Noname manuscript No.
(will be inserted by the editor)

Uniform Proofs of ACC Representations

Sam Buss

the date of receipt and acceptance should be inserted later

Abstract We give a uniform proof of the theorems of Yao and Beigel-Tarui representing ACC predicates as constant depth circuits with MOD_m gates and a symmetric gate. The proof is based on a relativized, generalized form of Toda's theorem expressed in terms of closure properties of formulas under bounded universal, existential and modular counting quantifiers. This allows the main proofs to be expressed in terms of formula classes instead of Boolean circuits. The uniform version of the Beigel-Tarui theorem is then obtained automatically via the Furst-Saxe-Sipser and Paris-Wilkie translations. As a special case, we obtain a uniform version of Razborov and Smolensky's representation of $\mathrm{AC}^0[p]$ circuits. The paper is partly expository, but is also motivated by the desire to recast Toda's theorem, the Beigel-Tarui theorem, and their proofs into the language of bounded arithmetic. However, no knowledge of bounded arithmetic is needed.

Keywords uniform circuits, constant depth circuits, modular counting, Toda Theorem, Biegel-Tarui Theorem

Mathematics Subject Classification (2000) 03D15,03D20,68Q15

1 Introduction

Yao [35] and Beigel-Tarui [7] proved a representation theorem for ACC showing that ACC circuits can be transformed into constant depth quasipolynomial size

Supported in part by NSF grants DMS-1101228 and CCR-1213151, Simons Foundation Grant 306202 and Skolkovo Institute of Science and Technology. Part of this work was performed during the Special Semester program in Complexity in St. Petersburg, Russia.

Sam Buss
Department of Mathematics, University of California, San Diego, La Jolla, CA 92093-0112, USA
E-mail: sbuss@ucsd.edu

circuits. The easiest version of their representation applies to prime moduli m and states that any $\mathrm{AC}^0[m]$ circuit can be converted into a quasipolynomial size, depth three circuit with the bottom (input) gates being polylogarithmic fanin $\wedge$ gates, the middle level being MOD_m gates, and the top level containing a symmetric gate as the output gate. A closely related, alternate representation of $\mathrm{AC}^0[m]$ uses depth four circuits, by replacing the symmetric gate with a depth two layer of $\wedge/\vee$ gates that perform approximate counting. The strongest version of the representation applies to arbitrary $m \geq 2$; it eliminates the MOD_m gates, and converts any ACC circuit into a quasipolynomial size, depth two circuit with the bottom gates being polylogarithmic fanin $\wedge$ gates, and the top gate a symmetric function.

These ACC representations were inspired by an important theorem of Toda [31] on the containment of the polynomial time hierarchy (PH) in P^{PP}. It was quickly realized that the Furst-Saxe-Sipser [13] method (or, equivalently, the Paris-Wilkie method [23]) of translating PH predicates into circuits means that Toda's theorem implies results about bounded depth circuits. The first step towards the ACC representations was by Allender [1], who translated Toda's result into the setting of AC^0-circuits. After this, Yao [35] and Beigel-Tarui [7] formulated their representations for ACC. Further refinements were made by a number of researchers, including [3,4,8,14,18,30,32]. See also [29, 15] for constructions expressing conjunctions and disjunctions with constant-depth modular counting circuits. In addition, see [11] for improvements in circuit sizes.

We presume the reader is familiar with Boolean complexity, but as a reminder, for fixed $m > 1$, $\mathrm{AC}^0[m]$ is the class of Boolean predicates which can be computed by constant depth circuits using negation gates and unbounded fanin $\wedge$, $\vee$, and MOD m gates. ACC is the union of the classes $\mathrm{AC}^0[m]$ for $m > 1$. The *size* of a circuit is the number of wires in the circuit. For more information, the reader may consult the papers cited in the previous paragraph, or the textbook [5].

We next state two forms of the Beigel-Tarui theorem. The first version (Theorem 1) is equivalent to the characterization of Razborov and Smolensky [24,28], showing that for prime m, any $\mathrm{AC}^0[m]$ predicate can be represented by a probabilistic low degree polynomial over $\mathbb{F}_m$. At the end of the paper, we state a weaker form of Theorem 1 which has already found applications via bounded arithmetic to propositional proof complexity [10].

