Small Error Versus Unbounded Error Protocols in the NOF Model

Arkadev Chattopadhyay∗ and Nikhil S. Mande†

†School of Technology and Computer Science, TIFR, Mumbai

Abstract

We show that a simple function has small unbounded error communication complexity in the $k$-party number-on-forehead (NOF) model but every probabilistic protocol that solves it with sub-exponential advantage over random guessing has cost essentially $\Omega\left(\sqrt{\frac{n}{k}}\right)$ bits. This yields the strongest known explicit separation up to $k \leq \delta \log n$ players, where $\delta < 1$ is a constant. After an initial manuscript of our work was published, Sherstov [31] pointed out to us that such a strong separation can also be obtained by carefully combining a result implicit in [29] with explicit results of [28, 30]. The analysis done in our work, inspired by the work of Goldmann, Håstad and Razborov [17] from the nineties, is more elementary and direct. The alternate route of combining results and ideas of much more recent work requires the use of approximation theory and other tools.

Our result has the following consequence for boolean Threshold circuits: let THR and MAJ denote respectively the classes of linear threshold functions that have unbounded weights and polynomially bounded weights. Further, let $\text{PAR}_k$ ($\text{ANY}_k$) denote the class of functions that are parities of $k$ bits (any $k$-junta). For every $2 \leq k \leq \delta \log n$, we show that there exists a function in linear size $\text{THR} \circ \text{PAR}_k$ that needs $2^{\Omega(1)}$ size to be computed by every circuit in the class $\text{MAJ} \circ \text{SYM} \circ \text{ANY}_{k-1}$, where $\text{SYM}$ represents the class of all symmetric functions. Applying a result of Goldmann et al. [17] to the above, similar lower bounds on the size of circuits of the form $\text{MAJ} \circ \text{THR} \circ \text{ANY}_{k-1}$ for computing the function follow.

The main technical ingredient of our result is to show that a composed function of the form $f \circ \text{XOR}$ has inverse exponentially small discrepancy while $f$ has sign degree just 1. An interesting aspect of our work is that the block size of the inner XOR function is fixed to 1, independent of the number of players $k$.

∗arkadev.c@tifr.res.in
†nikhil.mande@tifr.res.in
1 Introduction

Chandra, Furst and Lipton [10] introduced the “number-on-forehead” (NOF) model of multi-party communication, over thirty years ago, to obtain lower bounds on the size of branching programs. In this model, there are $k$ players each having an input that is metaphorically held on their foreheads. Every forehead is visible to a player except her own. The two features that make this model much more subtle than its classical two-party counterpart, are the mutual overlap of information and the fact that as $k$ grows, each player misses less information. Indeed, starting with the surprising work of Grolmusz [18], several work (see for example [3, 1, 15]) have shown that there are very counter-intuitive protocols especially when $k$ is larger than $\log n$. This makes proving multi-party lower bounds on the cost of protocols quite challenging. However, researchers have been well motivated to take on this challenge due to many well known applications of such lower bounds in diverse areas like circuit complexity, proof complexity, and pseudo-random generators. More recently new applications have emerged in areas like data-structures [23] and distributed computing [16].

In a seminal work, Babai, Frankl and Simon [2] introduced communication complexity classes for the 2-party model. Corresponding to polynomial time being the notion of efficient computation on a Turing machine, [2] argues that protocols with poly-log (of input length) communication cost is a natural notion of efficient protocols. Armed with this notion, most computational complexity classes have their analogues in communication complexity. This also extends easily to the NOF model and gives rise to complexity classes $P_k^c$, $BPP_k^c$, $NP_k^c$, $PP_k^c$ etc. While it is very hard to separate Turing machine complexity classes, many separation in the communication world is known when $k = 2$. For instance Equality function easily separates $P_2^c$ from $BPP_2^c$. Set-Disjointness famously separates $BPP_2^c$ from $PP_2^c$. However, for $k \geq 3$ things become much more delicate. While for $k \geq 3$ Beame et al. [5] separated $P_k^c$ from $BPP_k^c$ not too long ago, it is still outstanding to find an explicit function witnessing this separation for even $k = 3$. A very recent line of work [22, 13, 12, 28, 30, 24] showed that Set-Disjointness also separates $BPP_k^c$ and $PP_k^c$ for $k \leq \delta \cdot \log n$ for some constant $\delta < 1$.

In this paper, we consider the class $PP_k^c$. Babai et al. realized that the Turing machine complexity class $PP$ has two different natural versions in the communication world. Let $\epsilon$ be the advantage of a probabilistic protocol over random guessing. Then, one way to measure cost of the protocol is to sum up the total number of bits communicated in the worst case with $\log \frac{1}{\epsilon}$. Functions that admit $k$-party probabilistic protocols of poly-logarithmic cost in this model form the class $PP_k^c$. The other model is unrestricted: it does not penalize by adding the $\log \frac{1}{\epsilon}$ term to the cost, i.e. the cost is just the total number of bits communicated in the worst case. Protocols in this model are allowed to use only private random coins (see Section 2.1) and must, on each input, have non-zero advantage over random guessing. Functions that have efficient $k$-party protocols in this model form the class $UPP_k^c$. It is not difficult to see $PP_k^c \subseteq UPP_k^c$. The fact this inclusion is strict for $k = 2$ was relatively recently shown independently by Buhrman, Vereshchagin and de Wolf [8] and by Sherstov [26]. The two works use two different functions. However the corresponding separation question for $k \geq 3$ players remained unaddressed in the literature.

Our main theorem in this work separates $PP_k^c$ from $UPP_k^c$ for $k = \Theta(\log n)$. The function we use to achieve this is a simple and natural extension of the function defined by Goldmann, Hästad and Razborov [17], which is as follows:

**Definition 1.** Let  

$$P(x, y_1, \ldots, y_k) \equiv \sum_{i=0}^{n-1} \sum_{j=0}^{\lfloor \frac{k}{2} \rfloor - 1} 2^j y_{1j} \cdots y_{kj} (x_{i, 2j} + x_{i, 2j+1})$$
where \( x \in \{ \pm 1 \}^{2n^2 4^k} \), \( y_i \in \{ \pm 1 \}^{n^4 4^k} \) for each \( i \).

Then, \( \text{GHR}_k^N(x, y_1, \ldots, y_k) \equiv \text{sgn}(2^P(x, y_1, \ldots, y_k) + 1) \), where \( N = 2n^2 4^k \).

Note that the function \( \text{GHR}_k^N \) is a \( k \) + 1-partite function for which in a \( k \) + 1-party communication game the inputs are assigned to players in the following natural way: inputs \( x, y_1, \ldots, y_k \) are held on foreheads of Player 1, Player 2, . . ., Player \( k + 1 \) respectively. Our main theorem is given below.

