Upper Bound for Torus Polynomials

Vaibhav Krishan[1]

Indian Institute of Technology Bombay, Powai, Mumbai 400076, India
vkrishan@cse.iitb.ac.in

Abstract. We prove that all functions that have low degree torus polynomials approximating them with small error also have MidBit^+ circuits computing them. This serves as a partial converse to the result that all ACC functions have low degree torus polynomials approximating them with small error, by Bhrushundi, Hosseini, Lovett and Rao (ITCS 2019).

Keywords: Torus polynomials · ACC · MidBit

1 Introduction

Proving lower bounds for boolean circuits has been a major quest in complexity theory. Much of the recent work, for example [6–9, 13, 14, 17, 18] and reference therein, has focused on proving lower bounds for constant-depth circuits.

The first lower bounds for constant-depth circuits consisting of AND, OR, NOT gates were proven by Furst, Saxe and Sipser [10], and independently by Ajtai [1]. The lower bound was improved by Yao [19], and further improved by Håstad [12]. Razborov [15] and Smolensky [16] proved lower bounds for constant-depth circuits that additionally contain MOD_p gates where p is a prime. Barrington [3] first posed the question of extending lower bounds to the class ACC, where MOD_m gates are allowed for a general m.

Williams [18], in a breakthrough result, proved a lower bound against ACC circuits, where the hard function comes from non-deterministic exponential time NEXP. The lower bound was improved to average case by Chen, Oliveira and Santhanam [8]. The hard function was brought down to non-deterministic quasi-polynomial time NQP by Murray and Williams [14], which was then improved to an average case lower bound by Chen [7].

All the above lower bounds for ACC use a conversion of ACC circuits to SYM^+ circuits, a result first proven by Beigel and Tarui [4] and subsequently improved upon by Allender and Gore [2]. SYM^+ circuits are depth-two size-$O(2^{(\log n)^{O(1)}})$ circuits where the top gate is a symmetric function and the bottom layer has AND gates of fan-in $(\log n)^{O(1)}$. Green, Köbler and Torán [11] improved the result to show that the symmetric function implemented by the top gate can be the MidBit function. We define the MidBit function below.

Definition 1 (MidBit)**.** *The* MidBit *function over the input* $(x_1, \ldots, x_n)$ *behaves as follows. Consider the sum of inputs* $\sum_{i=1}^{n} x_i$ *and consider it's binary expansion* $b_{\ell-1}b_{\ell-2}\ldots b_0$ *with* $\ell = \lfloor \log_2(n) \rfloor + 1$ *many bits,* b_0 *being the least significant bit. Then* $\mathsf{MidBit}(x_1, x_2, \ldots, x_n) = b_{\lfloor \ell/2 \rfloor}$.

SYM^+ circuits where the top gate implements the MidBit function are called MidBit^+ circuits. The formal definition follows.

Definition 2 (MidBit^+)**.** *A* MidBit^+ *circuit is a depth-two circuit, with a* MidBit *gate of fan-in* $2^{(\log n)^{O(1)}}$ *at the top, and* AND *gates of fan-in* $(\log n)^{O(1)}$ *at the bottom.*

Green, Köbler and Torán [11] proved that all ACC circuits can be converted into an equivalent MidBit^+ circuit. MidBit^+ circuits are seemingly simpler in structure than ACC circuits, as MidBit^+ circuits have a fixed depth of two while ACC circuits can be of arbitrary constant depth. Understanding the power of MidBit^+ can be important for obtaining ACC lower bounds, as a lower bound for MidBit^+ circuits will automatically translate to a lower bound against ACC circuits.

1.1 Torus Polynomials

While lower bounds against ACC are known from "high" classes, such as NQP, such lower bounds are not known from "low" circuit classes, such as TC^0, the class of constant-depth polynomial size circuits consisting of Majority gates. In other words, it is not yet known whether the containment $\mathsf{ACC} \subseteq \mathsf{TC}^0$ is strict or not. Also the lower bounds for some classes contained in ACC were obtained by using algebraic methods, for example in [15, 16], where it was proved that the concerned class has low degree polynomial approximations of a particular type. Such algebraic methods were not known to carry over to ACC until Bhrushundi et al. [5] introduced torus polynomials, which they proposed as an approach to solving the ACC vs TC^0 problem.

