Electronic Colloquium on Computational Complexity, Report No. 74 (2024)

Towards ACC Lower Bounds using Torus Polynomials

Vaibhav Krishan * Sundar Vishwanathan †

Abstract

The class ACC consists of Boolean functions that can be computed by constant-depth circuits of polynomial size with AND, NOT and MOD_m gates, where m is a natural number. At the frontier of our understanding lies a widely believed conjecture asserting that MAJORITY does not belong to ACC. The Boolean function MAJORITY outputs one if more than half of its inputs are one, and zero otherwise.

In a paper presented at ITCS 2019, Bhrushundi, Hosseini, Lovett and Rao reduced the conjecture that $\mathsf{MAJORITY} \notin \mathsf{ACC}$ to a conjecture concerning the non-existence of low degree torus polynomials that approximate MAJORITY. Torus polynomials approximate Boolean functions when the fractional part of their value on Boolean points is close to half the value of the function. Our contribution takes this a step further by reducing it to a seemingly more manageable conjecture regarding the ℓ^2 norm of specific vectors. The crux of our work lies in constructing machinery inspired by the method of *dual polynomials* to establish lower bounds on the degree of torus polynomials approximating Boolean functions. Along the way, we prove several key results, which include:

- A lower bound on the degree of *symmetric* torus polynomials approximating Δ_w functions, i.e., functions that output one if and only if there are exactly w input variables that are one, which includes the AND function. Consequently, we prove that asymmetric torus polynomials are *strictly more powerful* than their symmetric counterparts, addressing a question arising from Bhrushundi, Hosseini, Lovett and Rao (ITCS 2019).
- An error-degree trade-off for symmetric torus polynomials approximating MAJORITY, extending the corresponding result of Bhrushundi, Hosseini, Lovett and Rao (ITCS 2019).
- The existence of symmetric torus polynomials of degree at most half the number of variables, approximating MAJORITY within inverse exponential error, when the number of variables is one more than a power of two. This surprising aspect of our machinery shows its versatility in proving upper bounds, despite being initially developed for lower bounds.
- The *first* known lower bounds against asymmetric torus polynomials, showcasing the power of the machinery we develop. We prove that any torus polynomial approximating AND within an error of $\frac{1}{2n}$ must have degree at least $\Omega(\log(n))$. Compare this with an upper bound of $\log^2(n)$, which follows from Bhrushundi, Hosseini, Lovett and Rao (ITCS 2019). Hence, we get an almost complete characterization of the torus polynomial approximation degree of AND.
- Lower bounds against asymmetric torus polynomials approximating MAJORITY, or AND, in the very low error regime. Here, we prove that when the degree is one less than the number of variables, symmetric and asymmetric torus polynomials are equivalent in their power.

The machinery we have developed has significant room for further analysis, and leads to numerous open problems. There are various combinatorial questions, interesting in their own right, that also arise from our work.

*Department of Computer Science and Engineering, IIT Bombay, Mumbai, India. vaibhkrishan@iitb.ac.in
†Department of Computer Science and Engineering, IIT Bombay, Mumbai, India. sundar@cse.iitb.ac.in

ISSN 1433-8092

Additionally, we apply our machinery to prove lower bounds on the degree of real polynomials approximating Boolean functions. We are able to prove strong lower bounds for a large set of functions with minimal effort.

1 Introduction

Proving that a complexity class is not contained in another, is the prime focus of complexity theorists and such questions make up some of the hardest problems in Computer Science. We study such a question at the frontier of our knowledge about Boolean circuit complexity classes. To state the question, we first need to define the two classes of Boolean circuits we consider.

The first class, called ACC, consists of constant-depth Boolean circuits of polynomial size comprising $\mathsf{AND}, \mathsf{NOT}$, and MOD_m gates. A MOD_m gate outputs one if and only if the count of ones in the input is divisible by the natural number m. The second class consists of constant-depth Boolean circuits of polynomial size comprising linear threshold gates. A linear threshold gate outputs one if and only if a specific linear combination of its inputs crosses a predetermined threshold. This class is called TC^0. It is easy to see, and is a well-known folklore result, that ACC is a subset of TC^0. Nearly 35 years ago, Yao [Yao90] conjectured that this containment is strict. This question has remained unanswered since.

Conjecture 1 ([Yao90])**.** $\mathsf{ACC} \subsetneq \mathsf{TC}^0$.

This conjecture can be restated as follows: there is a Boolean function $f \in \mathsf{TC}^0$ which cannot be computed by circuits in ACC. A candidate function for this is the majority function, i.e. the Boolean function that outputs one if and only if ones are in the majority in the input, denoted by MAJ_n. Barrington [Bar89] hypothesized that the majority function is not in ACC.

Conjecture 2 ([Bar89])**.** $\mathsf{MAJ}_n \notin \mathsf{ACC}$.

The majority function is one of the simplest threshold functions, as it outputs one if and only if $\sum_{i=1}^{n} x_i > \frac{n}{2}$. Hence, the majority function belongs to TC^0. Therefore, this conjecture is a refinement of Conjecture 1. Our work is towards the goal of proving this conjecture, i.e. Conjecture 2.

Hence, our task is to prove that a particular circuit class cannot compute a certain function. In the literature, this task is referred to as proving *lower bounds* against that class. We outline a few major approaches that have led to lower bounds in the past.

One of the approaches is based on "simplification", counting, and the probabilistic method, for example: using *random restrictions*. This is a classical technique, developed in the 80s. Håstad [Hås86] proved a landmark result, that constant-depth circuits, composed of AND and NOT gates, even when quite large, simplify considerably when a significant fraction of the input variables are assigned values randomly. It is easy to see that the parity function does not undergo such simplification. This intuition was formalized to prove that constant-depth circuits comprising AND and NOT gates require large size to compute parity. This result concluded a line of work by Ajtai [Ajt83], Furst, Saxe and Sipser [FSS84], and Yao [Yao85], by proving nearly optimal lower bounds against such circuits. Note that parity is in ACC, hence applying the technique of random restrictions does not seem useful for proving Conjecture 2.

Another approach is based on the *easy witness lemma*, proved by Impagliazzo, Kabanets and Wigderson [IKW02] in a groundbreaking work. Williams [Wil13], in a remarkable recent work, used this result to devise a clever approach for proving lower bounds. They proved that non-trivial algorithms for determining whether a circuit from a particular circuit class ever outputs one, leads

to lower bounds against that class. Then, in a subsequent breakthrough, Williams [Wil14] used this approach to prove that the class of functions computable by exponential time non-deterministic algorithms is not contained in ACC. Subsequent works, such as by Chen, Oliveira and Santhanam [COS18], Murray and Williams [MW18], Chen [Che19], Chen, Lyu and Williams [CLW20], and Chen [Che23] have improved the lower bound considerably. This method, although remarkably successful, seems unable to prove that deterministic classes are not contained in ACC. In particular, it does not seem to yield an approach to prove that P, the class of functions computable by polynomial time deterministic algorithms, is not contained in ACC, or to prove Conjecture 2. Recall that our main objective is to prove that a much simpler function, namely the majority function, which is contained in both P and TC^0, is not contained in ACC.

This leaves us with another classical approach, based on the so-called *polynomial method*. In this framework, researchers study various notions of representing Boolean functions using polynomials. This framework is quite powerful, and has numerous applications, such as in interactive and probabilistically checkable proofs, coding theory, circuit lower bounds, communication complexity, quantum complexity theory, learning theory, and much more. See this survey by Aaronson [Aar08] for an interesting and insightful account. We describe two such notions of approximations by polynomials that have found numerous uses in the study of Boolean circuits.

The first notion, referred to as real polynomial approximation, uses polynomials over the reals to approximate Boolean functions pointwise. Here is a formal definition.

Definition 1 (Real Polynomial Approximation)**.** *Consider $P \in \mathbb{R}[x_1, \ldots, x_n]$, a polynomial over the reals, and $f : \{0,1\}^n \to \{0,1\}$, a Boolean function. Then, P is said to approximate f within an error of ε if for all Boolean points $x \in \{0,1\}^n$, $|P(x) - f(x)| \leq \varepsilon$.*

Nisan and Szegedy [NS94], in a seminal work, studied the minimum degree of a polynomial that approximates a given function within an error of $\frac{1}{3}$, called the real polynomial approximation degree. They proved surprising connections between this degree and other complexity measures, such as the *decision tree complexity*. Their work provided considerable impetus to the study of real polynomial approximations, and firmly established their use in mainstream complexity theory. Shortly thereafter, Paturi [Pat92] completely characterized the real polynomial approximation degree of *symmetric*[1] functions. Subsequently, researchers have developed this line of study in a myriad of ways. Several works have devised techniques to prove upper and lower bounds on the real polynomial approximation degree in various special cases, discovered connections with other areas in computation complexity, found uses in constructing algorithms, etc. Today, the study of approximating Boolean functions by real polynomials is a subfield by itself. See this survey by Bun and Thaler [BT22], and references therein, for a comprehensive introduction.

The second notion of polynomial approximation uses polynomials over finite fields to compute Boolean functions on *most* Boolean points. A polynomial over a finite field is said to approximate a Boolean function within an error of ε, if the value of the polynomial and the function differ on at most an ε fraction of the Boolean points. Razborov [Raz87], and Smolensky [Smo87], in independent works, pioneered the use of polynomials over finite fields. Take a function f computable by constant-depth circuits of polynomial size consisting of $\mathsf{AND}, \mathsf{NOT}$ and MOD_p gates, for a prime p. Then, they proved that there exists a "low" degree polynomial over $\mathbb{F}_p$, the finite field of size p, that approximates f within an error of $\frac{1}{3}$. They also proved that the same does not hold for the majority function, or the MOD_q function, for a prime $q \neq p$. Hence, they proved that constant-depth circuits of polynomial size consisting of $\mathsf{AND}, \mathsf{NOT}$ and MOD_p gates cannot compute the majority function, or the MOD_q function.

[1] A Boolean function is symmetric if it remains unchanged under permutations of its variables.

Broadly speaking, in order to prove that a function f is not contained in a class $\mathcal{C}$, the theme here is to find a *distinguisher*. A distinguisher is a function μ that maps f to a point outside the image of $\mathcal{C}$ under μ, proving $f \notin \mathcal{C}$. That is, proving $\mu(f) \notin \mu(\mathcal{C})$ implies $f \notin \mathcal{C}$.

Neither of these methods seem useful to prove lower bounds against ACC. This is evident by considering the MOD_6 function. It requires a high degree to approximate using either real polynomials [Pat92], or polynomials over finite fields [Smo87], within a reasonable error. In fact, for a long time, there were no known polynomial method based approaches for resolving Conjecture 1.

Then, in a recent pivotal work, Bhrushundi, Hosseini, Lovett and Rao [BHLR19] made an inspired suggestion of using the degree of torus polynomials as a distinguisher for the ACC vs TC^0 question. They proved that torus polynomial approximations extend both real polynomial approximations and approximation using polynomials over finite fields (See [BHLR19, Lemma 14]). In fact, they have even more power than both of them combined. For example, torus polynomials can approximate the MOD_6 function. We give a definition of torus polynomial approximation, which is an equivalent restatement of Definition 1 in [BHLR19].

Definition 2 (Torus Polynomial Approximation)**.** *Let $f : \{0,1\}^n \to \{0,1\}$ be a boolean function and $P \in \mathbb{R}[x_1, \ldots, x_n]$ be a real polynomial. We say that P is a torus polynomial that approximates f within an error of ε if the following holds:*

There exist functions $Z : \{0,1\}^n \to \mathbb{Z}$ and $\delta : \{0,1\}^n \to [-\varepsilon, \varepsilon]$ such that:

$$\forall x \in \{0,1\}^n, P(x) = \frac{f(x)}{2} + Z(x) + \delta(x)$$

In other words, the fractional part of $P(x)$ is within ε of $\frac{f(x)}{2}$.

Given any function $f : \{0,1\}^n \to \{0,1\}$, one can simply consider the unique polynomial that exactly matches f on all Boolean points[2]. This would require degree n for most functions. Note that this leads to an approximation with zero error. Hence, the question is, for which functions f, does there exist a torus polynomial of much smaller degree, say $\log^2(n)$, approximating it within a reasonably small error, say $\frac{1}{n^2}$. Bhrushundi et al. [BHLR19, Corollary 20] proved that something similar holds for all functions belonging to ACC.

Theorem 1.1 ([BHLR19])**.** *Consider any function $f : \{0,1\}^n \to \{0,1\}$ belonging to the class ACC. Then, for any $\varepsilon = \frac{1}{n^{O(1)}}$, there exists a torus polynomial of degree at most $\log(n)^{O(1)}$ that approximates f within an error of ε.*

Hence, proving that the same does not hold for the majority function will resolve the ACC vs TC^0 question. This is precisely the goal of our work in this paper. While we could not take this task to completion, we do make progress towards it. Before discussing our contributions, we look at what is already known about torus polynomials.

1.1 Previous Work

Bhrushundi et al. [BHLR19] defined torus polynomials with the goal of proving Conjecture 2, that the majority function does not belong to ACC. Now, majority is a symmetric function. Hence, it is natural to study *symmetric*[3] torus polynomials that approximate the majority function as the first step. They indeed studied the same, and proved that any symmetric torus polynomial approximating the majority function within inverse linear error must have degree at least $\Omega\left(\sqrt{\frac{n}{\log(n)}}\right)$.

[2]The existence of such a polynomial is folklore.

[3]A polynomial is symmetric if it is invariant under permutation of its variables.

Theorem 1.2 ([BHLR19]). *Any symmetric torus polynomial approximating the majority function within an error of* $\frac{1}{20n}$ *must have degree* $\Omega\left(\sqrt{\frac{n}{\log n}}\right)$.

Now, a priori, this does not lead to a resolution of Conjecture 2. It requires proving an analogous statement for *asymmetric* torus polynomials. One would expect that as the majority function is symmetric, one need not consider asymmetric torus polynomials. This is certainly the case with real polynomials, which can be proved using a well known symmetrization technique. Given a real polynomial P over n variables, that approximates a symmetric function, obtain other polynomials by permuting its variables. Now, average over all polynomials obtained this way by considering all permutations over n variables. The degree, and the error of approximation, remains the same after this procedure, and one obtains a symmetric real polynomial.

This procedure does not work for asymmetric torus polynomials. When one averages over all permutations of variables, the average of the integer parts may not turn out to be an integer. Then, it will end up contributing to the fractional part, destroying any hope of retaining the approximation error. Regardless, Bhrushundi et al. [BHLR19, Conjecture 5] conjectured that a lower bound similar to Theorem 1.2 holds for asymmetric torus polynomials as well.

Conjecture 3 ([BHLR19]). *Any torus polynomial approximating the majority function within error* $\frac{1}{20n}$ *must have degree* $\Omega\left(\sqrt{\frac{n}{\log n}}\right)$.

This conjecture, if true, proves Conjecture 2. Moreover, it will separate P from ACC, improving our knowledge well beyond what is currently known. In fact, in a recent work, Chen, Lu, Lyu and Oliveira [CLLO21] proved that it leads to an even stronger statement. They proved that if the previous conjecture is true, then no ACC circuit matches the majority function's value on significantly more than half the inputs.

Before we proceed further, we briefly discuss the proof of Theorem 1.2, as proved by Bhrushundi et al. [BHLR19]. They employ a three step combinatorial argument, roughly outlined below. They consider the set of all symmetric functions, and take $d = o\left(\sqrt{\frac{n}{\log(n)}}\right)$. In the first step, they prove that there exists a symmetric function, such that no symmetric torus polynomial of degree at most d can approximate it within an error of $\frac{1}{10}$. Then, they consider Δ_w functions. The Δ_w function, for a $0 \leq w \leq n$, outputs one if and only if the sum of its inputs equals w. They prove that the statement above implies that there exists a w such that no torus polynomial of degree at most d can approximate the Δ_w function within an error of $\frac{1}{20n}$. Then, they use this to prove the same for the majority function.

This leads to the following natural question. The second step of the proof is existential. It only guarantees the existence of w for which the lower bound holds. Can we explicitly identify such a w? In particular, what about the simplest such function, $w = n$, i.e. the AND function? Nisan and Szegedy [NS94], in one of the earliest works on real polynomial approximations, prove a lower bound against real polynomial approximations for AND. On the other hand, there are *no* known lower bounds against torus polynomial approximations for the AND function. We fill this gap in our work in a strong sense.

Also, what if we require a smaller error of approximation. Does it lead to a higher degree lower bound? We answer this in the affirmative by proving an error-degree trade-off.

1.2 Our Contribution

Bhrushundi et al. [BHLR19] revived the polynomial method as an approach towards proving lower bounds against ACC. They reduced this question to proving lower bounds against torus polynomials,

an algebraic object more amenable to mathematical techniques. Still, proving lower bounds is asking for proving non-existence. We take the next step, and convert this into an existence question. This is often an important step in proving influential results, as seen in [HMP+93, MS08, TS08, Spa08, CM22] for just a few examples.

Our main contribution in this work is considerably extending the method of *dual polynomials*, and applying it to torus polynomials. The dual polynomial approach is a highly successful method, and has led to several interesting results about real polynomial approximations. We refer the reader to this survey by Bun and Thaler [BT22] for a comprehensive discussion of this method, and its applications. This method can be summarized as follows.