**Theorem 1** (Main Theorem). Let \( \Pi \) be any \( k \) + 1-party probabilistic public-coin protocol solving the \( \text{GHR}_k^N \) function with advantage \( \epsilon > 0 \). Then,

\[
\text{Cost}(\Pi) + \log \left( \frac{1}{\epsilon} \right) \geq \Omega \left( \frac{\sqrt{N}}{4^k} - \log N - k \right).
\]

Observe that Theorem 1 lower bounds precisely the cost of a \( \text{PP}^{cc}_{k+1} \) protocol computing \( \text{GHR}_k^N \). On the other hand, note that \( \text{GHR}_k^N \) is a composition of a linear threshold function with \( N \) parities of arity \( k + 1 \). A well known simple fact (refer to Section 3 for a proof) says that every such function has a \( \text{UPP}^{cc}_{k+1} \) protocol of cost \( O(\log N) \). This immediately yields the following separation result:

**Corollary 1.** For all \( 1 \leq k \leq \delta \cdot \log n \), the \( \text{GHR}_k^N \) function is not in \( \text{PP}^{cc}_{k+1} \) but is in the class \( \text{UPP}^{cc}_{k+1} \), where \( \delta > 0 \) is some constant.

**Remark 1.** After an initial manuscript of our work appeared as [14], Sherstov [31] pointed out to us that a separation of \( \text{PP}^{cc}_{k} \) and \( \text{UPP}^{cc}_{k} \) can be also derived in a different and shorter way by a careful combination of results from previous works. However, this route uses more general results from multi-party communication complexity and additional tools from approximation theory. Our argument is direct and self-contained using first principles.

An additional motivation for our work comes from the study of constant-depth circuits with Threshold gates. There are two types of Threshold gates that have been considered in the literature. The first one is with unbounded weights and the class of such gates is denoted by THR. The second is with polynomially bounded weights, called Majority gates. We denote the class of such gates by MAJ. Goldmann et al. [17] showed that although THR is strictly contained in MAJ  ⊙ MAJ, a simple function computable by linear size THR  ⊙ PAR needs exponential size to be computed by MAJ  ⊙ SYM circuits, where SYM denotes gates computing arbitrary symmetric functions. We strengthen their result to depth-three circuits as follows:

**Theorem 2.** For each \( k \geq 2 \), the function \( \text{GHR}_k^N \in \text{THR} \circ \text{PAR}_{k+1} \) needs size

\[
2^\Omega \left( \frac{\sqrt{N}}{4^k} - \log N \right)
\]

to be computed by depth-three circuits of the form \( \text{MAJ} \circ \text{SYM} \circ \text{ANY}_k \).

Let us remark that Theorem 2 continues to yield non-trivial bounds as long as \( k < \delta \log n \), for a certain constant \( \delta > 0 \). It is also worth noting that a result of [17] immediately yields, from the above theorem, the following interesting result:

**Corollary 2.** The function \( \text{GHR}_k^N \) can be computed very efficiently by THR  ⊙ PAR_{k+1} circuits but requires \( 2^\Omega \left( \frac{\sqrt{N}}{4^k} - \log N \right) \) size to be computed by depth-three circuits of the form \( \text{MAJ} \circ \text{THR} \circ \text{ANY}_k \).

### 1.1 Related Work

In this section, we present an overview of earlier work that could also be used to derive, in a completely different way, a separation of \( \text{PP}^{cc}_{k} \) and \( \text{UPP}^{cc}_{k} \). This was brought to our notice by an anonymous reviewer and Alexander Sherstov in private communication [31] after an
initial manuscript of our work was published as a technical report [14]. The reader who is more interested in the technique that we use, may skip to the next section.

The standard way to lower bound the cost of \( PP^c_k \) protocols for computing a function \( f \) is to establish upper bounds on the discrepancy over cylinder intersections of \( f \) under an appropriately chosen input distribution. Doing this for \( k \geq 3 \) is delicate and essentially the only known method is due to Babai, Nisan and Szegedy [4]. This general prescription has been refined in Raz [25] and then successfully applied for some functions [11, 6, 28, 30, 1]. However, trying to directly use these ideas faces the following problem: First, almost all of these works bound the discrepancy of a composed function of the form \( h \circ g \), where \( g \) is some nicely behaved function and \( h \) crucially has high sign degree. The fact that \( h \) has high sign degree seems to make the composed function difficult for \( UPP^c_k \) protocols. In particular, when \( k = 2 \) and \( h \) is symmetric, Sherstov [27] proves that such functions have high sign rank and consequently are hard for even \( UPP^c_2 \) protocols. This gives rise to a natural challenge of proving multi-party discrepancy bounds when \( h \) has low sign degree.

However, some results are now known when \( h \) has low sign degree. In a recent work, Theorem 5.7 of Sherstov [30] upper bounds the discrepancy of a composed function \( F = f \circ g \) in terms of the \( \epsilon \)-approximate degree of \( f \) denoted by \( \deg_{\epsilon}(f) \) and a quantity called the repeated discrepancy of \( g \). After an initial manuscript of our work was put out, an anonymous reviewer and Sherstov [31] pointed out that this line of work could also be combined with results implicit in earlier work to provide a more off-the-shelf separation of \( UPP^c_k \) from \( PP^c_k \). The best \( PP^c_k \) lower bound that one would get using functions obtained this way is \( \Omega(n^{2/5}) \) which is weaker than the \( n^{1/2} \) bound obtained in our Theorem 1. However, Sherstov [31] observed that by using additional tools from approximation theory one can use earlier works to obtain stronger lower bounds. In particular, one can probabilistically infer the existence of functions with sign degree 1 and \( \deg_{1-2^{-\lambda(n)}} = \Omega(n) \).

There is a basic technical difference between our method and the ones outlined above. The route of combining earlier work uses unique-disjointness as the inner function whose block size grows exponentially fast with \( k \). Here, the block size is defined to be \( \frac{1}{k} \) times the size of the input to each copy of the inner function. With such an inner function, the previous techniques work with any outer function, like ODD-MAX-BIT, that has large approximation error for any polynomial of degree sufficiently smaller than \( n \). This is in contrast to our use of XOR as the simple inner function with fixed block size of just 1 for all \( k \). It is not very difficult to see that ODD-MAX-BIT \( \circ \) XOR has very efficient \( PP^c_k \) protocols for all \( k \geq 2 \). Thus, our argument has to exploit some feature of the outer function that is not possessed by functions like ODD-MAX-BIT. We find this an independently interesting aspect of the technique used in this work.

In summary, progress on separating communication complexity classes in the NOF model has been slow. This work is the first one to explicitly address the question of separating \( PP^c_k \) and \( UPP^c_k \) for \( k > 2 \). After an initial manuscript of our work was put up [14], it was pointed out [31] that a super-polynomial separation can also be obtained by combining the works of [7, 32], and a very recent work of [30]. These routes use tools from approximation theory. The best of these separations\(^2\) yields \( PP^c_k \) lower bounds of \( \Omega(n^{2/5}) \) that is quantitatively weaker than our lower bound of \( \Omega((\sqrt{n})^3) \).\(^3\) In a private communication [31], it was pointed out that one can match our \( \sqrt{n} \) bounds by taking a more indirect route to combine a few other explicit and implicit results from previous works. Our argument, on the other hand, is direct, requires less background and provides one of the best known separations through

---

\(^1\) A very recent result of Bun and Thaler [9], which was published recently after an initial version of this work came out [14], constructs an explicit but more involved function of low sign degree which still require \( \Omega(n^{2/3-\delta}) \) degree to be \( \epsilon \)-approximated even when \( \epsilon \) is exponentially close to 1. This yields functions with \( \Omega(n^{2/3-\delta}) \) \( PP_k \) cost, but are in \( UPP^c_k \).