Torus polynomials are polynomials that approximate a function in its fractional part, the integer part is ignored. We first define the torus, based on which we define torus polynomials.

Definition 3 (Torus). *For* $\alpha, \beta \in \mathbb{R}$*, define* $\alpha \equiv \beta \mod 1$ *when* $\alpha - \beta \in \mathbb{Z}$ *(here* $\mod 1$ *is an abuse of notation).*

Define $\alpha \mod 1$ *to be the unique* $\beta \in [-1/2, 1/2)$ *such that* $\alpha \equiv \beta \mod 1$*.*

Definition 4 (Torus Polynomial). *Let* $P \in \mathbb{R}[X_1, \ldots, X_n]$ *be a real multilinear polynomial and* $f : \{0,1\}^n \to \{0,1\}$ *be a boolean function.* P *is a torus polynomial approximating* f *with error* ε *if*

$$\forall x \in \{0,1\}^n, \left| P(x) - \frac{f(x)}{2} \mod 1 \right| \leq \varepsilon$$

Denote by $\overline{deg}_\varepsilon(f)$ *as the smallest possible degree of a torus polynomial approximating* f *with error* ε*.*

All boolean functions have degree n torus polynomials approximating them with error 0, by considering their unique multilinear extension. Bhrushundi et al. [5] proved that all functions in ACC have polylogarithmic degree torus polynomials approximating them with small error.

Theorem 1 ([5]). *Let $f : \{0,1\}^n \to \{0,1\}$ be a boolean function such that $f \in \mathsf{ACC}$, and $n^{-O(1)} \leq \varepsilon < 1/4$. Then $\overline{deg}_\varepsilon(f) \leq (\log n)^{O(1)}$.*

This makes it important to understand the power of torus polynomials, as proving a lower bound on the degree of torus polynomials approximating a certain function with small error will prove a lower bound against ACC. In particular, proving that the Majority function doesn't have low degree torus polynomials approximating it with small error will prove the separation $\mathsf{ACC} \subsetneq \mathsf{TC}^0$. Some evidence towards this was given by Bhrushundi et al. [5] as they proved that *symmetric torus polynomials* need high degree to approximate the Majority function with small error.

1.2 Our Results

Theorem 1 proves that torus polynomials are powerful enough to capture all of ACC. On the other hand, an upper bound on the power of torus polynomials was not yet known. We prove that all functions that have low degree torus polynomials approximating them also have MidBit^+ circuits computing them.

Theorem 2. *Let $n^{-O(1)} \leq \varepsilon < 1/8$ and let $f : \{0,1\}^n \to \{0,1\}$ be a boolean function. Then $\overline{deg}_\varepsilon(f) \leq (\log n)^{O(1)} \implies f \in \mathsf{MidBit}^+$.*

The proof of Theorem 1 by Bhrushundi et al. [5] has two steps, where they first convert an ACC circuit into an equivalent MidBit^+ circuit and then convert the MidBit^+ circuit into a torus polynomial. Therefore our result can be thought of as a partial converse to Theorem 1, which we prove in Section 2. We also prove that the parity of two functions with low degree torus polynomial approximations, also has low degree torus polynomial approximations. We prove this result in Section 3.

2 Torus Polynomials are Equivalent to MidBit^+

We prove an upper bound on the power of torus polynomials, by constructing a MidBit^+ circuit for any function that has a low degree torus polynomial approximation. Any MidBit^+ circuit can be converted into a low degree torus polynomial, as is evident from Bhrushundi et al. [5], but the error of approximation cannot be bounded. We prove that the MidBit^+ circuits we construct satisfy an additional property about the number of AND gates that evaluate to "true" on a particular input. This property, using ideas from Bhrushundi et al. [5], can be used to convert the MidBit^+ circuits we construct into low degree torus polynomials with *small error*.