- Consider any real polynomial P of degree d. Assume that it approximates a Boolean function f within an error of ε. This implies that the values of P satisfy certain conditions, based on f and ε.
- Write the conditions that P must satisfy as a linear program, so that this linear program is feasible if and only if P approximates f within an error of ε.
- Use duality in linear programming to obtain another linear program, called the *dual*. The dual is unbounded if and only if the original program is infeasible.
- Prove that the dual is indeed unbounded. This proves the non-existence of such a P.

We start by doing the same for torus polynomials, by interpreting a torus polynomial as a real polynomial that can have arbitrary integral parts. This leads to a family of linear programs, for each combination of integral parts, and we convert each of them into their corresponding dual. For proving the non-existence of a torus polynomial, we need to prove that the whole family of dual linear programs is feasible. This is a key challenge we confront.

The starting point of our approach is very similar to how researchers have constructed dual polynomials in the literature. However, once we construct the dual, our approach for proving the feasibility is completely different, based on geometric arguments, rather than the earlier analytical approaches. Also, the linear program developed in the literature is constructed without assuming anything about the symmetry of the polynomial. Our first observation is that in the symmetric case, the linear program simplifies considerably. While it may not make a difference in the real polynomials case, this path is imperative while dealing with torus polynomials. Also, the resulting dimension reduction leads to a significant ease in analysis. We prove several results using this approach, including contributing to the theory of real polynomials. Following is a discussion of these results.

Consider the following quote, from the fourth paragraph of [BHLR19, Section 1.2]. *"Unfortunately, the aforementioned idea of symmetrization cannot be used in the setting of torus polynomials in a straightforward manner and so it's unclear how powerful non-symmetric torus polynomials are compared to their symmetric counterparts."*

They raise the question of the relative power of symmetric torus polynomials versus their asymmetric counterparts. We first prove that for an inverse linear error a statement similar to Theorem 1.2 holds for *all* Δ_w functions, extending the results of Bhrushundi et al. [BHLR19] in a strong form. This simultaneously answers the symmetric versus asymmetric question. Indeed, our lower bound holds for $\mathsf{AND} = \Delta_n$. However, Bhrushundi et al. [BHLR19] proved that low degree torus polynomials can approximate AND within an inverse linear error. Hence, asymmetric torus polynomials afford more power than symmetric torus polynomials, even when approximating symmetric Boolean functions.

Next, we consider a natural question, viz, does assuming a smaller error bound result in a higher degree lower bound. We answer this in the affirmative for the majority function, serving another extension of Theorem 1.2. Following are the formal statements of these results.

Theorem 1.3. *The following holds for any large n.*

1. ***All Delta Functions**. Let $0 \leq w \leq n$. Any symmetric torus polynomial that approximates the Δ_w function within an error of $\frac{1}{2n}$ must have degree d at least $\Omega\left(\sqrt{\frac{n}{\log(n)}}\right)$.*

2. ***Error-Degree Trade-off**. Fix $c \in \mathbb{R}$, $c \geq 0$. Let $\varepsilon = \frac{1}{2^{1+\log(n)+\log^{c+1}(n)}}$. Then there exists a w such that any symmetric torus polynomial that approximates Δ_w within an error of ε must have degree d at least $\Omega\left(\sqrt{n\log^c(n)}\right)$.*

Note that we get a degree lower bound of $\sqrt{n}$, by setting $c = 0$ in the error-degree trade-off result, when ε is inverse polynomial, slightly improving the lower bound by Bhrushundi et al. [BHLR19].

As a corollary of the previous result, we get the following statement.

Corollary 1. *There are symmetric functions, such as the* AND *function, that have torus polynomials of low degree approximating them within a small error. On the other hand, any symmetric torus polynomial approximating them within a small error must have large degree.*

Also, consider the fact that there is a real polynomial of degree $O(\sqrt{n})$ approximating AND within an error of $\frac{1}{3}$ [Pat92]. Using standard error reduction for real polynomials, such as from [BT22, Theorem 10], we get that there is a real polynomial of degree $O(\sqrt{n}\log(n))$ approximating AND within an error of $\frac{1}{2n}$. Using the standard technique of symmetrization, we can assume that this real polynomial is symmetric. Hence, this real polynomial is also a symmetric torus polynomial of degree $O(\sqrt{n}\log(n))$ approximating AND within an error of $\frac{1}{2n}$. Therefore, our degree lower bound of $\sqrt{\frac{n}{\log(n)}}$ is tight within logarithmic factors.

Next, we consider an unexplored error regime, where the error is inverse exponentially small. We prove that all symmetric functions have symmetric torus polynomials of degree at most $n-1$ approximating them within this miniscule error. Then, we prove a matching lower bound for AND and the majority function. Moreover, as opposed to when the degree is lower, we prove that symmetrization holds when the degree is $n-1$. However, the argument is not as simple as the one for real polynomials, and requires additional tricks. This adds even more intrigue to torus polynomials, as they behave quite differently from real polynomials.

One interesting aspect of our proof of the above results is that we find a single certificate for the feasibility of a whole family of dual linear programs. This leaves room for exploring multiple certificates, each of which works for a subset of the family, such that each dual in the family is covered by some certificate. We develop a theory, and propose a geometric method, to do exactly this. In fact, we show that this method leads to almost matching upper and lower bounds in the case of symmetric torus polynomials. Then, we exhibit the power of our method by proving upper bounds, i.e. the existence of a symmetric torus polynomial approximating the majority function within an inverse exponential error, for a careful choice of n and the degree d.

Theorem 1.4. *Consider $d = 2^t$ for some natural number $t \in \mathbb{N}$, and $n = 2d+1$. Then, there exists a symmetric torus polynomial of degree at most d that approximates MAJ_n within an error of $\frac{1}{2^{\Omega(n)}}$.*

Then, we shift our attention towards the main goal of this paper, i.e. proving Conjecture 2. We extend the geometric method we develop for the symmetric case to the asymmetric case as well. This allows us to reduce Conjecture 2 to a concrete combinatorial conjecture, that seems much more tractable. This conjecture appears as Conjecture 13. Our method for the asymmetric case also leads to lower bounds against torus polynomials approximating the AND function. These lower bounds lead to tight bounds on the torus polynomial degree of AND, up to a quadratic factor.

Theorem 1.5. *Any torus polynomial approximating the* AND *function within an error of* $O\left(\frac{1}{2^{\log^c(n)}}\right)$ *must have degree* $\Omega(\log^c(n))$.

For example, if we choose $c = 1$, this implies that any torus polynomial approximating the AND function within an error of $O\left(\frac{1}{n}\right)$ must have degree $\Omega(\log(n))$. Note that [BHLR19, Lemma 14], when combined with [All89, Lemma 1], implies a degree upper bound of $\log^2(n)$ for an error of $O\left(\frac{1}{n}\right)$. Hence, the bound is tight up to a quadratic factor. In general, the two works cited above imply a degree upper bound of $\log^{c+1}(n)$ for an error of $O\left(\frac{1}{2^{\log^{c+1}(n)}}\right)$. Hence, our lower bound of $\log^c(n)$ is tight up to a logarithmic factor. We leave it as an open problem to bring the lower bound and the upper bound closer.

Problem 4. *Find* $d : \mathbb{N} \to \mathbb{N}$ *as a function of* n*, such that each of the following holds:*

- *There exists a torus polynomial of degree* $O(d(n))$ *approximating* AND *within an error of* $O\left(\frac{1}{2^{\log^c(n)}}\right)$.
- *Any torus polynomial approximating* AND *within an error of* $O\left(\frac{1}{2^{\log^c(n)}}\right)$ *must have degree at least* $\Omega(d(n))$.

Finally, if we consider real polynomials, it becomes much simpler to apply our methods, as we have a single dual to deal with, namely with all integral parts being zero. Hence, our methods are applicable even for real polynomials. In fact, once we set up the machinery, it becomes exceedingly easy to prove lower bounds against real polynomials with an artful choice of n and d. We demonstrate this ease by proving lower bounds for a large family of Boolean functions. While these lower bounds do not improve known lower bounds significantly, we prove them using our method with the hope that it may be useful for more functions.

Theorem 1.6. *Fix a constant* $0 < c \leq 1$. *Consider an* odd n. *Take* $f : \{0,1\}^n \to \{0,1\}$ *to be any function that evaluates to* 0 *on any Boolean point* $x \in \{0,1\}^n$ *of even Hamming weight. If* f *evaluates to* 1 *on a* c *fraction of the inputs with odd Hamming weight, then the following holds. Any real polynomial approximating* f *within an error of* $\varepsilon < \frac{\sqrt{c}}{2}$ *must have degree at least* $\frac{n+1}{2}$.

To compare, the dual polynomial for parity, which we take from [BT22], implies the following. For a function f such as the one described above, it implies that the degree must be n if the real polynomial approximates f within an error of $\varepsilon < \frac{c}{2}$. Our result improves the error quadratically, a minor improvement, while decreasing the degree by half. Hence, these results seem incomparable.

1.3 Organization

We start with some preliminaries in Section 2. Then, we present our lower bounds against symmetric torus polynomials in Section 3. There, we also develop a method for potentially proving stronger lower bounds. Next, in Section 4, we prove matching lower and upper bounds for torus

polynomials approximating the majority function, as well as the AND function, within an inverse exponential error. In Section 5, we use the method developed in the previous sections to prove the existence of symmetric torus polynomials that approximate the majority function within an inverse exponential error. After this, in Section 6, we develop our methods further towards the goal of proving Conjecture 2. In Section 7, we show how to use our method to easily prove lower bounds against real polynomials approximations for a large family of Boolean functions. We close with a discussion on some future directions in Section 8.

2 Preliminaries

We begin with some standard definitions.

Definition 3 (Hamming weight)**.** *The* Hamming weight *of a Boolean point* $\mathbf{x} \in \{0,1\}^n$ *is defined as* $|\mathbf{x}| = \sum_{i=1}^{n} x_i$*. That is, the Hamming weight of the vector equals the number of ones in it.*

Definition 4 (ℓ^p norm)**.** *For a vector* $\mathbf{v} \in \mathbb{R}^n$ *and a real number* $p \neq 0$*, the* ℓ^p *norm* $\|\mathbf{v}\|_p$ *is defined as* $(\sum_{i=1}^{n} \mathbf{v}_i^p)^{\frac{1}{p}}$*.*

We will also need the following inequality.

Lemma 2.1 (Folklore)**.** *The* ℓ^1 *and* ℓ^2 *norms of a vector* $\mathbf{v} \in \mathbb{R}^n$ *satisfy* $\|\mathbf{v}\|_2 \leq \|\mathbf{v}\|_1 \leq \sqrt{n}\|\mathbf{v}\|_2$*.*

Definition 5 (Integer Lattice)**.** *Consider a matrix* $B \in \mathcal{M}_{n\times m}(\mathbb{R})$ *of dimensions* $n \times m$ *with real entries, with linearly independent columns* $\mathbf{b}_1, \ldots, \mathbf{b}_m$*. Then, the* lattice *generated by this matrix is defined as all integer combinations of the columns, i.e.* $\mathcal{L}(B) = \{\sum_{i=1}^{m} z_i \mathbf{b}_i : z_i \in \mathbb{Z}\}$*.*

The following theorem provides an upper bound on the length of the shortest vector in a lattice.

Theorem 2.2 (Minkowski [Min10])**.** *Consider a real matrix* $B \in \mathcal{M}_{n\times m}(\mathbb{R})$*. Then, the lattice* $\mathcal{L}(B)$ *contains a vector* $\mathbf{v}$ *with its* ℓ^2 *norm bounded by* $\|\mathbf{v}\|_2 \leq \sqrt{m}\, det(B^T B)^{\frac{1}{2m}}$*.*

The following corollary is immediate from Lemma 2.1. See also [Ngu09].

Corollary 2. *The lattice* $\mathcal{L}(B)$ *contains a vector* $\mathbf{v}$ *with* $\|\mathbf{v}\|_1 \leq \sqrt{n}\sqrt{m}\, det(B^T B)^{\frac{1}{2m}}$*.*

3 Limitations of Symmetric Torus Polynomials

In this section, we generalize the method of *dual polynomials*, a method previously applied to obtain lower bounds against real polynomials, and apply it to torus polynomials. Although we take the same initial step, our method deviates significantly afterward. One key observation that makes it easier for us to prove our results is that the linear program we construct is different and much smaller in case of symmetric polynomials.

3.1 Symmetric Torus Polynomial Lower Bounds

We begin by recalling the statement of Theorem 1.3.

Theorem 1.3. *The following holds for any large* n*.*

1. ***All Delta Functions****. Let* $0 \leq w \leq n$*. Any symmetric torus polynomial that approximates the* Δ_w *function within an error of* $\frac{1}{2n}$ *must have degree* d *at least* $\Omega\left(\sqrt{\frac{n}{\log(n)}}\right)$*.*

2. ***Error-Degree Trade-off***. *Fix $c \in \mathbb{R}$, $c \geq 0$. Let $\varepsilon = \frac{1}{2^{1+\log(n)+\log^{c+1}(n)}}$. Then there exists a w such that any symmetric torus polynomial that approximates Δ_w within an error of ε must have degree d at least $\Omega\left(\sqrt{n\log^c(n)}\right)$.*

Proof. The proof consists of two parts. Part 1 is common to both the claims.

Part 1: We begin by describing a general method for proving lower bounds on the degree of a symmetric torus polynomial that approximates a symmetric Boolean function. This is similar to the method of dual polynomials, popular in the study of real polynomials. We write the conditions that a torus polynomial must fulfill in order to approximate a Boolean function, within a certain error, as a linear program. The linear program we consider is much smaller than what is considered in the literature, precisely because the polynomial is symmetric. This program is feasible if and only if such a polynomial exists. Then, we convert this linear program into its dual linear program. By a theorem of alternatives, the dual linear program is feasible if and only if the required polynomial does not exist. The formal statement is as follows.

Lemma 3.1. *Define the matrix $A \in \mathcal{M}_{2(n+1)\times(n-d)}(\mathbb{R})$ with entries $A_{i,j} = \binom{i}{j}$, $A_{i+n+1,j} = -\binom{i}{j}$ for $0 \leq i \leq n, 0 \leq j \leq d$. For a tuple $(z_0, \ldots, z_n) \in \mathbb{Z}^{n+1}$, define $\mathbf{b}_i = z_i + \varepsilon + \frac{f(i)}{2}$ and $\mathbf{b}_{i+n+1} = -z_i + \varepsilon - \frac{f(i)}{2}$ for $0 \leq i \leq n$. Then, exactly one of the following statements holds.*

- *There exists a symmetric torus polynomial of degree d approximating f within an error of ε.*
- *The following linear program, called the dual, is feasible for all tuples $(z_0, \ldots, z_n)$.*

$$\begin{aligned} A^T\beta &= 0 \\ \beta &\geq 0 \\ \mathbf{b}^T \cdot \beta &< 0 \end{aligned}$$

Proof. Consider a symmetric torus polynomial P of degree d that approximates a symmetric Boolean function $f : \{0,1\}^n \to \{0,1\}$ within an error of ε. As P is symmetric, monomials of the same degree have the same coefficient. For $0 \leq j \leq d$, denote by α_j the coefficient of the degree j monomials of P. Then, for a point x with Hamming weight i for $0 \leq i \leq n$, $P(x)$ evaluates to $\sum_{j=0}^{d} \binom{i}{j}\alpha_j$.

Now, Definition 2 implies that there exist functions $Z : \{0,1\}^n \to \mathbb{Z}$ and $\delta : \{0,1\}^n \to [-\varepsilon, \varepsilon]$ such that $P(x) = \frac{f(x)}{2} + Z(x) + \delta(x)$ for any Boolean vector $x \in \{0,1\}^n$. As P is a symmetric polynomial, it evaluates to the same value on Boolean points of the same Hamming weight. This implies that Z and δ are symmetric functions as well, and evaluate to the same value on Boolean points of the same Hamming weight.

For each $0 \leq i \leq n$, denote by z_i the value of $Z(x)$ for a vector x with Hamming weight $|x| = i$. Then, the condition for the existence of a torus polynomial approximation implies that $z_i - \varepsilon + \frac{f(i)}{2} \leq P(x) \leq z_i + \varepsilon + \frac{f(i)}{2}$, for each $0 \leq i \leq n$. We collect these as a system of linear inequalities as follows.

$$A_{2(n+1)\times(d+1)}\alpha \leq \mathbf{b}$$

Here, $A_{i,j} = \binom{i}{j}$, $\mathbf{b}_i = z_i + \varepsilon + \frac{f(i)}{2}$, $A_{i+n+1,j} = -\binom{i}{j}$ and $\mathbf{b}_{i+n+1} = -z_i + \varepsilon - \frac{f(i)}{2}$ for $0 \leq i \leq n, 0 \leq j \leq d$. Note that $\mathbf{b}$ is a function of Z, while A is independent of Z.

We note that each tuple $(z_0, z_1, \ldots, z_n)$ defines a linear program. Hence, to prove that such an approximation is not possible, it suffices to prove that the corresponding linear program is infeasible for each $(z_0, z_1, \ldots, z_n)$.

By Farkas' lemma [Far02], this system is infeasible[4] if and only if the following system is feasible:

$$\begin{aligned} A^T\beta &= 0 \\ \beta &\geq 0 \\ \mathbf{b}^T \cdot \beta &< 0 \end{aligned}$$

This completes the proof of the statement. ■

We will prove that the dual is feasible for all choices of $f = \Delta_w$, for any $(z_0, \ldots, z_n)$, and the given regime of d and ε.