\(^2\)Barring the use of very recent result by Bun and Thaler[9] which just appeared as a technical report

\(^3\)See Footnote 1.
an explicit function whose NOF complexity has not been analyzed before. This is also an arguably simpler function which uses composition with XOR functions. Previous techniques do not seem to work for functions that use XOR as an inner function.

1.2 Our Proof Technique and Organization

We work with the GHR function which is easily seen to be the composition of the universal threshold function [20] and Parity. The universal threshold function derives its name from the fact that by setting some of its bits appropriately one can induce any arbitrary threshold function. In that sense, it is the hardest function of sign degree 1. To estimate the discrepancy of GHR^N_k, we extend ideas from [17] who estimated this in the setting of two players. The basic intuition can be seen after observing that for a given setting of y_1, . . . , y_k the GHR^N_k function essentially depends on the sign of a plus-minus combination of A_j’s for 0 ≤ j ≤ n4^k − 1, where

\[ A_j = \frac{1}{2} \sum_{i=0}^{n-1} 2^i (x_{2i,j} + x_{2i+1,j}). \]

The relevant sign of each A_j depends on the parity of the bits y_{1,j}, . . . , y_{k,j}. Further, the set of bits in x that each A_j depends on is disjoint from the set of bits that A_{j'} depends on, whenever j ≠ j'. We sample x such that each A_j are i.i.d. binomial distributions centered at 0 with range [−2^n + 1, 2^n − 1]. Let this distribution be \( \mu_x \). We sample each y_i uniformly at random. We want to ensure that the GHR^N_k function, under this distribution, behaves in a way that leaves the players with little clue about the outcome unless the relevant sign to be associated with each A_j is determined. To do this, as done in [17], one is forced to sample in a slightly more involved way: first sample y’s uniformly at random. Then sample x according to \( \mu_x \), conditioned on the fact that \( P = \sum_{j=0}^{n4^k-1} A_j y_{1,j} \cdots y_{k,j} \) is very close to its mean compared to its standard deviation (which is as high as 2^\( \Omega(n) \)). Note that the median of each A_j is 0, which gives us plenty of room to exploit. This turns out to be the hard distribution but to establish this requires technical work.

Organization: Section 2 develops the basic notions and lemmas. Section 3 establishes our main technical result, Theorem 3, which upper bounds the k-wise discrepancy of the GHR function. Using this, we prove Theorem 1 and Corollary 1. Section 4 derives the circuit consequences of Theorem 2 and Corollary 2. Finally, Section 5 concludes with some open problems.

2 Preliminaries

2.1 The NOF model

In the k-party model of Chandra et al.[10], k players with unlimited computational power wish to compute a function \( f : X_1 \times \cdots \times X_k \to \{-1, 1\} \) on some input \( x = (x_1, \ldots, x_k) \). For the purpose of this paper, we consider inputs of the form \( X_i \in \{-1, 1\}^{n_i} \). On input \( x \), player \( i \) is given \( (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_k) \), which is why it is figuratively said that \( x_i \) is on the \( i \)th player’s forehead. Players communicate by writing on a blackboard, so every player sees every message. We denote by \( D_k(f) \) the deterministic k-party communication complexity of \( f \), namely the number of bits exchanged in the best deterministic protocol for \( f \) on the worst case input.

A probabilistic protocol \( \Pi \) with access to public (private) randomness computes \( f \) with advantage \( \epsilon \) if the probability that \( \Pi \) and \( f \) agree is at least \( 1/2 + \epsilon \) for all inputs. The cost of \( \Pi \) is the maximum number of bits it communicates over it’s internal random choices in

\footnote{Here we consider the most natural case where the inputs to the various instances of the inner function are pairwise disjoint.}
the worst case. Let us define \(R_ε(f)\) to be the cost of the best such protocol. Note that for convenience, we deviate from the notation defined in [21]. We now define two other notions.

\[
PP_k(f) ≡ \min_ε \left[ R_ε^{pub}(f) + \log \left( \frac{1}{ε} \right) \right], \quad
UPP_k(f) ≡ \min_ε \left[ R_ε^{priv}(f) \right]
\]  

(1)

Note that privateness of the random coins is essential in the definition of \(UPP_k\). It is a simple exercise to show that every function can be computed using 2 bits if allowed public coins. Define \(PP_{cc}^k = \{ f : PP_k(f) = \text{polylog}(n) \}\) and \(UPP_{cc}^k = \{ f : UPP_k(f) = \text{polylog}(n) \}\), where \(n\) is the maximum length of an input to a player.

2.2 Cylinder intersections, discrepancy and the cube norm

Let \(f : X_1 \times \cdots \times X_k \to \{-1, 1\}\). A subset \(S_i \subseteq X_1 \times \cdots \times X_k\) is a cylinder in the \(i\)th direction if membership in \(S\) does not depend on the \(i\)th coordinate. A subset \(S\) is called a cylinder intersection if it can be represented as the intersection of \(k\) cylinders, \(S = \bigcap_{i=1}^k S_i\), where \(S_i\) is a cylinder in the \(i\)th direction.

**Definition 2.** Let \(µ\) be a distribution on \(X_1 \times \cdots \times X_k\). The discrepancy of \(f\) according to \(µ\), \(Disc_µ^k(f)\) is

\[
\max_S \left| \Pr_µ[f(x_1, \ldots, x_k) = 1 \land (x_1, \ldots, x_k) \in S] - \Pr_µ[f(x_1, \ldots, x_k) = -1 \land (x_1, \ldots, x_k) \in S] \right|
\]

where the maximum is taken over all cylinder intersections \(S\).

The \(k\) in \(Disc_µ^k\) denotes the dimension of the underlying cylinder intersections. We will drop this superscript when it is clear from the context what \(k\) is. Let \(Disc(f) = \min_µ Disc_µ^k(f)\).

The discrepancy method is a powerful tool that lower bounds the randomized communication complexity in terms of the discrepancy. The following lemma can be found for example in [21].

**Lemma 1.** \(R_ε(f) \geq \log(2ε/\text{Disc}(f))\).

We now recall a useful technique that upper bounds the discrepancy of a function under a product distribution. It is a standard lemma and can be found in [12] and [25] for example.

**Lemma 2.** Let \(f : X \times Y_1 \times \cdots \times Y_k \to \{-1, 1\}\), and \(µ\) any product distribution. Then,

\[
(Disc_µ^{k+1}(f))^{2^k} \leq \mathbb{E}_{y_1, y_1', \ldots, y_k, y_k'} \left[ \mathbb{E}_x \Pi_{a \in \{0, 1\}^k} f(x, y_a^0, \ldots, y_a^{2^k}) \right]
\]

2.3 The binomial distribution

**Definition 3.** Let \(B(N)\) denote the distribution obtained as the sum of \(2N\) independent Bernoulli variables, each of which take values \(1/2, -1/2\) with probability \(1/2\) each.

A few important things to observe are that \(B(N)\) takes only integral values, it is centered and symmetric around 0, so \(B(N)\) is identically distributed to \(-B(N)\). Its range is \([-N, N]\).

Let us denote \(\Pr[B(N) = 0]\) by \(p_0\). It is a well known fact that \(p_0 = \binom{2N}{N}/2^N = \Theta \left( \frac{1}{N^{1/2}} \right) \).

The following lemma tells us that the probability of a binomial distribution taking any value close to its mean is significantly high.