Theorem 3. *Let $\varepsilon < 1/8$. Let $f : \{0,1\}^n \to \{0,1\}$ have a degree d torus polynomial approximating it with error ε. Then there is a MidBit^+ circuit computing f of the following form:*

- *Fan-in of each AND gate is bounded by d.*

- *Fan-in of the* MidBit *gate is* $2^{2k-1}-1$ *where* $k = O(d \log n + log(1/\varepsilon))$ *(the fan-in puts the middle bit at position* k*).*
- *For* $x \in \{0,1\}^n$*, let* $A(x)$ *be the number of* AND *gates that output* 1*. Then* $A(x) \equiv f(x)2^{k-1} + E(x) \mod 2^k$*, where* $E(x)$ *can be thought of as an error term and is bounded by* $0 \leq E(x) \leq 4\varepsilon 2^k$*.*

Proof. Let $P = \sum_\alpha c_\alpha X^\alpha$ be a degree d torus polynomial approximating f with error ε. The first step is to perturb the coefficients by a small amount to make them rational with a power of 2 in the denominator. Let k be a natural number, the value of which will be chosen later. Find $q_\alpha \in \{0, 1/2^k, \ldots, (2^k-1)/2^k\}$ for each c_α such that

$$c_\alpha - q_\alpha \mod 1 \leq \frac{1}{2^k}$$

Consider $P_{disc} = \sum_\alpha q_\alpha X^\alpha$. Let M be the number of monomials in P. Each monomial can contribute at most $1/2^k$ additional error after changing the coefficients, therefore P_{disc} is a torus polynomial that approximates f with error $\varepsilon' = \varepsilon + M/2^k$. Therefore the following holds:

$$\forall x \in \{0,1\}^n, \left| P_{disc}(x) - \frac{f(x)}{2} \mod 1 \right| \leq \varepsilon'$$

The next step is to add a small rational number so that the resulting polynomial is always "more" than the function when considered $\mod 1$. Find the least natural number q such that $\varepsilon' \leq q/2^k$. Construct the polynomial $P_{pos} = P_{disc} + q/2^k$. The following holds now:

$$\forall x \in \{0,1\}^n, 0 \leq P_{pos}(x) - \frac{f(x)}{2} \mod 1 \leq \varepsilon' + \frac{q}{2^k}$$

Now choose a value of k such that

$$\varepsilon' + \frac{q}{2^k} \leq 4\epsilon$$

Note that $q/2^k \leq \varepsilon' + 1/2^k$ as per the choice of q. Also $M \leq (d+1)n^d$. Substitute these as well as the value of ε' to get that the following inequality suffices for the inequality above to hold.

$$\frac{2(d+1)n^d + 1}{2^k} \leq 2\varepsilon$$

A value of $k = O(d \log n + \log(1/\varepsilon))$ with a large enough constant suffices for this inequality. All coefficients of P_{pos} have 2^k as the common denominator. The next step is to clear out this common denominator to make the coefficients integral. Consider $P_{int} = 2^k \cdot P_{pos}$. It is easy to see that all coefficients of P_{int} are integers. For $x \in \{0,1\}^n$, the following holds:

- $f(x) = 0 \iff P_{int}(x) \equiv E(x) \mod 2^k$,
- $f(x) = 1 \iff P_{int}(x) \equiv 2^{k-1} + E(x) \mod 2^k$,

where $0 \leq E(x) \leq 4\varepsilon 2^k < 2^{k-1}$. The last inequality holds because $\varepsilon < 1/8$.

Hence the value of $f(x)$ is determined exactly by the k^{th} bit of $P_{int}(x)$. Note that P_{int} can be written as $P_{int} = \sum_\alpha n_\alpha X^\alpha$ where $n_\alpha \in \mathbb{Z}$ and $0 \leq n_\alpha < 2^k$.

Use this polynomial to construct the following circuit. For each monomial indexed by α, create n_α many copies of an AND gate, the variables fed to the gate being the variables in the support of α. Note that the fan-in of these AND gates is bounded by d. Feed all these AND gates to a MidBit gate.