First, we focus on the first two expressions in the dual, i.e. $A^T\beta = 0, \beta \geq 0$. This system is known as a *polyhedral cone*. Geometrically, to prove that the dual is feasible, we need to prove the existence of a ray in the cone which forms an obtuse angle with $\mathbf{b}$. It suffices to identify the extreme rays of the cone since, if there exists a ray in the cone forming an obtuse angle with $\mathbf{b}$, one of the extreme rays would also form an obtuse angle with vector $\mathbf{b}$.

However, obtaining a description of the extreme rays looks like a difficult exercise. Instead, we leverage the fact that A^T has a special structure, and show that it suffices to characterize the nullspace of a suitable matrix. To demonstrate this special structure, we write A^T below.

$$A^T = \begin{bmatrix} \binom{0}{0} & \cdots & \binom{i}{0} & \cdots & \binom{n}{0} & -\binom{0}{0} & \cdots & -\binom{i}{0} & \cdots & -\binom{n}{0} \\ \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ \binom{0}{j} & \cdots & \binom{i}{j} & \cdots & \binom{n}{j} & -\binom{0}{j} & \cdots & -\binom{i}{j} & \cdots & -\binom{n}{j} \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ \binom{0}{d} & \cdots & \binom{i}{d} & \cdots & \binom{n}{d} & -\binom{0}{d} & \cdots & -\binom{i}{d} & \cdots & -\binom{n}{d} \end{bmatrix}_{(d+1)\times 2(n+1)}$$

First, note that A^T is upper triangular with $d+1$ non-zero entries on the main diagonal. Hence, it has rank $d+1$. Therefore, the nullspace of A^T has dimension $2(n+1) - (d+1)$. We describe a basis for the nullspace of A^T.

Next, note that A^T has form $A^T = [M \mid -M]$. Hence, $A^T \cdot [e_i \mid e_i] = 0$ for each $0 \leq i \leq n$. There are $n+1$ of these, and each of them forms an element of our chosen basis.

We need $n - d$ additional basis vectors to describe the nullspace of A^T. We obtain these from the nullspace of M, denoted henceforth as $null(M)$, using a procedure that we will describe later. First, we construct a basis for $null(M)$.

Lemma 3.2. *A basis for $null(M)$ consists of columns of the matrix $B \in \mathcal{M}_{(n+1)\times(n-d)}(\mathbb{R})$ where $B_{i,k} = (-1)^{i-k}\binom{d+1}{i-k}$.*

Proof. We need to prove that $MB = 0$, and that the columns of B are linearly independent First, note that B is a lower triangular matrix with non-zero entries on the main diagonal. Also, B has fewer columns than rows. Therefore, B has full column rank. This implies that the columns of B are linearly independent.

Now, consider any $0 \leq j \leq d$ and $0 \leq k < n - d$. Then, we need to prove that the product of j^{th} row of M and k^{th} row of B is 0.

$$(MB)_{j,k} = \sum_{i=0}^{n} \binom{i}{j}\binom{d+1}{i-k}(-1)^{i-k} = 0$$

A proof of this identity will complete the proof. We prove this in the next lemma. ■

[4] Alternatively, one can put $\mathbf{b}^T \cdot \beta$ as the cost function, and require the linear program to be unbounded.

Lemma 3.3.

$$\sum_{i=0}^{n} \binom{i}{j}\binom{d+1}{i-k}(-1)^{i-k} = 0$$

Proof. In order to prove this identity, consider another identity

$$x^j(1+x)^{d+1+k-j} = \sum_{m\geq 0}\binom{d+1+k-j}{m}x^{m+j}$$

Differentiate the equation k times and substitute $x=-1$. Then, the limits on j and k imply that each term in the LHS will be some positive power of $1+x$. This implies that the LHS will be 0 after the substitution. That is,

$$0 = \sum_{m\geq 0}\binom{d+1+k-j}{m}(-1)^{m+j-k}((m+j)\cdot\ldots\cdot(m+j+1-k)) \tag{1}$$

$$= \sum_{i\geq j}\binom{d+1+k-j}{i-j}(-1)^{i-k}(i\cdot\ldots\cdot(i+1-k)) \tag{2}$$

$$= \sum_{i\geq j}\binom{d+1}{i-k}(-1)^{i-k}\frac{(d+1+k-j)\cdot\cdots\cdot(d+2)}{(i-j)\cdot\ldots\cdot(i+1-k)}(i\cdot\ldots\cdot(i+1-k)) \tag{3}$$

$$= ((d+1+k-j)\cdot\ldots\cdot(d+2))\sum_{i\geq j}\binom{d+1}{i-k}(-1)^{i-k}\frac{i\cdot\ldots(i+1-k)}{(i-j)\cdot\ldots\cdot(i+1-k)} \tag{4}$$

$$\tag{5}$$

Hence,

$$\sum_{i\geq j}\binom{d+1}{i-k}(-1)^{i-k}\frac{i!}{j!(i-j)!} \tag{6}$$

$$= \sum_{i\geq 0}\binom{i}{j}\binom{d+1}{i-k}(-1)^{i-k} \tag{7}$$

Equation 1 is obtained by setting LHS equal to the RHS. Then, equation 2 is obtained by substituting $i=m+j$. Next, equation 3 is obtained by using the binomial identity $\binom{n}{r}=\frac{n}{r}\binom{n-1}{r-1}$. Furthermore, equation 4 is obtained by simple rearrangement of terms. Following that, equation 6 is obtained by dividing both sides by $(d+1)\cdot\ldots\cdot(d+2)$ and $j!$. Finally, equation 7 is obtained by using the definition of binomial coefficients and simple rearrangement of terms.

This completes the proof. ■

We continue with the task of constructing the remaining non-negative vectors in the nullspace of A^T. We show how to construct a non-negative $\beta \geq 0$ in the nullspace of A^T using a vector $\gamma \in null(M)$. Note that γ may have both positive and negative entries.

Lemma 3.4. *For a vector $\gamma \in null(M)$, define two $n+1$ dimensional vectors. Define γ^+ as the $n+1$ dimensional vector, where $\gamma_i^+ = \begin{cases}\gamma_i & \gamma_i \geq 0\\ 0 & \gamma_i < 0\end{cases}$. Similarly, define γ^- as the $n+1$ dimensional vector, where $\gamma^- = \begin{cases}-\gamma_i & \gamma_i \leq 0\\ 0 & \gamma_i > 0\end{cases}$. Denote by β the $2n+2$ dimensional vector obtained by concatenating γ^+ and γ^-, i.e. $\beta = [\gamma^+ \mid \gamma-]$. Then, $A^T\beta = 0, \beta \geq 0$.*

Proof. Consider a vector γ in $null(M)$. Define β as per the statement. Then, $A^T\beta = M\gamma^+ - M\gamma^- = M\gamma = 0$. Also, $\beta \geq 0$ by construction. ■

For each column of B, apply this procedure to obtain a non-negative vector in the nullspace of A^T. It is easy to check that these newly obtained vectors are also linearly independent, and that there are exactly enough of them to form a basis for the nullspace of A^T along with $[e_i \mid e_i]$. Note that $\mathbf{b}^T \cdot [e_i \mid e_i] = 2\varepsilon \geq 0$, hence, to get a β with $\mathbf{b}^T \cdot \beta < 0$, we focus on the βs obtained from γs.

Now, for $\beta = [\gamma^+ \mid \gamma^-]$, a straightforward calculation shows that

$$\mathbf{b}^T \cdot \beta = \sum_{i=0}^{n} \left(z_i + \frac{f(i)}{2} \right) \gamma_i + \varepsilon \|\gamma\|_1 \tag{8}$$

Our objective then is to find, for each $(z_0, \ldots, z_n)$, a $\gamma \in null(M)$ such that $\mathbf{b}^T \cdot \beta < 0$. Consider the second term in the expression of $\mathbf{b}^T \cdot \beta$, i.e. $\varepsilon\|\gamma\|_1$. This term will always evaluate to a positive value. Hence, to make $\mathbf{b}^T \cdot \beta$ negative, the first term must dominate the second. Therefore, we would like to make the second term as small as possible while ensuring the first term does not become small.

In the next lemma, we prove that it is also sufficient to show that $\mathbf{b}^T \cdot \beta$ is "large enough". This will prove useful as we proceed.

Lemma 3.5. *The dual defined in Lemma 3.1 is feasible, if there exists a $\gamma \in null(M)$, such that for $\beta = [\gamma^+ \mid \gamma^-]$, the quantity $\mathbf{b}^T \cdot \beta > 2\|\gamma\|_1$.*

Proof. Define $\beta' = [\gamma^- \mid \gamma^+]$. Notice that β' is also in the nullspace of A^T. Now, note that $\mathbf{b}^T \cdot (\beta + \beta') = 2\varepsilon\|\gamma\|_1$. As $\mathbf{b}^T \cdot \beta > 2\varepsilon\|\gamma\|_1$, it implies $\mathbf{b}^T \cdot \beta' < 0$. ■

Finally, we show how to complete our proof by finding a γ with integer entries.

Lemma 3.6. *Consider an integral $\gamma \in \mathbb{Z}^{n+1} \cap null(M)$. If γ_w is* odd*, then for $f = \Delta_w$, the dual is feasible for $\varepsilon < \frac{1}{\|\gamma\|_1}$.*

Proof. First, notice that $\sum_{i=0}^{n} \left(z_i + \frac{f(i)}{2} \right) \gamma_i = z + \frac{1}{2}$, where z is an integer. Now, if $\varepsilon\|\gamma\|_1 < \frac{1}{2}$, then we argue as follows. If $z \geq 0$, $\mathbf{b}^T \cdot \beta \geq 1 > 2\varepsilon\|\gamma\|_1$. Else $z \leq -1$, for which $\mathbf{b}^T \cdot \beta < 0$. Hence, we get that the dual is feasible for $\varepsilon < \frac{1}{2\|\gamma\|_1}$. □

This completes the first part of the proof.

Part 2: Till this point, we follow the same steps for both item 1 and 2. Now, in part 2A, we prove the **All Delta Functions** claim (item 1), and in part 2B, we prove the **Error-degree Trade-off** claim (item 2).

Part 2A: Below, we prove that for each $f = \Delta_w$, and $\varepsilon < \frac{1}{2n}$, there exists a $\gamma \in null(M)$ that allows us to prove that the dual is feasible for any small $d = O\left(\sqrt{\frac{n}{\log(n)}}\right)$.

Lemma 3.7. *There exist a constant c, such that for any large n, and $d \leq c\sqrt{\frac{n}{\log(n)}}$, the following holds. For any $0 \leq w \leq n$, there exists a $\overline{\gamma} \in null(M)$ with the following properties.*

G.1 $\overline{\gamma}_i \in \{-1, 0, 1\}$ for any $0 \leq i \leq n$.

G.2 $\overline{\gamma}_w \neq 0$.

G.3 $\|\overline{\gamma}\|_1 \leq n$.

Proof. Before we begin the proof, we remark that the statement looks similar to Siegel's Lemma [HS13, Lemma D.4.1]. In fact, it will be evidence that Siegel's Lemma suffices to prove items G.1 and G.3. We provide a proof, very similar to the proof of Siegel's Lemma, not only for completeness, but to prove item G.2 as well.

Assume that $w \leq \frac{n}{2}$. The case where $w \geq \frac{n}{2}$ is analogous. Consider an $n+1$ dimensional vector $\mathbf{g} \in \{0,1\}^{n+1}$, where $\mathbf{g}_w, \ldots, \mathbf{g}_{w+\lceil \frac{n}{2} \rceil}$ is either 0 or 1, and all others are 0. There are exactly $2^{\lceil \frac{n}{2} \rceil + 1}$ such $\mathbf{g}$s. Now, consider $M\mathbf{g}$ for such a $\mathbf{g}$.

For a fixed $\mathbf{g}$, denote by the label of $\mathbf{g}$, $\ell(\mathbf{g})$, the vector $M\mathbf{g}$. This is a vector of $d+1$ dimensions. Note that, $|M\mathbf{g}|_j \leq \sum_{i=0}^{n} |M_{i,j}|$ upper bounds the i^{th} entry in the label. This itself is upper bounded by $\sum_{i=0}^{n} \binom{i}{j} \leq n\binom{n}{j} \leq n\binom{n}{d} \leq n \cdot n^d = n^{d+1}$. Hence, the total number of distinct labels is at most $n^{(d+1)^2}$.

If the number of possible $\mathbf{g}$s is larger than the number of possible labels, there have to be two $\mathbf{g}$s with the same label. Say the two $\mathbf{g}$s are $\mathbf{g}, \mathbf{g}'$, such that $M\mathbf{g} = M\mathbf{g}'$. Then, $\overline{\gamma} = \mathbf{g} - \mathbf{g}'$ satisfies $M\overline{\gamma} = 0$, hence $\overline{\gamma} \in null(M)$. For the above to hold, we need $n^{(d+1)^2} < 2^{\lceil \frac{n}{2} \rceil + 1}$. This is satisfied if we choose a $d = O\left(\sqrt{\frac{n}{\log(n)}}\right)$ which is small enough.

Hence, we can obtain a $\overline{\gamma} \in null(M)$ through this procedure. By construction, $\overline{\gamma}$ satisfies Property G.1. As the number of non-zeros entries in $\mathbf{g}$ is at most $\lceil \frac{n}{2} \rceil$, it satisfies Property G.3.

It remains to prove that $\overline{\gamma}$ satisfies Property G.2. If $\overline{\gamma}_w \neq 0$, then we are done. Otherwise, we use the fact that the columns of B are essentially the same vector shifted downwards. This allows us to "shift" $\overline{\gamma}$, to make it satisfy Property G.2, while still satisfying Property G.1 and G.3. We formalize this intuition in the next lemma, which will complete the proof.

We need the following lemma to complete the proof of the previous lemma.

Lemma 3.8. *Consider a vector $\gamma \in null(M)$ such that the minimum index i where $\gamma_i \neq 0$ is i_0. Then, for any $0 \leq w' < i_0$, the vector γ' given below is also in $null(M)$.*

$$\gamma'_i = \begin{cases} 0 & 0 \leq i < w' \\ \gamma_{i+i_0-w'} & w' \leq i \leq n - i_0 + w' \\ 0 & n - i_0 + w' < i \leq n \end{cases}$$

Proof. Recall that the columns of B are a basis for $null(M)$. Suppose $\mathbf{a} \in \mathbb{R}^{n-d}$ is the vector such that $B\mathbf{a} = \gamma$. As B is lower triangular, and $\gamma_0 = \ldots = \gamma_{i_0-1} = 0$, it follows that $a_0 = \ldots = a_{i_0-1} = 0$.

Now, consider the equality $(Ba)_{i_0} = \gamma_{i_0}$. The LHS is $\sum_{k=0}^{n-d-1} (-1)^{i_0-k} \binom{d+1}{i_0-k} \mathbf{a}_k$. For any $k < i_0$, $\mathbf{a}_k = 0$. On the other hand, for any $k > k_0$, the difference $i_0 - k < 0$ is negative, hence $\binom{d+1}{i_0-k} = 0$. Therefore, the only non-zero term in the summation is $(-1)^{i_0-i_0}\binom{d+1}{i_0-i_0}\mathbf{a}_{i_0} = \mathbf{a}_{i_0}$.

This yields $\mathbf{a}_{i_0} = \gamma_{i_0} \neq 0$. Now, define $\mathbf{a}'$ as

$$\mathbf{a'}_k = \begin{cases} 0 & k < w' \\ 0 & k > n - d - 1 - i_0 + w' \\ \mathbf{a}_{k+k_0-w'} & w' \leq k \leq n - d - 1 - i_0 + w' \end{cases}$$

Essentially, a is shifted upwards by $i_0 - w'$ places, and padded with 0s at the bottom to get a'. Then, for $\gamma' = B\mathbf{a}'$, we have $\gamma'_{w'} \neq 0$.

Finally, note that each column of B is a shift of the first column, i.e. $B_{i,k} = B_{i+1,k+1}$ for any $0 \leq i \leq n-1$ and $0 \leq k \leq n-d-2$. Hence, $B_{i,k}\mathbf{a'}_k = B_{i+k_0-w'}\mathbf{a}_{k+k_0-w'}$ for any $w' \leq i \leq n+w'-i_0$

and $w' \leq k \leq n-d-1+i_0+w'$. Therefore, $\gamma'_i = \gamma_{i+k_0-w'}$ for any $w' \leq i \leq n+w'-i_0$. All other entries of γ' are 0 by construction. Therefore, γ' is as desired.

This completes the proof of this statement. ■

This allows us to complete the proof of Lemma 3.7. Hence, putting everything together, we get that the dual is feasible any small enough $d = O\left(\sqrt{\frac{n}{\log(n)}}\right)$, all $f = \Delta_w$, and $\varepsilon < \frac{1}{2n}$. Therefore, the lower bound holds as claimed in item 1.

Part 2B: We begin by noticing that columns of B are integral vectors. Hence, any integer combination of columns of B will lead to an integral vector in $null(M)$. Therefore, it is natural to consider the *lattice* generated by the columns of B, denoted by $\mathcal{L}(B)$.

Our goal is again to find vectors with small ℓ^1 norm. This is a very well studied pursuit (see [Yas21] for a nice recent survey). The most general upper bound known for the length of the shortest vector in a lattice comes from the theorem of Minkowski, and it is this we use (see Corollary 2). Let $\overline{\gamma}$ be a lattice vector with the smallest ℓ^1 norm. Then, it must have an *odd* entry $\overline{\gamma}_w$. Unfortunately, Minkowski's theorem only guarantees the existence of a vector, but it does not allow us to determine the w for which γ_w is odd.

