The communication complexity of addition

Emanuele Viola[*]

December 12, 2013

Abstract

Suppose each of $k \leq n^{o(1)}$ players holds an n-bit number x_i in its hand. The players wish to determine if $\sum_{i \leq k} x_i = s$. We give a public-coin protocol with error 1% and communication $O(k \lg k)$. The communication bound is independent of n, and for $k \geq 3$ improves on the $O(k \lg n)$ bound by Nisan (Bolyai Soc. Math. Studies; 1993).

Our protocol also applies to addition modulo m. In this case we give a matching (public-coin) $\Omega(k \lg k)$ lower bound for various m. We also obtain some lower bounds over the integers, including $\Omega(k \lg \lg k)$ for protocols that are one-way, like ours.

We give a protocol to determine if $\sum x_i > s$ with error 1% and communication $O(k \lg k) \lg n$. For $k \geq 3$ this improves on Nisan's $O(k \lg^2 n)$ bound. A similar improvement holds for computing degree-$(k-1)$ polynomial-threshold functions in the number-on-forehead model.

We give a (public-coin, 2-player, tight) $\Omega(\lg n)$ lower bound to determine if $x_1 > x_2$. This improves on the $\Omega(\sqrt{\lg n})$ bound by Smirnov (1988). Troy Lee informed us in January 2013 that an $\Omega(\lg n)$ lower bound may also be obtained by combining a result in learning theory by Forster et al. (2003) with a result by Linial and Shraibman (2009).

As an application, we show that polynomial-size AC^0 circuits augmented with $O(1)$ threshold (or symmetric) gates cannot compute cryptographic pseudorandom functions, extending the result about AC^0 by Linial, Mansour, and Nisan (J. ACM; 1993).

[*] Supported by NSF grants CCF-0845003, CCF-1319206. Email: `viola@ccs.neu.edu`

Contents

1 Introduction

Consider the following problem.

Definition 1. In the k-player Sum-Equal problem, each of k players (or parties) holds in its hand an integer x_i of magnitude $|x_i| < 2^n$. The players want to determine if $\sum_{i \le k} x_i = s$. Both n and s, and a public random string, are known to all players.

Variants of Sum-Equal have been studied since at least the seminal paper by Chandra, Furst, and Lipton [CFL83]. In addition to its fundamental nature, this problem has several applications, as we shall see.

In the case of $k = 2$ players, Sum-Equal reduces (with no communication) to checking the equality of x_1 and $s - x_2$. Using public-coins, this can be solved with error, say, 1% and communication $O(1)$.[KN97, Example 3.13] Remarkably, the communication is independent of n. In the more challenging case $k \ge 3$, Nisan gives in his beautiful paper [Nis93] a randomized protocol with $O(k \lg(n + \lg k))$ communication. Note now the communication depends (logarithmically) on n. In his protocol, Player i communicates x_i modulo a small, randomly-chosen prime.

In this work we obtain a protocol whose communication is $O(k \lg k)$, independent of n. This improves on Nisan's result for $k = n^{o(1)}$, which we assume throughout.

Theorem 2. The k-player Sum-Equal problem has a randomized communication protocol with error ϵ and communication $O(k \lg(k/\epsilon))$. Moreover, there is no false negative.

In fact, the protocol is simultaneous: each players sends $O(\lg(k/\epsilon))$ bits to a referee who then outputs the answer. The referee is the only one who needs to know s. With some changes, the protocol also works when the addition is performed modulo m, for any m.

The Sum-Equal problem also arises in the following problem.

Definition 3. In the k-player Sum-Greater problem each of k players holds in its hand an integer number x_i of magnitude $|x_i| < 2^n$. The players want to determine if $\sum_{i \le k} x_i > s$. Both n and s, and a public random string, are known to all players.

In the case of $k = 2$ players, Nisan and Safra give a clever protocol with communication $O(\lg n)$.[Nis93] Their protocol is not explicit; along the way we give an explicit protocol. For $k \ge 3$ players, Nisan gives a protocol with communication $O(\lg^2 n)k$.[Nis93]

We improve Nisan's protocol to $O(\lg n)k \lg k$.

Theorem 4. The k-player Sum-Greater problem has a randomized communication protocol with error ϵ and communication $O(\lg n/\epsilon)k \lg k$.

The model considered so far, where Player i knows only the input x_i, is known as *number-in-hand.* Nisan applies protocols for Sum-Greater to the *number-on-forehead* model, where x_i figurately resides on the i-th player forehead; formally Player i knows all the inputs except x_i. A *degree-d polynomial-threshold function* $f : \{0,1\}^n \to \{0,1\}$, abbreviated d-PTF, is a function that can be written as the sign of a real-valued polynomial of degree d. Nisan uses the protocol for Sum-Greater to compute d-PTF with communication $O(d^3 \lg^2 n)$, in the number-on-forehead model with $k := d+1$ players. We improve this to $O(d^2 \lg d) \lg n$.

Theorem 5. Let $f : \{0,1\}^n \to \{0,1\}$ be a d-PTF. Then, under any partition, and for any $k \geq d+1$, f can be computed with error ϵ by a k-player number-on-forehead protocol with communication $O(d \lg d)(d \lg n + \lg 1/\epsilon)$.

Nisan uses the upper bound for d-PTF to obtain three lower bounds for functions with high $(d+1)$-player number-on-forehead communication, such as Generalized Inner Product (GIP) [BNS92]. He proves that for any d, to compute GIP: any circuit of d-PTF requires $\Omega(n/\lg^2 n)$ gates, any tree of d-PTF requires depth $\Omega(n/\lg^2 n)$, and any majority of d-PTF requires $2^{\Omega(n/\lg n)}$ gates.

Using our Theorem 5 one immediately improves the first two to $\Omega(n/\lg n)$ and the last one to $2^{\Omega(n)}$.

Below we develop another application of Theorem 5 to the study of the complexity of computing pseudorandom functions.

Cobham asked [Cob64] whether multiplication is harder than addition. We show that the variants of Sum-Equal and Sum-Greater with addition replaced by multiplication do not require more communication.

Lower bounds. In the special case of *private* randomness, it is known that 2-player Sum-Greater requires communication $\Omega(\lg n)$.[KN97] This bound was not known for public randomness (as in Def. 3). However, in [Smi88] Smirnov proves a round-communication tradeoff for 2-player Sum-Greater, according to Miltersen, Nisan, Safra, and Wigderson who reprove it in [MNSW98]. This tradeoff implies an $\Omega(\sqrt{\lg n})$ lower bound.

In this paper we obtain a tight $\Omega(\lg n)$ lower bound.

Theorem 6. The communication complexity of 2-player Sum-Greater is $\Omega(\lg n)$.

Troy Lee informed us in January 2013 that an $\Omega(\lg n)$ lower bound may also be obtained by combining a result in learning theory by Forster et al. [FSSS03] with a result by Linial and Shraibman [LS09]. After our work, Braverman and Weinstein obtain [BW12] yet another proof.

Together with our previous upper bound (Theorem 4), we obtain that for every $k = O(1)$, k-player Sum-Greater has communication complexity $\Theta(\lg n)$.

We then move to lower bounds for Sum-Equal. In light of its many applications, it is desirable to establish if our $O(k \lg k)$ upper bound (Theorem 2) is tight. As mentioned before, our $O(k \lg k)$ upper bound also works when the sum is modulo m, for any m. In this case we can show a matching lower bound, for various choices of m.

Theorem 7. The communication complexity of k-player Sum-Equal modulo m is $\Omega(k \lg k)$, when $m \geq k^{1/4}$ is either a prime or a power of 2.

Over the integers $\mathbb{Z}$ we prove lower bounds under additional restrictions. We define a type of protocol that generalizes one-way (and so even simultaneous): a k-player protocol is *sententious* if Player k speaks only once to announce the output.

Theorem 8. For sententious protocols, the communication complexity of k-player Sum-Equal over $\mathbb{Z}$ is $\Omega(k \lg\lg k)$.

As a corollary, we can rule out general protocols in which every player communicates $o(\lg\lg\lg k)$ bits.

Corollary 9. Let Π be a protocol for k-player Sum-Equal over $\mathbb{Z}$. Then some player must communicate $\geq 0.9 \lg\lg\lg k$ bits.

We have been unable to prove a stronger lower bound even for simultaneous protocols. We have also been unable to prove an $\omega(k)$ lower bound for general protocols. We note that our protocol for Sum-Equal is simultaneous, and in it every player communicates $O(\lg k/\epsilon)$ bits.

Pseudorandom functions. As an application of the previous Theorem 5, we show that truth-tables of poly(n)-size constant-depth AC^0 circuits augmented with $O(1)$ threshold gates cannot support cryptographic pseudorandom functions. Introduced in the seminal work [GGM86] of Goldreich, Goldwasser, and Micali, a pseudorandom function is a random function $F : \{0,1\}^n \to \{0,1\}$ such that for every c and sufficiently large n, any oracle algorithm M running time $\leq n^c$ has advantage $\leq 1/n^c$ in distinguishing F from a uniform random function from n bits to 1 bit. A threshold gate is a gate that computes a degree-1 polynomial-threshold function.

Our impossibility result generalizes the one by Linial, Mansour, and Nisan [LMN93] that applies to AC^0 circuits. We mention that the distinguisher in [LMN93] is simpler than ours. Our impossibility result also complements the well-known candidate pseudorandom functions by Naor and Reingold [NR04], which are computable by poly(n)-size, constant-depth AC^0 circuits augmented with poly(n) threshold gates, a.k.a. TC^0 circuits. (Cf. [GHR92] for the equivalence between threshold and majority gates.)

In fact our impossibility result also applies to circuits augmented with $O(1)$ arbitrary symmetric gates, such as parity or majority. (For this, previous communication lower bounds suffice.) In the case of a single majority gate, this improves on the quasipolynomial-time distinguisher by Razborov and Rudich [RR97], based on [ABFR94]; cf. [KL01].

Theorem 10. Let F be a distribution on functions from $\{0,1\}^n$ to $\{0,1\}$ such that each function in the support is computable by a circuit of size n^d, depth d, with d threshold (or arbitrary symmetric) gates, where d is a constant and n is sufficiently large.

Then there is a randomized, oracle algorithm D that runs in time n^b such that

$$\left|\Pr[D^F = 1] - \Pr[D^U = 1]\right| \geq 1 - o(1),$$

where b depends on d only, and U is a uniform function on n bits.

Table 1: Pseudorandom functions $F : \{0,1\}^n \to \{0,1\}$ computable by circuits of size poly(n) and depth $O(1)$.

Complexity class	Security	Reference
TC^0	Secure against time $t = t(n)$ under assumptions in [NR04] against times poly$(n)t(n)$	[NR04, NRR02]
AC^0 with Mod m gates, any m; CC^0	Secure against time $n^{\lg^c n}$ under assumptions in [NR04] against time $2^{n^{\Omega(1)}}$ (circuit depth depends on c)	Theorem 11
AC^0 with Mod m gates, prime m	Breakable in time $n^{\lg^c n}$ (c depends on circuit depth)	[RR97, KL01]
AC^0 with $O(1)$ threshold gates and $O(1)$ symmetric gates (e.g. parity, majority)	Breakable in time poly(n)	Theorem 10
AC^0	Breakable in time poly(n)	[LMN93]

Motivated by the result above, we ask how far one can push poly-time distinguishers. The papers [RR97, KL01] also give a quasipolynomial-time distinguisher for AC^0 circuits with (unboundedly many) Mod m gates, m a fixed prime. We show that in this model the running time cannot be improved to polynomial. This holds under standard cryptographic assumptions, such as the "2^{n^ϵ}-factoring assumption" which is reviewed later, and essentially says that one cannot factor n-bit Blum integers in time $< 2^{n^\epsilon}$. We show the same for the model of poly-size circuits consisting exclusively of Mod m gates, m a fixed composite (i.e., CC^0 circuits).

Theorem 11. Assume there is an $\epsilon > 0$ such that the 2^{n^ϵ}-factoring assumption holds. Then for every m, c, there is a pseudorandom function $F : \{0,1\}^n \to \{0,1\}$ computable by poly(n)-size AC^0 circuits with Mod m gates such that adversaries running in time $t(n) := 2^{\lg^c n} \geq n^{\omega(1)}$ have advantage $1/t(n)$ in distinguishing F from uniform.

The results on pseudorandom functions are summarized in Table 1.

Organization In §2 we give an overview of our techniques. The communication upper bounds for k-player Sum-Equal, Sum-Greater, and polynomial-threshold functions are proved in §3. In that section we also give upper bounds for multiplication, an explicit version of Nisan and Safra's 2-player protocol for Sum-Greater, and we prove an upper bound of a more technical nature, concerning computing Sum-Equal in the number-on-forehead model under an arbitrary partition. The lower bounds for 2-player Sum-Greater and k-player Sum-Equal are in §4. The results about pseudorandom functions are in §5.

2 Overview of techniques

Sum-Equal. We now discuss our Sum-Equal protocol, for simplicity in the case where the sum is modulo 2^n. The key idea is to use a family of hash functions $h : \mathbb{Z}_{2^n} \to \mathbb{Z}_{2^r}$ analyzed in the work [DHKP97, §2.2] by Dietzfelbinger, Hagerup, Katajainen, and Penttonen. On input x and seed a, the hash function outputs certain r bits of the product $a \cdot x$ over the integers. The protocol works as follows. Using the public randomness, the players agree on a hash function mapping n bits to $r := O(\lg k/\epsilon)$. Player i then communicates $y_i := h(x_i)$. One accepts if $\sum y_i - h(s)$ is close to 0, that is if $\sum y_i - h(s) \in \{-k+1, -k+2, \ldots, 0\}$.

The properties that are used of h are:

1. *almost linearity*: $\forall x, y : h(x) + h(y) = h(x+y) - b$ for some $b \in \{0, 1\}$,

2. $\forall z \neq 0$, $h(z)$ is unlikely to land around 0, that is, in the set $\{-k, -k+1, \ldots, k\}$.

In the case $\sum x_i = s$, by applying (1) several times, we see that the protocol accepts.

