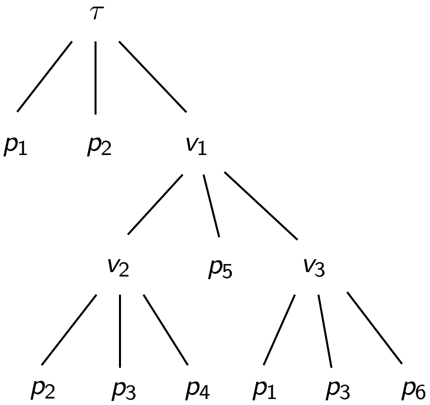
This is trivial in the presence of a **trusted** party τ (ideal protocol).

Definition (Security)

Protocol P is A-secure if the adversary "can do no better" in the absence of a trusted party τ. ($\mathcal{S}$-secure if A-secure $\forall A \in \mathcal{S}$).

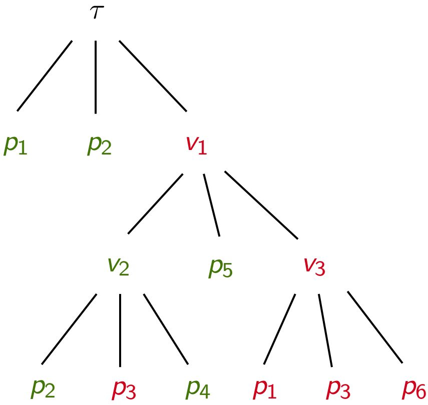
Player Emulation

[Hirt-Maurer'00] presented 3-party secure protocols for an honest majority. To construct n-party protocols, we can recursively apply 3-party secure protocols to *emulate* the trusted party τ in the ideal protocol:



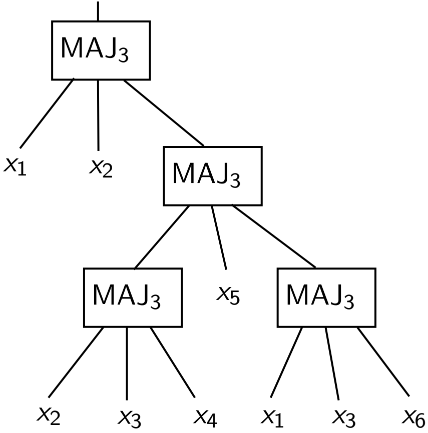
Player Emulation

Q. What structure $\mathcal{S}$ is this protocol secure against?
Is it secure against the adversary $A = \{p_1, p_3, p_6\}$?



Player Emulation

Q. What structure $\mathcal{S}$ is this protocol secure against?
A. The function f computed by the threshold formula below!



Low-depth threshold formulas $\iff$ efficient SMPC protocols.

Q_k Functions

Definition (Q_k Functions)

$f : \{0,1\}^n \to \{0,1\}$ is a Q_k function if $\forall x^1, x^2, \ldots, x^k \in f^{-1}(0)$
$\exists i \in [n] : x_i^1 = x_i^2 = \cdots = x_i^k = 0$.

Example. $\mathrm{THR}_{n+1}^{kn+1}$ is a Q_k function on $kn+1$ bits.

Theorem (folklore)

$\forall f : \{0,1\}^n \to \{0,1\}$ such that f is Q_k, $\exists$ formula F consisting of only THR_2^{k+1} gates and variables such that $F \leq f$. Equivalently, $\forall x \in \{0,1\}^n \; f(x) = 0 \implies F(x) = 0$.

Literature Review (Upper Bounds)

Of specific interest is the problem of computing MAJ_{2n+1} using MAJ_3-formulas (security against **honest-majority** structures).

1. $O(\log n)$-depth monotone formula for MAJ_{2n+1} [Ajtai-Komlos-Sz'12]
2. Poly-time computable $O(n)$-depth MAJ_3-formula for MAJ_{2n+1} [Hirt-Maurer'00].
3. Poly-time computable $O(\log n)$-depth MAJ_3-formula for computing MAJ_{2n+1} $\forall x \in \{0,1\}^{kn+1} : |\mathrm{wt}(x) - \frac{1}{2}| \geq 2^{-O(\sqrt{\log n})}$ [Cohen'13].

Literature Review (Upper Bounds)

Of specific interest is the problem of computing MAJ_{2n+1} using MAJ_3-formulas (security against **honest-majority** structures).