Instead, we work around this by turning the problem over on its head. We choose $f = \Delta_w$ for the w where $\overline{\gamma}_w$ is odd. Note that $\overline{\gamma}$ does not depend on $(z_0, \ldots, z_n)$, hence neither does our choice of f. This leaves us with the task of bounding $\|\overline{\gamma}\|_1$ from above. This is precisely what we do next.

Lemma 3.9. *Fix $c \in \mathbb{R}$, $c \geq 0$. For any large n, and small enough $d \leq O\left(\sqrt{n\log^c(n)}\right)$, $\mathcal{L}(B)$ contains a vector $\overline{\gamma}$ with $\|\overline{\gamma}\|_1 \leq 2^{\log(n)+\log^{c+1}(n)}$.*

Proof. We invoke Corollary 2 on $\mathcal{L}(B)$. It implies the existence of a vector $\overline{\gamma}$ with $\|\overline{\gamma}\|_1 \leq \sqrt{n+1}\sqrt{n-d}(det(B^TB))^{\frac{1}{2(n-d)}}$. This requires calculating $det(B^TB)$, which we do next.

Consider row k_1 of B^T and column k_2 of B for some $0 \leq k_1, k_2 < n-d$. Then, $(B^TB)_{k_1,k_2} = \sum_{i=0}^{n}(-1)^{i-k_1}\binom{d+1}{i-k_1}(-1)^{i-k_2}\binom{d+1}{i-k_2}$. By rearranging these terms, and replacing $\binom{d+1}{i-k_2}$ with $\binom{d+1}{d+1-i+k_2}$, we get $(B^TB)_{k_1,k_2} = \sum_{i=0}^{n}(-1)^{k_1+k_2}\binom{d+1}{i-k_1}\binom{d+1}{d+1-i+k_2}$. We use the identity $\sum_{i=0}^{d+1}\binom{d+1}{i}\binom{d+1}{d+1-i+k_1} = \binom{2(d+1)}{d+1-k_1}$ to get $(B^TB)_{k_1,k_2} = (-1)^{k_1+k_2}\binom{2(d+1)}{d+1+k_1-k_2}$. This proves that B^TB looks like the following.

$$B^TB = \begin{bmatrix} \binom{2(d+1)}{d+1} & \cdots & (-1)^{k_1}\binom{2(d+1)}{d+1-k_1} & \cdots \\ \vdots & \cdots & \vdots & \cdots \\ (-1)^{k_2}\binom{2(d+1)}{d+1+k_2} & \cdots & (-1)^{k_2+k_1}\binom{2(d+1)}{d+1+k_2-k_1} & \\ \vdots & \cdots & \vdots & \cdots \end{bmatrix}_{(n-d)\times(n-d)}$$

Now we calculate the determinant of B^TB. We simplify the task by multiplying row k_1 and column k_2 of B^TB with -1 for each odd value of k_1, k_2 in the range $0 \leq k_1, k_2 < n-d$. Note that this makes each entry of the transformed matrix non-negative while, as -1 gets multiplied an even number of times, it has the same determinant as B^TB. Therefore,

$$det(B^TB) = det\left(\begin{bmatrix} \binom{2(d+1)}{d+1} & \binom{2(d+1)}{d} & \binom{2(d+1)}{d-1} & \cdots & \cdots & \cdots \\ \binom{2(d+1)}{d+2} & \binom{2(d+1)}{d+1} & \binom{2(d+1)}{d} & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & & & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ & & & \binom{2(d+1)}{d+3} & \binom{2(d+1)}{d+2} & \binom{2(d+1)}{d+1} \end{bmatrix}\right)$$

Grinberg [Gri22, Gri] proved the following formula for this determinant:

$$det(B^TB) = \frac{H(d+1)H(d+1)H(n-d)H(n+d+2)}{H(n+1)H(n+1)H(2(d+1))}$$

where $H(m+1) = m! \cdot \ldots \cdot 2!1!0!$ for any $m \in \mathbb{Z}_{\geq 0}$. We substitute this into the above equation.

$$\begin{aligned}
det(B^TB) &= \frac{(d! \cdot \ldots \cdot 2!)^2((n-d-1)! \cdot \ldots \cdot 2!)((n+d+1)! \cdot \ldots \cdot 2!)}{(n! \cdot \ldots \cdot 2!)^2((2d+1)! \cdot \ldots \cdot 2!)} \\
&= \frac{(d! \cdot \ldots \cdot 2!)((n+d+1)! \cdot \ldots \cdot (n+1)!)}{(n! \cdot \ldots \cdot (n-d)!)((2d+1)! \cdot \ldots \cdot (d+1)!)} && (9) \\
&= \frac{d! \cdot \ldots \cdot 2!}{(d+2)^d(d+3)^{d-1} \cdot \ldots \cdot (2d+1)} \cdot \frac{(n+d+1)!}{n!(d+1)!} \cdot \ldots \cdot \frac{(n+1)!}{(n-d)!(d+1)!} && (10) \\
&\leq \binom{n+d+1}{d+1} \cdot \binom{n+d}{d+1} \cdot \ldots \cdot \binom{n+1}{d+1} && (11) \\
&\leq \binom{n+d+1}{d+1}^{(d+1)} && (12) \\
&\leq (n+d+1)^{(d+1)^2} && (13)
\end{aligned}$$

Equations 9 and 10 are obtained by rearranging terms and cancellation of common terms. Inequalities 11, 12 and 13 are obtained as follows:

- ignore the first fraction from the previous expression and convert the other fractions into binomial coefficients,
- $\binom{n+d+1}{d+1}$ is the largest binomial coefficient among the ones that appear in the RHS,
- $\binom{n}{d} \leq n^d$.

This proves that the shortest vector has length at most $\sqrt{n+1}\sqrt{n-d} \cdot (n+d+1)^{\frac{(d+1)^2}{2(n-d)}}$. For small enough $d = O\left(\sqrt{n \log^c(n)}\right)$, the shortest vector has length at most $n \cdot 2^{\log^{c+1}(n)} = 2^{\log(n)+\log^{c+1}(n)}$. ■

We give another proof for the previous statement, using an argument similar to the one given in the proof of Lemma 3.7.

Proof. Consider an $n+1$ dimensional $\mathbf{g} \in \{0, \ldots, \ell-1\}^{n+1}$. There are exactly ℓ^{n+1} such $\mathbf{g}$s. Now, consider $M\mathbf{g}$ for such a $\mathbf{g}$.

For a fixed $\mathbf{g}$, denote by the label of $\mathbf{g}$, $\ell(\mathbf{g})$, the vector $M\mathbf{g}$. This is a vector of $d+1$ dimensions. Note that, $|M\mathbf{g}|_j \leq \sum_{i=0}^{n} \ell |M_{i,j}|$ upper bounds the i^{th} entry in the label. This itself is upper bounded by $\ell \sum_{i=0}^{n} \binom{i}{j} \leq \ell n \binom{n}{j} \leq \ell n \binom{n}{d} \leq \ell n \cdot n^d = \ell n^{d+1}$. Hence, the total number of distinct labels is at most $\ell^{d+1} n^{(d+1)^2}$.

If the number of possible $\mathbf{g}$s is larger than the number of possible labels, there have to be two $\mathbf{g}$s with the same label. Say the two $\mathbf{g}$s are $\mathbf{g}, \mathbf{g}'$, such that $M\mathbf{g} = M\mathbf{g}'$. Then, $\overline{\gamma} = \mathbf{g} - \mathbf{g}'$ satisfies $M\overline{\gamma} = 0$, hence $\gamma \in null(M)$. For the above to hold, we need $\ell^{d+1} n^{(d+1)^2} < \ell^{n+1}$. For $\ell = 2^{\log^{c+1}(n)}$, this is satisfied if we choose a $d = O\left(\sqrt{n \log^c(n)}\right)$ which is small enough.

Hence, we can obtain a $\overline{\gamma} \in null(M)$ through this procedure. Note that, by construction, $|\gamma|_i \leq \ell - 1$ for each i. Therefore, we get $\|\gamma\|_1 = \sum_{i=0}^{n} |\gamma|_i \leq 2^{\log(n)+\log^{c+1}(n)}$. This completes the proof. ■

Using the bound on the ℓ^1 norm, we achieve the claimed lower bound for some $f = \Delta_w$ and $\varepsilon < \frac{1}{2^{1+\log(n)+\log^{c+1}(n)}}$. This completes the proof of both the claimed statements. □

The result above implies that symmetric torus polynomials are significantly less powerful than asymmetric torus polynomials.

Here, we conjecture that the error-degree trade-off actually holds for all Δ_w functions. The main hurdle we encounter in proving this is that, while Lemma 3.9 gives us an upper bound on the ℓ^1 norm of the shortest vector, we are not able to argue about which position in the shortest vector must contain an odd entry. If we can prove that the last entry in the shortest vector is one, then by using an argument similar to Lemma 3.8, we can prove the lower bound for all Δ_w functions. Hence, we conjecture that the last entry is indeed one.

Conjecture 5. *Consider the basis B for any n, d. There exists a vector $\overline{\gamma} \in \mathcal{L}(B)$ with the smallest ℓ^2 norm such that $\overline{\gamma}_n = 1$.*

3.2 Lower Bounds for the Majority Function

So far, the error-degree trade-off we have proved is for Δ_w. Next, we borrow a reduction by Bhrushundi et al. [BHLR19] to prove the lower bound for the majority function. They proved that a symmetric torus polynomial approximation for the majority function can be used to obtain a symmetric torus polynomial approximation for each Δ_w function. We give a proof for completeness.

Claim 3.1 ([BHLR19])**.** *Let $d : \mathbb{N} \to \mathbb{N}$ and $\varepsilon : \mathbb{N} \to \mathbb{R}$ be functions over natural numbers. Suppose $\varepsilon = O\left(\frac{1}{n}\right)$. Consider a symmetric torus polynomial of degree at most $d(n)$ that approximates MAJ_n within an error of $\varepsilon(n)$. Then, for any $0 \leq w \leq n$, there is a symmetric torus polynomial of degree $d(2n+1)$ that approximates Δ_w within an error of $\varepsilon(n)$.*

Proof. Fix n and w, $0 \leq w \leq n$. Define $\Delta_{\geq w} : \{0,1\}^n \to \{0,1\}$ as $\Delta_{\geq w}(x) = 1$ if and only if $|x| \geq w$. Following is a way to compute it using MAJ_n:

$$\Delta_{\geq w}(x_1, \ldots, x_n) = \mathsf{MAJ}_{2n+1}(x_1, \ldots, x_n, c_1, \ldots, c_{n+1})$$

Here, $c_1 = \ldots = c_{n-w+1} = 1$ and the rest are 0.

Let $P(x_1, \ldots, x_{2n+1})$ be the symmetric torus polynomial over $2n+1$ variables of degree at most $d(2n+1)$ that approximates MAJ_{2n+1} within an error of $\varepsilon(2n+1)$ ($d(2n+1)$, and similarly $\varepsilon(2n+1)$, are the value of functions d and ε evaluated at $2n+1$). Set the last $n+1$ variables to obtain $P_{\geq w}$ as

$$P_{\geq w} = P(x_1, \ldots, x_n, c_1, \ldots, c_{n+1})$$

Then $P_{\geq w}$ is a symmetric torus polynomial of degree at most $d(2n+1)$ that approximates $\Delta_{\geq w}$ within an error of $\varepsilon(2n+1)$. Similarly, obtain a symmetric torus polynomial $P_{\geq w+1}$ of degree at most $d(2n+1)$ that approximates $\Delta_{\geq w+1}$ within an error of $\varepsilon(2n+1)$. Note that

$$\Delta_w = \Delta_{\geq w} - \Delta_{\geq w+1}$$

Therefore, $P_w = P_{\geq w} - P_{\geq w+1}$ is a symmetric torus polynomial that approximates Δ_w within $2\varepsilon(2n+1)$ error. The degree of P_w is bounded by $d(2n+1)$ as desired.

Now use $\varepsilon = O\left(\frac{1}{n}\right)$ to get that $2\varepsilon(2n+1) \leq \varepsilon(n)$. Therefore, P_w is a torus polynomial that approximates Δ_w within an error of $\varepsilon(n)$. This completes the proof. □

Hence, we get the following as a corollary of the previous result and Theorem 1.3.

Corollary 3. *Fix $c \in \mathbb{R}$, $c \geq 0$. Consider $\varepsilon < \frac{1}{2^{1+\log(n)+\log^{c+1}(n)}}$. Then, any symmetric torus polynomial that approximates MAJ_n within an error of ε must have degree $\Omega\left(\sqrt{n \log^c(n)}\right)$.*

3.3 Towards Stronger Lower Bounds

Several questions arise from our approach. Here, we answer some of them, and pose others as open questions. We first prove that it is necessary to look beyond the shortest vector to obtain stronger lower bounds. Moreover, we conjecture that the upper bound on the shortest vector we obtain is indeed tight. If the conjecture holds, it becomes imperative to look beyond the shortest vector to improve the lower bounds we have proved. Then, we propose an approach and develop a theory that indeed looks beyond the shortest vector. In fact, our approach allows proving almost tight lower bounds against symmetric torus polynomials.

First, we prove that simply considering the shortest vector cannot lead to stronger lower bounds.

Claim 3.2. *Consider a pair (f, γ) such that $\langle f, \gamma \rangle$ is odd, and the entries of γ have GCD 1. Then, there exists a $Z = (z_0, \ldots, z_n) \in \mathbb{Z}^{n+1}$ such that $\left\langle Z + \frac{f}{2}, \gamma \right\rangle = \frac{1}{2}$.*

Proof. Say $\left\langle \frac{f}{2}, \gamma \right\rangle = z + \frac{1}{2}$. As the GCD of the entries of γ is 1, there exists a vector $Z = (z_0, \ldots, z_n) \in \mathbb{Z}^{n+1}$ such that $\langle Z, \gamma \rangle = -z$. Hence, $\left\langle Z + \frac{f}{2}, \gamma \right\rangle = \frac{1}{2}$. □

Hence, simply looking at the shortest vector cannot improve the lower bound beyond what we have proved. Therefore, we would like to understand whether there is an improvement to our lower bound using the shortest vector method by simply finding an even shorter vector. We do not know whether shorter vectors exist, which leads us to the following open problem.

Problem 6. *Prove a lower bound on the ℓ^1 norm of the shortest vector in $\mathcal{L}(B)$.*

The standard method for this is to perform Gram-Schmidt orthogonalization on B. Then, the smallest ℓ^2 norm in the orthogonalized basis is a lower bound on the smallest ℓ^2 norm of a vector in $\mathcal{L}(B)$. Using the fact that the ℓ^1 norm is at least as much as the ℓ^2 norm, it will also lead to a lower bound on the smallest ℓ^1 norm. We leave it as an open problem to perform the Gram-Schmidt orthogonalization.

Problem 7. *Describe the shortest vector obtained after applying the Gram-Schmidt orthogonalization process on B.*

Naturally, one may ask whether the bound we obtain using Minkowski's theorem is tight. Here, we provide some intuition that it is probably almost tight. Consider the columns of B when d is much smaller than n. Then, more often than not, a pair of columns is orthogonal to each other. Hence, we expect Minkowski's theorem to be *almost* tight. Therefore, we conjecture exactly that. Note that we state this conjecture for the ℓ^2 norm, not the ℓ^1 norm. This is because $\mathcal{L}(B)$ is not full rank, hence Minkowski's theorem is directly applicable only for the ℓ^2 norm. The conjecture is as follows.

Conjecture 8. *Consider the case when $d = o(n)$. Then, the smallest ℓ^2 norm of a vector $\gamma \in \mathcal{L}(B)$ is $\|\gamma\|_2 = \Theta(\sqrt{n-d}(det(B^T B))^{\frac{1}{2(n-d)}})$.*

3.4 Beyond the Shortest Vector

In this subsection, we propose a method that looks beyond the shortest vector, with a view towards stronger lower bounds against symmetric torus polynomials. Recall Equation 8 from the proof of

Theorem 1.3 for a given f, n, d, ε. The dual is feasible if and only if, for each $Z = (z_0, \ldots, z_n)$, there exists a $\gamma \in null(M)$ such that

$$\left\langle Z + \frac{f}{2}, \gamma \right\rangle + \varepsilon \|\gamma\|_1 < 0$$

Define

$$\varepsilon_0 = \min_{Z \in \mathbb{Z}^{n+1}} \max_{\gamma \in null(M)} \frac{\left\langle Z + \frac{f}{2}, \gamma \right\rangle}{\|\gamma\|_1}$$

Note that ε_0 is a function of f, n, d. Also, the dual is feasible for each $Z \in \mathbb{Z}^{n+1}$ if and only if $\varepsilon < \varepsilon_0$. This implies the following:

- If $\varepsilon < \varepsilon_0$, then there exists no degree d symmetric torus polynomial that approximates f within an error of ε.
- If $\varepsilon \geq \varepsilon_0$, then there exists a degree d symmetric torus polynomial that approximates f within an error of ε.