In the case $\sum x_i = z \neq s$, by (1) the protocol accepts only if $h(\sum x_i - s) = h(z)$ lands around 0. By (2) this is unlikely. This concludes the overview of our protocol for Sum-Equal.

It is instructive to compare the above hash function h with the hash function h' used in [Nis93], which takes the input modulo a small prime p. While h' is linear, the prime p must be $n^{\Omega(1)}$ regardless of the target error, resulting in $\lg n$-bit hashes. With h, we give up linearity and only have almost-linearity, but we can trade more accurately the size of the hashes with the error.

Given the many applications of taking numbers modulo small primes in complexity theory, especially "low-level" complexity, one may hope that the hash function h may find more applications. It works well when summing few, large numbers.

Sum-Greater. Our k-player Sum-Greater protocol builds on protocols by Nisan and Safra in [Nis93]. To explain our contribution, in the next paragraphs we discuss:

(1) a sub-optimal protocol with communication $O(\lg n) \lg \lg n$, obtained by combining [Nis93] and our Sum-Equal protocol,

(2) an additional idea by Nisan and Safra that reduces the communication to $O(\lg n)$ in the special case $k = 2$,

(3) the obstacle to utilizing the additional idea to the setting $k \geq 3$, and finally

(4) our protocol with communication $O(\lg n)$ for any k.

For simplicity we focus on the case of constant k and error, and non-negative $x_i, s \in \{0, 1\}^n$.

(1) in [Nis93] it is shown how to reduce an instance $\sum_i x_i >^? s$ where the x_i have n bits to another instance $\sum_i x'_i >^? s'$ where the x'_i have $n/2$ bits. This gives a recursive protocol with recursion depth $O(\lg n)$. Each reduction involves solving $k-1$ Sum-Equal problems. For this, one can use a randomized protocol for k-player Sum-Equal with error ϵ. To withstand the possible accumulation of the error in the $\lg n$ reductions, a naive implementation requires

$\epsilon \leq 1/\lg n$. Plugging our protocol for k-player Sum-Equal results in a protocol for Sum-Greater with communication $O(\lg n)\lg\lg n$.

(2) The special case $k = 2$ is simplified by the fact that 2-player Sum-Equal reduces easily to equality of x_1 and $s - x_2$. It turns out that this makes the recursive protocol in (1) correspond to a binary search for the most significant bit where x_1 and $s - x_2$ differ. Again, each comparison during this binary search corresponds to a 2-player Sum-Equal problem. If instead of solving these problems with error $\epsilon \leq 1/\lg n$ one solves them with a larger, constant error, then one is in the setting of performing *binary search with noisy comparisons.* For a recent account of the latter, and the many solutions available, we refer the reader to [BOH08]. Nisan and Safra use the cute random-walk-with-backtrack algorithm by Feige, Raghavan, Peleg, and Upfal [FRPU94]. The algorithm [FRPU94] shows that such a search can be accomplished with $O(\lg n)$ comparisons. The idea is to start each recursive call with a check that the target element is contained in the current interval, and if not backtrack. This yields an $O(\lg n)$ protocol for 2-player Sum-Greater.

(3) For $k \geq 3$ the situation is more complex because we actually have to deal with sums and carries. The above protocol (1) does not correspond anymore to performing binary search for the most significant bit where $\sum_{i \leq k} x_i$ and s (or $\sum_{i<k}$ and $s - x_k$) differ: the transcript of the protocol does not determine that bit. (This is proved at the end of §3.2.)

(4) *Our* $O(\lg n)$ *protocol for any* k. We show that the ideas in [FRPU94] can be modified to again execute Nisan's protocol while solving the Sum-Equal problems with error as large as constant. At a high level, our approach is similar to [FRPU94]: we start each recursive call with a check that all answers along the current branch of the recursion tree are correct; and if they are not, we backtrack.

But our implementation of this approach is new and crucially exploits two things that are specific to our setting. First, we exploit that our protocol for Sum-Equal has no false negatives, so any Sum-Equal question that was answered negatively is correct. It remains to verify all the questions that were answered affirmatively. This means that we have some ℓ equations of the form $\sum_{i \leq k} y_i^j = u^j$, for $j = 1, 2, \ldots, \ell$ and we need to check if all of them are correct. We use the simple fact that this reduces to solving a single Sum-Equal question. For example write each number in one digit over a sufficiently large basis. Alternatively, check the sum of a random subset of the ℓ equations. We pay $O(1)$ communication per check, and walk for $O(\lg n)$ steps, resulting in an $O(\lg n)$-communication protocol.

Lower bounds. The high-level strategy of the proofs of our lower bounds is the same, and standard. We define two distributions: G (for good) and B (for bad) so that any correct, randomized protocol satisfies, say,

$$\Pr_G[\Pi(G) = 1] \geq \Pr_B[\Pi(B) = 1] + 0.4. \tag{1}$$

By an averaging argument, there is a *deterministic* protocol Π' that maintains the same probability gap. Our proofs show that the latter is impossible.

To prove this impossibility, we use the standard fact that any k-player, deterministic protocol with communication $\leq c$ partitions the inputs in $C \leq 2^c$ monochromatic rectangles $R = R_1 \times R_2 \times \cdots \times R_k$. ("Monochromatic" means that the output of the protocol is constant.) We then show that for each large rectangle R we have

$$\Pr[B \in R] \geq \Pr[G \in R] - 1\%/C. \tag{2}$$

Summing over rectangles we contradict Equation (1).

For a k-tuple T such as R, G, or B, we use the notation T_i for the i-th coordinate, and T_{-k} for all coordinates but the k-th.

To show Equation (2) we bound the statistical distance between the distributions

$$G'_k := G_k \text{ conditioned on } G_{-k} \in R_{-k},$$
$$B'_k := B_k \text{ conditioned on } B_{-k} \in R_{-k}.$$

In each case we use the fact that since R is large we are conditioning over a noticeable event. The definitions of G and B, and the associated arguments to prove the statistical closeness of G'_k and B'_k, are the main novelty in our proofs. They depend on the problem.

2-player Sum-Greater. For simplicity we prove a lower bound for deciding if $x_1 \geq x_2$ (as opposed to $x_1 > -x_2$). We define G and B as follows. G_1 and B_1 are a uniform n-bit integer. G_2 is obtained from G_1 by selecting a random index $i \in [n]$, and by setting all bits less significant than the i-th to 0. B_2 is obtained similarly from B_1, but in addition the i-th bit is flipped. In other words, if $G_1 = B_1 = X_1X_2 \ldots X_n$, where X_1 is the most significant bit, and if we choose the same i for both G_2 and B_2, we can write

$$G_2 = X_1X_2 \ldots X_{i-1}X_i0 \ldots 0,$$
$$B_2 = X_1X_2 \ldots X_{i-1}(1 - X_i)0 \ldots 0.$$

It is easy to check that $G_1 \geq G_2$ always, but $B_1 \geq B_2$ with probability $1/2$.

Using the fact that the bits less significant than the i-th are set to 0 in both G_2 and B_2, and that the bits more significant than the i-th are the same, we prove that the distance between B'_2 and G'_2 is at most the probability of predicting X_i given the prefix $X_1, X_2, \ldots, X_{i-1}$. Since we are conditioning over $G_1 \in R_1$, which w.l.o.g. has probability $\geq 1/n^\gamma$ for a small γ, we can bound this prediction probability for a typical i, using the chain rule of entropy in a fashion similar to [Vio09, SV10].

k-player Sum-Equal modulo a prime p. Here we simply define G_i and B_i to be uniform and independent in $\mathbb{Z}_p$ for $i < k$. Then we define

$$G_k := -\sum_{i<k} G_i,$$
$$B_k := -\sum_{i<k} B_i + 1.$$

We show that both G'_k and B'_k are the sum of $\Omega(k)$ independent variables. For $p = \Theta(k^{1/4})$, we use a relatively standard Fourier argument to show that G'_k and B'_k are both close to the uniform distribution, with error exponentially small in k.

k-player Sum-Equal modulo 2^n. For this the previous argument does not work, as the players could just communicate the parity of theirs bits.

We again define G_i and B_i to be uniform and independent in $\mathbb{Z}_{2^n}$. But then we define

$$G_k := -\sum_{i<k} G_i,$$
$$B_k := -\sum_{i<k} B_i + 2^{n-1}.$$

Again we show that both G'_k and B'_k are the sum of $\Omega(k)$ independent variables. We then show both are close to a distribution over $\mathbb{Z}_{2^n}$ whose most significant bits are uniform. To any such distribution, adding 2^{n-1}, which amounts to flipping the most significant bit, is immaterial. Hence G'_k and B'_k are close.

k-player Sum-Equal over the integers. For this the previous arguments do not work, as the players could just communicate their numbers modulo a small prime.

We define G_i and B_i to be uniform and independent in a small range $\{1, \ldots, a\}$ for $a = \Theta(\lg k)$. Then we define

$$t := \prod_{\text{prime } p \leq a} p^{\lfloor \lg_p a \rfloor},$$
$$G_k := -\sum_{i<k} G_i,$$
$$B_k := -\sum_{i<k} B_i - t.$$

Again we show that both G'_k and B'_k are the sum of $\Omega(k)$ independent variables. This allows us to write (after appropriate conditioning)

$$G'_k = b - mS$$

where b and m are fixed, $m \leq a$, and S is a binomial distribution.

By using the shift-invariance of binomial distributions, we note that S is close to $S + t/m$. Hence we derive:

$$G'_k = b - mS \approx b - m(S + t/m) = b - mS - t = B'_k$$

yielding the desired closeness.

The error incurred by the use of shift-invariance is (only) polynomial in k. This prevents us from summing over all rectangles of a general protocol. But we show that the rectangles corresponding to sententious protocols have additional structure which allows us to sum.

Pseudorandom functions. We now explain the ideas behind our poly(n)-time distinguisher for functions $f : \{0,1\}^n \to \{0,1\}$ computed by poly(n)-size AC^0 circuits augmented with 1 threshold gate.

The main idea in the proof is to use a quantity $R_k(f)$ which measures the correlation between the function f and k-player number-on-forehead protocols. In this section all protocols are number-on-forehead. The quantity $R_k(f)$ is implicit in the seminal work [BNS92] by Babai, Nisan, and Szegedy, and was later made explicit in a series of works [CT93, Raz00, VW08]. We use the notation and results from the latter. Specifically, we use that

(1) $R_k(f)$ is exponentially small for random functions, while

(2) if f correlates with a $O(\lg n)$-communication protocol among $k = O(1)$ players, then $R_k(f) \geq 1/\mathrm{poly}(n)$.

The distinguisher D hits the oracle with a random restriction (cf. [FSS84, Hås87]), it then computes an approximation of R_k up to an additive polynomial error, and accepts if it is large. This takes polynomial time.

In the pseudorandom case, after being hit by the restriction, the circuit collapses to a polynomial-threshold function of constant degree. By Theorem 4, a protocol with $k = O(1)$ players can compute the restricted circuit (under any partition of the input) using communication $O(\lg n)$ and error 1%. Hence by (2) $R_k \geq 1/\mathrm{poly}(n)$.

In the random case, just note that a restricted random function is still random, and use (1).

This concludes the overview of the proof in the case where the circuit is augmented with a single threshold gate. We mention that using this proof but plugging the previous $O(\lg^2 n)$ bound on the communication complexity of Sum-Greater from [Nis93] one gets a distinguisher running in time $n^{O(\lg n)}$ instead of $n^{O(1)}$.

To handle more threshold gates we still show that with high probability the restricted circuit is computable by a low-communication protocol, using an idea from [Vio07]. We note that the result [Bei94] of Beigel shows that $O(1)$ symmetric gates can be reduced to 1 with only a polynomial blow-up. However no equivalent of Beigel's result holds even for two threshold gates, as was shown by Gopalan and Servedio in [GS10].

3 Upper bounds

In this section we prove our upper bounds for Sum-Equal, Sum-Greater, and polynomial-threshold functions.

3.1 Sum-Equal

We define the hash function originating in the work [DHKP97, §2.2] by Dietzfelbinger, Hagerup, Katajainen, and Penttonen. For a bit string $x = x_1 x_2 \cdots x_n \in \{0,1\}^n$ and a positive integer s we use the following notation for the strings obtained by shifting x to the right or left by s bits:

$$x \gg s := x_1 x_2 \cdots x_{n-s},$$
$$x \ll s := x_1 x_2 \cdots x_n \underbrace{00\cdots 0}_{s}.$$

Definition 12 (§2.2 in [DHKP97])**.** The hash function $h : \{0,1\}^n \to \{0,1\}^r$ has a seed of $n-1$ bits and is defined as follows. Interpret the seed as an odd integer $a \in \{0,1\}^n$. Then

$$h_a(x) := (a \cdot x \mod 2^n) \gg n - r.$$

Here $(a \cdot x \mod 2^n)$ is multiplication modulo 2^n, (i.e., multiply as integers but only keep the n least significant bits) and $\gg n-r$ amounts to taking the r most significant bits of $(a \cdot x \mod 2^n)$.

Equivalently, $h_a(x) = ((a \cdot x) \gg n - r) \mod 2^r$, where the product $a \cdot x$ is over the integers.

So the hash function outputs a fixed subset of r bits of the product $a \cdot x$ over the integers.

Notation. The domain and range of the hash function are identified with the groups $\mathbb{Z}_{2^n}$ and $\mathbb{Z}_{2^r}$ respectively. Operations are interpreted accordingly. For example, $h_a(x+y)+1 \in \{-1,0,1\}$ means that the sum $x+y$ is over $\mathbb{Z}_{2^n}$, the sum $h_a(x+y)+1$ is over $\mathbb{Z}_{2^r}$, and the membership $\in \{-1,0,1\}$ is over $\mathbb{Z}_{2^r}$. The latter means that elements are interpreted modulo 2^r.

The next lemma is essentially from [DHKP97, §2.2]. However for our protocol we need somewhat different properties than those used in that work.