**Lemma 3.** Let \(W\) be a binomial random variable distributed according to \(B(N)\). Let \(p_0\) denote \(\Pr[W = 0]\). Then for all \(j \in [-N, N]\),

\[
p_0 - O \left( \frac{j^2}{N^{3/2}} \right) \leq \Pr[W = j] \leq p_0
\]
Proof. Note that for $|j| \geq N/2$, the bound to be proved is trivial. Thus we assume $|j| < N/2$.

$$\Pr[W = j - 1] - \Pr[W = j] = \left[ \binom{2N}{N + j - 1} - \binom{2N}{N + j} \right] \cdot \frac{1}{2^{2N}}$$

$$= \left[ \frac{(2N)!}{(N + j - 1)! (N - j + 1)!} - \frac{(2N)!}{(N + j)! (N - j)!} \right] \cdot \frac{1}{2^{2N}}$$

$$= \frac{(N + j - 1)! (N - j)!}{(2N)!} \cdot \frac{2j - 1}{2j + 1} \cdot \frac{1}{2^{2N}}$$

$$\leq \left( \frac{2N}{N} \right) \cdot \frac{1}{2^{2N}} \cdot \frac{2j}{N - j}$$

since the middle binomial coefficient is the maximum. Thus, we have $\forall i, |i| \leq j$,

$$\Pr[W = i - 1] - \Pr[W = i] \leq \left( \frac{2N}{N} \right) \cdot \frac{2j}{N - j} \cdot \frac{1}{2^{2N}}$$

Since $\left( \frac{2N}{N} \right) = \Theta \left( \frac{1}{N^{1/2}} \right)$,

$$\Pr[W = 0] - \Pr[W = j] \leq \sum_{i=1}^{j} |\Pr[W = i - 1] - \Pr[W = i]| \leq \frac{2j^2}{N - j} \cdot O \left( \frac{1}{N^{1/2}} \right)$$

$$\leq \frac{2 \cdot 2j^2}{N} \cdot O \left( \frac{1}{N^{1/2}} \right)$$

$$\leq O \left( \frac{j^2}{N^{3/2}} \right)$$

Since $|j| \leq N/2$.

\[\square\]

3 A discrepancy upper bound for the multiparty GHR function

In this section, we prove essentially an $O \left( 2^{-\sqrt{N/4^k}} \right)$ upper bound on the discrepancy of the GHR$_k^N$ function where the first player gets $N$ input bits. This gives us an inverse exponential upper bound on the discrepancy if $k < \epsilon \log(N)$ for any constant $\epsilon$. Goldmann et al. [17] showed that when $k = 2$, if there is a low cost one-way protocol for GHR$_2^N$, then it must have low advantage. Sherstov [26] noted that the same proof technique can be adapted to prove an upper bound on the discrepancy on GHR$_2^N$. We generalize this for higher $k$. In particular, we show

**Theorem 3.** For any $k \geq 1$,

$$\text{Disc}(\text{GHR}_k^N) = O \left( \frac{(8e)^k N^{1/4}}{2^{\sqrt{N/4^k}} \cdot 2^{k/2}} \right),$$

where GHR$_k^N$ is defined as in Definition 1, and $N$ is the maximum number of bits a player gets (in this case the first player).

**Proof of Theorem 1.** It follows directly from Theorem 3 and Lemma 1.  \[\square\]
Proof of Corollary 1. From Theorem 1, it follows that for all \(1 \leq k \leq \delta \cdot \log n\), the \(\text{GHR}_k\) function is not in \(\text{PF}_{k+1}^c\) where \(\delta > 0\) is some constant. Let us see an easy unbounded error protocol for \(\text{GHR}_k\). Note that all the weights of the top threshold are positive. One player chooses and announces a bottom layer Parity gate with probability proportional to it’s corresponding weight. The cost of announcing this is \(O(\log(N))\). The probability of success equals \(\sum w_i^+ / w\), where \(w_i^+\)’s are the weights of the gates which agree with the output. Since \(\sum w_i^+ > \sum w_i^-\) (the weights of the gates which disagree with the output), the probability of success is strictly greater than 1/2.

Recall that \(N = 2n^24^k\). The proof technique of Theorem 3 is inspired from that of Goldmann et al. [17].

Proof of Theorem 3. Let \(A_j = \frac{1}{2} \sum_{i=0}^{n-1} 2^i(x_{i,2j} + x_{i,2j+1})\). It is easy to see that \(A_j\) can take any integer value in \([-2^n + 1, 2^n - 1]\). Let \(\mu_x\) be a distribution on the \(x\)’s that make the \(A_j\)’s independent and binomially distributed according to \(B(2^n - 1)\) as defined in Definition 3. Such a distribution exists because each \(A_j\) depends on a disjoint set of variables. For each \(i \in \{1, \ldots, k\}\), let \(\mu_i\) be the uniform distribution on the \(y_i\). We choose a tuple \((x, y_1, \ldots, y_k)\) by first picking \(y_i \sim \mu_i\) independently for each \(i\), and then picking \(x \sim \mu_x\) under the condition that \(|P(x, y_1, \ldots, y_k)| = 2^k\). Let us define \(\mu\) to be the distribution obtained by this sampling procedure.

We will now show an upper bound on the discrepancy of \(\text{GHR}_k\) under the distribution \(\mu\). Let \(S\) denote the characteristic function \((0-1\) valued\) of a cylinder intersection. By Definition 2, the discrepancy of \(\text{GHR}_k\) according to \(\mu\) is

\[
\text{Disc}_\mu(\text{GHR}_k) = \max_S \left( |\mathbb{E}_\mu[\text{GHR}_k(x, y_1, \ldots, y_k)S(x, y_1, \ldots, y_k)]| \right)
\]

The following lemma will enable us to switch to working with a product distribution on the inputs, for which we have nice techniques for proving discrepancy upper bounds, namely Lemma 2.

Lemma 4. For \(\mu_x, \mu_1, \ldots, \mu_k\) as defined above,

\[
\Pr_{\mu_x, \mu_1, \ldots, \mu_k} [ |P(x, y_1, \ldots, y_k)| = 2^k ] \geq \Omega \left( \frac{1}{\sqrt{n}2^{(n+2k)/2}} \right)
\]

Proof. We will show that for any fixed \(y_1, \ldots, y_k\), if we sample \(x\) according to \(\mu_x\), then

\[
\Pr_{\mu_x} [ |P(x, y_1, \ldots, y_k)| = 2^k ] \geq \Theta \left( \frac{1}{(n+2k)(n+1)/2} \right) - O \left( \frac{4^k}{(n+2k)(n+1)^3/2} \right)
\]

We could discard the second term since it equals \(O \left( \frac{1}{(n+2k)(n+1)^3/2} \right)\), and is dominated by the first term.