This gives us an exact characterization. Hence, the parameter ε_0 is precisely what we would like to estimate. For each $Z \in \mathbb{Z}^{n+1}$, consider the *projection* of $Z + \frac{f}{2}$ onto $null(M)$. The standard method of determining the projection of a vector onto a subspace V is by determining the *projection matrix* for V. We denote the projection matrix for projecting onto $null(M)$ by P. The projection of $Z + \frac{f}{2}$ onto $null(M)$ is then $P\left(Z + \frac{f}{2}\right)$. Then, by virtue of being the projection, $\frac{\left\langle Z + \frac{f}{2}, P\left(Z + \frac{f}{2}\right) \right\rangle}{\left\| P\left(Z + \frac{f}{2}\right) \right\|_2} \geq \frac{\left\langle Z + \frac{f}{2}, \gamma \right\rangle}{\|\gamma\|_2}$ for any $\gamma \in null(M)$. Using the inequalities $\|\gamma\|_2 \leq \|\gamma\|_1 \leq \sqrt{n+1}\|\gamma\|_2$ for any $\gamma \in \mathbb{R}^{n+1}$, we get the following observation.

Observation 1.

$$\min_{Z \in \mathbb{Z}^{n+1}} \frac{\left\langle Z + \frac{f}{2}, P\left(Z + \frac{f}{2}\right) \right\rangle}{\left\| P\left(Z + \frac{f}{2}\right) \right\|_2} \geq \varepsilon_0 \geq \min_{Z \in \mathbb{Z}^{n+1}} \frac{\left\langle Z + \frac{f}{2}, P\left(Z + \frac{f}{2}\right) \right\rangle}{\sqrt{n+1}\left\| P\left(Z + \frac{f}{2}\right) \right\|_2}$$

Therefore, calculating the projection gives us almost tight bounds on the optimum error ε_0.

Now, $null(M)$ is a subspace of $\mathbb{R}^{n+1}$ with B as the basis. Hence, the projection of a vector onto $null(M)$ can be calculated using the projection matrix $P = B(B^T B)^{-1} B^T$. Note that the projection matrix is idempotent, i.e. $P^2 = P$, and symmetric, i.e. $P^T = P$. Hence, we get $\left\langle Z + \frac{f}{2}, P\left(Z + \frac{f}{2}\right) \right\rangle = \left\| P\left(Z + \frac{f}{2}\right) \right\|_2^2$. Therefore, $\frac{\left\langle Z + \frac{f}{2}, P\left(Z + \frac{f}{2}\right) \right\rangle}{\left\| P\left(Z + \frac{f}{2}\right) \right\|_2} = \left\| P\left(Z + \frac{f}{2}\right) \right\|_2$. This shows that estimating the smallest possible projection over all possible Zs is useful for not only obtaining degree lower bounds, but also upper bounds. In fact, in Section 5, we use this machinery to prove an upper bound.

Here, we derive an explicit formula for the projection matrix.

Theorem 3.10. *For a given n, d, the projection matrix $P \in \mathcal{M}_{(n+1)\times(n+1)}(\mathbb{R})$ has the following entries.*

$$(-1)^{i+j} P_{i,j}$$
$$= \sum_{k_1=0}^{i} \sum_{k_2=0}^{j} (-1)^{k_1+k_2} \binom{d+1}{i-k_1} \binom{d+1}{j-k_2} \binom{k_1+d+1}{d+1} \binom{k_2+d+1}{d+1} \sum_{k=\max(k_1,k_2)}^{n-d-1} \frac{\binom{k+d-k_1}{d} \binom{k+d-k_2}{d}}{\binom{k+d+1}{d+1} \binom{k+2(d+1)}{d+1}}$$

Proof. The expression for $P = B(B^TB)^{-1}B^T$ requires calculating $(B^TB)^{-1}$ first. We have already calculated (B^TB) in the proof of Claim 3.9. For $0 \leq k_1, k_2 < n-d$, $(B^TB)_{k_1,k_2} = (-1)^{k_1+k_2}\binom{2(d+1)}{d+1+k_1-k_2}$. We derive the inverse of this matrix using a formula given by Lemma 5 of Hoskins and Ponzo [HP72, Lemma 5], but omit the tedious calculations. Hence, we get

$$(B^TB)^{-1}_{k_1,k_2} = \binom{k_1+d+1}{d+1}\binom{k_2+d+1}{d+1}\sum_{k=\max(k_1,k_2)}^{n-d-1}\frac{\binom{k+d-k_1}{d}\binom{k+d-k_2}{d}}{\binom{k+d+1}{d+1}\binom{k+2(d+1)}{d+1}}$$

After this, a straightforward multiplication with B on the left, and B^T on the right, gives the formula for $P_{i,j}$ as claimed. Again, we omit the calculations. □

4 Tight Bounds for Torus Polynomials for Very Small Error

Usually, polynomial approximations are studied in the context of inverse polynomial error regime. We believe it is fruitful to study them in other regimes as well, where the error is much smaller. In this section, we consider the case when the error is miniscule – less than $\frac{1}{2^{n+1}}$. The interesting result is that for an error this small, the degree has to be n. This holds even for *asymmetric* torus polynomials, for the AND and MAJ_n functions, making it the first lower bound known in the asymmetric world. On the other hand, degree $n-1$ suffices for any larger error.

Lower bounds for the asymmetric case do not automatically follow from the symmetric case, as per Corollary 1, as opposed to real polynomials. In this section, we prove a surprising result that lower bounds for the symmetric case imply lower bounds for the asymmetric case when the degree is $n-1$. The proof though is not as simple as real polynomials, it requires additional tricks to make it work.

We start with the proof of upper and lower bound for symmetric torus polynomials, which we then lift to bounds for asymmetric torus polynomials.

Theorem 4.1. *Depending on the value of ε, the following cases hold.*

- *If $\varepsilon < \frac{1}{2^{n+1}}$, then the following holds for $f = \mathsf{MAJ}_n$ as well as $f = \mathsf{AND}$, and infinitely many n. There does not exist a symmetric torus polynomial of degree at most $n-1$ approximating f within an error of ε.*
- *If $\varepsilon \geq \frac{1}{2^{n+1}}$, then the following holds for any symmetric Boolean function $f : \{0,1\}^n \to \{0,1\}$. There exists a symmetric torus polynomial P of degree at most $n-1$ approximating f within an error of ε.*

Proof. We prove the lower bound first. Consider a symmetric torus polynomial of degree $n-1$ approximating $f = \mathsf{MAJ}_n$. The proof proceeds along the lines of the proof of Theorem 1.3 till Lemma 3.2. In this case, $null(M)$ has dimension one, spanned by the vector $\overline{\gamma} = [\binom{n}{0}, \ldots, (-1)^n\binom{n}{n}]$.

For $n = 2^t$ for some natural number t, $\binom{n}{i}$ is even for all $1 \leq i \leq n-1$. Hence, $\langle f, \gamma\rangle = \sum_{i>\frac{n}{2}}\binom{n}{i}$ is odd. Therefore, the lower bound holds for $\varepsilon < \frac{1}{2\|\overline{\gamma}\|_1} = \frac{1}{2^{n+1}}$

A similar argument proves both the bounds for AND as well.

To prove the upper bound, take f as any symmetric function. Now, consider the following expressions from the dual.

$$A^T\beta = 0$$
$$\beta \geq 0$$

We prove that it is enough to look at β that arise out of $null(M)$, by converting $\gamma \in null(M)$ to $\beta = [\gamma^+ \mid \gamma^-]$.

Lemma 4.2. *If $\mathbf{b}^T \cdot [\gamma^+ \mid \gamma^-] \geq 0$ for all $\gamma \in null(M)$, then $\mathbf{b}^T \cdot \beta \geq 0$ for any β satisfying $A^T\beta = 0, \beta \geq 0$.*

Proof. Consider any β that satisfies $A^T\beta = 0, \beta \geq 0$. Define the vector β' such that

$$\beta'_i = \begin{cases} \min(\beta_i, \beta_{i+n+1}) & 0 \leq i \leq n \\ \min(\beta_{i-n-1}, \beta_i) & n+1 \leq i \leq 2n+1 \end{cases}$$

Then, $\beta' = \sum_{i=0}^{n} \min(\beta_i, \beta_{i+n+1})[e_i \mid e_i]$. Note that each $[e_i \mid e_i]$ is in the nullspace of A^T. Hence, $\beta'' = \beta - \beta'$ is in the nullspace of A^T as well. Also, $\mathbf{b}^T \cdot \beta'' = \mathbf{b}^T \cdot \beta - \mathbf{b}^T \cdot \beta' = \mathbf{b}^T \cdot \beta - \varepsilon \left(\sum_{i=0}^{n} \min(\beta_i, \beta_{i+n+1})\right) \leq \mathbf{b}^T \cdot \beta$. Therefore, $\mathbf{b}^T \cdot \beta'' \geq 0$ implies $\mathbf{b}^T \cdot \beta \geq 0$.

Now, we construct a vector $\gamma \in null(M)$ using β'' as follows:

$$\gamma_i = \begin{cases} \beta''_i & \beta''_i > 0 \\ \beta''_{i+n+1} & \beta''_{i+n+1} > 0 \\ 0 & \beta''_i = \beta''_{i+n+1} = 0 \end{cases}$$

The construction for β'' ensures that for any $0 \leq i \leq n$, either $\beta''_i = 0$ or $\beta''_{i+n+1} = 0$. Hence, the construction for γ is well-defined. Therefore, $\beta'' = [\gamma^+ \mid \gamma^-]$.

Now, it is easy to check that $\gamma \in null(M)$. This completes the proof. ■

Denote by $\overline{\beta} = [\overline{\gamma}^+ \mid \overline{\gamma}^-], \overline{\beta}' = [\overline{\gamma}^- \mid \overline{\gamma}^+]$. If $\varepsilon\|\overline{\gamma}\|_1 = \frac{1}{2}$, then, we claim that there exists $(z_0, \ldots, z_n)$ such that $\mathbf{b}^T \cdot \overline{\beta} \geq 0, \mathbf{b}^T \cdot \overline{\beta}' \geq 0$. This will prove the upper bound.

There are two cases to consider.

- Let $\sum_{i=0}^{n} \frac{f(i)}{2}\overline{\gamma}_i = z$ for some integer z. Choose $z_0 = -z, z_1 = \ldots = z_n = 0$. Then, $\sum_{i=0}^{n} \left(z_i + \frac{f(i)}{2}\right)\overline{\gamma}_i = 0$.

  Hence, $\mathbf{b}^T \cdot [\overline{\gamma}^+ \mid \overline{\gamma}^-] = \mathbf{b}^T \cdot [\overline{\gamma}^- \mid \overline{\gamma}^+] = \varepsilon\|\overline{\gamma}\|_1 = \frac{1}{2}$.

- Let $\sum_{i=0}^{n} \frac{f}{2}\overline{\gamma} = z + \frac{1}{2}$ for some integer z. Choose $z_0 = -z, z_1 = \ldots = z_n = 0$. Then, $\sum_{i=0}^{n} \left(z_i + \frac{f(i)}{2}\right)\overline{\gamma}_i = \frac{1}{2}$.

  Hence, $\mathbf{b}^T \cdot [\overline{\gamma}^+ \mid \overline{\gamma}^-] = \frac{1}{2} + \varepsilon\|\overline{\gamma}\|_1 = 1$, $\mathbf{b}^T \cdot [\overline{\gamma}^- \mid \overline{\gamma}^+] = -\frac{1}{2} + \varepsilon\|\overline{\gamma}\|_1 = 0$.

Hence, $\mathbf{b}^T \cdot \beta \geq 0$ for any $\beta \geq 0$ in the nullspace of A^T. Therefore, the dual is infeasible. This proves that the primal is feasible. Noting that $\|\overline{\gamma}\|_1 = 2^n$ finishes the proof. □

The proof of the upper bound above implies the existence of symmetric torus polynomials with low error. We leave it as an open problem to construct these polynomials.

Problem 9. *For $\varepsilon = \frac{1}{2^{n+1}}$ and any symmetric Boolean function $f : \{0,1\}^n \to \{0,1\}$, construct a symmetric torus polynomial of degree $n-1$ that approximates f within an error of ε.*

Now, we prove a conversion from torus polynomials of degree $n-1$ to symmetric torus polynomials of degree $n-1$, given that the torus polynomial approximates a symmetric function. This conversion preserves the error bound as well. First, we describe the intuition behind this proof.

The hurdle with applying symmetrization for torus polynomials is the average of integral parts leading to a fractional value. This hurdle does not appear if all the integral parts over a level are the same. Now, note that adding integers to the coefficients of a torus polynomial does not change its behavior vis-à-vis approximating a Boolean function. Moreover, a change in the coefficient of a certain degree monomial does not affect the value of the polynomial on points of Hamming weight smaller than the degree. Hence, we can inductively change the coefficients for each layer to ensure that the integral parts for that layer become the same.

This does not work for a general degree d, as the procedure may end up changing the coefficients of a monomial with degree higher than d. Regardless, if we look at Boolean points with Hamming weight n, there is only one such point. Hence, over all permutations, it adds up this single value, and therefore the average remains an integer. Therefore, the procedure need not do anything for degree n, it can stop after considering monomials of degree $n-1$. This makes the procedure work for degree $n-1$. Below, we make this argument formal.

Claim 4.1. *Let $P \in \mathbb{R}[X_1, \ldots, X_n]$ be a torus polynomial of degree at most $n-1$ that approximates a symmetric function $f : \{0,1\}^n \to \{0,1\}$ within an error of ε. Then, there exists a symmetric torus polynomial $P_s \in \mathbb{R}[X_1, \ldots, X_n]$ of degree at most $n-1$ that approximates f within an error of ε.*

Proof. Let P have the form $P(X) = \sum_{S \subseteq [n], |S| \leq d} \alpha_S X^S$. The symmetric polynomial P_s will be of the form $P_s(X) = \sum_{S \subseteq [n], |S| \leq n-1} \alpha_{|S|} X^S$. We determine the α_js using an inductive procedure.

Consider Boolean points of Hamming weight i, starting with $i = 1$. Fix a Boolean point x of Hamming weight i and consider all other Boolean points y of Hamming weight i. Denote by $S_x = \{i \mid x_i = 1\}$ the set of indices where x has 1. Similarly, denote by S_y the corresponding set for y. Then, $P(x) = \sum_{S \subseteq S_x} \alpha_S$ and $P(y) = \sum_{S \subseteq S_y} \alpha_S$. Let their integral and fractional parts be $Z(x), \delta(x)$ and $Z(y), \delta(y)$. Set $\alpha'_{S_y} = Z(x) - Z(y) + \alpha_{S_y}$. As this doesn't change the fractional part when evaluated at y, the new polynomial correctly approximates $f(y)$. Do this for all Boolean points $y \neq x$ of Hamming weight i. Then start the same procedure for Hamming weight $i+1$.

Note that this procedure needs to be done only up to $i \leq n-1$ as there is only one point of Hamming weight n. Hence, if $\alpha_{[n]} = 0$, then $\alpha'_{[n]} = 0$ as well. This proves that $deg(P') \leq n-1$.

Now, take $P_s = \sum_{\pi \in \mathfrak{S}_n} \frac{P' \circ \pi}{n!}$ as the average of all the polynomials obtained by permuting the variables of P'. As the integral part of all points with the same Hamming weight is the same, it adds up to a multiple of $n!$ when summed over all permutations $\pi \in \mathfrak{S}_n$ over n variables. Hence, the fractional part of P_s only gets contribution from the fractional parts of P', which are all bounded by ε in their absolute values. Therefore, the fractional part of P_s is also bounded by ε in its absolute value.

This proves that P_s is a symmetric torus polynomial approximating f within an error of ε. The degree bound has already been proved. □

As a corollary, we obtain the following result.

Corollary 4. *Depending on the value of ε, the following cases hold.*

- *If $\varepsilon < \frac{1}{2^{n+1}}$, then the following holds for $f = \mathsf{MAJ}_n$ as well as $f = \mathsf{AND}$, and infinitely many n. There does not exist a torus polynomial of degree at most $n-1$ approximating f within an error of ε.*
- *If $\varepsilon \geq \frac{1}{2^{n+1}}$, then the following holds for any symmetric Boolean function $f : \{0,1\}^n \to \{0,1\}$. There exists a symmetric torus polynomial P of degree at most $n-1$ approximating f within an error of ε.*

5 Power of Symmetric Torus Polynomials

While the paper has largely focused on proving lower bounds, in this section, we use the machinery we have developed, along with some additional ideas, to prove an upper bound. To us, it is surprising that our machinery facilitates proving of upper bounds.

In Theorem 4.1, we proved the existence of a degree $n-1$ symmetric torus polynomial that approximates MAJ_n within an error of $\frac{1}{2^{n+1}}$. We prove that in certain cases of n, we can bring the degree down to $\lfloor \frac{n}{2} \rfloor$ without significantly increasing the error of approximation. An interesting aspect of the proof is that we choose n, d based on divisibility criteria, so that the calculations simplify substantially.

Theorem 1.4. *Consider $d = 2^t$ for some natural number $t \in \mathbb{N}$, and $n = 2d+1$. Then, there exists a symmetric torus polynomial of degree at most d that approximates MAJ_n within an error of $\frac{1}{2^{\Omega(n)}}$.*

Proof. Take $d = 2^t$ for some natural number t, $n = 2d+1$, and $f = \mathsf{MAJ}_n$. Our goal is to find a tuple $Z = (z_0, \ldots, z_n) \in \mathbb{Z}^{n+1}$ such that $\left\| P\left(Z + \frac{f}{2}\right) \right\|_2 \leq \frac{1}{2^{\Omega(n)}}$. First, we prove that there exists a tuple $Z = (z_0, \ldots, z_n)$ such that $B^T\left(Z + \frac{f}{2}\right) = \left[-\frac{1}{2}, 0, \ldots, 0, \frac{1}{2}\right]$. Then, we prove that $\left\| P\left(Z + \frac{f}{2}\right) \right\|_2 \leq \frac{1}{2^{\Omega(n)}}$ for this Z.