Lemma 13. The hash function in Definition 12 satisfies the following:

(1) For every seed a, and for every $x, y \in \{0,1\}^n$:

-(1.1) $h_a(x) + h_a(y) \in \{h_a(x+y), h_a(x+y) - 1\}$, and

-(1.2) $h_a(x) - h_a(y) \in \{h_a(x-y), h_a(x-y) + 1\}$,

(2) For every $y \in \{0,1\}^n, y \neq 0$, and every $t \geq 0$, $\Pr_a[h_a(y) \in \{-t, -t+1, \ldots, 0, 1, 2, \ldots, t\}] \leq O(t^2)/2^r$.

Proof: (1.1) Write $h_a(x)$ as $(ax \gg n-r) \mod 2^r$, where the product ax is over the integers. Note that for either $t=0$ or $t=1$ we have

$$(ax \gg n-r) + (ay \gg n-r) + t = (ax + ay) \gg n - r,$$

since all we can lose is one carry bit. The result follows by taking modulo 2^r.

(1.2) Follows from (1.1) by replacing x with $x - y$.

(2) Let $y = b2^c$. So $h_a(y) = (ab2^c \mod 2^n) \gg n - r$. Note $(ab2^c \mod 2^n) = (ab \mod 2^{n-c}) \ll c$. When a is a uniform odd integer $\in \{0,1\}^n$, so is ab Mod 2^n. Hence $(ab \mod 2^{n-c})$ is a uniform odd integer $\in \{0,1\}^{n-c}$. Hence $(ab \mod 2^{n-c}) \ll c$ is an n-bit string of the form $U1 \ll c$, where U is a uniform $(n-c-1)$-bit string.

The output consists of the r most significant bits of this string. If these contain $s > \lg t$ of the c low-order zeros, then the output $\in B := \{-t, -t+1, \ldots, 0, 1, 2, \ldots, t\}$ with probability 0. This is because from an r-bit string of the form $u1 \ll s$ we cannot reach 0 mod 2^r by subtracting less than $2^s > t$, nor by adding less than $2^s > t$.

Otherwise, it contains $\geq r - \lg t - 1$ uniform bits. By a union bound, the probability of falling in B, a set of size $O(t)$, is

$$\leq O(t)/2^{r - \lg t - 1} = O(t^2/2^r).$$

■

For concreteness, we define all variants of Sum-Equal we study.

Definition 14. In the k-player Sum-Equal problem over $\mathbb{Z}$, each of k players knows an integer number x_i of magnitude $|x_i| < 2^n$. The players want to determine if $\sum_{i \leq k} x_i = s$. Both n and s, and a public random string, are known to all players.

In the k-player Sum-Equal problem over $\mathbb{Z}_m$, each of k players knows an element $x_i \in \mathbb{Z}_m$. The players want to determine if $\sum_{i \leq k} x_i = s$, where $s \in \mathbb{Z}_m$ and the summation is in $\mathbb{Z}_m$. Both m and s, and a public random string, are known to all players.

Theorem 15. The k-player Sum-Equal problem has a randomized communication protocol with error ϵ and communication $O(k \lg(k/\epsilon))$, both over $\mathbb{Z}$ for any n and over $\mathbb{Z}_m$ for any m. Moreover, the protocol has one-sided error with no false negatives.

Proof: We first show the protocol over $\mathbb{Z}_m$ for m a power of 2, then over $\mathbb{Z}$, and finally over $\mathbb{Z}_m$ for any m.

Over $\mathbb{Z}_m$ for $m = 2^n$. The players use the public randomness to agree on a hash function h_a with a suitable range of $r = \lg(k^{O(1)}/\epsilon)$ bits. Player i on input x_i then broadcasts the value $y_i = h_a(x_i)$. A referee may then accept if

$$\sum_{i \leq k} y_i - h_a(s) \in \{0, -1, \ldots, -k+1\}.$$

By definition this protocol has communication $kr = O(k \lg(k/\epsilon))$. We now claim this protocol has error ϵ, and no false negative.

Indeed, if $\sum x_i = s$, then, by applying Lemma 13.(1.1) $k-1$ times we have that $\sum y_i = h(\sum x_i) - t = h(s) - t$ for some $t \in \{0, -1, \ldots, -k+1\}$, and thus the referee always accepts.

Otherwise, if $\sum x_i = z \neq s$ then:

$$\begin{aligned}
&\Pr[\sum_{i \leq k} y_i - h_a(s) \in \{0, -1, \ldots, -k+1\}] \\
&\leq \Pr[h_a(z) - h_a(s) \in \{k-1, k-2, \ldots, -k+1\}] \qquad \text{(Lemma 13.(1.1))} \\
&\leq \Pr[h_a(z-s) \in \{k-1, k-2, \ldots, -k\}] \qquad \text{(Lemma 13.(1.2))} \\
&\leq O(k^2)/2^r \qquad \text{(Lemma 13.(2), since } z - s \neq 0\text{)} \\
&\leq \epsilon.
\end{aligned}$$

Over $\mathbb{Z}$. It is easy to reduce to the case in which both the x_i and s are not negative, n'-bit numbers, by having each player privately add to its input an appropriate quantity that depends only on n and s.

On input $x_1, \ldots, x_k \in \{0,1\}^{n'}$, and for common $s \in \{0,1\}^{n'}$, the players run the protocol for sum in $\mathbb{Z}_m$ for $m = 2^{n''}$ where $n'' \geq n' \lg k$. Since the sum is $\leq (2^{n'} - 1)k < 2^{n''}$, this works.

Over $\mathbb{Z}_m$ for any m. Each player i communicates an integer a_i that is within $< m/2k$ of x_i. Discretizing, this costs $O(\lg k)$ bits per player. Let $A := \sum_i a_i$, and let M be the multiple of m that is closest to $A - s$. The players run the protocol for $\sum x_i =^? s + M$ over $\mathbb{Z}$. Note this protocol still has communication $O(k \lg k)$.

To see that the protocol is correct, first note that if $\sum x_i = s + M$ over $\mathbb{Z}$ then $\sum x_i = s$ over Z_m.

Conversely, start by noting that $|\sum(x_i) - A| \leq \sum |x_i - a_i| < km/2k = m/2$, and that $|A - s - M| \leq m/2$. Now, if $\sum x_i = s$ over $\mathbb{Z}_m$ then $\sum x_i = s + M + tm$ over $\mathbb{Z}$ for some integer t. We have

$$\begin{aligned} |tm| &= |\sum x_i - s - M| \\ &= |\sum x_i - A + A - s - M| \\ &< m/2 + m/2 \end{aligned}$$

and so $t = 0$. Hence, $\sum x_i = s + M$ over $\mathbb{Z}$. ■

Note that in all variants the protocol remains simultaneous, and the referee is the only one who needs to know s.

3.2 Sum-Greater

In this section we prove our upper bound for Sum-Greater:

Theorem 4. The k-player Sum-Greater problem has a randomized communication protocol with error ϵ and communication $O(\lg n/\epsilon)k \lg k$.

Proof of Theorem 4: It is easy to see that we can reduce to the case of non-negative integers with $O(n)$ bits. So we assume that all the x_i are in $\{0,1\}^n$ and $s \geq 0$.

To describe one reduction the following definition and result help.

Definition 16. The problem *carry matters* (*CM*) on input $y_1, \ldots, y_k \in \{0,1\}^n$ and t is the problem of computing $h \in \{0, 1, \ldots, k-2\}$ such that

$$\sum_{i \leq k} y_i = t - h, \tag{3}$$

or if no such h exists computing $h = \perp$. This output is denoted $CM(y_1, \ldots, y_k, t)$. Here Player i knows y_i. Both t and n are known to all players.

The next claim saves a factor k over repeating our protocol for SUM-EQUAL many times.

Claim 17. *$CM(y_1,\ldots,y_k,t)$ can be solved by a k-player protocol with communication $O(k\lg k)$ with error 1%. Moreover, when the protocol outputs $\perp$, it is always correct.*

Proof: The players essentially run the protocol for SUM-EQUAL with error $1\%/k$, which takes communication $O(k\lg k)$. More specifically, the players agree on a hash function family $h:\{0,1\}^{n'}\to\{0,1\}^r$, where n' is large enough so that $|\sum y_i-(t-h)|<2^{n'}$ for any $h\in\{0,1,\ldots,k-2\}$, and the range is $r=O(\lg k)$ to guarantee error probability $1\%/k$ in Lemma 13.(2) (where the "t" in that lemma is the current k). For a public random a, Player i sends $h_a(y_i)$. The protocol then outputs the first $h\in\{0,1,\ldots,k-2\}$ such that

$$\sum_{i\le k}h_a(y_i)-h_a(t-h)\in\{0,-1,\ldots,-k+1\},$$

and $\perp$ if there is no such h.

Let z be the (random) output of the protocol on the instance $(y_1,\ldots,y_k,t)$ Let also $CM(y_1,\ldots,y_k,t)=h$.

Note $h\neq\perp\Rightarrow z\neq\perp$, by Lemma 13.(1.1). So when the protocol outputs $\perp$ it is correct.

When $h\neq\perp$, there still is some probability that $z=h'\neq h$. But this probability is $\le k1\%/k\le 1\%$ by Lemma 13.(1-2) and a union bound, similarly to the analysis of the SUM-EQUAL protocol.

Finally, a similar analysis shows that if $h=\perp$ then the probability that $z\neq\perp$ is $\le 1\%$. ■

We now describe one reduction in the protocol. Suppose the current SUM-GREATER instance is $\sum_{i\le k}x_i>^? s$ where the x_i have n bits. We denote by x_i^L the least significant $n/2$ bits of x_i, and by x_i^M the most significant $n/2$ bits of x_i. We also denote by s^L the least significant $n/2$ bits of s, and by s^M all the other bits. We note that s^M may consist of more than $n/2$ bits. We also note that the range of the $\cdot^M$ and $\cdot^L$ operations depends on the bit-length of the x_i corresponding to the instance we are dealing with. However we refrain from indicating the range in the notation to avoid clutter.

First the players solve the problem $CM(x_1^M,\ldots,x_k^M,s^M)=h$. We refer to this problem as *the CM problem associated with the* SUM-GREATER *instance* $\sum_i x_i>^? s$. If $h\in\{0,1,\ldots,k-2\}$ then the SUM-GREATER instance is reduced to $\sum x_i^L>^? s^L+h2^{n/2}$. The correctness is obvious. Otherwise if $h=\perp$, the instance is reduced to $\sum x_i^M>^? s^M$. Here correctness follows from the fact that the carry from $\sum x_i^L$ into the most significant $n/2$ bits is $\le k-1$. This completes the description of one reduction.

To describe the whole protocol, consider the tree T_N corresponding to repeating the above reduction. Each internal node is labeled with a SUM-GREATER instance and its associated CM instance. Each leaf is labeled with a SUM-GREATER instance only. This tree has depth $\lg n$ and each internal node has k edges, each labeled with one of the $(k-1)+1$ possible answers to the CM problem at the node.

Our efficient protocol works on another tree which we denote T. T is obtained from T_N be replacing each leaf with a sufficiently long chain, say of length n. The high-level structure

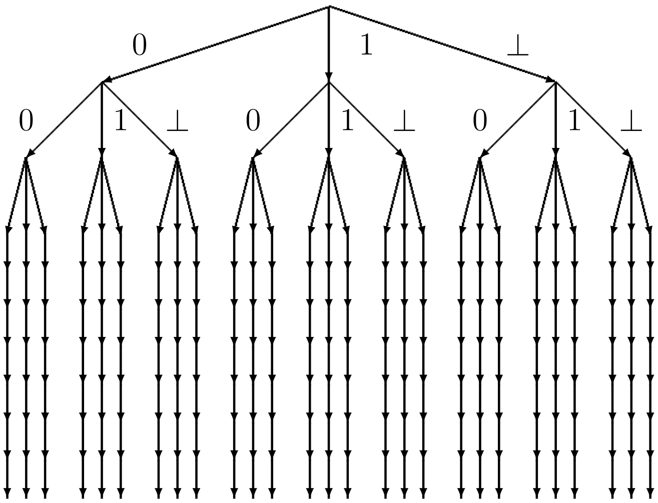

Figure 1: High-level view of tree T for $k = 3$.

of T is depicted in Figure 1. A more detailed and partial view of T is in Figure 2. Each node on the chain is labeled only with the same Sum-Greater instance as the leaf in T_N it corresponds to.

The *good path* is defined as the root-leaf path where we follow edges corresponding to the values of CM. We now claim that during an error-prone execution, the players can verify if the node they reached is along the good path, with constant communication and constant error.

Claim 18. Let v be a node in T. Consider the path taking the root to v. Suppose that if it follows an edge labeled $\perp$, then the corresponding CM problem has indeed value $\perp$. Then the players can decide if v is along the good path with communication $O(k \lg k)$ and error 1%. Here Player i knows x_i, n, s, and the path.