1. $O(\log n)$-depth monotone formula for $\mathrm{MAJ}_{2\mathrm{n}+1}$ [Ajtai-Komlos-Sz'12]
2. Poly-time computable $O(n)$-depth MAJ_3-formula for MAJ_{2n+1} [Hirt-Maurer'00].
3. Poly-time computable $O(\log n)$-depth MAJ_3-formula for MAJ_{2n+1} $\forall x \in \{0,1\}^{kn+1} : |\mathrm{wt}(x) - \frac{1}{2}| \geq 2^{-O(\sqrt{\log n})}$ [Cohen'13].
4. Poly-time computable $O(\log n)$-depth MAJ_3-formula for MAJ_{2n+1} [Kozachinskiy-Podolskii'20].

Literature Review (Upper Bounds)

Generalization: computing $\mathrm{THR}_{n+1}^{kn+1}$ using THR_2^{k+1}-formulas.

1. Exponential-time computable $O(n)$-depth $\mathrm{THR}_2^{\mathrm{k}+1}$-formula for $\mathrm{THR}_{n+1}^{kn+1}$ [Cohen'13].
2. Poly-time computable $O(\log n)$-depth $\mathrm{THR}_2^{\mathrm{k}+1}$-formula for $\mathrm{THR}_{n+1}^{kn+1}$ $\forall x \in \{0,1\}^{kn+1} : |\mathrm{wt}(x) - \frac{1}{k}| \geq \Omega(\frac{1}{\sqrt{\log n}})$ [Cohen'13].
3. Poly-time computable $O(\log^2 n)$-depth $\mathrm{THR}_2^{\mathrm{k}+1}$-formula for $\mathrm{THR}_{n+1}^{kn+1}$ [Kozachinskiy-Podolskii'20].

Literature Review (Upper Bounds)

Generalization: computing $\mathrm{THR}_{n+1}^{kn+1}$ using THR_2^{k+1}-formulas.

1. Exponential-time computable $O(n)$-depth THR_2^{k+1}-formula for $\mathrm{THR}_{n+1}^{kn+1}$ [Cohen'13].
2. Poly-time computable $O(\log n)$-depth THR_2^{k+1}-formula for $\mathrm{THR}_{n+1}^{kn+1}$ $\forall x \in \{0,1\}^{kn+1} : |\mathrm{wt}(x) - \frac{1}{k}| \geq \Omega(\frac{1}{\sqrt{\log n}})$ [Cohen'13].
3. Poly-time computable $O(\log^2 n)$-depth THR_2^{k+1}-formula for $\mathrm{THR}_{n+1}^{kn+1}$ [Kozachinskiy-Podolskii'20].

... via **communication complexity!**

(Monotone) Karchmer-Wigderson Games

First, consider monotone boolean functions.

Definition (Monotone Functions)

$f : \{0,1\}^n \to \{0,1\}$ is **monotone** if $\forall x, y \in \{0,1\}^n$ we have:
$\forall i \in [n]\ x_i \leq y_i \implies f(x) \leq f(y)$

Idea. We can relate the formula depth-complexity of monotone boolean functions with the communication cost of strategies in a game (called the mKW_f game).

(Monotone) Karchmer-Wigderson Games

First, consider monotone boolean functions.

Definition ($\mathrm{mKW_f}$ Game)

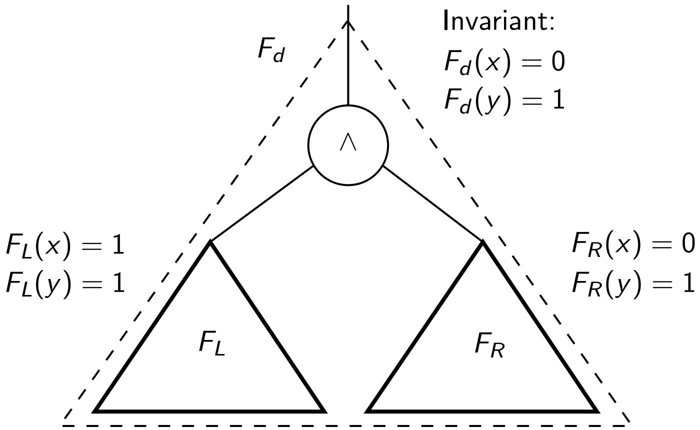
For some monotone function f, we define a two-player communication game as follows: Alice gets $x \in f^{-1}(0)$, Bob gets $y \in f^{-1}(1)$. The goal is to output any $i : 0 = x_i \neq y_i = 1$ (this is always possible).