By a result of Kummer [Kum52], or the use of Lucas' theorem [Luc78] (see also [Dic52, Fin47]), $\binom{d+1}{i}$ is odd if and only if $i \in \{0, 1, d, d+1\}$.

We start by considering the first row of B^T. Its entries are $B^T_{0,i} = (-1)^i \binom{d+1}{i}$ for $0 \leq i \leq n$. We choose $z_0 = \ldots = z_{d+1} = 0$. For $f = \mathsf{MAJ}_n$, $f(0) = \ldots = f(d) = 0$ and $f(d+1) = 1$. Hence, the first entry of $B^T\left(Z + \frac{f}{2}\right)$ is $\frac{(-1)^{d+1}}{2} = -\frac{1}{2}$.

Now, we apply an iterative procedure to obtain $z_{d+2}, \ldots, z_{n-1}$. Assume that we have found values for $z_{d+1}, \ldots, z_{d+1+k}$ for some $0 \leq k \leq n-d-3$. Consider the $(k+1)^{\text{th}}$ row of B^T, denoted by B^T_{k+1}. When this row is multiplied by f, among all the entries that coincide with 1, there are exactly two odd numbers, namely $\binom{d+1}{d}$ and $\binom{d+1}{d+1}$. Hence, when $\frac{f}{2}$ is multiplied by B^T_{k+1}, it evaluates to an integer. Also, the $(k+d+2)^{\text{th}}$ entry of B^T_{k+1} is -1. Moreover, all entries of B^T_{k+1} for $k' > k+d+2$ are 0. Hence, the product of Z with B^T_{k+1} uses exactly one indeterminate, namely z_{k+d+2}, with -1 as the coefficient. Therefore, we can choose z_{k+d+2} such that $\left\langle Z + \frac{f}{2}, B^T_{k+1} \right\rangle = 0$.

Finally, consider the $n-d-1^{\text{th}}$ row of B^T, denoted by B^T_{n-d-1}. When this row is multiplied by f, among all the entries that coincide with 1, there are exactly three odd numbers, namely $\binom{d+1}{1}$, $\binom{d+1}{d}$ and $\binom{d+1}{d+1}$. Hence, when $\frac{f}{2}$ is multiplied by B^T_{n-d-1}, it evaluates to half plus an integer. Also, the n^{th} entry of B^T_{n-d-1} is -1. Hence, the product of Z with B^T_{k+1} uses exactly one indeterminate, namely z_n, with -1 as the coefficient. Therefore, we can choose z_n such that $\left\langle Z + \frac{f}{2}, B^T_{k+1} \right\rangle = \frac{1}{2}$.

Now, using the fact that P is symmetric, and idempotent, we get:

$$\begin{aligned}\left\|P\left(Z+\frac{f}{2}\right)\right\|_2^2 &= \left\langle P\left(Z+\frac{f}{2}\right), P\left(Z+\frac{f}{2}\right)\right\rangle \\ &= \left\langle \left(Z+\frac{f}{2}\right), P\left(Z+\frac{f}{2}\right)\right\rangle \\ &= \left\langle \left(Z+\frac{f}{2}\right), B(B^TB)^{-1}B^T\left(Z+\frac{f}{2}\right)\right\rangle \\ &= \left\langle B^T\left(Z+\frac{f}{2}\right), (B^TB)^{-1}B^T\left(Z+\frac{f}{2}\right)\right\rangle\end{aligned}$$

By construction, we have $B^T\left(Z+\frac{f}{2}\right) = \left[-\frac{1}{2}, 0, \ldots, 0, \frac{1}{2}\right]$. Hence, the last quantity can be written as

$$\frac{\left(\frac{(B^TB)^{-1}_{n-d-1,n-d-1} - (B^TB)^{-1}_{n-d-1,0}}{2}\right)}{2} - \frac{\left(\frac{(B^TB)^{-1}_{0,n-d-1} - (B^TB)^{-1}_{0,0}}{2}\right)}{2} \tag{14}$$

Now, we simplify this expression by determining some special properties of $(B^TB)^{-1}$. First, we recall the entries of B^TB:

$$B^TB = \begin{bmatrix} \binom{2(d+1)}{d+1} & \cdots & (-1)^{k_1}\binom{2(d+1)}{d+1-k_1} & \cdots \\ \vdots & \cdots & \vdots & \cdots \\ (-1)^{k_2}\binom{2(d+1)}{d+1+k_2} & \cdots & (-1)^{k_2+k_1}\binom{2(d+1)}{d+1+k_2-k_1} & \\ \vdots & \cdots & \vdots & \cdots \end{bmatrix}_{(n-d)\times(n-d)}$$

Note that B^TB is symmetric with respect to both diagonals, hence $(B^TB)^{-1}$ is symmetric with respect to both diagonals as well. Hence, $(B^TB)^{-1}_{0,0} = (B^TB)^{-1}_{n-d-1,n-d-1}$, and $(B^TB)^{-1}_{0,n-d-1} = (B^TB)^{-1}_{n-d-1,0}$. Therefore, we can simplify equation 14 to

$$\frac{(B^TB)^{-1}_{n-d-1,n-d-1} - (B^TB)^{-1}_{n-d-1,0}}{2}$$

Recall the expression for $(B^TB)^{-1}$, as calculated in the proof of Theorem 3.10.

$$(B^TB)^{-1}_{k_1,k_2} = \binom{k_1+d+1}{d+1}\binom{k_2+d+1}{d+1}\sum_{k=\max(k_1,k_2)}^{n-d-1}\frac{\binom{k+d-k_1}{d}\binom{k+d-k_2}{d}}{\binom{k+d+1}{d+1}\binom{k+2(d+1)}{d+1}}$$

Hence, we get

$$\begin{aligned}(B^TB)^{-1}_{n-d-1,n-d-1} &= \frac{\binom{n}{d+1}\binom{n}{d+1}}{\binom{n}{d+1}\binom{n+d+1}{d+1}} \\ (B^TB)^{-1}_{n-d-1,0} &= \frac{\binom{n}{d+1}\binom{n-1}{d}}{\binom{n}{d+1}\binom{n+d+1}{d+1}}\end{aligned}$$

Therefore, we get

$$(B^TB)^{-1}_{n-d-1,n-d-1} - (B^TB)^{-1}_{n-d-1,0} = \frac{\binom{n-1}{d+1}}{\binom{n+d+1}{d+1}}$$

For $n = 2d+1$, it is easy to see that $\frac{\binom{n-1}{d+1}}{\binom{n+d+1}{d+1}} = \frac{1}{2^{\Omega(n)}}$. Hence, we get that the error of approximation is at most $\frac{1}{2^{\Omega(n)}}$. This completes the proof. □

The tuple $(z_0, \ldots, z_n)$ we have constructed leads to low error, but does it lead to the smallest error? We do not have an answer to this, hence we leave this as another open problem.

Problem 10. *Consider $d = 2^t$ for some natural number $t \in \mathbb{N}$, and $n = 2d + 1$. Is there another $(z_0, \ldots, z_n)$ that leads to lower error as compared to the tuple constructed in the proof of Theorem 1.4?*

Moreover, we have proved the upper bound for a special case of n, d, but we are unable to extend this for more pairs of n, d. Hence, we leave this as an open problem.

Problem 11. *Are there other n, d pairs for which we can find symmetric torus polynomials of degree at most d that approximate MAJ_n within $\frac{1}{2^{\Omega(n)}}$ error?*

Finally, it will be interesting to understand how much we can decrease the degree and continue to find symmetric torus polynomials that approximate MAJ_n within an error of $\frac{1}{2^{\Omega(n)}}$. Note that we can apply the proof of Corollary 3 to get that the degree must be $\Omega(n)$. Hence, essentially we are asking for $d = cn$, where $c < \frac{1}{2}$.

Problem 12. *What is the smallest d, as a function of n, such that there exists a degree d symmetric torus polynomial that approximates MAJ_n within an error of $\frac{1}{2^{\Omega(n)}}$.*

6 Extending the Method for Asymmetric Torus Polynomials

The holy grail we are working towards is proving that $\mathsf{MAJ} \notin \mathsf{ACC}$, a problem at the frontier of our knowledge about Boolean functions. To do this, we need to extend the techniques of the previous sections to deal with asymmetric torus polynomials. While the overall methodology is similar, this project turns out to be much harder, as expected. The details and the technical challenges are very different, and require different approaches. Hence, we develop a framework with the aim of tackling exactly these challenges.

The objective here is proving Conjecture 3 [BHLR19, Conjecture 5]. They conjecture that there does not exist a torus polynomial of degree $o\left(\sqrt{\frac{n}{\log(n)}}\right)$ that approximates MAJ_n within an error of $\frac{1}{20n}$. Similar to the symmetric case, we write down an exponentially larger linear program, and use Farkas' lemma. The reader will notice that the details differ. Like in the symmetric case, we go through all the steps, including building up the machinery to construct the projection matrix. Using this, we reduce Conjecture 2 to proving a lower bound on the ℓ^2 norm of a family of vectors, explicitly defined. The linear program, and the dual, in this section are similar to the ones constructed in the literature [BT22].

6.1 The Dual in the Asymmetric Case

We begin with some notation. First, fix some ordering on the subsets of $[n] = \{1, \ldots, n\}$. We choose the reverse lexicographic ordering. For a set S_i, define $x^{(i)}$ as the Boolean point with:

$$x^{(i)}_{i'} = \begin{cases} 1 & i' \in S_i \\ 0 & i' \notin S_i \end{cases}$$

The proof of the following theorem follows along the broad lines of Theorem 1.3, however the details, such as the matrices A, M, B, are different.

Theorem 6.1. *Consider a Boolean function $f : \{0,1\}^n \to \{0,1\}$, an integer $1 \leq d < n$, and a real number $\varepsilon \in [0, \frac{1}{4})$. Define $B \in \mathcal{M}_{2^n \times \sum_{w=d+1}^{n} \binom{n}{w}}(\mathbb{R})$ as the following matrix. Its rows are labelled with all subsets S_1 of $\{1, \ldots, n\}$. Its columns are labelled with all subsets S_2 of $\{1, \ldots, n\}$ with size at least $d+1$. The entries of B are:*

$$B_{S_1,S_2} = \begin{cases} (-1)^{|S_1|+|S_2|} & S_1 \subseteq S_2 \\ 0 & S_1 \not\subseteq S_2 \end{cases}$$

Then, the following statements are equivalent:

1. *Any torus polynomial approximating f within an error of ε must have degree more than d.*
2. *For any function $Z : \{0,1\}^n \to \mathbb{Z}$, there exists a γ in the span of columns of B such that*

$$\left(\sum_{i=1}^{n} \left(Z(x^{(i)}) + \frac{f(x^{(i)})}{2} \right) \gamma_i \right) + \varepsilon \|\gamma\|_1 < 0$$

Proof. Consider a torus polynomial P of degree d that approximates a Boolean function f within an error of ε. Let the coefficient of the monomial $\prod_{j \in S_i} X_j$ corresponding to the set S_i be α_i. For the Boolean vector $x^{(i)}$, the polynomial evaluates to $P(x^{(i)}) = \sum_{S_j \subseteq S_i} \alpha_j$.

Definition 2 implies that there exists functions $Z : \{0,1\}^n \to \mathbb{Z}$ and $\delta : \{0,1\}^n \to [-\varepsilon, \varepsilon]$ such that $P(x) = \frac{f(x)}{2} + Z(x) + \delta(x)$.

As P is a torus polynomial that approximates f within an error of ε, the following inequalities must hold for all $x \in \{0,1\}^n$:

$$Z(x) + \frac{f(x)}{2} - \varepsilon \leq P(x) \leq Z(x) + \frac{f(x)}{2} + \varepsilon$$

We collect these as a system of linear inequalities as follows.

$$A\alpha \leq \mathbf{b}$$

Here, A is a matrix of dimension $2 \cdot 2^n \times \sum_{w=0}^{d} \binom{n}{w}$ with $A_{i,j} = -A_{i+2^n,j} = \begin{cases} 1 & S_j \subseteq S_i \\ 0 & \text{otherwise.} \end{cases}$ Also, $\mathbf{b}$ is a vector of size $2 \cdot 2^n$ with $b_{i,j} = Z(x^{(i)}) + \frac{f(x^{(i)})}{2} + \varepsilon$ and $b_{i+2^n,j} = -Z(x^{(i)}) - \frac{f(x^{(i)})}{2} + \varepsilon$. Here, $1 \leq i \leq 2^n$ and $1 \leq j \leq \sum_{w=0}^{d} \binom{n}{w}$.

We note that each function Z defines a linear program. Hence, proving that the linear program is infeasible for each Z proves the desired lower bound.

By Farkas' lemma [Far02], this system is infeasible if and only if the following system is feasible:

$$\begin{aligned} A^T \beta &= 0 \\ \mathbf{b}^T \cdot \beta &< 0 \\ \beta &\geq 0 \end{aligned}$$

Showing that this linear program is feasible for any $Z : \{0,1\}^n \to \{0,1\}$ is equivalent to proving the lower bound for the given f, d and ε.

The first step to understand the latter linear program is to determine non-negative vectors in the nullspace of A^T. We construct a basis for the nullspace of A^T. It is easy to see that $A^T = [M \mid -M]$ for a matrix M. The entries of M are $\begin{cases} M_{j,i} = 1 & S_j \subseteq S_i \\ 0 & \text{otherwise.} \end{cases}$ Hence, $A^T \cdot [e_i \mid e_i] = 0$ for each $1 \leq i \leq 2^n$.

The remaining vectors in the basis can be obtained from $null(M)$ using the following procedure. Let γ be a vector in $null(M)$. Define $\gamma^+ = \max(\gamma, 0)$ and $\gamma^- = \max(-\gamma, 0)$, where max is taken coordinate wise. Then, $\beta = [\gamma^+ \mid \gamma^-]$ is a non-negative vector in the nullspace of A^T. Hence, choose a basis B for $null(M)$ and construct β from the elements of this basis. These vectors, along with $[e_i \mid e_i]$, form a basis for the nullspace of A^T (we leave it to the reader to verify this).

Now, note that $\mathbf{b}^T \cdot [e_i, e_i] = 2\varepsilon$, while we need $\mathbf{b}^T \cdot \beta < 0$. Also, for any β, define β' such that $\beta'_i = \beta_i - \min(\beta_i, \beta_{i+n+1}), \beta'_{i+n+1} = \beta_i - \min(\beta_i, \beta_{i+n+1})$ for all $1 \leq i \leq 2^n$. Then, $\mathbf{b}^T \cdot \beta < 0$ implies $\mathbf{b}^T \cdot \beta' < 0$. Note that these are exactly the vectors that can be obtained as $\beta = [\gamma^+ \mid \gamma^-]$ from $\gamma \in null(M)$. Therefore, the focus is on $null(M)$.

We construct a basis for $null(M)$. This can be achieved using an extension trick. Consider the matrix D of dimension $2^n \times 2^n$ where $D_{j,i} = \begin{cases} 1 & S_j \subseteq S_i \\ 0 & S_j \not\subseteq S_i \end{cases}$. This extends the matrix M to a square matrix.

Lemma 6.2. *A basis B for $null(M)$ consists of the last $\sum_{w=d+1}^{n} \binom{n}{w}$ columns of D^{-1}.*

Proof. M consists of the first $\sum_{w=0}^{d} \binom{n}{w}$ rows of D, while B consists of the last $\sum_{w=d+1}^{n} \binom{n}{w}$ columns of D^{-1}. Hence, each row of M multiplied by any column of B equals 0. Therefore, B is a basis for $null(M)$. ■

We describe D^{-1} explicitly.

Lemma 6.3. $D^{-1}_{i,j} = \begin{cases} (-1)^{|S_i|+|S_j|} & S_j \subseteq S_i \\ 0 & \textit{otherwise.} \end{cases}$

Proof. Towards a proof, consider $(DD^{-1})_{k,i} = \sum_{j=1}^{2^n} 1_{S_k \subseteq S_j} (-1)^{|S_i|+|S_j|} 1_{S_j \subseteq S_i}$. If $k = i$, the only non-zero entry on the RHS is when $S_j = S_i = S_k$, which is 1, therefore $(DD^{-1})_{i,i} = 1$. When $k > i$, there is no subset S_j such that $S_j \subseteq S_i$ and $S_k \subseteq S_j$, therefore $(DD^{-1})_{k,i} = 0$.

For $k < i$, consider two cases.

- Let $S_k \subseteq S_i$ with $|S_i| = |S_k| + d$ where $d \geq 1$. Then, for any $0 \leq d' \leq d$, there are $\binom{d}{d'}$ many sets such that $|S_j| = |S_k| + d'$ and $S_k \subseteq S_j \subseteq S_i$. Hence, $DD^{-1}_{k,i} = \sum_{d'=0}^{d} (-1)^{d'} \binom{d}{d'} = 0$.
- Let $S_k \not\subseteq S_i$. Consider any set S_j such that $S_k \subseteq S_j$. Then, there is $s \in S_j$ such that $s \notin S_i$. Hence, $S_j \not\subseteq S_i$. Similarly, if $S_j \subseteq S_i$, then $S_k \not\subseteq S_j$. Therefore, $DD^{-1}_{k,i} = 0$.