Proof: Since the edges labeled $\perp$ are taken correctly, we just need to verify all other edges. Let there be ℓ of them. The j-th such edge is labeled with a value $h^j \in \{0, \ldots, k-2\}$ that corresponds to an equation $\sum y_i^j = u^j := t^j - h^j$. We need to verify that

$$\bigwedge_{j \le \ell} \left(\sum_{i \le k} y_i^j = u^j \right).$$

This is reduced to verifying the single equation

$$\sum_{j \le \ell} C^j \cdot \left(\sum_{i \le k} y_i^j \right) = \sum_j C^j \cdot u^j,$$

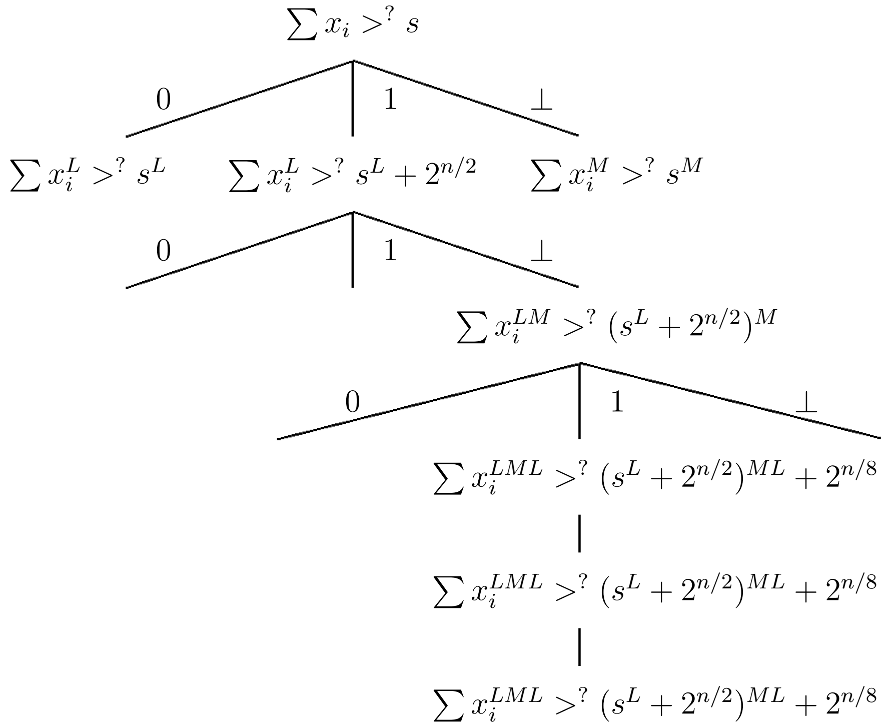
$\sum x_i >^? s$
0
1
$\perp$
$\sum x_i^L >^? s^L$
$\sum x_i^L >^? s^L + 2^{n/2}$
$\sum x_i^M >^? s^M$
0
1
$\perp$
$\sum x_i^{LM} >^? (s^L + 2^{n/2})^M$
0
1
$\perp$
$\sum x_i^{LML} >^? (s^L + 2^{n/2})^{ML} + 2^{n/8}$
$\sum x_i^{LML} >^? (s^L + 2^{n/2})^{ML} + 2^{n/8}$
$\sum x_i^{LML} >^? (s^L + 2^{n/2})^{ML} + 2^{n/8}$

Figure 2: Low-level view of tree T for $k = 3$. We show the SUM-GREATER instances at distance 1 from the root, and also along a path.

for a large enough value C that "pads" the numbers to avoid the carry from one sum to interfere with another sum. The players can determine C without communication. Note this is the same as the SUM-EQUAL instance

$$\sum_{i \le k} \left(\sum_{j \le \ell} C^j \cdot y_i^j \right) =^? \sum_j C^j \cdot w^j,$$

which can be verified with the protocol for SUM-EQUAL. Note Player i can compute privately $\sum_{j \le \ell} C^j \cdot y_i^j$ from x_i and the path.

Alternatively, the players use the public randomness to select a random subset W of the ℓ equations, and check their sum. This gives error probability $\le 51\%$ which can be amplified by repetition. ■

We can now complete the description of the protocol. The players perform a random walk on T of length $w := b \lg(n/\epsilon)$, for an appropriate constant b to be determined below, as follows. Upon entering a node, they first verify that the node lies along the good path via Claim 18. If the answer is negative, they backtrack, that is, they move to the parent of the node. If the answer is affirmative they proceed as follows. If they are not on a chain, they solve the corresponding CM problem via Claim 17, and move to the corresponding children. If on a chain, they move on to the child. At the end of the walk: if the players are on a chain, they output the solution of the corresponding $O(1)$-size instance; if not, players output something arbitrary.

Bounding the communication. By Claim 18 and 17, the communication is $O(wk \lg k) + O(1) = O(\lg(n/\epsilon)k \lg k)$.

Correctness. To establish correctness we only need to bound the probability of ending up in the correct chain. Consider the leaf along the good path. Mark each node with the distance from that leaf in T. Note that it is enough that at the end of the walk we have decreased our starting distance by $\geq \lg n$, since in that case we must be on the correct chain. Note that at each step we have $\geq 98\%$ probability of decreasing the distance. Since we take $w = b\lg(n/\epsilon)$ steps, by a Chernoff bound, for a suitably large b, with probability $\geq 1-\epsilon$ we have increased the distance at most $3\%w$ steps. In this case we have decreased the distance by at least $w - 2 \cdot 3\%w \geq \lg n$, for a large enough b. ■

To conclude, we verify that Nisan and Safra's protocol does not determine the most significant bit where $\sum_{i\leq k} x_i$ and s differ. By setting some x_i to 0 one can reach the same conclusion for $\sum_{i\in I} x_i$ and $s - \sum_{i\in \bar{I}} x_i$ for any I. Consider the execution of the protocol on a k-player Sum-Greater instance where $s = 0$ and for $x_i \in \{0,1\}^n$, $\sum_i x_i^M = 2^{n/2} - 1$. The protocol will focus on x_i^M and s^M, ignoring the least significant bits. This is indeed sufficient to solve the instance, but gives no information on the most significant bit where $\sum_i x_i$ and s differ, which depends on the carry of $\sum_i x_i^L$ into the most significant bits.

3.3 Polynomial-threshold functions

In this section we collect the pieces to prove the following upper bound for polynomial-threshold functions:

Theorem 5. Let $f : \{0,1\}^n \to \{0,1\}$ be a d-PTF. Then, under any partition, and for any $k \geq d+1$, f can be computed with error ϵ by a k-player number-on-forehead protocol with communication $O(d \lg d)(d \lg n + \lg 1/\epsilon)$.

Proof: It is known that every polynomial-threshold function of degree d in n variables can be written using integer weights of magnitude $n^{O(dn^d)}$,[MTT61, Mur71] and this is tight up to the $O(.)$ for any constant d.[Hås94, Pod09]

Without loss of generality we prove the result for $k := d+1$. As cleverly exploited by Håstad and Goldmann in [HG91], for each monomial there is a player that sees all its variables. Thus the players on input x can privately compute k numbers $z_1, \ldots, z_k$ such that $f(x) = 1 \Leftrightarrow \sum_i z_i > s$, for some integer s known to all players. The magnitude of each integer z_i is $\leq O(n)^d n^{O(dn^d)} = n^{O(dn^d)}$, so each number can be expressed with $m = O(dn^d \lg n) \leq n^{O(d)}$ bits.

By Theorem 4 the players can then decide if $\sum_i z_i > s$ with error ϵ using communication $O(k \lg k) \lg(m/\epsilon) = O(d \lg d)(d \lg n + \lg 1/\epsilon)$. ■

3.4 A derandomization of Nisan and Safra's protocol

In this section we derandomize the $O(\lg n)$-communication protocol for 2-player Sum-Greater by Nisan and Safra in [Nis93] and obtain the following result.

Theorem 19. The 2-player SUM-GREATER problem has an explicit, randomized, private-coin communication protocol with error 1% and communication $O(\lg n)$.

Before giving the simple arguments that prove this theorem building on [Nis93], we make some remarks. Since the core of Nisan and Safra's protocol [Nis93] is a binary search for the most significant digit that is different, a minor variant of the argument below gives an explicit, randomized, private-coing algorithm with communication $O(\lg n)$ to solve the following seemingly basic problem: Each of two players is given a subset of the elements $\{1, 2, \dots, n\}$; compute an element in the symmetric difference (i.e., an element that one player has but the other has not). The work [DPR$^+$13] extends and applies these results to the study of information spreading in dynamic networks.

An interesting open problem is to make explicit our SUM-GREATER protocol for $k > 2$, for which it would be sufficient to derandomize the hash function in Definition 12.

Proof: The proof is a relatively standard composition of Naor and Naor's small-bias generator [NN93, AGHP92] and Nisan and Zuckerman's generator [NZ96] for one-way, small-space algorithms. Specifically, first note that for 2-player SUM-EQUAL one can use small-bias generators. By communicating the corresponding seeds at each invocation, two players can explicitly decide SUM-GREATER with communication $O(\lg^2 n)$ following the protocol [Nis93]. In turn, for fixed inputs x_1, x_2 to SUM-GREATER, this protocol can be modeled as an algorithm using $O(\lg n)$ bits of space and reading $O(\lg^2 n)$ random bits in a one-way fashion. The bits of space are used to store the current node in the tree with $\leq n^2$ nodes corresponding to the protocol; see §3.2. (The algorithm is so that transitions can be determined with low communication, but that is irrelevant for this derandomization.) The result in [NZ96] can be used to reduce the randomness complexity to $O(\lg n)$. The protocol in the theorem starts with one player selecting at random these $O(\lg n)$ bits and communicating them to the other. The two players then expand the seed into the needed $O(\lg^2 n)$ bits of randomness and proceed as before. ■

3.5 Multiplication

In this section we show that the communication complexity of SUM-EQUAL and SUM-GREATER does not increase if we replace addition by multiplication in the definition of the problems.

Definition 20. In the k-player MUL-EQUAL problem, each of k players holds in its hand an integer x_i such that $2 \leq x_i \leq 2^n$. The players want to determine if $\prod_{i \leq k} x_i = s$, where s is an integer satisfying $2 \leq s \leq 2^n$. Both n and s, and a public random string, are known to all players.

The MUL-GREATER problem is like MUL-EQUAL except that the condition $\prod_{i \leq k} x_i = s$ is replaced with $\prod_{i \leq k} x_i \geq s$.

Theorem 21. For every integer k,

the k-player MUL-EQUAL problem has a randomized communication protocol with error ϵ and communication $O(\lg(1/\epsilon))$;

the k-player Mul-Greater problem has a randomized communication protocol with error ϵ and communication $O(\lg(n/\epsilon))$.

Proof: Mul-Equal: This proof uses the uniqueness of prime factorization of integers. For a prime p and an integer $x \geq 2$ define $e(p,x)$ to be the exponent of p in the prime factorization of x. First, if any player has $e(p,x_i) > e(p,s)$ for any prime p, the protocol rejects; this costs $O(k) = O(1)$ communication. Then, similarly to the alternative proof of Claim 18, the players select using public randomness a uniform subset S of the primes dividing s. They then decide if $\sum_{i \leq k}(\sum_{p \in S} e(p,x_i)) = \sum_{p \in S} e(p,s)$ with error 1/4. This protocol has error $\leq 1/2 + 1/4 < 1$, and the result follows by repeating it $O(\lg 1/\epsilon)$ times.

Mul-Greater: By a result of Baker and Wüstholz on logarithmic forms, Theorem 2.15 in [BW07], we have that the quantity $|\sum_{i \leq k} \lg x_i - \lg s|$ is either 0 or at least $\beta := 1/2^{n^{O(k)}}$ in absolute value. Player i computes privately approximations x'_i and s' to $\lg x_i$ and $\lg s$ respectively, to within precision $< \beta/2(k+1)$ and thus with bit-length $n^{O(k)}$. The players then run the protocol for Sum-Greater on inputs x'_i and target $s' - \beta/2$.

To argue correctness note that if $\sum x'_i \geq s' - \beta/2$ then $\sum \lg x_i > \lg s - \beta/2 - (k+1)\beta/2(k+1) = \lg s - \beta$, and hence by the aforementioned theorem in fact $\sum \lg x_i \geq \lg s$.

Conversely, if $\sum x'_i < s' - \beta/2$ then $\sum \lg x_i < \lg s - \beta/2 + (k+1)\beta/2(k+1) = \lg s$. ■

3.6 Number-on-forehead sums under any partition

The variant of Sum-Equal where the numbers are on the players' foreheads, rather then in their hands, is studied in the seminal paper [CFL83] by Chandra, Furst, and Lipton. [CFL83] gives a characterization of the deterministic communication complexity in terms of certain Ramsey numbers, in particular obtaining an $\omega(1)$ lower bound for any fixed number k of players.

For randomized, public-coin protocols, the problem reduces to 2-player equality, and the communication drops to $O(1)$.

We note that this efficient protocol exploits the specific partition. We ask what happens under different partitions.

We show that for any $k \leq 10^4$, k players can decide if $\sum_{i \leq k} x_i = -1$ modulo 2^n with $O(1)$ communication, under every partition. And so in particular $k+1$ players can decide if $\sum_{i \leq k} x_i = s$ with $O(1)$ communication. A partition for which this result may be surprising is the one in which Player j misses the ℓ-th bit of every number for every $\ell \equiv j$ modulo k. The result may hold for every k. The current proof however does show that for every d there is a k such that k players can decide $\sum_{i \leq d} x_i = s$ modulo 2^n under any partition.

Theorem 22. For every n and $k \leq 10^4$, and for every partition of $[n \cdot k]$ in k inputs, there is a number-on-forehead protocol that given $x_1, \ldots, x_k \in \{0,1\}^n$ decides if $\sum_i x_i = -1$ modulo 2^n with communication $O(k \lg 1/\epsilon)$ and error ϵ.

The proof uses the so-called two-out-of-three construction to reduce the sum of k elements to the sum of 2 elements whose bits are GF(2) polynomials of degree $< k$ in the inputs. We then compare the inner products of these polynomials with a public random string, which are again GF(2) polynomials. Håstad and Goldmann show in [HG91] that these can be computed with the desired resources.

Claim 23 (Two-out-of-three)**.** For every n there are maps $f, g : (\{0,1\}^n)^3 \to \{0,1\}^n$ such that:

(1) Each output bit of f is a linear polynomial over GF(2);

(2) Each output bit of g is a quadratic polynomial over GF(2);

(3) For every $x_1, x_2, x_3 \in \{0,1\}^n$, $f(x_1,x_2,x_3) + 2g(x_1,x_2,x_3) = x_1 + x_2 + x_3$.

Proof: The i-th bit of f is the xor of the i-th bits of x_1, x_2, and x_3. The i-th bit of g is the majority of the i-th bits of x_1, x_2, x_3, corresponding to the carries of the sum $x_1 + x_2 + x_3$. Majority over 3 bits has degree 2. ■

Proof of Theorem 22: Iterating the construction in Claim 23, we end up with two multi-bit polynomial maps $p(x_1, \ldots, x_k)$ and $q(x_1, \ldots, x_k)$ with the property that, letting $a := p(x)$ and $b := q(x)$ be the integers whose binary representation is given by $p(x)$ and $q(x)$, $a + b = \sum_i x_i$.

By always applying the claim to the three numbers with least degree, we verified with computer search that each output bit of p and q has degree $< k$ for every $k \leq 10^4$.