Theorem

$CC(\mathrm{mKW_f}) = d(f)$ *(min # bits exchanged = monotone depth of f)*

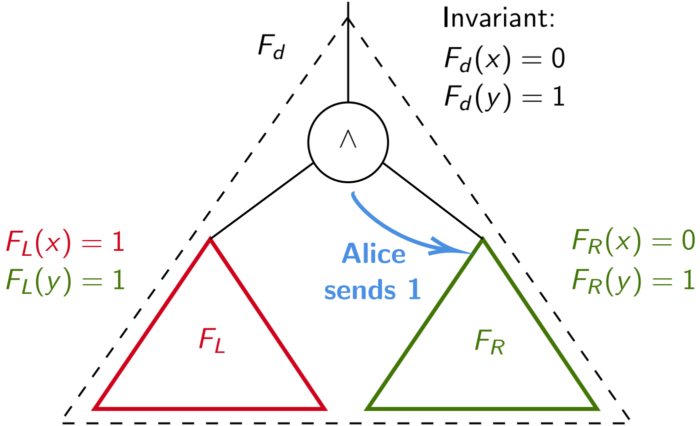
Monotone Formula $\implies$ Protocol

For every depth-d formula F_0 for monotone f, we can specify a d-round communication strategy for mKW_f game. After d-rounds:



Monotone Formula $\implies$ Protocol

Invariant holds on the right, so Alice sends a 1.



After d-rounds, $F_d = x_i$ (variable). Alice, Bob output i.

Q_2 Communication Games

Remark. Q_2 functions are equivalent to monotone and *self-dual*.

Definition (Self-dual Functions)

$f : \{0,1\}^n \to \{0,1\}$ is **self-dual** if $\forall x \in \{0,1\}^n$ we have $f(\neg x) = \neg f(x)$.

The following game is equivalent to the mKW_f game for Q_2 functions.

Definition (Q_2 Game)

For some Q_2 function f, we define a two-player communication game as follows: Alice, Bob get $x, y \in f^{-1}(0)$. The goal is to output any $i : x_i = y_i = 0$ (this is always possible).

Multiparty (NIH) Q_k Communication Game

We can generalize Q_2 games to the multi-party setting; private inputs and 'shared blackboard' communication (Number-In-Hand model).

Definition (Q_k Communication Game)

For some Q_k function f, we define a k-player communication game (in the NIH model) as follows: $(p_1, p_2, \ldots, p_k) \leftarrow (z^1, z^2, \ldots, z^k) \in (f^{-1}(0))^k$. The goal is to output any $i : z_i^1 = z_i^2 = \cdots = z_i^k = 0$.

Such an index i is guaranteed to exist by definition of a Q_k function.

Multiparty (NIH) Q_k Communication Game

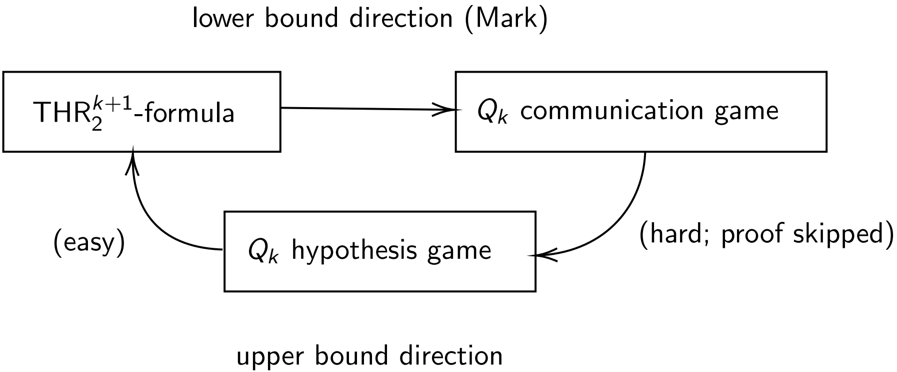
[Kozachinskiy-Podolskii'20] proved this generalized Q_k game has the desirable KW-like property:

Theorem

For any Q_k function f, the communication complexity of its Q_k communication game is equal (upto a constant factor) to the minimum depth of a THR_2^{k+1}-formula that lower bounds f.

Remark. We treat k as a constant; the number of players is always small.