This completes the proof. ■

The remaining step in the proof is to find a γ such that for $\beta = [\gamma^+ \mid \gamma^-]$, $\mathbf{b}^T \cdot \beta < 0$. The expansion of $\mathbf{b}^T \cdot \beta$ can be seen to be $\mathbf{b}^T \cdot \beta = \left(\sum_{i=1}^{n} \left(Z(x^{(i)}) + \frac{f(x^{(i)})}{2} \right) \gamma_i \right) + \varepsilon' \|\gamma\|_1$. Hence, finding a β such that $A^T\beta = 0, \beta \geq 0, \mathbf{b}^T \cdot \beta < 0$ is precisely equivalent to the statement of the theorem.

This completes the proof. □

6.2 Torus Polynomial Degree Lower Bounds for the AND Function

Here, we prove Theorem 1.5. We recall the statement here.

Theorem 1.5. *Any torus polynomial approximating the* AND *function within an error of* $O\left(\frac{1}{2^{\log^c(n)}}\right)$ *must have degree* $\Omega(\log^c(n))$.

Proof. First, we prove the lower bound for the OR function. Then, we use De Morgan's law to prove that the same lower bound holds for the AND function.

To see the lower bound for OR, apply Theorem 6.1 with $f = \mathsf{OR}, d = \log^c(n) - 3, \varepsilon = \frac{1}{2^{\log^c(n)}}$. Choose γ as the first column of B. Then, $\langle f, \gamma \rangle = -1$, which is odd. Moreover, $\|\gamma\|_1 = 2^{d+1} = 2^{\log^c(n)-2}$. Hence, the lower bound holds for $\varepsilon < \frac{1}{2\|\gamma\|_1} = \frac{1}{2^{\log^c(n)-1}}$

Now, consider a torus polynomial $P(x_1, \ldots, x_n)$ of degree d, approximating AND within an error of ε. Then, $\frac{1}{2} + P(1 - x_1, \ldots, 1 - x_n)$ is a polynomial of degree d, approximating OR within an error of ε. Hence, we get the desired lower bound. □

6.3 Limits to using Minkowski's Theorem

In the asymmetric case, the matrix B has dimensions $2^n \times \sum_{w=d+1}^{n} \binom{n}{w}$. If we apply Minkowski's theorem to $\mathcal{L}(B)$, we get that it contains a vector γ with its ℓ^2 norm bounded by

$$\|\gamma\|_1 \leq \sqrt{\sum_{w=d+1}^{n} \binom{n}{w}} \, det(B^T B)^{\frac{1}{2\sum_{w=d+1}^{n}\binom{n}{w}}}$$

For $d \leq \frac{n}{2}$, the first term in the product is already quite large. Hence, this will imply a really large upper bound on the smallest ℓ^2 norm, not small enough to prove Conjecture 2. Regardless, we calculate $det(B^T B)$, as it has a closed form solution and is an interesting combinatorial problem in its own right. The quantity $det(B^T B)$ denotes the volume of the fundamental parallelepiped associated with the lattice.

The matrix $B^T B$ has the following entries. Its rows and columns are indexed by sets of size at least $d + 1$. The entry indexed by S_1, S_2 is $(B^T B)_{S_1,S_2} = (-1)^{|S_1|+|S_2|} 2^{|S_1 \cap S_2|}$. We did not find it easy to calculate the determinant using this form. We notice that B has a recursive structure, which considerably simplifies the calculation of the determinant. Following is the recursive structure.

Lemma 6.4. *Denote by* $B(n, d)$ *the basis of the nullspace for* n, d. *Then,* $B(n, d)$ *has the following recursive structure.*

$$B(n,d) = \begin{bmatrix} B(n-1,d-1) & 0 \\ -B(n-1,d-1) & B(n-1,d) \end{bmatrix}$$

Proof. Consider an entry in $B(n, d)$, with its row and column indexed by sets S_1, S_2 respectively. We consider four cases, based on which set contains 1.

- $1 \in S_1, 1 \in S_2$. In this case, $S_1 \subseteq S_2$ if and only if $S_1 \setminus \{1\} \subseteq S_2 \setminus \{1\}$. Note that this reduces the size of both the row set and the column set by 1. Hence, $B(n,d)_{S_1,S_2} = B(n-1,d-1)_{S_1\setminus\{1\},S_2\setminus\{1\}}$. Therefore, the top-left block of $B(n, d)$ is exactly $B(n-1, d-1)$.
- $1 \in S_1, 1 \notin S_2$. In this case, $S_1 \subseteq S_2$ never holds. Hence, the top-right block of $B(n, d)$ is 0.
- $1 \notin S_1, 1 \in S_2$. In this case, $S_1 \subseteq S_2$ if and only if $S_1 \subseteq S_2 \setminus \{1\}$. Note that this reduces the size of the column set by 1. Moreover, as $1 \notin S_1$, we can consider $S_1 \subseteq \{2, \ldots, n\}$. Hence, $B(n,d)_{S_1,S_2} = -B(n-1,d-1)_{S_1,S_2\setminus\{1\}}$. Therefore, the top-left block of $B(n, d)$ is exactly $-B(n-1, d-1)$.

- $1 \notin S_1, 1 \notin S_2$. In this case, we can consider both $S_1, S_2 \subseteq \{2, \ldots, n\}$. Hence, $B(n,d)_{S_1,S_2} = B(n-1,d)_{S_1\setminus\{1\},S_2\setminus\{1\}}$. Therefore, the top-left block of $B(n,d)$ is exactly $B(n-1,d)$.

This completes the proof. □

With this recursive structure in place, it becomes much easier to calculate $det(B^TB)$.

Theorem 6.5. $det(B^TB) = 2^{n\binom{n-1}{d}}$.

Proof. We use the following notation to shorten the expressions.

$$B = B(n,d) \qquad B_d = B(n-1,d) \qquad B_{d-1} = B(n-1,d-1)$$

Using the recursive structure of B, we get,

$$B^TB = \begin{bmatrix} 2B_{d-1}^TB_{d-1} & -B_{d-1}^TB_d \\ -B_d^TB_{d-1} & B_d^TB_d \end{bmatrix}$$

Using the Schur formula [Sch18] for the determinant of block matrices, we get

$$det(B^TB) = det(2B_{d-1}^TB_{d-1}) * det(B_d^TB_d - B_d^TB_{d-1}(2B_{d-1}^TB_{d-1})^{-1}B_{d-1}^TB_d)$$

Now, note that $B_{d-1}(B_{d-1}^TB_{d-1})^{-1}B_{d-1}^T$ is the matrix for projecting on the column-space of B_{d-1}. Moreover, each column of B_d is also a column of B_{d-1}. Hence, a column of B_d remains unchanged when projected on the column-space of B_{d-1}. Therefore, $B_{d-1}(B_{d-1}^TB_{d-1})^{-1}B_{d-1}^TB_d = B_d$.

With this observation, we can simplify the formula as

$$det(B^TB) = det(2B_{d-1}^TB_{d-1})\, det\left(\frac{B_d^TB_d}{2}\right)$$

Now, the number of columns in $B_{d-1}^TB_{d-1}$ is $\sum_{w=d}^{n-1}\binom{n-1}{w}$. Hence,

$$det(2B_{d-1}^TB_{d-1}) = 2^{\sum_{w=d}^{n-1}\binom{n-1}{w}}\, det(B_{d-1}^TB_{d-1})$$

Similarly, the number of columns in $B_d^TB_d$ is $\sum_{w=d+1}^{n-1}\binom{n-1}{w}$. Hence,

$$det\left(\frac{B_d^TB_d}{2}\right) = 2^{-\sum_{w=d+1}^{n-1}\binom{n-1}{w}}\, det(B_d^TB_d)$$

Therefore,

$$det(B^TB) = 2^{\binom{n-1}{d}}\, det(B_{d-1}^TB_{d-1})\, det(B_d^TB_d)$$

Define $LD(n,d) = \log_2(det(B^TB))$ as the log of the determinant of B^TB. Then, the equation above leads to the following recursion:

$$LD(n,d) = LD(n-1,d) + LD(n-1,d-1) + \binom{n-1}{d}$$

There are two base cases of this recursion.

- $B(n,-1) = I$. Hence, $B^TB(n,-1) = I$. Therefore, $LD(n,-1) = \log_2(1) = 0$.
- $B(n,n-1)$ is a single column, of length 2^n, with 1 and -1 entries. Hence, $B^TB(n,n-1) = [2^n]$. Therefore, $LD(n,n-1) = n$.

With these base cases, it is easy to verify that $LD(n,d) = n\binom{n-1}{d}$ satisfies the recursion. Hence, $det(B^TB) = 2^{n\binom{n-1}{d}}$. □

Now, if we apply Corollary 2, we get that there exists a vector γ with its ℓ^1 norm bounded by

$$\|\gamma\|_1 \leq \sqrt{2^n}\sqrt{\sum_{w=d+1}^{n}\binom{n}{w}}2^{\frac{n\binom{n-1}{d}}{2\left(\sum_{w=d+1}^{n}\binom{n}{w}\right)}}$$

Note that there are basis columns with ℓ^1 norm 2^{d+1}, which is better than the previous bound for most values of d. Hence, the error-degree trade-off we get here is not enough to resolve Conjecture 1.

6.4 The Projection Matrix in the Asymmetric Case

Inspired by the symmetric case, we study the projections here as well. We first establish the importance of such a study. Consider the projection of $Z + \frac{f}{2}$, i.e. $P\left(Z + \frac{f}{2}\right)$. Then, it is easy to observe the following:

Observation 2.

$$\min_{Z\in\mathbb{Z}^{2^n}}\left\|P\left(Z+\frac{f}{2}\right)\right\|_2 \geq \min_{Z\in\mathbb{Z}^{2^n}}\max_{\gamma\in null(M)}\frac{\left\langle Z+\frac{f}{2},\gamma\right\rangle}{\|\gamma\|_1} \geq \min_{Z\in\mathbb{Z}^{2^n}}\frac{\left\|P\left(Z+\frac{f}{2}\right)\right\|}{2^{\frac{n}{2}}}$$

Hence, for any Z, the ℓ^2 norm of the projection has to be large in order for us to prove a lower bound against asymmetric torus polynomials. Moreover, if it is at least an inverse polynomial fraction of $2^{\frac{n}{2}}$, then it is sufficient to prove an inverse polynomial lower bound for the error. Therefore, it is beneficial to study the projection en route to proving lower bounds against asymmetric torus polynomials.

Next, we calculate an explicit formula for the projection matrix in the asymmetric case.

Theorem 6.6. *Denote the symmetric difference of two sets S_1, S_2 by $S_1\Delta S_2$, i.e. $S_1\Delta S_2 = (S_1 \cap \overline{S_2}) \cup (S_2 \cap \overline{S_1})$. Given an n and d, the projection matrix $P \in \mathcal{M}_{2^n\times 2^n}(\mathbb{R})$ has the following entries:*

$$P_{S_1,S_2} = \begin{cases} \frac{\sum_{w=d+1}^{n}\binom{n}{w}}{2^n} & S_1 = S_2 \\ \frac{\sum_{i=0}^{k-1}(-1)^{i+1}\binom{k-1}{i}\binom{n-k}{d-i}}{2^n} & |S_1\Delta S_2| = k \end{cases}$$

Proof. We denote by $B(n,d)$ the basis of the nullspace, and by $P(n,d)$ the projection matrix, for n, d. Then, we use the recursive structure of $B(n,d)$, as described in Lemma 6.4.

$$B(n,d) = \begin{bmatrix} B(n-1,d-1) & 0 \\ -B(n-1,d-1) & B(n-1,d) \end{bmatrix}$$

We use the following notation to shorten the expressions.

$$\begin{aligned} &B = B(n,d) && B_d = B(n-1,d) && B_{d-1} = B(n-1,d-1) \\ &A = B^TB && A_d = B_d^TB_d && A_{d-1} = B_{d-1}^TB_{d-1} \\ &P = P(n,d) = BA^{-1}B^T && P_d = B_dA_d^{-1}B_d^T && P_{d-1} = B_{d-1}A_{d-1}^{-1}B_{d-1}^T \end{aligned}$$

Hence,

$$B^T B = \begin{bmatrix} 2B_{d-1}^T B_{d-1} & -B_{d-1}^T B_d \\ -B_d^T B_{d-1} & B_d^T B_d \end{bmatrix}$$

Next, we calculate A^{-1} using a formula for the inverse of block matrices, as described by Lu and Shiou [LS02].

$$A^{-1} = \begin{bmatrix} A_{d-1}^{-1}/2\left(I + B_{d-1}^T P_d B_{d-1} A_{d-1}^{-1}\right) & A_{d-1}^{-1} B_{d-1}^T B_d A_d^{-1} \\ \\ A_d^{-1} B_d^T B_{d-1} A_{d-1}^{-1} & 2A_d^{-1} \end{bmatrix}$$

Multiplying by B on the left gives:

$$BA^{-1} = \begin{bmatrix} B_{d-1} A_{d-1}^{-1}/2\left(I + B_{d-1}^T P_d B_{d-1} A_{d-1}^{-1}\right) & P_{d-1} B_d A_d^{-1} \\ \\ P_d B_{d-1} A_{d-1}^{-1} - B_{d-1} A_{d-1}^{-1}/2\left(I + B_{d-1}^T P_d B_{d-1} A_{d-1}^{-1}\right) & 2B_d A_d^{-1} - P_{d-1} B_d A_d^{-1} \end{bmatrix}$$

Now, we make an observation that simplifies our calculations considerably. We note that each column of B_d is also a column of B_{d-1}. Hence, a column of B_d remains unchanged when projected on the column space of B_{d-1}. Therefore, $P_{d-1}B_d$ is equal to B_d. This yields $2B_d A_d^{-1} - P_{d-1} B_d A_d^{-1} = B_d A_d^{-1}$. Now, by multiplying B^T on the right we get:

$$P = \begin{bmatrix} (P_{d-1} + P_{d-1} P_d P_{d-1})/2 & P_d - (P_{d-1} + P_{d-1} P_d P_{d-1})/2 \\ \\ P_d P_{d-1} - (P_{d-1} + P_{d-1} P_d P_{d-1})/2 & P_d - P_d P_{d-1} + (P_{d-1} + P_{d-1} P_d P_{d-1})/2 \end{bmatrix}$$

Again, $P_{d-1}B_d = B_d$ implies $P_{d-1}P_d = P_d$. Taking transpose on both sides, and noting that both P_d, P_{d-1} are symmetric matrices, we get $P_d P_{d-1} = P_d$. Therefore, the final expression we get is

$$P(n,d) = \begin{bmatrix} (P(n-1,d) + P(n-1,d-1))/2 & (P(n-1,d) - P(n-1,d-1))/2 \\ \\ (P(n-1,d) - P(n-1,d-1))/2 & (P(n-1,d) + P(n-1,d-1))/2 \end{bmatrix}$$

The base cases for this recursion occur when $n = d$, $d = -1$, or $n = 1, d = 0$. For these, we have the following basis:

$$B(n,n) = 0_{0 \times 2^n}$$
$$B(n,-1) = I_{2^n \times 2^n}$$
$$B(1,0) = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Hence, the base cases are:

$$P(n,n) = 0_{2^n \times 2^n}$$
$$P(n,-1) = I_{2^n \times 2^n}$$
$$P(1,0) = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

It is easy to observe that the base cases satisfy the claimed formula. Now, it is an easy exercise of induction to obtain the claimed formula for all n, d. □

The main conjecture of this paper is now given below.

Conjecture 13. *Consider* $f = \mathsf{MAJ}_n$ *and* $d = o\left(\sqrt{\frac{n}{\log(n)}}\right)$. *Then,*

$$\min_{Z \in \mathbb{Z}^n} \left\| P(n,d)\left(Z + \frac{f}{2}\right) \right\| \geq \frac{2^{\frac{n}{2}}}{n^{O(1)}}$$

A positive resolution of this conjecture will lead to a positive resolution of Conjecture 3, thereby proving Conjecture 1.

7 Results for Real Polynomials

This is a "harvest section", a phrase we borrow from Babai and Frankl. Coincidentally, their section is focused on inclusion matrices, which are also our focus (see of [BF92, Section 7.3]). We prove lower bound results against real polynomials, using the machinery we have developed, almost effortlessly.

7.1 Lower Bounds via the Projection Theory

Here, we showcase the power of the framework we have developed in the previous section. We consider a large family of *asymmetric* functions, and prove degree lower bounds on real polynomials approximating them within small error. Formally, we prove the following.

Theorem 1.6. *Fix a constant* $0 < c \leq 1$. *Consider an* odd n. *Take* $f : \{0,1\}^n \to \{0,1\}$ *to be any function that evaluates to* 0 *on any Boolean point* $x \in \{0,1\}^n$ *of even Hamming weight. If* f *evaluates to* 1 *on a* c *fraction of the inputs with odd Hamming weight, then the following holds. Any real polynomial approximating* f *within an error of* $\varepsilon < \frac{\sqrt{c}}{2}$ *must have degree at least* $\frac{n+1}{2}$.

Proof. Again, we follow the proof of Theorem 6.1 using $Z = 0$. Hence, we get that $\varepsilon \geq \frac{\|P(n,d)f\|_2}{2^{\frac{n}{2}}}$.