We want to verify that $a + b = -1$ modulo 2^n. Hence we can consider the n least significant bits of a and b. We also note that $-1 - b$ just amounts to complementing the bits of b. So we see that we want to verify if, restricted to the n least significant bits, $p(x) = \bar{q}(x)$, where the i-bit of $\bar{q}(x)$ is the i-th bit of $q(x)$ plus one modulo 2. For this task the players compare $\langle p(x), r\rangle$ and $\langle \bar{q}(x), r\rangle$, where $\langle ., . \rangle$ denotes inner product, and r is a public random string. For fixed r, this amounts to computing publicly available GF(2) polynomials $\langle p(x), r\rangle, \langle \bar{q}(x), r\rangle : (\{0,1\}^n)^k \to \{0,1\}$ on an input x.

These polynomials are sums of polynomials of degree $< k$ and hence also have degree $< k$. Each can be computed by k players under any partition of the input by sending 1 bit per player.[HG91] ■

4 Lower bounds

In this section we prove our lower bounds for the Sum-Equal and Sum-Greater problems. Our lower bounds hold even for the special case $s = 0$. Common to many of the proofs is the following standard fact about protocols inducing decompositions by monochromatic rectangles, cf. [KN97, Lemma 1.16].

Claim 24. A k-player (number-in-hand) deterministic protocol using communication $\leq c$ partitions the inputs in $C \leq 2^c$ sets of inputs $R^1, R^2, \ldots, R^C$ such that

(1) the protocol outputs the same value on inputs in the same set, and

(2) the sets are rectangles: each R^i can be written as $R^i = R^i_1 \times R^i_2 \times \cdots \times R^i_k$ where R^i_j is a subset of the inputs of Player j.

When dealing with rectangles, we use the notation

$$R^i_{-k} := R^i_1 \times R^i_2 \times \cdots \times R^i_{k-1}$$

for the first $k-1$ coordinates.

We also recall that we can amplify the success probability of a protocol to any constant by increasing the communication by a corresponding constant factor. Hence we can and will prove our lower bounds under the assumption that the error probability is, say, 1%.

4.1 Sum-Greater

Theorem 6. The communication complexity of 2-player Sum-Greater is $\Omega(\lg n)$.

4.1.1 Proof

Let Π be a randomized protocol with error $\leq 1\%$ that on input $(x_1, x_2) \in \{0,1\}^n \times \{0,1\}^n$ decides if $x_1 \geq x_2$. (The Sum-Greater problem is about deciding $x_1 > -x_2$. The current variant is sufficient and slightly more convenient for the lower bound.)

We show that if the protocol uses $c < \gamma \lg n$ bits of communication, for a sufficiently small constant γ, we reach a contradiction.

We define two distributions $G = (G_1, G_2)$ and $B = (B_1, B_2)$ as follows. To obtain a sample $G = (G_1, G_2)$, let G_1 be a uniform n-bit number. Then select $I \in [n]$ uniformly, and let G_2 be equal to G_1 but with all bits less significant than the I-th set to 0.

To obtain a sample $B = (B_1, B_2)$, again let B_1 be a uniform n-bit number. Then again select $I \in [n]$ uniformly. Now let B_2 be equal to B_1 but with the I-th bit complemented, and all bits less significant than the I-th set to 0.

Note G_1 and B_1 are the same distribution.

Pictorially, if $G_1 = B_1 = X_1 X_2 \ldots X_n$, where X_1 is the most significant bit, we can write

$$\begin{aligned} G_2 &= X_1 X_2 \ldots X_{I-1} X_I 0 \ldots 0, \\ B_2 &= X_1 X_2 \ldots X_{I-1} (1 - X_I) 0 \ldots 0. \end{aligned}$$

Now observe that $G_1 \geq G_2$ always, while $B_1 \geq B_2$ with probability 1/2. The latter happens exactly when the I-th bit of B_1 is 1. By an averaging argument, there exists a deterministic protocol Π' that gives the correct answer on the distribution $G/2 + B/2$ except with error probability $\leq 1\%$. In particular, the error is $\leq 2\%$ on each of G and B. We obtain

$$\Pr_G[\Pi'(G) = 1] \geq 1 - 2\% \geq 98\%, \tag{4}$$

$$\Pr_B[\Pi'(B) = 1] \leq 1/2 + 2\% = 52\%. \tag{5}$$

We will reach a contradiction by showing that, for small enough γ,

$$\Pr_B[\Pi'(B) = 1] \geq \Pr_G[\Pi'(G) = 1] - 2\%. \tag{6}$$

Indeed, together with equations (4) and (5), Equation (6) gives the contradiction

$$52\% \geq \Pr_B[\Pi'(B) = 1] \geq \Pr_G[\Pi'(G) = 1] - 2\% \geq 98\% - 2\%.$$

To show Equation (6) we prove the following lemma.

Lemma 25. Let $R = R_1 \times R_2 \subseteq \{0,1\}^n \times \{0,1\}^n$, be a rectangle such that $\Pr[G \in R] \geq 1/n$. Then $\Pr[B \in R] \geq \Pr[G \in R] - 1/n^{1/3}$.

In fact the lemma holds under the weaker assumption that $\Pr[G \in R] \geq 1/2^{n^\alpha}$ for a suitable constant α.

We now claim the above Lemma 25 implies Equation (6). Indeed, for $i = 1, 2, \ldots$ let $R^i = R^i_1 \times R^i_2$, where $R^i_j \subseteq \{0,1\}^n$, be the $C \leq 2^c$ rectangles obtained applying Claim 24 to the deterministic protocol Π'. Further let $\mathcal{R}$ be the subset of these rectangles on which Π' outputs 1. Recalling $C \leq 2^{\gamma \lg n}$, note:

$$\begin{aligned}
\Pr[\Pi'(B) = 1] &\geq \sum_{R \in \mathcal{R} : \Pr[G \in R] \geq 1\%/C} \Pr[B \in R] \qquad \text{(by disjointness of the rectangles)} \\
&\geq \sum_{R \in \mathcal{R} : \Pr[G \in R] \geq 1\%/C} (\Pr[G \in R] - 1/n^{1/3}) \qquad \text{(by Lemma 25, for any } \gamma < 1) \\
&\geq \Pr[\Pi'(G) = 1] - 1\% - C/n^{1/3} \\
&\geq \Pr[\Pi'(G) = 1] - 2\%. \qquad \text{(For small enough } \gamma)
\end{aligned}$$

So it only remains to prove Lemma 25 to conclude the proof of the theorem.

We denote by $\Delta(X, Y)$ the statistical distance between two distributions. For a distribution X and an event E, we denote by $X|E$ the distribution of X conditioned on E. The shannon entropy of X is denoted $H(X)$. We recall the following facts:

Fact 26 (Chapter 3, Exercise 17 in [CK82])**.** Let V be a random variable taking values in a set S. Let U be a uniform variable over S. Then: $\Delta(V, U) \leq 4\sqrt{\lg |S| - H(V)}$.

Fact 27. Let V be a random variable taking values in a set S. Let U be a uniform variable over S. Let $\pi : S \to S$ be a permutation. Then $\Delta(V, \pi(V)) \leq 2\Delta(V, U)$.

We use the above in the simple case where $S = \{0, 1\}$ and $\pi(V) = 1 - V$.

Fact 28. Let V, W be two random variables, and let $E_1, E_2, \ldots, E_t$ be mutually exclusive events. Then, $\Delta(V, W) \leq \sum_{i \leq t} \Pr[E_i]\Delta(V|E_i, W|E_i)$.

Proof of Lemma 25:

Fix such a large rectangle R. We shall prove the inequality

$$\Pr[B_2 \in R_2 | B_1 \in R_1] \geq \Pr[G_2 \in R_2 | G_1 \in R_1] - 1/n^{1/3} \ (\star),$$

from which we easily obtain the lemma as follows:

$$\begin{aligned}
\Pr[B \in R] &= \Pr[B_2 \in R_2 | B_1 \in R_1] \Pr[B_1 \in R_1] \\
&\geq (\Pr[G_2 \in R_2 | G_1 \in R_1] - 1/n^{1/3}) \Pr[B_1 \in R_1] \qquad \text{(By } (\star)\text{)} \\
&\geq \Pr[G_2 \in R_2 | G_1 \in R_1] \Pr[G_1 \in R_1] - 1/n^{1/3} \qquad \text{(Since } G_1 \text{ and } B_1 \text{ are equal)} \\
&= \Pr[G \in R] - 1/n^{1/3}.
\end{aligned}$$

To show $(\star)$, we shall prove that

$$\Delta(G_2 | G_1 \in R_1, B_2 | B_1 \in R_1) \leq 1/n^{1/3}.$$

Let $X = X_1 X_2 \ldots X_n$ denote the distribution $G_1 | G_1 \in R_1$. Note the shannon entropy of X is

$$H(X) = \lg |R_1| \geq n - \lg n,$$

since our assumption that $\Pr[G \in R] \geq 1/n$ implies $\Pr[G_1 \in R_1] = |R_1|/2^n \geq 1/n$.

Denote by $X_{<i}$ the variables $X_1 X_2 \ldots X_{i-1}$. By the chain rule of entropy,

$$\sum_{i=1}^{n} H(X_i | X_{<i}) = H(X) \geq n - \lg n.$$

Denoting by I a uniform choice for $i \in \{1, 2, \ldots, n\}$, we have

$$E_I H(X_I | X_{<I}) \geq 1 - (\lg n)/n. \tag{7}$$

We can now bound:

$$
\begin{aligned}
&\Delta\left(G_2|G_1 \in R_1, B_2|B_1 \in R_1\right)\\
&\le \sum_i \Pr[I=i]\Delta\left(X_1X_2\ldots X_{i-1}X_i0\ldots 0, X_1X_2\ldots X_{i-1}(1-X_i)0\ldots 0\right) \qquad \text{(By Fact 28)}\\
&= \sum_i \Pr[I=i]\Delta\left(X_1X_2\ldots X_{i-1}X_i, X_1X_2\ldots X_{i-1}(1-X_i)\right)\\
&\le \sum_i \Pr[I=i]\sum_z \Pr[X_{<i}=z]\\
&\qquad \Delta\left(X_1X_2\ldots X_{i-1}X_i|X_{<i}=z, X_1X_2\ldots X_{i-1}(1-X_i)|X_{<i}=z\right) \qquad \text{(By Fact 28)}\\
&\le \sum_i \Pr[I=i]\sum_z \Pr[X_{<i}=z]\Delta\left(X_i|X_{<i}=z,(1-X_i)|X_{<i}=z\right)\\
&\le O(1)\sum_i \Pr[I=i]\sum_z \Pr[X_{<i}=z]\Delta(X_i|X_{<i}=z,U) \qquad \text{(by Fact 27, for U a uniform bit)}\\
&\le O(1)\sum_i \Pr[I=i]\sum_z \Pr[X_{<i}=z]\sqrt{1-H(X_i|X_{<i}=z)} \qquad \text{(by Fact 26)}\\
&\le O(1)\sqrt{\sum_i \Pr[I=i]\sum_z \Pr[X_{<i}=z](1-H(X_i|X_{<i}=z))} \qquad \text{(by Jensen's inequality)}\\
&= O(1)\sqrt{1-\sum_i \Pr[I=i]H(X_i|X_{<i}))}\\
&= O(1)\sqrt{(\lg n)/n} \qquad \text{(By Equation (7))}\\
&\le 1/n^{1/3} \qquad \text{(for large enough n.)}
\end{aligned}
$$

■

4.2 Sum-Equal modulo a prime

In this section we prove our lower bound for the Sum-Equal problem modulo a prime p.

Theorem 29. The communication complexity of k-player Sum-Equal modulo p a prime between $k^{1/4}$ and $2k^{1/4}$ is $\Omega(k \lg k)$.

The same lower bound holds modulo any number that has a prime factor $\ge k^{\Omega(1)}$. (Details omitted.)

4.2.1 Proof of Theorem 29

Let Π be a randomized protocol with error $\le 1/3$ for the k-player Sum-Equal modulo p problem with $s = 0$. We show that if the protocol uses $c < \gamma k \lg k$ bits of communication, for a sufficiently small constant γ, we reach a contradiction. By decreasing γ, we can and will assume that k is sufficiently large.

We define two distributions G and B:

$$G := (G_1, \ldots, G_{k-1}, -\sum_i G_i)$$
$$B := (B_1, \ldots, B_{k-1}, -\sum_i B_i + 1)$$

for uniform and independent $G_i, B_i \in \mathbb{Z}_p$. Note $\Pi(G)$ is supposed to output 1 while $\Pi(B)$ is supposed to output 0. Also note G_{-k} and B_{-k} are the same distribution.

By an averaging argument, there exists a deterministic protocol Π' that gives the correct answer on the distribution $G/2 + B/2$ except with error probability $\leq 1\%$. In particular,

$$\Pr_G[\Pi'(G) = 0] \leq 2\%, \tag{8}$$
$$\Pr_B[\Pi'(B) = 1] \leq 2\%. \tag{9}$$

We will contradict this fact by showing that, for small enough γ,

$$\Pr_B[\Pi'(B) = 1] \geq 0.5 \Pr_G[\Pi'(G) = 1] - 1\%. \tag{10}$$

Indeed this is a contradiction because equations (8) and (10) imply $\Pr_B[\Pi'(B) = 1] \geq 0.5 \cdot 98\% - 1\% = 48\% > 2\%$, contradicting Equation (9).

To show Equation (10) we prove a lemma. For $i = 1, 2, \ldots$ let $R^i = R^i_1 \times R^i_2 \times \cdots \times R^i_k$, where $R^i_j \subseteq \mathbb{Z}_p$, be the $C \leq 2^c$ rectangles obtained applying Claim 24 to the deterministic protocol Π'. Further let $\mathcal{R}$ be the subset of these rectangles on which Π' outputs 1.

Lemma 30. Let $R = R_1 \times R_2 \times \cdots \times R_k$, where $R_j \subseteq \mathbb{Z}_p$, be a rectangle such that $\Pr[G \in R] \geq 1\%/C$. Then $\Pr[B \in R] \geq 0.5 \Pr[G \in R]$.