There are at most $(d+1)n^d$ many distinct AND gates and each AND gate has at most 2^k many copies. Therefore the fan-in of the MidBit gate is bounded by $2^k(d+1)n^d < 2^{2k-1}$. Add dummy gates which output 0, if needed, to ensure the fan-in of the MidBit gate becomes $2^{2k-1} - 1$.

This circuit will now compute $f(x)$. That the circuit satisfies the third property is easy to see. □

This can now be used to prove Theorem 2 by substituting $d = (\log n)^{O(1)}$ and $n^{-O(1)} \leq \varepsilon$, and observing the fan-in of the top MidBit gate and AND gates is as required for the function to be in MidBit^+.

Proof (Theorem 2 of Subsection 1.2). Substitute $d = (\log n)^{O(1)}$ in Theorem 3. Observe that the fan-in of the AND gates is bounded by $d = (\log n)^{O(1)}$. The fan-in of the top MidBit gate is $2^{2k-1} - 1$ for $k = O(d \log n + \log(1/\varepsilon)) = (\log n)^{O(1)}$. This implies that the fan-in of the MidBit is bounded by $2^{(\log n)^{O(1)}}$. This proves $f \in \mathsf{MidBit}^+$. □

3 Closure under Parity

We prove that if two functions have low degree torus approximations with small error, then the parity of these functions also has low degree torus approximations with slightly larger error. The error grows in an additive fashion.

Theorem 4. *Let $f_1 : \{0,1\}^n \to \{0,1\}$ be a boolean function that has a degree d_1 torus polynomial approximating it with error ε_1. Let $f_2 : \{0,1\}^n \to \{0,1\}$ be another boolean function that has a degree d_2 torus polynomial approximating it with error ε_2. Then $f_1 \oplus f_2$ has a degree $\max(d_1, d_2)$ torus polynomial approximating it with error $\varepsilon_1 + \varepsilon_2$.*

Proof. Let P_1, P_2 be the torus polynomials approximating f_1, f_2 respectively. The polynomial that approximates $f_1 \oplus f_2$ is simply $P_1 + P_2$. Consider these three cases to prove that this approximation is correct:

- Let $f_1(x) = f_2(x) = 0$, hence $f_1(x) \oplus f_2(x) = 0$. In this case

$$|P_1(x) \mod 1| \leq \varepsilon_1, |P_2(x) \mod 1| \leq \varepsilon_2$$

  Therefore

$$|P_1(x) + P_2(x) \mod 1| \leq \varepsilon_1 + \varepsilon_2$$

– Let $f_1(x) = 1, f_2(x) = 0$, hence $f_1(x) \oplus f_2(x) = 1$. In this case

$$|P_1(x) - 1/2 \mod 1| \leq \varepsilon_1, |P_2(x) \mod 1| \leq \varepsilon_2$$

Therefore

$$|P_1(x) + P_2(x) - 1/2 \mod 1| \leq \varepsilon_1 + \varepsilon_2$$

Similar analysis works for $f_1(x) = 0, f_2(x) = 1$.

– Let $f_1(x) = f_2(x) = 1$, hence $f_1(x) \oplus f_2(x) = 0$. In this case

$$|P_1(x) + 1/2 \mod 1| \leq \varepsilon_1, |P_2(x) + 1/2 \mod 1| \leq \varepsilon_2$$

Therefore

$$|P_1(x) + 1/2 + P_2(x) + 1/2 \mod 1| \leq \varepsilon_1 + \varepsilon_2$$

Note that $P_1(x) + 1/2 + P_2(x) + 1/2 \equiv P_1(x) + P_2(x) \mod 1$. Hence

$$|P_1(x) + P_2(x) \mod 1| \leq \varepsilon_1 + \varepsilon_2$$

This proves that $P_1 + P_2$ approximates $f_1 \oplus f_2$ within error $\varepsilon_1 + \varepsilon_2$ in all possible cases. Note that the degree of $P_1 + P_2$ is $\max(d_1, d_2)$. Hence this is the desired polynomial to approximate $f_1 \oplus f_2$. □

4 Future Directions

We have proved that the parity of two functions, that have torus approximations, has a torus approximation as well. It will be interesting to see whether the same can be proven for other ACC functions, such as $\mathsf{AND}, \mathsf{OR}, \mathsf{MOD}_m$. If all these can be proven, it may provide an alternate proof to the fact that ACC has low degree torus approximations.