Proof Structure [KP'20]



Hypothesis Games

Definition (Q_k Hypothesis Game)

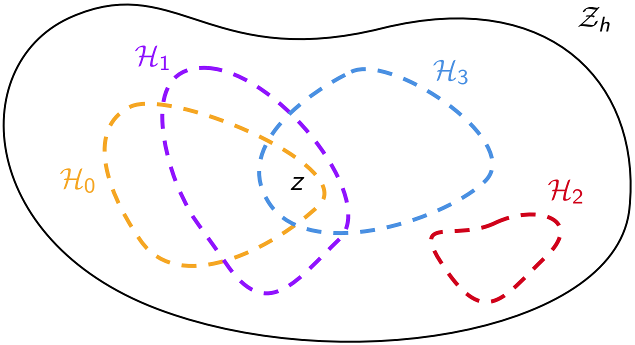
For some Q_k function f, we define the Q_k hypothesis game as follows. There are two players: Nature and Learner. Nature chooses some $z \in f^{-1}(0)$ that is kept private from Learner. In each round, Learner makes the following $k+1$ hypotheses:

$$z \in \mathcal{H}_0, z \in \mathcal{H}_1, \ldots, z \in \mathcal{H}_k$$

If $< k$ of these hypotheses are satisfied, Learner loses the game. Else, Nature communicates the index j of some true hypothesis $\mathcal{H}_j$ to Learner and the game continues. Learner's goal is to output an index $i : z_i = 0$.

Hypothesis Games (e.g. $k = 4$)

Let $\mathcal{Z}_h$ be the set of 'alive' strings after h-rounds. In round $(h+1)$, Learner guesses $\mathcal{H}_0, \ldots, \mathcal{H}_3$. Nature may reply with 0, 1, or 3 but **not** 2.



Hypothesis Strategy $\implies$ THR_2^{k+1} Formula (easy)

Any d-round hypothesis strategy for a Q_k function f can be converted to a depth-d THR_2^{k+1}-formula that lower-bounds f (the converse also holds).

Proof.

Assume that we have a d-round hypothesis strategy for f. We can interpret the strategy as a $(k+1)$-ary tree by considering each round as a node and the 'outputs' corresponding to the bits sent by Nature. We can transform this tree to a formula by placing a THR_2^{k+1}-gate at each node and the variable x_i at each terminal where Learner outputs index i.

We claim that this formula lower bounds f. This is because, for any path p from root s to a node v, $\forall z$ 'alive' after p, $F_v(z) = 0$. □

MAJ_3 formula for MAJ_{2n+1} [KP'20]

Poly-time computable $O(\log n)$-depth MAJ_3-formula for MAJ_{2n+1}.

Proof.

1. There exists an $O(\log n)$-depth monotone formula for computing $\mathrm{MAJ_{2n+1}}$ (AKS sorting network).
2. Can construct an $O(\log n)$-size communication protocol for mKW game for $\mathrm{MAJ_{2n+1}}$.
3. Reduction from Q_2 game for $\mathrm{MAJ_{2n+1}}$ to mKW game.
4. Can construct $O(\log n)$-depth MAJ_3 formula for computing $\mathrm{MAJ_{2n+1}}$ from the Q_2 protocol.

□

THR_2^{k+1} formula for $\mathrm{THR}_{n+1}^{kn+1}$ [KP'20]

Poly-time computable $O(\log n)$-depth MAJ_3-formula for MAJ_{2n+1}.

Proof.

Suffices to show an efficient $O(\log^2 n)$-bit communication protocol for Q_k game for $\mathrm{THR}_{\mathrm{n+1}}^{\mathrm{kn+1}}$. The idea is that we can perform a binary search over indices. Let all players maintain indices $\ell, r \in [kn+1]$.

Initially, $\ell \leftarrow 1, r \leftarrow kn+1$. In each round, players share count of 1's on both halves ($O(\log n)$ bits) and choose the half that satisfies the invariant:

$$\sum_{i=1}^{k} |\mathrm{supp}_{\ell:r}(z^i)| < |r - \ell + 1|$$

After $O(\log n)$ rounds, ℓ, r narrow to an index i, which players output. $\square$

Open Problems (Upper Bounds)

Open Problem #1

$O(\log n)$-depth (with **small constant**), deterministic poly-time computable MAJ_3-formula for computing MAJ_{2n+1}.

Open Problem #2

$o(\log^2 n)$-depth deterministic poly-time computable THR_2^{k+1}-formula for computing $\mathrm{THR}_{n+1}^{kn+1}$.