The above Lemma 30 implies Equation (10) because:

$$\begin{aligned}
\Pr[\Pi'(B) = 1] &\geq \sum_{R \in \mathcal{R} : \Pr[G \in R] \geq 1\%/C} \Pr[B \in R] \qquad \text{(by disjointness of the rectangles)} \\
&\geq 0.5 \sum_{R \in \mathcal{R} : \Pr[G \in R] \geq 1\%/C} \Pr[G \in R] \qquad \text{(by Lemma 30)} \\
&\geq 0.5(\Pr[\Pi'(G) = 1] - 1\%).
\end{aligned}$$

So it only remains to prove Lemma 30 to conclude the proof of the theorem.

Proof of Lemma 30: First note that, for every rectangle R:

$$\begin{aligned}
\Pr[G \in R] &= \Pr[G_{-k} \in R_{-k}] \Pr[G_k \in R_k | G_{-k} \in R_{-k}], \\
\Pr[B \in R] &= \Pr[B_{-k} \in R_{-k}] \Pr[B_k \in R_k | B_{-k} \in R_{-k}] \\
&= \Pr[G_{-k} \in R_{-k}] \Pr[G_k + 1 \in R_k | G_{-k} \in R_{-k}].
\end{aligned}$$

Now suppose that $\Pr[G \in R] \geq 1\%/C = 1\%/k^{\gamma k}$. We are going to show the inequality

$$\Pr[G_k + 1 \in R_k | G_{-k} \in R_{-k}] \geq 0.5 \Pr[G_k \in R_k | G_{-k} \in R_{-k}], \qquad (\star)$$

which gives the lemma. First, if $R_k = \emptyset$ then both probabilities are 0, and we are done. To handle the more interesting $R_k \neq \emptyset$ case, we are going to write the distribution G conditioned on $G_{-k} \in R_{-k}$, denoted $G|G_{-k} \in R_{-k}$, as a convex combination of distributions $G^{(v)}$: $G|G_{-k} \in R_{-k} = \sum_v \alpha_v G^{(v)}$. Then we show that for each v, $\Pr[G_k^{(v)} + 1 \in R_k] \geq 0.5 \Pr[G_k^{(v)} \in R_k]$. This implies $(\star)$.

To define the $G^{(v)}$, note that among the first $k-1$ indices there must be a set I of $\geq (k-1)/2 \geq k/3$ indices i such that $\Pr[G_i \in R_i] \geq 1\%/k^{2\gamma}$ for every $i \in I$. This is because $G_1, \ldots, G_{k-1}$ are independent random variables uniform in $\mathbb{Z}_p$, and so if there are not that many indices, we would have $\Pr[G \in R] < (1\%/k^{2\gamma})^{k/2} < 1\%/k^{\gamma k} = 1\%/C$. Now note that the condition $\Pr[G_i \in R_i] \geq 1\%/k^{2\gamma}$ implies that $|R_i| \geq 2$, by making γ sufficiently small and using the fact that $p \geq k^{1/4}$. Hence, for $i \in I$, G_i conditioned on being in R_i is a random variable that is uniform on ≥ 2 elements. Any such variable is a convex combination of random variables uniform on 2 elements. (Proof: pick a random pair, output a random element.) Hence we can write $G|G_{-k} \in R_{-k}$ as a convex combination of distributions $G^{(v)}$ such that for each v: the first $k-1$ coordinates of $G^{(v)}$ are independent, and each is either uniform over two values if it belongs to I, or else it is fixed.

For any v we can write:

$$G_k^{(v)} = a + \sum_{i \leq k/3} X_i$$
$$G_k^{(v)} + 1 = a + 1 + \sum_{i \leq k/3} X_i,$$

where the variables X_i are independent variables, and X_i is uniform over $\{a_i, b_i\}$ for some $a_i \neq b_i$. Note the "$-$" sign in the definition of G_k and B_k is incorporated in the a_i and b_i.

We then use the following relatively standard claim, whose proof uses Fourier analysis and is below.

Claim 31. Let p be a prime number. Let X be the sum of t independent random variables each uniform over $\{a_i, b_i\} \subseteq \mathbb{Z}_p$ for $a_i \neq b_i$. Then X modulo p is $\epsilon \leq 0.5\sqrt{p}e^{O(t/p^2)}$ close to uniform.

We apply the claim to our case where $X = \sum_{i \leq k/3} X_i$, $t \geq k/3$, and $p \leq 2k^{1/4}$. This gives statistical distance $\epsilon \leq \exp(-k^{\Omega(1)})$. This means that replacing $G_k^{(v)}$ or $G_k^{(v)} + 1$ with a uniform random variable in $\mathbb{Z}_p$ we only lose an additive ϵ in probabilities. Hence, recalling $|R_k| > 0$, and letting $\mu := |R_k|/p \geq 1/p$, we get

$$\Pr[G_k^{(v)} + 1 \in R_k] \geq 0.5 \Pr[G_k^{(v)} \in R_k] \Leftarrow \mu - \epsilon \geq 0.5(\mu + \epsilon) \Leftrightarrow \mu \geq 3\epsilon, \qquad (11)$$

which is true for sufficiently large k.

This proves the lemma, and hence the theorem, assuming the claim. ■

Proof of Claim 31: First we use a relatively standard claim that the statistical distance between X and uniform is

$$\leq 0.5\sqrt{p} \max_{s \neq 0} |E_X[e(-sX)]|,$$

where $e(z) := e^{2\pi\sqrt{-1}z/p}$ outputs the pth primitive root of unity raised to the input; see e.g. [BV10, §B].

In our case, for every s, we can write $|E_X[e(-sX)]| = \prod_{i \leq t} |E_{X_i}[e(-sX_i)]|$.

Each term $|E_{X_i}[e(-sX_i)]|$ equals $|0.5(e(a) + e(b))|$ for some $a, b \in \mathbb{Z}_p$ such that $a \neq b$, using the fact that multiplying by s is a permutation modulo p. Letting $\alpha := 2\pi a/p$ and $\beta := 2\pi b/p$ we get:

$$\begin{aligned}
|0.5(e(a) + e(b))| &= 0.5\sqrt{(\cos(\alpha) + \cos(\beta))^2 + (\sin(\alpha) + \sin(\beta))^2} \\
&= \sqrt{0.5(1 + \cos(\alpha - \beta))} \qquad \text{(By trigonometric equalities)} \\
&\leq \sqrt{0.5(2 - (2\pi/p)^2/3)} \qquad \text{(By small-angle approximations)} \\
&\leq 1 - O(1/p^2).
\end{aligned}$$

So multiplying over the t indices we get

$$|E_X[e(-sX)]| \leq e^{O(t/p^2)}.$$

And overall the statistical distance is

$$\leq 0.5\sqrt{p}e^{O(t/p^2)}.$$

■

Conceivably one can also prove $\Pr[B \in R] \geq \Pr[G \in R] - 1\%/C$. Our proof does not give this. For example when $\Pr[G_{-k} \in R_{-k}] = \Omega(1)$, because of the loss in Claim 31, our additive error is exponential in $k^{1-\Omega(1)}$, rather than in $c = \gamma k \lg k$.

4.3 Sum-Equal modulo 2^n

In this section we prove our lower bound for the Sum-Equal problem modulo a power of two.

Theorem 32. The communication complexity of k-player Sum-Equal modulo $2^n = k^{1/4}$ is $\Omega(k \lg k)$.

The same lower bound holds modulo any power of 2 that is larger than $k^{1/4}$. (Simple details omitted.)

The proof of the above theorem is a bit more complicated than the previous proof of Theorem 29 for the case of $\mathbb{Z}_p$, p prime. In particular, the choice of distributions G and B from the previous proof does not work anymore: players can just send the least significant bit.

4.3.1 Proof of Theorem 32

Let $N := 2^n = k^{1/4}$. Let Π be a randomized protocol with error $\leq 1\%$ for the k-player SUM-EQUAL modulo N problem with $s = 0$. We define two distributions G and B:

$$G := (G_1, \ldots, G_{k-1}, -\sum_i G_i)$$
$$B := (B_1, \ldots, B_{k-1}, -\sum_i B_i + 2^{n-1})$$

for uniform and independent $G_i, B_i \in \mathbb{Z}_N$. Note $\Pi(G)$ is supposed to output 1 while $\Pi(B)$ is supposed to output 0.

The proof then proceeds like the proof of Theorem 29 until we write

$$G_k^{(v)} = a + \sum_{i \leq k/3} X_i$$
$$G_k^{(v)} + 2^{n-1} = a + 2^{n-1} + \sum_{i \leq k/3} X_i,$$

where the variables X_i are independent variables, and X_i is uniform over $\{a_i, b_i\}$ for some $a_i \neq b_i$.

We now write X_i as $a_i + (b_i - a_i)Y_i$ where Y_i is uniform over $\{0, 1\}$. Since $b_i - a_i \in \mathbb{Z}_N$, among the $\geq k/3$ indices $i \in I$ for which X_i takes at least 2 values, we must have $t \geq |I|/N = (k/3)/N = k^{3/4}/3$ indices I' such that for any $i \in I'$ the value $b_i - a_i$ is the same value m.

Further write $G^{(v)}$ as a convex combination of distributions $G^{[v]}$ where, among the first $k-1$ coordinates, only those in I' are not fixed. Note $G|G_{-k} \in R_{-k}$ is a convex combination of the $G^{[v]}$. Fix any v and denote $G^{[v]}$ by G'. Let S denote the sum of t 0/1 variables. So we can now write

$$G'_k = a + mS$$
$$B'_k = G'_k + 2^{n-1} = a + 2^{n-1} + mS.$$

We will show that

$$\Pr[B'_k \in R_k] \geq 0.5 \Pr[G'_k \in R_k]. \tag{12}$$

The argument is a bit more complicated than for the case of $\mathbb{Z}_p$, because we will not be comparing G'_k and B'_k to the uniform distribution. Consequently it is not only the size, but the structure of R_k that matters.

Write $m = d2^b$, where d is odd and $b < n$. And consider the distribution $Y := a + 2^b U$, where U is uniform over $n - b$ bits.

If $\text{Support}(Y) \cap R_k = \emptyset$, then also $\text{Support}(G'_k) \cap R_k = \emptyset$, and so $\Pr[G'_k \in R_k] = 0$ and Equation (12) holds.

Otherwise, $\text{Support}(Y) \cap R_k \neq \emptyset$. Since Y is uniform on its support, $\mu := \Pr[Y \in R_k] \geq 1/N$. We show that both distributions G'_k and B'_k are $\epsilon \leq \exp(-k^{\Omega(1)})$ close to Y. This allows us to obtain Inequality (12) as in the proof of Theorem 29, Inequality (11).

To show closeness, we use the following claim that is similar to Claim 31.

Claim 33. Let N be a power of 2. Let X be the sum of t independent random variables each uniform over $\{0,1\}$. Then X modulo N is $\epsilon \le 0.5\sqrt{N}e^{O(t/N^2)}$ close to uniform.

In our case $t/N^2 \ge k^{3/4}/3\sqrt{k} = k^{\Omega(1)}$, so the error is bound as desired.

To apply the claim to show that both B'_k and G'_k are close Y, reason as follows:

$$G'_k \equiv a + 2^b dS \equiv_\epsilon a + 2^b dU \equiv a + 2^b U \equiv Y$$
$$B'_k \equiv a + 2^{n-1} + 2^b dS \equiv_\epsilon a + 2^{n-1} + 2^b dU \equiv a + 2^b(dU + 2^{n-b-1}) \equiv a + 2^b U \equiv Y.$$

Above the symbol $\equiv$ stands for "same distribution," and ϵ is the error in statistical distance. The errors come from the claim. And we also use the fact that multiplying by the odd integer d is a permutation modulo any power of 2. This allows us to write $dU \equiv U$ and $dU + 2^{n-b-1} \equiv U$.

It only remains to verify Claim 33.

Proof sketch of Claim 33: The proof is nearly identical to that of Claim 31. We replace p by N throughout. The only conceptual difference is that we write the terms $|E_{X_i}[e(-sX_i)]|$ as $|0.5(e(0) + e(s))|$, where recall $0 \ne s < N$. This is the point where we use that X_i is boolean as opposed to uniform on two values. ■

4.4 Sum-Equal over the integers

In this section we prove our lower bounds for the Sum-Equal problem over the integers.

Definition 34. A k-player protocol is *sententious* if Player k speaks only once to announce the output.

Theorem 8. For sententious protocols, the communication complexity of k-player Sum-Equal over $\mathbb{Z}$ is $\Omega(k \lg\lg k)$.

Corollary 9. Let Π be a protocol for k-player Sum-Equal over $\mathbb{Z}$. Then some player must communicate $\ge 0.9 \lg\lg\lg k$ bits.

We only use the following property of sententious protocols.

Claim 35. Let Π be a deterministic k-player sententious protocol using c bits of communication. Claim 24 holds with the following additional guarantee:

for any $i \ne j$ such that the protocols outputs 1 on both R^i and R^j, $R^i_{-k} \bigcap R^j_{-k} = \emptyset$.

In particular, let $\mathcal{R}$ the rectangles on which Π outputs 1. Then for every distribution D on inputs:

$$\sum_{R \in \mathcal{R}} \Pr[D_{-k} \in R_{-k}] \le 1.$$

For general protocols using c bits of communication, the sum in the "in particular" part of the claim can be as large as 2^c.

Proof: Consider the protocol tree. Since the protocol is sententious, each leaf has a sibling that is also a leaf, and the parent corresponds to the single bit sent by Player k. Each rectangle R^i is the set of inputs taking to a leaf. Since the protocol outputs two different values on two sibling leaves, one of which may be empty, R^i and R^j correspond to leaves that are not siblings. R^i_{-k} and R^j_{-k} are the inputs taking to the parents x and y of these two leaves. Neither the path from x to the root, is an extension of the path from y to the root, nor vice versa. This implies $R^i_{-k} \bigcap R^j_{-k} = \emptyset$. ■

The connection with general protocols is the following.

Claim 36. Let Π be a protocol with communication c where Player k always communicates $\leq c_k$ bits. Then Π has an equivalent sententious protocol with communication $\leq 2^{c_k} c + 1$.