Acknowledgements

We would like to thank Nutan Limaye for a key suggestion that made our results cleaner to state as well as for helpful feedback on earlier drafts. We would also like to thank Srikanth Srinivasan and Shachar Lovett for useful discussions. Finally we would like to thank anonymous reviewers from CSR for their helpful comments.

References

1. Ajtai, M.: Σ_1^1-formulae on finite structures. Annals of pure and applied logic **24**(1), 1–48 (1983)
2. Allender, E., Gore, V.: On strong separations from AC^0. In: International Symposium on Fundamentals of Computation Theory. pp. 1–15. Springer (1991)
3. Barrington, D.A.: Bounded-width polynomial-size branching programs recognize exactly those languages in NC^1. Journal of Computer and System Sciences **38**(1), 150–164 (1989)

4. Beigel, R., Tarui, J.: On ACC (circuit complexity). In: [1991] Proceedings 32nd Annual Symposium of Foundations of Computer Science. pp. 783–792. IEEE (1991)
5. Bhrushundi, A., Hosseini, K., Lovett, S., Rao, S.: Torus polynomials: An algebraic approach to ACC lower bounds. In: 10th Innovations in Theoretical Computer Science Conference (ITCS 2019). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2018)
6. Chattopadhyay, A., Mande, N.: A short list of equalities induces large sign rank. In: 2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS). pp. 47–58. IEEE (2018)
7. Chen, L.: Non-deterministic quasi-polynomial time is average-case hard for ACC circuits. In: 2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS). pp. 1281–1304. IEEE (2019)
8. Chen, R., Oliveira, I.C., Santhanam, R.: An average-case lower bound against ACC^0. In: Latin American Symposium on Theoretical Informatics. pp. 317–330. Springer (2018)
9. Chen, R., Santhanam, R., Srinivasan, S.: Average-case lower bounds and satisfiability algorithms for small threshold circuits. arXiv preprint arXiv:1806.06290 (2018)
10. Furst, M., Saxe, J.B., Sipser, M.: Parity, circuits, and the polynomial-time hierarchy. Mathematical systems theory **17**(1), 13–27 (1984)
11. Green, F., Köbler, J., Torán, J.: The power of the middle bit. In: [1992] Proceedings of the Seventh Annual Structure in Complexity Theory Conference. pp. 111–117. IEEE (1992)
12. Håstad, J.: Computational limitations of small-depth circuits. MIT press (1987)
13. Kane, D.M., Williams, R.: Super-linear gate and super-quadratic wire lower bounds for depth-two and depth-three threshold circuits. In: Proceedings of the forty-eighth annual ACM symposium on Theory of Computing. pp. 633–643 (2016)
14. Murray, C., Williams, R.: 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. pp. 890–901 (2018)
15. Razborov, A.A.: 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)
16. Smolensky, R.: Algebraic methods in the theory of lower bounds for boolean circuit complexity. In: Proceedings of the nineteenth annual ACM symposium on Theory of computing. pp. 77–82 (1987)
17. Williams, R.: New algorithms and lower bounds for circuits with linear threshold gates. In: Proceedings of the forty-sixth annual ACM symposium on Theory of computing. pp. 194–202 (2014)
18. Williams, R.: Nonuniform ACC circuit lower bounds. Journal of the ACM (JACM) **61**(1), 1–32 (2014)
19. Yao, A.C.C.: Separating the polynomial-time hierarchy by oracles. In: 26th Annual Symposium on Foundations of Computer Science (sfcs 1985). pp. 1–10. IEEE (1985)