Define a function \(q\) by

\[
q(x, y_1, \ldots, y_k) = \begin{cases} P(x, y_1, \ldots, y_k)/2^k & \text{if } |P(x, y_1, \ldots, y_k)| = 2^k \\ 0 & \text{otherwise} \end{cases}
\]
This means that if \((x, y_1, \ldots, y_k)\) is chosen according to the distribution \(\mu_x \times \mu_1 \times \cdots \times \mu_k\), then \(q(x, y_1, \ldots, y_k) = \text{GHR}_k^N(x, y_1, \ldots, y_k)\) on the domain of \(\mu\), and 0 otherwise. Using Lemma 4,

\[
\text{Disc}_\mu(\text{GHR}_k^N) \leq \max_S \left| \mathbb{E}_{\mu_x, \mu_1, \ldots, \mu_k} [q(x, y_1, \ldots, y_k) S(x, y_1, \ldots, y_k)] \right| \cdot O\left(\sqrt{n/2}^{n+1}2^{k} \right) \tag{3}
\]

where \(S\) denotes a cylinder intersection. This can be seen by expanding the expectation in the above equation as a sum, and comparing the sum to that obtained when expanding as in Equation 2 term by term. We can then use the definition of conditional probability to obtain the above inequality. It suffices to show that for all cylinder intersections \(S\),

\[
\left| \mathbb{E}_{\mu_x, \mu_1, \ldots, \mu_k} [q(x, y_1, \ldots, y_k) S(x, y_1, \ldots, y_k)] \right| \leq O\left(2^{- \frac{n+2k}{2}} \epsilon\right) \tag{4}
\]

for some constant \(\epsilon > 0\) to give us an inverse exponential discrepancy. For notational convenience, we may switch between the notations \(\mathbb{E}_x\) and \(\mathbb{E}_{x \sim \mu_x}\) from now on. Now that we have a product distribution, we can use Lemma 2,

\[
\left| \mathbb{E}_{\mu_x, \mu_1, \ldots, \mu_k} [q(x, y_1, \ldots, y_k) S(x, y_1, \ldots, y_k)] \right| \leq \left( \mathbb{E}_{\mu_x, \mu_1, \ldots, \mu_k} \left| q(x, y_{a_1}, \ldots, y_{a_k}) \prod_{a_1, \ldots, a_k \in \{0,1\}} \right| \right)^{1/2} \tag{5}
\]

We will now upper bound the RHS of the above equation by splitting the outer expectation into two terms, the first of which has low probability. We will require certain properties of Hadamard matrices to upper bound the second term. Let \(\alpha \in \{0,1\}^k\). Define \(2^k\) subsets of indices as \(I_\alpha = \{ j \in [n^k] : \forall i \in [k], (y_1^0)_j = 1^{\alpha_i}.(y_1^1)_j \}\). Note that \(\{I_\alpha : \alpha \in \{0,1\}^k\}\) forms a partition of the indices. Since our distribution on \(y_1^0, y_1^1\)'s are uniform and independent, the probability of each \(I_\alpha\) being empty is equal. An easy counting gives us the probability of \(I_\alpha\) being empty as \((2^k - 1)2^{k-1} n^{4k}\). By a union bound, the probability that any one of them is empty is at most \(2^k \cdot \left(2^k - 1 \right)2^{k-1} n^{4k}\). We have the following.

\[
\left( \mathbb{E}_{y_1^0, y_1^1, \ldots, y_k^0, y_k^1} \left| \mathbb{E}_x \left| \prod_{a_1, \ldots, a_k \in \{0,1\}} q(x, y_{a_1}, \ldots, y_{a_k}) \right| \right| \right)^{1/2} \leq \left( 2^k \left( 1 - \frac{1}{2^k} \right) n^{4k} + Z \right)^{1/2k}
\]

where \(Z = \mathbb{E}_{y_1^0, y_1^1, \ldots, y_k^0, y_k^1} \mathbb{P}_{\alpha, I_\alpha \neq \emptyset} \mathbb{E}_x \prod_{a_1, \ldots, a_k \in \{0,1\}} q(x, y_{a_1}, \ldots, y_{a_k})\|

**Claim 1.** For all \(y_1, \ldots, y_k, y_1^0, \ldots, y_k^1\) such that \(I_\alpha\) is non-empty for each \(\alpha \in \{0,1\}^k\), we have

\[
\left| \mathbb{E}_x \left| \prod_{a_1, \ldots, a_k \in \{0,1\}} q(x, y_{a_1}, \ldots, y_{a_k}) \right| \right| \leq O\left(2^k \log(\epsilon)2^{k} \cdot 2^{2k} \cdot \frac{1}{(2n/2)^{2k-1}} \cdot \frac{2^{(k+1)2^{k+1}}}{2^{3n/2}} \right)
\]
Let us assume the claim to be true for now. We have from Equation 3 that

\[
\begin{align*}
\text{Disc}_\mu(\text{GHR}^N_k) &\leq |E_{\mu,\mu_1,\ldots,\mu_k}[q(x, y_1, \ldots, y_k)S(x, y_1, \ldots, y_k)]| O\left(\sqrt{n}2^{n/2k}\right) \\
&\leq \left(2^k \left(1 - \frac{1}{2^k}\right)^{n/2} + O\left(2^{k\log(\epsilon)2^k} \cdot 2^{2k} \frac{1}{(2n/2)^{2k-1}} \cdot \frac{2(k+1)2^{k+1}}{23n/2}\right)\right)^{1/2^k} \\
&\leq \left(2^{k/2^k} \left(1 - \frac{1}{2^k}\right)^{n/2^k} + O\left(\frac{(4\epsilon)^k}{(2\epsilon^2)^{1/2^k} \cdot 2^{n/2k} \frac{1}{2^k}}\right)\right) O\left(\sqrt{n}2^{n/2k}\right) \\
&\leq O\left((e^{-1/2^k})^{n/2^k} \cdot 2^{n/2+k+k/2} \cdot \sqrt{n} + \frac{(8\epsilon)^k \sqrt{n}}{2^{n/2^k}}\right)
\end{align*}
\]

Using the fact that \((1 - \frac{1}{\beta}) < e^{-1/\beta}\)

\[
= O\left(e^{-n} \cdot 2^{n/2+k+k/2} \cdot \sqrt{n} + \frac{(8\epsilon)^k \sqrt{n}}{2^{n/2^k}}\right) = O\left(\frac{(8\epsilon)^k \sqrt{n}}{2^{n/2^k}}\right)
\]

Assuming \(k < n/3\)

which proves Theorem 3. Assuming \(k \leq \epsilon \log(n)\) for any constant \(\epsilon < 1\) gives us an inverse exponential upper bound on the discrepancy. \(\square\)

Now it only remains to prove Claim 1.

### 3.1 Proof of Claim 1

Recall that we need to show the following. For all \(y_1^0, \ldots, y_k^0, y_1^1, \ldots, y_k^1\) such that \(I_\alpha\) is non-empty for each \(\alpha\), we want

\[
\left|E_x \left[ \prod_{a_1, \ldots, a_k \in \{0,1\}} q(x, y_1^{a_1}, \ldots, y_k^{a_k}) \right] \right| \leq O\left(2^{k\log(\epsilon)2^k} \cdot 2^{2k} \frac{1}{(2n/2)^{2k-1}} \cdot \frac{2(k+1)2^{k+1}}{23n/2}\right)
\]

Fix any such \(y_1^0, \ldots, y_k^0, y_1^1, \ldots, y_k^1\). Note that the LHS of the above equation is

\[
\left|Pr_x \left[ \prod_{a_1, \ldots, a_k \in \{0,1\}} q(x, y_1^{a_1}, \ldots, y_k^{a_k}) = 1 \right] - Pr_x \left[ \prod_{a_1, \ldots, a_k \in \{0,1\}} q(x, y_1^{a_1}, \ldots, y_k^{a_k}) = -1 \right] \right|
\]