Proof: The first $k-1$ players run 2^{c_k} instances of Π, one for each possible output of Player k. Then Player k can privately decide which run matches its output and announce the output. ■

Proof of Corollary 9: Suppose there is a protocol in which each player communicates $<$ $0.9 \lg\lg\lg k$ bits. By Claim 36 there is a sententious protocol with communication $(\lg\lg k)^{0.9} \cdot k(0.9 \lg\lg\lg k)$, violating the $\Omega(k \lg\lg k)$ lower bound in Theorem 8. ■

4.4.1 Proof of Theorem 8

Let Π be a sententious, randomized protocol with error $\leq 1\%$ for the k-player SUM-EQUAL over $\mathbb{Z}$ problem with $s = 0$. We show that if the protocol uses $c < \gamma k \lg\lg k$ bits of communication, for a sufficiently small constant γ, we reach a contradiction. By decreasing γ, we can and will assume that k is sufficiently large.

To define the distributions G and B we need a certain integer t. Let $a := \gamma \lg k$. Define t to be the smallest number that is divisible by any integer $\leq a$.

Claim 37. $t \leq 2^{O(a)}$.

Proof: Let $t := \prod_{\text{prime } p \leq a} p^{\lfloor \lg_p a \rfloor}$. Clearly any number less than a divides t, since it is the produce of primes $p \leq a$ raised to an integer exponent $\leq \lg_p a$.

Now bound

$$t \leq \prod_{\text{prime } p \leq a} p^{\lg_p a} = a^{\pi(a)} \leq 2^{O(a)},$$

where $\pi(a)$ is the number of primes less than a, and $\pi(a) = \Theta(a/\lg a)$ by the prime number theorem. ■

We define two distributions G and B:

$$G := (G_1, \ldots, G_{k-1}, -\sum_i G_i)$$
$$B := (B_1, \ldots, B_{k-1}, -\sum_i B_i - t)$$

for uniform and independent $G_i, B_i \in \{1, \ldots, a\}$. Note $\Pi(G)$ is supposed to output 1 while $\Pi(B)$ is supposed to output 0.

By an averaging argument, there exists a deterministic protocol Π' that gives the correct answer on the distribution $G/2 + B/2$ except with error probability $\leq 1\%$. In particular,

$$\Pr_G[\Pi'(G) = 0] \leq 2\%, \tag{13}$$
$$\Pr_B[\Pi'(B) = 1] \leq 2\%. \tag{14}$$

We will contradict this fact by showing that, for small enough γ,

$$\Pr_B[\Pi'(B) = 1] \geq \Pr_G[\Pi'(G) = 1] - 2\%. \tag{15}$$

Indeed this is a contradiction because equations (13) and (15) imply $\Pr_B[\Pi'(B) = 1] \geq 98\% - 2\% = 96\% > 2\%$, contradicting Equation (14).

To show Equation (15) we prove a lemma; cf. Lemma 30. Here we use that the protocol is sententious. For $i = 1, 2, \ldots$ let $R^i = R^i_1 \times R^i_2 \times \cdots \times R^i_k$, where $R^i_j \subseteq \mathbb{Z}_p$, be the $C \leq 2^c$ rectangles obtained applying Claim 35 to the deterministic, sententious protocol Π'. Further let $\mathcal{R}$ be the subset of these rectangles on which Π' outputs 1.

Lemma 38. Let $R = R_1 \times R_2 \times \cdots \times R_k$, where $R_j \subseteq \mathbb{Z}_p$, be a rectangle such that $\Pr[G \in R] \geq 1\%/C$. Then

$$\Pr[B \in R] \geq \Pr[G \in R] - o(1) \Pr[G_{-k} \in R_{-k}].$$

Note the above Lemma 38 implies Equation (15):

$$\begin{aligned}
\Pr[\Pi'(B) = 1] &\geq \sum_{R \in \mathcal{R} : \Pr[G \in R] \geq 1\%/C} \Pr[B \in R] \qquad \text{(by disjointness of the rectangles)} \\
&\geq \sum_{R \in \mathcal{R} : \Pr[G \in R] \geq 1\%/C} \Pr[G \in R] - o(1) \Pr[G_{-k} \in R_{-k}] \qquad \text{(by Lemma 38)} \\
&\geq \Pr[\Pi'(G) = 1] - 1\% - o(1) \sum_{R \in \mathcal{R}} \Pr[G_{-k} \in R_{-k}] \\
&\geq \Pr[\Pi'(G) = 1] - 1\% - o(1) \qquad \text{(By Claim 35).}
\end{aligned}$$

So it only remains to prove Lemma 30 to conclude the proof of the theorem.

Proof of Lemma 30: We are going to prove the equivalent fact that

$$\Pr[G_k - t \in R_k | G_{-k} \in R_{-k}] \geq \Pr[G_k \in R_k | G_{-k} \in R_{-k}] - o(1). \qquad (\star)$$

As before, we are going to write the distribution $G|G_{-k} \in R_{-k}$ as a convex combination of distributions $G^{[v]}$, and argue that, for each v, $\Pr[G_k^{[v]} - t \in R_k] \geq \Pr[G_k^{[v]} \in R_k] - o(1)$. This implies $(\star)$.

To define the $G^{[v]}$, let $\epsilon := \sqrt{2/a} = \sqrt{2/\gamma \lg k}$. Let $R = R_1 \times R_2 \times \cdots \times R_k$ be a rectangle where $\Pr[G \in R] \geq 1\%/C$. This means that there must be $\geq k/3$ coordinates $i \leq k-1$ such that $\Pr[G_i \in R_i] \geq \epsilon^2$. Indeed, otherwise $\Pr[G \in R]$ is at most

$$\left(\epsilon^2\right)^{k/2} \leq (2/\gamma \lg k)^{k/2} < (1/\lg k)^{\gamma k} \leq 1\%/C,$$

where the strict inequality holds for any $\gamma < 1/2$ and sufficiently large k. Note that $\Pr[G_i \in R_i] \geq \epsilon^2$ implies that $|R_i| \geq a\epsilon^2 = 2$. So, conditioned on falling in R_i, G_i is uniform on a set of size ≥ 2.

Hence, in a fashion analogous to the proof of Theorem 32, we can write G conditioned on $G_{-k} \in R_{-k}$ as a convex combinations of distributions $G^{[v]}$ such that, for any v, letting $G' := G^{[v]}$:

$$G'_k = b - mS$$
$$G'_k - t = b - mS - t,$$

where $0 < m < a$ and S is the sum of $(k/3)/a \geq \sqrt{k}$ uniform $0-1$ i.i.d. random variables.

Since $m \leq a$, m divides t by definition of t. So let

$$t = m \cdot q.$$

Note $q \leq t \leq 2^{O(a)}$, the latter by Claim 37. We now apply q times the shift-invariance of the binomial distribution, which is the following fact. (Simple proof omitted.)

Claim 39. Let S be the sum of ℓ uniform, i.i.d. boolean random variables. Then S and $S+1$ have statistical distance $\leq O(1/\sqrt{\ell})$.

This yields that the distributions

$$b - mS = G'_k$$
$$b - m(S+q) = b - mS - t = G'_k - t$$

have statistical distance

$$\leq qO(1/\sqrt{\sqrt{k}}) \leq 2^{O(a)}/k^{1/4} = k^{O(\gamma)-1/4} = o(1)$$

for a sufficiently small $\gamma > 0$. ■

We make some remarks about the above proof of the lower bound for sententious protocols. First, we note that the lower bound is proved for numbers G_i, $i < k$, of $\leq \lg \lg k$ bits. Nisan's protocol gives a simultaneous protocol where each player sends $O(k \lg(\lg \lg k + \lg k)) = O(k \lg \lg k)$. So our proof is tight for the distributions used, and to beat our lower bound one must use numbers on $(\lg k)^{\omega(1)}$ bits.

We also note that picking t small, or uniformly at random in an interval of integers, does not work: the players can just announce the numbers modulo a comparably small prime.

5 Pseudorandom functions

In this section we prove our results for pseudorandom functions. We start with the distinguisher for AC^0 circuits augmented with few threshold (or arbitrary symmetric) gates. Then we give our candidate pseudorandom function computable by AC^0 circuits with Mod m gates.

5.1 Distinguisher for AC^0 with few threshold (or arbitrary symmetric) gates

Theorem 10. Let F be a distribution on functions from $\{0,1\}^n$ to $\{0,1\}$ such that each function in the support is computable by a circuit of size n^d, depth d, with d threshold (or arbitrary symmetric) gates, where d is a constant and n is sufficiently large.

Then there is a randomized, oracle algorithm D that runs in time n^b such that

$$\left|\Pr[D^F = 1] - \Pr[D^U = 1]\right| \geq 1 - o(1),$$

where b depends on d only, and U is a uniform function on n bits.

We define the k-player norm.

Definition 40. The k-player norm of a function $f : (\{0,1\}^\ell)^k \to \{-1,1\}$ is

$$R_k(f) := E_{\substack{x_1^0, x_2^0, \ldots, x_k^0 \in \{0,1\}^\ell \\ x_1^1, x_2^1, \ldots, x_k^1 \in \{0,1\}^\ell}} \left[\prod_{\epsilon_1, \ldots, \epsilon_k \in \{0,1\}} f\left(x_1^{\epsilon_1}, x_2^{\epsilon_2}, \ldots, x_k^{\epsilon_k}\right) \right].$$

The next lemma shows that if we can compute f well on average using little communication, then $R_k(f)$ is large.

Lemma 41 (Corollary 3.11 in [VW08])**.** Let $f : (\{0,1\}^\ell)^k \to \{-1,1\}$ be a function. Let $\Pi : (\{0,1\}^\ell)^k \to \{-1,1\}$ be a function computable by a k-player number-on-forehead protocol using c bits of communication. Then

$$E_x[f(x) \cdot \Pi(x)] \leq 2^c R_k(f)^{1/2^k},$$

where the expectation is over a uniform $x \in (\{0,1\}^\ell)^k$.

Proof of Theorem 10: We call a gate *special* if it computes a threshold or an arbitrary symmetric function.

The distinguisher. The distinguisher D on oracle $f : \{0,1\}^n \to \{-1,1\}$ depends on two constants k and h which depend on d and will be bound later. D begins with selecting a random restriction ρ which leaves free exactly $\sqrt{n}$ variables. All queries to f are masked by this restriction, so that D is querying the restricted oracle $f' := f|_\rho : \{0,1\}^{\sqrt{n}} \to \{0,1\}$.

Then D divides the $\sqrt{n}$ free variables into k blocks of size $\ell := \sqrt{n}/k$. It then computes an approximation α to $R_k(f')$. Recall that $R_k(f')$ is an expectation of a random variable in $\{-1,1\}$. D takes $s := n^{2h+1}$ samples (i.e., a value of the argument of the expectation for a random, uniform choice for the variables in the subscript) and defines α as their average. Each sample can be computed in time $\mathrm{poly}(n, 2^k) = \mathrm{poly}(n)$. So α can be computed in time $\mathrm{poly}(n)$. D accepts iff $\alpha \geq 1/n^h$.

Analysis for random functions. In the case that f comes from the uniform distribution U, note that even the restricted oracle f' is uniform. The expectation of the k-player norm over f' is zero except in the case that $x_i^0 = x_i^1$ for some $i \in \{1, 2, \ldots, k\}$. By a union bound, the expectation of the k-player norm is $\leq k2^{-\ell}$. Hence the probability over f' that the norm is $\geq 0.5/n^h$ is $o(1)$.

When the norm is $\leq 0.5/n^h$, by a Chernoff bound the probability that the approximation α is $\geq 1/n^h$ is at most

$$2^{-D(1/2+0.25/n^h+0.25/n^h||1/2+0.25/n^h)s} \leq 2^{-2(0.25/n^h)^2 s} \leq o(1).$$

Hence the distinguisher accepts with probability $o(1)$.

Analysis for circuit functions. W.l.o.g. assume the output gate is special. Consider the $\leq d2^d$ subcircuits $\mathcal{C}$ obtained by taking a special gate as a root, and fixing all other special gates to any possible value.

Each circuit in $\mathcal{C}$ has 1 special gate taking as input $\leq n^d$ AC^0 circuits. Consider the $\leq d2^d n^d$ AC^0 circuits that feed into the special gate in some circuit in $\mathcal{C}$. For a constant k depending on d, by the switching lemma [Hås87] with probability $1 - o(1)$ all these circuits will collapse to decision trees of depth $k-1$. (One can use the version of the switching lemma in [Bea94], and apply it iteratively along the lines of the proof of [BS90, Theorem 3.6].) By writing a decision tree as a DNF, noting that no two terms accept the same input, we see that each circuit in $\mathcal{C}$, after the restriction, is computable by either a polynomial-threshold function with degree $k-1$, or a function of the output of a polynomial of degree $k-1$ with polynomially-bounded coefficients. (The latter is a.k.a. Sym$\circ$And$_{k-1}$.)

By using Theorem 5 in the case of threshold gates, and [HG91, Lemma 4] in the case of arbitrary symmetric gates, each circuit in $\mathcal{C}$, after the restriction, can be computed with error probability $\leq 1\%/d$ by a k-player number-on-forehead protocol with $O(\lg n)$ communication.

By computing one special gate at the time, we have a protocol with communication $dO(\lg n) = O(\lg n)$ and error $\leq 1\%$ for f'. By Lemma 41,

$$R_k(f') \geq (\Omega(1)/2^c)^{2^k} \geq 1/O(n^{O(kd)})^{2^k} \geq 2/n^h,$$

for a constant h depending on k and d.

By a Chernoff bound, the probability that the approximation α is less than $1/n^\alpha$ is

$$\leq 2^{-D(1/2+1/n^h-0.5/n^h||1/2+1/n^h)s} \leq 2^{-2(0.5/n^h)^2 s} \leq o(1).$$

Hence the distinguisher accepts with probability $\geq 1 - o(1)$. ■

5.2 Candidate pseudorandom function in AC^0 with Mod m gates

We now construct candidate PRF computable by poly-size constant-depth AC^0 circuits with Mod m gates, for any m. Here a Mod m gate outputs 1 iff the Hamming weight of the input is divisible by m.