First, we apply the formula obtained in Theorem 6.6 to calculate the projection matrix when n is odd and $d = \frac{n-1}{2}$. We recall the formula here.

$$P_{S_1,S_2} = \begin{cases} \frac{\sum_{w=d+1}^{n} \binom{n}{w}}{2^n} & S_1 = S_2 \\ \frac{\sum_{i=0}^{k-1} (-1)^{i+1} \binom{k-1}{i}\binom{n-k}{d-i}}{2^n} & |S_1 \Delta S_2| = k \end{cases}$$

The summation in the first case if $\sum_{w=\frac{n+1}{2}}^{n} \binom{n}{w} = 2^{n-1}$, as it sums up exactly the top half of the binomial coefficients. Hence, $P_{S,S} = \frac{1}{2}$.

Now, consider the second case when $k = |S_1 \Delta S_2|$ is non-zero and *even*. Here, we prove that $P_{S_1,S_2} = 0$ in this case. We have chosen n and d carefully to make further calculations almost trivial.

Lemma 7.1. *For any* $d \geq 0$, $n = 2d + 1$, *and* $S_1 \neq S_2$ *with* $|S_1 \Delta S_2|$ *being even,* $P_{S_1,S_2} = 0$.

Proof.

$$P_{S_1,S_2} = \sum_{i=0}^{k-1} (-1)^{i+1} \binom{k-1}{i} \binom{2d+1-k}{d-i}$$

Using the binomial identity $\binom{n}{r} = \binom{n}{n-r}$, we get,

$$P_{S_1,S_2} = \sum_{i=0}^{k-1} (-1)^{i+1} \binom{k-1}{k-1-i} \binom{2d+1-k}{d+1-k+i}$$

Now, by substituting $i' = k-1-i$, we get,

$$P_{S_1,S_2} = \sum_{i'=0}^{k-1} (-1)^{k-1-i'+1} \binom{k-1}{i'} \binom{2d+1-k}{d-i'}$$

As k is even,

$$P_{S_1,S_2} = \sum_{i'=0}^{k-1} (-1)^{i'} \binom{k-1}{i'} \binom{2d+1-k}{d-i'} = -P_{S_1,S_2}$$

Hence, $P_{S_1,S_2} = 0$. ■

Now, consider any set S such that $f(x^S) = 1$, where $x_i^S = \begin{cases} 1 & i \in S \\ 0 & i \notin S \end{cases}$. Then,

$$(Pf)_S = P_{S,S} f(S) + \sum_{S_1 : |S_1| \equiv 0 \mod 2} P_{S,S_1} f(S_1) + \sum_{S_2 : |S_2| \equiv 1 \mod 2} P_{S,S_2} f(S_2)$$

Now, $f(S_1) = 0$ for any set S_1 with even size. Also, for any set S_2 with odd size, we have $|S \Delta S_2|$ is even. Hence, $P_{S,S_2} = 0$. Therefore, $(Pf)_S = \frac{1}{2}$.

Now, there are 2^{n-1} sets S with odd size. Of these, a c fraction of them have $(Pf)_S = \frac{1}{2}$. Hence, $\|Pf\|_2 \geq \sqrt{c 2^{n-1} \frac{1}{2}} = 2^{\frac{n}{2}} \frac{\sqrt{c}}{2}$. Therefore, the dual, as defined in Theorem 6.1, is feasible for any $\varepsilon < \frac{\|Pf\|_2}{2^{\frac{n}{2}}} = \frac{\sqrt{c}}{2}$. The proof of the lower bound follows. □

Note how easily we have obtained a lower bound for asymmetric real polynomials. Hence, this suggests that if one is looking to prove lower bounds against real polynomials, an early step would be to simply check if the projection of the function has large ℓ^2 norm. An artful choice of n, d, like in the proof of the statement above, can also simplify the task significantly.

To compare with existing lower bounds, the method of dual polynomials implies a degree lower bound of n if the error is less than $\frac{c}{2}$. Hence, our result gives a quadratic improvement over the error by reducing the degree lower bound to $\frac{n+1}{2}$.

7.2 Maximal Bound for the Majority Function

We prove that for any large n, any real polynomial that approximates MAJ_n within an error of $O\left(\frac{1}{\sqrt{n}}\right)$ must have degree n.

Theorem 7.2. *There exists a $c \in \mathbb{R}$ such that the following holds for large n. Any real polynomial approximating the majority function within an error of $\frac{1}{c\sqrt{n}}$ must have degree n.*

Proof. The theory developed for torus polynomials applies to real polynomials as well. The only change is that we set $Z = 0$. This makes it much easier, and we exhibit this power by proving new lower bounds for real polynomials.

Consider the matrix A from the proof of Theorem 1.3. The nullspace of A^T contains the vector β where $\beta_i = \binom{n}{i}, \beta_{i+n+1} = 0$ for even $0 \leq i \leq n$ and $\beta_i = 0, \beta_{i+n+1} = \binom{n}{i}$ for odd $0 \leq i \leq n$. Then $\mathbf{b}^T \cdot \beta = \sum_{i=0}^{n} (-1)^i f(i) \binom{n}{i} + \varepsilon \left(\sum_{i=0}^{n} \binom{n}{i}\right)$.

Note that $f(i) = 1$ for $i > \frac{n}{2}$ and $f(i) = 0$ otherwise. Therefore, $b^T \cdot \beta = \sum_{i > \frac{n}{2}} (-1)^i \binom{n}{i} + \varepsilon \left(\sum_{i=0}^{n} \binom{n}{i}\right)$.

We use the identity $\sum_{i>n/2} (-1)^i \binom{n}{i} = (-1)^{\lceil \frac{n+1}{2} \rceil} \binom{n-1}{\lceil \frac{n-1}{2} \rceil}$ to get $b^T \cdot \beta = (-1)^{\lceil \frac{n+1}{2} \rceil} \binom{n-1}{\lceil \frac{n-1}{2} \rceil} + \varepsilon 2^n$.

Now, We consider the following two cases. If $\lceil \frac{n+1}{2} \rceil$ is odd, we use Stirling's approximation to get that $-\binom{n-1}{\lceil \frac{n-1}{2} \rceil} < -\frac{2^n}{c\sqrt{n}}$ for some $c \in \mathbb{R}$ for large n. Hence, $b^T \cdot \beta < 0$. If $\lceil \frac{n+1}{2} \rceil$ is even, we consider the vector β' such that $\beta'_i = \beta_{i+n+1}, \beta'_{i+n+1} = \beta_i$ for $0 \leq i \leq n$. Then, $b^T \cdot \beta' = -\binom{n-1}{\lceil \frac{n-1}{2} \rceil} + \varepsilon 2^n < 0$

This proves that the dual is feasible, hence proving the lower bound for symmetric real polynomials. Note that one can always assume that a real polynomial approximating MAJ_n is symmetric. If P is not symmetric, simply consider $P_s = \sum_{\pi \in \mathfrak{S}_n} \frac{P \circ \pi}{n!}$. Then, P_s is symmetric and the degree of P_s is at most the degree of P. Hence, the degree lower bound holds for real polynomials in general.

Note that a degree upper bound of n is well-known and trivial for any function f. This completes the proof. □

8 Future Directions

We have already described several problem statements arising out of our work. Here, we discuss some future directions to explore, that may be of independent interest.

Torus Polynomial Lower Bounds for MOD_m Functions. We have proved lower bounds for torus polynomials approximating the AND function, which is tight within polynomial factors. A torus polynomial for the MOD_2 function is simply $\frac{\sum_{i=1}^n x_i}{2}$, which approximates it within an error of 0. Note that this is a symmetric polynomial of degree 1. We believe that this is a special case, that any torus polynomial approximating the MOD_m function, for $m \neq 2$, within an error of, say $O\left(\frac{1}{n}\right)$, must have degree $\Omega(\log(n))$.

Conjecture 14. *Any torus polynomial approximating the MOD_m function, for $m \neq 2$, within an error of $O\left(\frac{1}{n}\right)$ must have degree $\Omega(\log(n))$.*

This will complete the theory by proving lower bounds for all constituent functions of ACC circuits.

More Real Polynomial Lower Bounds.

Problem 15. *Use the method we have described to prove lower bounds for a larger family of functions against real polynomials approximating them.*

Relating Dual Polynomials and Our Method. The method of dual polynomials has seen considerable success in the literature of real polynomials. Our starting point is the same, but we use the geometric interpretation to prove feasibility of the dual. Both the methods achieve the same objective, but with different viewpoints. Is there a way to relate the two?

Problem 16. *Re-interpret our methods, for example, the shortest vector based lower bound, as a dual polynomial.*

The Method of Dual Block Composition. One of the ways of proving lower bounds against real polynomial approximations for block composed function is via the method of dual block composition. See [She13, Lee09, SZ09] (also see [BT22] for an in-depth discussion). Is there such a method for torus polynomials?

Problem 17. *Find an analogue of the method of dual block composition for proving lower bounds against torus polynomials.*

The Bipartite Perfect Matching Function. In a beautiful paper, Benaimini and Nisan [BN21] studied real polynomials for the bipartite matching problem function. The input consists of n^2 variables $x_{i,j}$, where $x_{i,j}$ is 1 if there is an edge between i and j, and 0 otherwise. The function

outputs 1 if the graph contains a perfect matching, otherwise it outputs 0. They proved that the unique real polynomial computing the bipartite perfect matching function has degree n^2.

Now, this function is contained in P. We believe that this is another explicit function that does not belong to ACC. Hence, we conjecture that it does not permit low degree torus polynomial approximations. Note that this is an asymmetric function. Therefore, we need to study asymmetric torus polynomials.

Conjecture 18. *Any torus polynomial approximating the bipartite matching function within an error of* $O\left(\frac{1}{n}\right)$ *must have degree* $\log^{\omega(1)}(n)$.

Proving this would separate P from ACC.

References

[Aar08] Scott Aaronson. The polynomial method in quantum and classical computing. In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 3–3. IEEE, 2008.

[Ajt83] Miklós Ajtai. Σ_1^1-formulae on finite structures. *Annals of pure and applied logic*, 24(1):1–48, 1983.

[All89] Eric Allender. A note on the power of threshold circuits. In *30th Annual Symposium on Foundations of Computer Science*, pages 580–584. IEEE Computer Society, 1989.

[Bar89] David A Barrington. Bounded-width polynomial-size branching programs recognize exactly those languages in NC. *Journal of Computer and System Sciences*, 38(1):150–164, 1989.

[BF92] László Babai and Péter Frankl. *Linear algebra methods in combinatorics: with applications to geometry and computer science*. Department of Computer Science, univ. of Chicag, 1992.

[BHLR19] Abhishek Bhrushundi, Kaave Hosseini, Shachar Lovett, and Sankeerth Rao. Torus polynomials: An algebraic approach to acc lower bounds. *10th Innovations in Theoretical Computer Science*, 2019.

[BN21] Gal Beniamini and Noam Nisan. Bipartite perfect matching as a real polynomial. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1118–1131, 2021.

[BT22] Mark Bun and Justin Thaler. Approximate degree in classical and quantum computing. *Foundations and Trends® in Theoretical Computer Science*, 15(3-4):229–423, 2022.

[Che19] Lijie Chen. Non-deterministic quasi-polynomial time is average-case hard for ACC circuits. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1281–1304. IEEE, 2019.

[Che23] Lijie Chen. New Lower Bounds and Derandomization for ACC, and a De-randomization-Centric View on the Algorithmic Method. In *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2023.

[CLLO21] Lijie Chen, Zhenjian Lu, Xin Lyu, and Igor C Oliveira. Majority vs. approximate linear sum and average-case complexity below NC^1. In *48th International Colloquium on Automata, Languages, and Programming (ICALP 2021)*, volume 198, page 51. Leibniz International Proceedings in Informatics, 2021.

[CLW20] Lijie Chen, Xin Lyu, and R Ryan Williams. Almost-everywhere circuit lower bounds from non-trivial derandomization. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1–12. IEEE, 2020.

[CM22] Arkadev Chattopadhyay and Nikhil S Mande. A Short List of Equalities Induces Large Sign-Rank. *SIAM Journal on Computing*, 51(3):820–848, 2022.

[COS18] Ruiwen Chen, Igor C Oliveira, and Rahul Santhanam. An Average-Case Lower Bound Against ACC^0. In *Latin American Symposium on Theoretical Informatics*, pages 317–330. Springer, 2018.

[Dic52] Leonard Eugene Dickson. *History of the Theory of Numbers*. Number 256. Chelsea publishing company, 1952.

[Far02] Julius Farkas. Theorie der einfachen Ungleichungen. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 1902(124):1–27, 1902.

[Fin47] Nathan J Fine. Binomial coefficients modulo a prime. *The American Mathematical Monthly*, 54(10):589–592, 1947.

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

[Gri] Darij (https://math.stackexchange.com/users/586/darij-grinberg) Grinberg. Determinant of a Matrix with Binomial Coefficient Entries. Mathematics Stack Exchange. URL:https://math.stackexchange.com/q/2690256 (version: 2018-03-14).

[Gri22] Darij Grinberg. A Hyperfactorial Divisibility, 2022.

[Hås86] Johan Håstad. *Computational limitations for small depth circuits*. PhD thesis, Massachusetts Institute of Technology, 1986.

[HMP+93] András Hajnal, Wolfgang Maass, Pavel Pudlák, Mario Szegedy, and György Turán. Threshold circuits of bounded depth. *Journal of Computer and System Sciences*, 46(2):129–154, 1993.

[HP72] WD Hoskins and PJ Ponzo. Some properties of a class of band matrices. *Mathematics of Computation*, 26(118):393–400, 1972.

[HS13] M. Hindry and J.H. Silverman. *Diophantine Geometry: An Introduction*. Graduate Texts in Mathematics. Springer New York, 2013.

[IKW02] Russell Impagliazzo, Valentine Kabanets, and Avi Wigderson. In search of an easy witness: Exponential time vs. probabilistic polynomial time. *Journal of Computer and System Sciences*, 65(4):672–694, 2002.

[Kum52] Ernst Eduard Kummer. Über die ergänzungssätze zu den allgemeinen reciprocitätsgesetzen. *Journal für die reine und angewandte Mathematik*, 1852.

[Lee09] Troy Lee. A note on the sign degree of formulas. *arXiv preprint arXiv:0909.4607*, 2009.

[LS02] Tzon-Tzer Lu and Sheng-Hua Shiou. Inverses of 2× 2 block matrices. *Computers & Mathematics with Applications*, 43(1-2):119–129, 2002.

[Luc78] Édouard Lucas. Sur les congruences des nombres eulériens et des coefficients différentiels des fonctions trigonométriques suivant un module premier. *Bulletin de la Société mathématique de France*, 6:49–54, 1878.

[Min10] Hermann Minkowski. *Geometrie der zahlen*. BG Teubner, 1910.

[MS08] Ketan D Mulmuley and Milind Sohoni. Geometric complexity theory II: towards explicit obstructions for embeddings among class varieties. *SIAM Journal on Computing*, 38(3):1175–1206, 2008.

[MW18] Cody Murray and Ryan Williams. Circuit lower bounds for nondeterministic quasi-polytime: an easy witness lemma for NP and NQP. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 890–901, 2018.

[Ngu09] Phong Q Nguyen. Hermite's constant and lattice algorithms. In *The LLL Algorithm: Survey and Applications*, pages 19–69. Springer, 2009.

[NS94] Noam Nisan and Mario Szegedy. On the degree of Boolean functions as real polynomials. *Computational complexity*, 4:301–313, 1994.

[Pat92] Ramamohan Paturi. On the degree of polynomials that approximate symmetric boolean functions (preliminary version). In *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*, pages 468–474, 1992.

[Raz87] Alexander A Razborov. Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Mathematical Notes of the Academy of Sciences of the USSR*, 41(4):333–338, 1987.

[Sch18] J Schur. Über potenzreihen, die im innern des einheitskreises beschränkt sind. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 1918(148):122–145, 1918.

[She13] Alexander A Sherstov. The intersection of two halfspaces has high threshold degree. *SIAM Journal on Computing*, 42(6):2329–2374, 2013.

[Smo87] Roman Smolensky. Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pages 77–82, 1987.

[Spa08] Robert Spalek. A dual polynomial for OR. *arXiv preprint arXiv:0803.4516*, 2008.

[SZ09] Yaoyun Shi and Yufan Zhu. Quantum communication complexity of block-composed functions. *Quantum Information & Computation*, 9(5):444–460, 2009.

[TS08] Jacobo Toran and Alexander A Sherstov. Communication Lower Bounds Using Dual Polynomials. *Bulletin of the European Association for Theoretical Computer Science*, (95):59–93, 2008.

[Wil13] Ryan Williams. Improving Exhaustive Search Implies Superpolynomial Lower Bounds. *SIAM Journal on Computing*, 42(3):1218–1244, 2013.

[Wil14] Ryan Williams. Nonuniform ACC circuit lower bounds. *Journal of the ACM (JACM)*, 61(1):1–32, 2014.

[Yao85] Andrew Chi-Chih Yao. Separating the polynomial-time hierarchy by oracles. In *26th Annual Symposium on Foundations of Computer Science (SFCS 1985)*, pages 1–10. IEEE, 1985.

[Yao90] AC-C Yao. On ACC and threshold circuits. In *Proceedings [1990] 31st Annual Symposium on Foundations of Computer Science*, pages 619–627. IEEE, 1990.

[Yas21] Masaya Yasuda. A survey of solving SVP algorithms and recent strategies for solving the SVP challenge. In *International Symposium on Mathematics, Quantum Theory, and Cryptography: Proceedings of MQC 2019*, pages 189–207. Springer Singapore, 2021.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il