For convenience, for all \(a \in \{0,1\}^k\) let us denote \(P(x, y_1^{a_1}, \ldots, y_k^{a_k})\) by \(P_a\) and let \(S_a\) denote \(P_a/2\). By the definition of \(q\), we have

\[
\left|E_x \left[ \prod_{a \in \{0,1\}^k} q(x, y_1^{a_1}, \ldots, y_k^{a_k}) \right] \right| = \left| Pr_x \left[ \prod_{a \in \{0,1\}^k} \frac{P_a}{2^k} = 1 \right] - Pr_x \left[ \prod_{a \in \{0,1\}^k} \frac{P_a}{2^k} = -1 \right] \right|
\]

\[
= \left| Pr_x \left[ \prod_{a \in \{0,1\}^k} S_a = 2^{(k-1)2^k} \right] - Pr_x \left[ \prod_{a \in \{0,1\}^k} S_a = -2^{(k-1)2^k} \right] \right| \tag{6}
\]

Let \(W_\alpha = \sum_{j \in I_\alpha} A_j(y_1^0)_j \ldots (y_k^0)_j\). It will be useful to note here that \(W_\alpha\) only takes integral values. We will use this fact crucially later. Let \(P_k\) denote the \(2^k \times 1\) matrix whose rows are indexed by \(a = (a_1, \ldots, a_k) \in \{0,1\}^k\), and the \(a\)th row of \(P_k\) is \(P(x, y_1^{a_1}, \ldots, y_k^{a_k})\). Similarly define matrices \(S_k\) and \(W_k\), whose \(a\)'th entries are \(S_a\) and \(W_a\) respectively for all \(a \in \{0,1\}^k\).
Claim 2. The following holds true for all $k$

$$P_k = 2S_k = 2H_k \cdot W_k$$

where $H_k$ is a $2^k \times 2^k$ Hadamard matrix defined as $H_k = \begin{bmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & -H_{k-1} \end{bmatrix}$ and $H_0 = [1]$.

Let us first state a well known property of $H_k$ where $(H_k)_{ij} = (-1)^{i+j}$ for all $i, j \in \{0, 1\}^k$.

**Fact 1.** Let $H_k$ be as defined above. Then, $(H_k)_{ij} = (-1)^{(i+j)}$ for all $i, j \in \{0, 1\}^k$.

In other words, $H_k$ is the communication matrix of the inner product (modulo 2) function. Let us now prove Claim 2.

**Proof.** Let $a \in \{0, 1\}^k$. $P_a = 2 \sum_{j=1}^{n^k} A_j(y^{a_1}_j)_j \cdots (y^{a_k}_j)_j$ and $W_a = \sum_{j \in I_a} A_j(y^0_1)_j \cdots (y^0_k)_j$.

Say $j \in I_a$ where $\alpha \in \{0, 1\}^k$. Note that $(y^0_i)_j = -1 \cdot (y^0_i)_j$ iff $a_i = 1, \alpha_i = 1$. Hence $(y^1_i)_j \cdots (y^k_i)_j = (-1)^{(\sum_i a_i - \alpha_i)} (y^0_i)_j \cdots (y^0_k)_j = (-1)^{(\alpha_i)} (y^0_i)_j \cdots (y^0_k)_j$.

$$P_a = 2 \sum_{j=1}^{n^k} A_j(y^{a_1}_j)_j \cdots (y^{a_k}_j)_j = 2 \left( \sum_{\alpha \in \{0, 1\}^k} \sum_{j \in I_a} (-1)^{(\alpha_i)} A_j(y^0_1)_j \cdots (y^0_k)_j \right)$$

$$= 2(H_k)_a \cdot W_k$$

where $(H_k)_a$ denotes the ath row of $H_k$. Thus, $P_k = 2S_k = 2H_k \cdot W_k$. \hfill \Box

### 3.1.1 On integral solutions to Hadamard constraints

In this subsection, we will prove that the number of integral solutions to $W_k$ such that $\prod_{\alpha \in \{0, 1\}^k} S_a = 2^{(k-1)2^k}$ is equal to the number of integral solutions to $W_k$ such that $\prod_{\alpha \in \{0, 1\}^k} S_a = -2^{(k-1)2^k}$. Moreover, we show that the total number of integral solutions is small, and the values of $|W_a|$ are not too large in any integral solutions. Recall from Equation 6 that for all $y^0_1, \ldots, y^0_k, y^1_1, \ldots, y^1_k$ such that $I_a$ is non-empty for each $\alpha$, we have

$$\mathbb{E}_x \left[ \prod_{\alpha \in \{0, 1\}^k} q(x, y^{a_1}_1, \ldots, y^{a_k}_k) \right] = \Pr_x \left[ \prod_{\alpha \in \{0, 1\}^k} S_a = 2^{(k-1)2^k} \right] - \Pr_x \left[ \prod_{\alpha \in \{0, 1\}^k} S_a = -2^{(k-1)2^k} \right]$$

This allows us to pair the “positive” and “negative” solutions, and higher order terms in the difference of probabilities $\Pr_x \left[ \prod_{\alpha \in \{0, 1\}^k} S_a = 2^{(k-1)2^k} \right] - \Pr_x \left[ \prod_{\alpha \in \{0, 1\}^k} S_a = -2^{(k-1)2^k} \right]$ cancel out. We will require the following well known property of Hadamard matrices.

**Fact 2.** Let $H$ be an $N \times N$ Hadamard matrix. Then, $H$ is invertible, and $H^{-1} = \frac{1}{N} H$.

**Claim 3.** The number of integral solutions to $W_k$ such that $\prod_{\alpha \in \{0, 1\}^k} S_a = +2^{(k-1)2^k}$ equals the number of integral solutions such that $\prod_{\alpha \in \{0, 1\}^k} S_a = -2^{(k-1)2^k}$. 

Proof. The constraints we have are $H_k \cdot W_k = S_k$. Since $W_a$ is integral for all $a$, and $H_k$ is a $\pm 1$ matrix, this implies that $S_a$'s are integral as well. Thus, using Fact 2 we get $\frac{1}{2^k} H_k \cdot S_k = W_k$, or $H_k \cdot \frac{S_k}{2^k} = W_k$. Let us consider two cases, one where $\forall a \in \{0,1\}^k, |\frac{S_a}{2^k}| = 1/2$, and another where there exists an $a$ such that $|\frac{S_a}{2^k}| \neq 1/2$.

- Let us assume $\forall a, |\frac{S_a}{2^k}| = 1/2$. We show something slightly stronger, namely that every setting of each $\frac{S_a}{2^k}$ to $\pm 1/2$ gives us an integer solution to the $W_a$'s. Since $H_k$ is a $\pm 1$ matrix of even dimension, the parity of the number of appearances of $+1/2$ equals the parity of number of appearances of $-1/2$ in the sum $(H_k)_R \cdot \frac{S_a}{2^k}$, where $(H_k)_R$ is the $R$th row of $H_k$. This holds for every row $R$. Thus, $W_R$ is always an integer. This means the number of positive solutions equals the number of negative solutions in this case.