Such pseudorandom functions can be built from any PRF candidate $f : \{0,1\}^n \to \{0,1\}$ in space $O(\lg n)$, or even NL, with security $2^{n^{\Omega(1)}}$. We use the TC^0 candidates by Naor and Reingold, later with Rosen, in [NR99, NRR02]. For concreteness we focus on the candidate in [NRR02] which is based on the hardness of factoring. See [NRR02, §4.1,5] for background. For a bound $t(n)$, the *$t(n)$-factoring assumption* is the assumption that any algorithm running in time $t(n)$ has success probability $\leq 1/t(n)$ in factoring a uniformly-chosen n-bit Blum integer N, that is, an integer N that is equal to $P \cdot Q$ for P, Q two primes that are both congruent to 3 modulo 4. Conceivably, there is some $\epsilon > 0$ such that the 2^{n^ϵ}-factoring assumption holds. In fact, the (exponential) running time in this assumption is the type that is needed for the "Natural Proofs" connection by Razborov and Rudich [RR97].

Theorem 11. Assume there is an $\epsilon > 0$ such that the 2^{n^ϵ}-factoring assumption holds. Then for every m, c, there is a pseudorandom function $F : \{0,1\}^n \to \{0,1\}$ computable by $\mathrm{poly}(n)$-size AC^0 circuits with Mod m gates such that adversaries running in time $t(n) := 2^{\lg^c n} \geq n^{\omega(1)}$ have advantage $1/t(n)$ in distinguishing F from uniform.

The use of the Mod m gates in the above theorem is limited: for each seed, the function is computed by a circuit where all the Mod m gates are on the level closest to the input.

Proof: The work [NRR02] gives a pseudorandom function $f : \{0,1\}^n \to \{0,1\}$ computable by TC^0 circuits of size $\mathrm{poly}(n)$ such that no adversary running in time $t'(n) := 2^{\alpha n^\epsilon}$ has advantage $\geq 1/t'(n)$ in distinguishing the function from random, for a constant $\alpha > 0$.

Let $b = b(\epsilon, c)$ be a sufficiently large constant. Consider the pseudorandom function $f'_s : \{0,1\}^{n'} \to \{0,1\}$ which is the same as f above but on inputs of length $n' := \lg^b n$ and so with a key s of length $|s| = \mathrm{poly}(n') = \lg^{O(b)} n$. For a large enough b, the hardness of this function is as desired. That is, no algorithm running in time $2^{\lg^c n}$ can distinguish f'_s from a uniform random function $F' : \{0,1\}^{n'} \to \{0,1\}$ with advantage bigger than $2^{\lg^c n}$.

Using the fact that Majority on $\mathrm{poly} \lg n$ bits is computable by $\mathrm{poly}(n)$-size AC^0 circuits, or using alternatively a standard argument based on small-space computation which is usually attributed to Nepomnjaščiĭ [Nep70], f'_s is computable by $\mathrm{poly}(n)$-size AC^0 circuits.

What remains to do is to increase the input length from n' to n while maintaining security. Here we use the idea of Levin [Lev87] and combine f'_s with a hash function $h_t : \{0,1\}^n \to$

$\{0,1\}^{n'}$ with seed t to obtain the pseudorandom function $f_{s,t} : \{0,1\}^n \to \{0,1\}$ defined as $f_{s,t}(x) := f'_s(h_t(x))$. A hashing property that is sufficient is that for every $x \neq y$,

$$\Pr_t[h_t(x) = h_t(y)] \leq 1/2^{n'/\lceil \lg m \rceil}.$$

Below we construct such hash functions with the desired resources.

We now analyze the pseudorandomness of $f_{s,t}$. First, by enlarging b appropriately to take into account the complexity of computing the hash functions, any algorithm running in time $2^{\lg^c n}$ has advantage at most $0.5/2^{\lg^c n}$ in distinguishing $f_{s,t}$ from $F'(h_t(\cdot))$. To complete the analysis we need to argue that $F'(h_t(\cdot))$ and a uniform random function $F : \{0,1\}^n \to \{0,1\}$ cannot be distinguished with advantage bigger than $0.5/2^{\lg^c n}$. Indeed, the advantage is at most the probability of finding a collision of the hash function. To bound this collision probability, note that the probability that the queries result in *some* collisions is the same when the queries are made to $F'(h_t(\cdot))$ as when they are made to F, since until the first collision the query answers are the same. But for every fixed F the probability of a collision is at most

$$2^{2\lg^c n}/2^{n'/\lceil \lg m \rceil} = 2^{2\lg^c n - (\lg^b n)/\lceil \lg m \rceil} \leq 0.5/2^{\lg^c n}$$

for a sufficiently large b.

It remains to implement the hash function. First, it is easy to see, using padding, that even with boolean mod m gates one can compute, given a bit string, its Hamming weight modulo m, represented as a bit string of length $\lceil \lg m \rceil$. On an input x, the hash function outputs $\ell := n'/\lceil \lg m \rceil$ symbols modulo m corresponding to the sum modulo m of a random subset of the bits of x. It is easy to see that the probability that two distinct x, y hash to the same value is $\leq 2^\ell = 2^{-n'/\lceil \lg m \rceil}$ as desired. ■

Finally, we observe that the candidate pseudorandom function in Theorem 11 can also be implemented by polynomial-size, constant-depth circuits consisting exclusively of "generalized" Mod m gates when m is a fixed composite integer like 6, where a generalized Mod m gate is one whose output depends arbitrarily on the sum of the input bits modulo m. To see this one can use the fact that any function on $O(\lg n)$ bits can be computed by such circuits of size poly(n), see e.g. [ST06, Lemma 2 with no restriction on k]. Using in addition that an Or gate on poly $\lg n$ bits can be written as a constant-depth tree of Or gates on $\lg n$ bits, one sees that the function f'_s in the proof of Theorem 11 is computable with the desired resources. It is also not hard to see that the other component of the construction, the hash function, is computable with the desired resources.

References

[ABFR94] James Aspnes, Richard Beigel, Merrick Furst, and Steven Rudich. The expressive power of voting polynomials. *Combinatorica*, 14(2):135–148, 1994.

[AGHP92] Noga Alon, Oded Goldreich, Johan Håstad, and René Peralta. Simple constructions of almost k-wise independent random variables. *Random Structures & Algorithms*, 3(3):289–304, 1992.

[Bea94] Paul Beame. A switching lemma primer. Technical Report UW-CSE-95-07-01, Department of Computer Science and Engineering, University of Washington, November 1994. Available from http://www.cs.washington.edu/homes/beame/.

[Bei94] Richard Beigel. When do extra majority gates help? polylog(N) majority gates are equivalent to one. *Comput. Complexity*, 4(4):314–324, 1994.

[BNS92] László Babai, Noam Nisan, and Márió Szegedy. Multiparty protocols, pseudorandom generators for logspace, and time-space trade-offs. *J. of Computer and System Sciences*, 45(2):204–232, 1992.

[BOH08] Michael Ben-Or and Avinatan Hassidim. The bayesian learner is optimal for noisy binary search (and pretty good for quantum as well). In *IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 221–230, 2008.

[BS90] Ravi B. Boppana and Michael Sipser. The complexity of finite functions. In *Handbook of theoretical computer science, Vol. A*, pages 757–804. Elsevier, Amsterdam, 1990.

[BV10] Andrej Bogdanov and Emanuele Viola. Pseudorandom bits for polynomials. *SIAM J. on Computing*, 39(6):2464–2486, 2010.

[BW07] Alan Baker and Gisbert Wüstholz. *Logarithmic forms and Diophantine geometry*, volume 9 of *New Mathematical Monographs*. Cambridge University Press, 2007.

[BW12] Mark Braverman and Omri Weinstein. A discrepancy lower bound for information complexity. In *Workshop on Randomization and Computation (RANDOM)*, pages 459–470, 2012.

[CFL83] Ashok K. Chandra, Merrick L. Furst, and Richard J. Lipton. Multi-party protocols. In *15th ACM Symp. on the Theory of Computing (STOC)*, pages 94–99, 1983.

[CK82] Imre Csiszar and Janos Korner. *Information Theory: Coding Theorems for Discrete Memoryless Systems*. Academic Press, Inc., 1982.

[Cob64] Alan Cobham. The intrinsic computational di?culty of functions. In *Int. Congress for Logic, Methodology and Philosophy of Science*, page 2430, 1964.

[CT93] Fan R. K. Chung and Prasad Tetali. Communication complexity and quasi randomness. *SIAM J. Discrete Math.*, 6(1):110–123, 1993.

[DHKP97] Martin Dietzfelbinger, Torben Hagerup, Jyrki Katajainen, and Martti Penttonen. A reliable randomized algorithm for the closest-pair problem. *J. Algorithms*, 25(1):19–51, 1997.

[DPR$^+$13] Chinmoy Dutta, Gopal Pandurangan, Rajmohan Rajaraman, Zhifeng Sun, and Emanuele Viola. On the complexity of information spreading in dynamic networks. In *ACM-SIAM Symp. on Discrete Algorithms (SODA)*, 2013.

[FRPU94] Uriel Feige, Prabhakar Raghavan, David Peleg, and Eli Upfal. Computing with noisy information. *SIAM J. Comput.*, 23(5):1001–1018, 1994.

[FSS84] Merrick L. Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1):13–27, 1984.

[FSSS03] Jürgen Forster, Niels Schmitt, Hans-Ulrich Simon, and Thorsten Suttorp. Estimating the optimal margins of embeddings in euclidean half spaces. *Machine Learning*, 51(3):263–281, 2003.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. of the ACM*, 33(4):792–807, October 1986.

[GHR92] Mikael Goldmann, Johan Håstad, and Alexander A. Razborov. Majority gates vs. general weighted threshold gates. *Computational Complexity*, 2:277–300, 1992.

[GS10] Parikshit Gopalan and Rocco A. Servedio. Learning and lower bounds for AC^0 with threshold gates. In *Workshop on Randomization and Computation (RANDOM)*, pages 588–601, 2010.

[Hås87] Johan Håstad. *Computational limitations of small-depth circuits.* MIT Press, 1987.

[Hås94] Johan Håstad. On the size of weights for threshold gates. *SIAM J. Discrete Math.*, 7(3):484–492, 1994.

[HG91] Johan Håstad and Mikael Goldmann. On the power of small-depth threshold circuits. *Comput. Complexity*, 1(2):113–129, 1991.

[KL01] Matthias Krause and Stefan Lucks. On the minimal hardware complexity of pseudorandom function generators. In *Symp. on Theoretical Aspects of Computer Science (STACS)*, pages 419–430, 2001.

[KN97] Eyal Kushilevitz and Noam Nisan. *Communication complexity.* Cambridge University Press, 1997.

[Lev87] Leonid A. Levin. One way functions and pseudorandom generators. *Combinatorica*, 7(4):357–363, 1987.

[LMN93] Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, Fourier transform, and learnability. *J. of the ACM*, 40(3):607–620, 1993.

[LS09] Nathan Linial and Adi Shraibman. Learning complexity vs communication complexity. *Combinatorics, Probability & Computing*, 18(1-2):227–245, 2009.

[MNSW98] Peter Bro Miltersen, Noam Nisan, Shmuel Safra, and Avi Wigderson. On data structures and asymmetric communication complexity. *J. of Computer and System Sciences*, 57(1):37 – 49, 1998.

[MTT61] Saburo Muroga, Iwao Toda, and Satoru Takasu. Theory of majority decision elements. *J. Franklin Inst.*, 271:376–418, 1961.

[Mur71] Saburo Muroga. *Threshold logic and its applications.* Wiley-Interscience, New York, 1971.

[Nep70] Valery A. Nepomnjaščiĭ. Rudimentary predicates and Turing calculations. *Soviet Mathematics-Doklady*, 11(6):1462–1465, 1970.

[Nis93] Noam Nisan. The communication complexity of threshold gates. In *Combinatorics, Paul Erdős is Eighty, number 1 in Bolyai Society Mathematical Studies*, pages 301–315, 1993.

[NN93] Joseph Naor and Moni Naor. Small-bias probability spaces: efficient constructions and applications. *SIAM J. on Computing*, 22(4):838–856, 1993.

[NR99] Moni Naor and Omer Reingold. Synthesizers and their application to the parallel construction of pseudo-random functions. *J. Comput. Syst. Sci.*, 58(2):336–375, 1999.

[NR04] Moni Naor and Omer Reingold. Number-theoretic constructions of efficient pseudorandom functions. *J. of the ACM*, 51(2):231–262, 2004.

[NRR02] Moni Naor, Omer Reingold, and Alon Rosen. Pseudorandom functions and factoring. *SIAM J. Comput.*, 31(5):1383–1404, 2002.

[NZ96] Noam Nisan and David Zuckerman. Randomness is linear in space. *J. of Computer and System Sciences*, 52(1):43–52, February 1996.

[Pod09] Vladimir Podol′skiĭ. Perceptrons of large weight. *Problems of Information Transmission*, 45(1):46–53, 2009.

[Raz00] Ran Raz. The BNS-Chung criterion for multi-party communication complexity. *Comput. Complexity*, 9(2):113–122, 2000.

[RR97] Alexander Razborov and Steven Rudich. Natural proofs. *J. of Computer and System Sciences*, 55(1):24–35, August 1997.

[Smi88] D.V. Smirnov. Shannon's information methods for lower bounds for probabilistic communication complexity. Master's thesis, Moscow University, 1988.

[ST06] Howard Straubing and Denis Thérien. A note on $\mathrm{mod}_{\mathrm{p}}$ - $\mathrm{mod}_{\mathrm{m}}$ circuits. *Theory Comput. Syst.*, 39(5):699–706, 2006.

[SV10] Ronen Shaltiel and Emanuele Viola. Hardness amplification proofs require majority. *SIAM J. on Computing*, 39(7):3122–3154, 2010.

[Vio07] Emanuele Viola. Pseudorandom bits for constant-depth circuits with few arbitrary symmetric gates. *SIAM J. on Computing*, 36(5):1387–1403, 2007.

[Vio09] Emanuele Viola. Cell-probe lower bounds for prefix sums, 2009. arXiv:0906.1370v1.

[VW08] Emanuele Viola and Avi Wigderson. Norms, XOR lemmas, and lower bounds for GF(2) polynomials and multiparty protocols. *Theory of Computing*, 4:137–168, 2008.