- The absolute value of $S_a$ must equal a power of 2 for each $a$ since the product of them is a power of 2. If there exists an $S_a$ whose value is not $\pm 2^{k-1}$, then there must exist an $S_b$ (consider the last such one) which is a multiple of $2^k$ since $\prod_{a \in \{0,1\}^k} S_a = \pm 2^{(k-1)2^k}$. Since $S_b/2^k$ is an integer, and we had an integral solution to $W_k$, flipping the sign of $S_b$ can change the value of any $W_c$ to $W_c \pm 2 \cdot S_b/2^k$, which remains an integer. This is a bijection between positive and negative solutions.

The following lemmas just require $H_k$ to be the $2^k \times 2^k$ Hadamard matrix as defined in Claim 2, $S_a$'s to be integer valued such that $\prod_{a \in \{0,1\}^k} S_a = \pm 2^{(k-1)2^k}$, and $H_k \cdot W_k = S_k$.

Lemma 5. The number of integral solutions to $W_k$ is at most $2^{k \log(e)2^k}$.

We will require the following standard fact about binomial coefficients.

Fact 3. For all $n$ and for all $k \in [n], \left(\frac{n}{2}\right)^k \leq \left(\frac{n}{k}\right)^k$.

Proof of Lemma 5. Suppose $\prod_{a \in \{0,1\}^k} S_a = \pm 2^{(k-1)2^k}$. This means we have to distribute $(k-1)^2$ powers of 2 among $2^k S_a$'s (which are all integers). This equals the number of non-negative integer solutions to $m_1 + \cdots + m_{2^k} = (k-1)^2$, which equals $\binom{k^2 - 1}{(k-1)^2}$. This is at most $\frac{(k^2 - 1)!}{(k-1)^2!}$, which is at most $\left(\frac{k^2 - 1}{(k-1)^2}\right)^{(k-1)^2}$ by Fact 3. Now we will use the fact that $1 + x \leq e^x$ and conclude that this is bounded above by $e^{k^2}$, which equals $2^{k \log(e)2^k}$. Each of these can give at most 1 integral solution to the $W_a$'s because the system of constraints is linearly independent.

We now state an upper bound on the value of $|W_a|$ in every integral solution.

Lemma 6. For all $a \in \{0,1\}^k, |W_a| \leq 2^{(k+1)2^k}$.

Proof. First note that for each $a$, $|W_a| \leq \sum_{a \in \{0,1\}^k} |S_a|/2^k$ since $H_k \cdot S_k = W_k$. We show that $\sum_{a \in \{0,1\}^k} |S_a|$ is at most $2^{k^2}$. Suppose not. By a simple averaging argument, there must be an $b$ such that $|S_b| > \frac{2^{k^2}}{2^k}$, which is $2^{k(2^k-1)}$, which is at least $2^{(k-1)2^k}$ if $k \geq 1$. But this is not possible since $\prod_{a \in \{0,1\}^k} S_a = \pm 2^{(k-1)2^k}$ and $S_a$'s are integers.
3.1.2 Using properties of the binomial distribution

Recall from Equation 6 that for all \( y_1^0, \ldots, y_k^0, y_1^1, \ldots, y_k^1 \) such that \( I_{\alpha} \) is non-empty for each \( \alpha \), we want to upper bound \( \Pr_x \left[ \prod_{a \in \{0, 1\}^k} S_a = 2^{(k-1)2^k} \right] - \Pr_x \left[ \prod_{a \in \{0, 1\}^k} S_a = -2^{(k-1)2^k} \right] \).

Recall that we defined \( W_a = \sum_{j \notin I_{\alpha_a}} A_j (y_1^0)_{j} \ldots (y_k^0)_{j} \). For any \( \alpha \in \{0, 1\}^k \), note that \( W_a \) is always distributed according to \( B(c_{\alpha}(2^n - 1)) \), where \( c_{\alpha} = |I_{\alpha}| \neq 0 \). We can prove this in a manner similar to that in the proof of Lemma 4. In Claim 3, we showed that the number of integral solutions to \( W_a \) such that \( \prod_{a \in \{0, 1\}^k} S_a = 2^{(k-1)2^k} \) equals the number of integral solutions such that \( \prod_{a \in \{0, 1\}^k} S_a = -2^{(k-1)2^k} \). Note that if the solution to \( W_a \) is not integral, then it has probability 0 since for each \( a, W_a \) takes only integral values. Let us call a solution to \( W_a \) to be positive if the corresponding value of \( \prod_{a \in \{0, 1\}^k} S_a = 2^{(k-1)2^k} \), and negative if the value of \( \prod_{a \in \{0, 1\}^k} S_a = -2^{(k-1)2^k} \). Arbitrarily pair up the positive and negative solutions. We will bound the difference of probabilities of each pair.

\[
\left| \Pr_x \left[ \prod_{a \in \{0, 1\}^k} S_a = 2^{(k-1)2^k} \right] - \Pr_x \left[ \prod_{a \in \{0, 1\}^k} S_a = -2^{(k-1)2^k} \right] \right| 
\leq \sum_{w, w'} |\Pr_x[W_a = w] - \Pr_x[W_a = w']|
\]

where \( w = (w_a)_{a \in \{0, 1\}^k}, w' = (w'_a)_{a \in \{0, 1\}^k} \) are positive and negative solutions respectively to \( W_a \) such that \( \prod_{a \in \{0, 1\}^k} S_a = \pm 2^{(k-1)2^k} \). The term \( \Pr_x[W_a = w] \) equals \( \Pr_x[I_{\alpha_a} \left( w_a = w_{a_a} \right) \prod_{a \in \{0, 1\}^k} S_a \right] \). In Lemma 6 we showed that for each \( \alpha \), the absolute value of \( W_a \) in any integral solution can be at most \( 2^{(k-1)2^k} \). Each \( W_a \) is distributed according to \( B(c_{\alpha}(2^n - 1)) \), \( c_{\alpha} > 0 \), since \( \forall \alpha \in \{0, 1\}^k, |I_{\alpha}| > 0 \).

Now for a particular positive solution \( w \), negative solution \( w' \) and any \( y_1^0, \ldots, y_k^0, y_1^1, \ldots, y_k^1 \) such that \( I_{\alpha} \) is non-empty for each \( \alpha \),

\[
|\Pr_x[W_a = w] - \Pr_x[W_a = w']| = |\Pr_\big[ \prod_{a \in \{0, 1\}^k} W_a = w_a \big] - \Pr_\big[ \prod_{a \in \{0, 1\}^k} W_a = w'_a \big]|
\]

By plugging in \( N = c_{\alpha}(2^n - 1) \) and \( j = 2^{(k+1)2^k} \) in Lemma 3, we obtain \( p_0 \geq \Pr_x[W_a = w_a] \geq p_0 - O \left( \frac{2^{(k+1)2^k+1}}{2^{3n/2}} \right), \) where \( p_0 = \Pr[W_a = 0] = O \left( \frac{1}{2^{3n/2}} \right). \) For convenience in calculations, let us say \( \Pr_x[W_a = w_a] \in \left( p_0 \pm O \left( \frac{2^{(k+1)2^k+1}}{2^{3n/2}} \right) \right). \) Recall that \( W_{\alpha_a} \)'s are independent of each other since they depend on disjoint variables. Thus,

\[
\left| \Pr_\big[ \prod_{a \in \{0, 1\}^k} W_a = w_a \big] - \Pr_\big[ \prod_{a \in \{0, 1\}^k} W_a = w'_a \big] \right| 
\leq \left( p_0 \pm O \left( \frac{2^{(k+1)2^k+1}}{2^{3n/2}} \right) \right)^{2^k} - \left( p_0 \pm O \left( \frac{2^{(k+1)2^k+1}}{2^{3n/2}} \right) \right)^{2^k} 
\leq \frac{2^k}{2^{3n/2}} - \frac{2^k}{2^{3n/2}}
\]

The last inequality holds because the highest order term after binomially expanding both terms is \( \frac{1}{2^{2k}} \), which cancel each other. Note that the sum of the binomial coefficients is \( 2^{2k} \), and each term after the first is at most \( \frac{1}{(2n/2)^{2k-1}} \cdot \frac{2^{(k+1)2^k+1}}{2^{3n/2}} \). Thus, the sum of the remaining terms can be bounded above by \( 2^{2k} \cdot \frac{1}{(2n/2)^{2k-1}} \cdot \frac{2^{(k+1)2^k+1}}{2^{3n/2}} \). By Claim 5, the number
of solutions (and hence number of pairs) is at most $2^{k \log(2) 2^k}$. Thus,

$$\left| \Pr_x \left[ \prod_{a \in \{0,1\}^k} s_a = 2^{(k-1)2^k} \right] - \Pr_x \left[ \prod_{a \in \{0,1\}^k} s_a = -2^{(k-1)2^k} \right] \right| \leq \sum_{w,w'} \left| \Pr_x [W_k = w] - \Pr_x [W_k = w'] \right| \leq 2^{k \log(2) 2^k} \cdot 2^{2k} \cdot \frac{1}{(2n/2)^{2k-1} - 2^{(k+1)2^k+1}}$$

which proves Claim 1. Using Equation 3, this proves Theorem 3.

4 Circuit Lower Bounds

In this section, we will show how we obtain depth-3 circuit lower bounds on the class MAJ $\circ$ THR $\circ$ ANY$_k$ for the GHR$_k^N$ function, which is in THR $\circ$ PAR$_{k+1}$. First let us state the results that were known prior to this work.

**Lemma 7** (Folklore). Any function $f \in \text{SYM} \circ \text{ANY}_k$ of size $s$ has a deterministic simultaneous $(k + 1)$ player protocol of cost $O(k \log(s))$ for any partitioning of the input bits.

**Proof.** Since each of the bottom layer gates have fan-in at most $k$, there must exist a player who sees all the inputs to it. The protocol decides beforehand which gate ‘belongs’ to which player. All players simultaneously broadcast their contribution to the top SYM gate using at most $\log(s)$ bits each.

A consequence of this is an upper bound for randomized protocols for depth-3 circuits, which may be found in [11] for example and is stated below without proof.

**Lemma 8** (Folklore). Given any function $f \in \text{MAJ} \circ \text{SYM} \circ \text{ANY}_k$ of size $s$, and any partition of the input bits, there exists a randomized protocol computing $f$ with advantage $\Omega(1/s)$ and cost $O(k \log(s))$.

Let us now prove Theorem 2.

**Proof.** Suppose GHR$_k^N$ could be computed by MAJ $\circ$ SYM $\circ$ ANY$_k$ circuits of size $s = 2^{\Theta(\sqrt{N}/4^k)}$. Using the protocol mentioned in Lemma 8, the cost of the protocol is $O(k \log(s))$ and advantage $\Omega(1/s)$. Using Theorem 1, $O(k \log(s) + \log(s)) \geq \Omega\left(\frac{\sqrt{N}}{4^k} - \log(N) - k\right)$, which gives $\log(s) \geq \Omega\left(\frac{\sqrt{N}}{4^k} - \log(N) - 1\right)$. Thus, $s \geq 2^{\Omega\left(\frac{\sqrt{N}}{4^k} - \log(N) - 1\right)} \geq 2^{\Omega\left(\frac{\sqrt{N}}{4^k} - \log(N)\right)}$.

By definition, MAJ $\circ$ MAJ $\subseteq$ MAJ $\circ$ SYM. Also, Goldmann et al. [17] (Theorem 26) showed that MAJ $\circ$ THR circuits can be simulated by MAJ $\circ$ MAJ circuits with a polynomial blowup. More precisely, a MAJ $\circ$ THR circuit of size $s$ can be simulated by a MAJ $\circ$ MAJ circuit of size $s^{\alpha} \cdot n^\beta$ for some constants $\alpha, \beta$. Hence, Corollary 2 follows by a similar proof as that of Lemma 8.

5 Conclusion

We have shown that GHR$_k^N$ needs essentially $\Omega(\sqrt{N}/4^k)$ cost to be solved in the PP$^cc_{k+1}$ model. Since it follows almost from the definition of GHR$_k^N$ that it has $O(\log(N))$ cost UPP$^cc_{k+1}$ protocols, this provides a separation of PP$^cc_k$ from UPP$^cc_{k+1}$ for the NOF model when $k \leq \delta \cdot \log N$ for some constant $\delta > 0$. In general, current techniques do not allow us to go beyond $\log N$ number of players to prove lower bounds for the cost of even deterministic protocols.

This remains one of the most interesting problems in NOF complexity. However, let us remark that for many of the functions used in the literature (see for example [18, 3, 1, 15]),
there are surprisingly efficient protocols when $k > \log N$. Moreover these protocols are typically deterministic and either simultaneous or barely interactive. On the other hand, we do not immediately see an efficient randomized interactive protocol for $GHR^N_k$ at $k > \log N$. This raises the following question: Is $GHR^N_k$ a hard function for even $k > \log N$?

Another question that may be within reach to answer is the following: our work shows that the $PP^{cc}_k$ complexity of $GHR^N_k$ is $\Omega(\sqrt{N})$ for any constant $k$. As mentioned in Section 1.1, Sherstov [31] exhibits functions with $\Omega(N)$ cost in $PP_k$ but have efficient $UPP_k$ protocols. The same work implies an explicit function in $UPP^{cc}_k$ which requires $\Omega(\sqrt{N})$ $PP_k$ cost. Can one come up with an explicit function in $UPP^{cc}_k$ that requires $\Omega(N)$ $PP_k$ cost? 5

Finally, proving super-logarithmic lower bounds for $UPP^{cc}_k$ protocols for any explicit function remains a very interesting challenge for even $k = 3$. Hansen and Podolskii [19] have shown that meeting this challenge is enough to yield super-polynomial lower bounds for THR ° THR circuits.

Acknowledgements

We are grateful to an anonymous reviewer for pointing out to us that the results of Sherstov [30] and Beigel [7] can be combined to get a separation between $PP^{cc}_k$ and $UPP^{cc}_k$ for $k$ at most $O(\log \log n)$. We would like to thank Alexander Sherstov [31] for communicating to us after we published an initial manuscript [14] that carefully combining results from previous works in fact yields a separation between $PP^{cc}_k$ and $UPP^{cc}_k$ for $k$ up to $\Theta(\log n)$ players, and that there exist functions with $\Omega(n)$ cost in $PP_k$ but have efficient $UPP_k$ protocols (this also implies an explicit function in $UPP^{cc}_k$ with $\Omega(\sqrt{n})$ $PP_k$ cost). We would also like to thank Kristoffer Hansen for directing our attention to the result of Goldmann, Håstad and Razborov [17].

References


5See Footnote 1