Theorem 1 *Fix $d \geq 1$ and a prime $m \geq 2$. Suppose C is a depth d ACC circuit of size S which uses MOD m gates. (That is, C is an $\mathrm{AC}^0[m]$ circuit.) Then there is an equivalent circuit C' of size $2^{(\log S)^{O(1)}}$ which has depth three. The first (input) level of C' contains $\wedge$ gates of fanin $(\log S)^{O(1)}$, the middle level contains MOD m gates, and the output gate is a symmetric gate. The output gate can be taken to be an approximate majority gate.*

Theorem 2 *Fix d and $m \geq 2$. Suppose C is a depth d ACC circuit of size S which uses MOD m gates. Then there is an equivalent circuit C' of* *size $2^{(\log S)^{O(1)}}$ of depth two. The first level of C' contains $\wedge$ gates of fanin $(\log S)^{O(1)}$ and the output gate is a symmetric gate.*

The constants implicit in the $O(1)$ terms in the bounds on size and fanin are independent of S, but do depend on d and m. Note that C' is quasipolynomial size, not polynomial, size. This is the usual situation for the Paris-Wilkie and Furst-Saxe-Sipser translations. Indeed, in the setting of circuit complexity, Barrington [6] argues that quasipolynomial size is more natural than polynomial size. Likewise, in the setting of bounded arithmetic, it is widely recognized that $\mathrm{I}\Delta_0 + \Omega_1$ and the S^i_2 and T^i_2 fragments of bounded arithmetic correspond to bounded depth quasipolynomial size circuits. (Paris and Wilkie [23], however, worked with the fragment $\mathrm{I}\Delta_0$, and obtained polynomial size circuits.)

The output symmetric gate for Theorem 2 can be expressed as an approximate majority of iterated modular counting operators. For this, see Theorem 31 and Section 4.

An important aspect of the Beigel-Tarui theorem is that the circuit C' can be described uniformly in terms of the circuit C. Suppose uniform circuits are represented with the "direct connection" representation in which there is an algorithm which runs in time logarithmic in the size of the circuit, and identifies indices i which are valid gate numbers, identifies the type of gate i, and computes the index of the j-th input to gate i. The Beigel-Tarui theorem holds in a strongly uniform way: namely, there is a quasilogarithmic time (w.l.o.g., logarithmic time in the size of C') algorithm, which, given access to the direct connection language for C, computes the direct connection language for C'.

This paper gives a new proof of this strongly uniform version of the Beigel-Tarui theorem below. It should be noted there are already several versions of this result, including Barrington [6] and Allender-Gore [3] and more recently Williams [34]. The novel aspects of our proof are threefold. First, we take a detour through formulas in the modular polynomial time hierarchy ModPH [2,14]. The ModPH formulas are defined to use modular counting quantifiers over any prime modulus m. Counting over composite moduli m can be simulated using by using multiple counting quantifiers with prime moduli. This means that ModPH is a uniform version of ACC. At the same time, we consider the modular polynomial time hierarchy ModPH_m in which only the fixed prime value m may be used for modular counting quantifiers. ModPH_m is a uniform version of $\mathrm{AC}^0[m]$. Second, we do not directly prove the Beigel-Tarui theorem, nor do our main proofs deal with circuits. Instead, we use uniform descriptions of ModPH and ModPH_m predicates in terms of *formulas.* We deal with subclasses, $\oplus$-ptime and $1\oplus_m$-ptime, of formulas which are expressed in a prenex form with modular counting quantifiers, but no universal or existential quantifiers. We also consider probabilistic versions of these classes, called BP$\cdot\oplus$-ptime and BP$\cdot 1\oplus_m$-ptime. Our main theorems prove closure properties for these classes of formulas. Third, all our results about formula classes relativize. Using these relativized results, we obtain the uniform Beigel-Tarui theorems in the forms of Theorems 1 and 2 as corollaries via the Furst-Saxe-

Sipser and Paris-Wilkie translation. Prior proofs all used delicate, local transformations of circuits to obtain uniformity; our proof avoids these in favor of manipulations of ModPH and ModPH_m formulas.

Our underlying proof techniques are based closely on the earlier proofs of Toda's theorem and the Beigel-Tarui theorem. The reader can consult Arora and Barak [5] for the standard version of the proof of Toda's theorem. Another simple proof has been given by [12]. In particular, our proof still uses the Valiant-Vazirani theorem. Similarly, we still use the modulus amplifying polynomials of Toda, Yao, and Beigel-Tarui. Our use of the Paris-Wilkie, Furst-Saxe-Sipser translation is also standard; see for instance [19,10].

Chen and Papakonstantinou [11] have recently given improved size bounds, showing in particular that the dependence of the "$O(1)$" exponents in Theorems 1 and 2 can be reduced to only $O(d)$. The present paper does not examine dependence on the depth, but our constructions do not achieve bounds as small as those of [11]. It would be interesting to investigate whether they can be improved. It is a bit surprising where the growth rates appear in our proofs: the quasipolynomial growth rate of Theorems 1 and 2 arises from the conversion of ModPH formulas to BP$\cdot\oplus$-ptime formulas; however, there is only polynomial size increase in going from BP$\cdot\oplus$-ptime formulas to the representation of Theorem 2.

Our motivations arise from expressibility and provability in bounded arithmetic, notably how results in computational complexity relate to bounded arithmetic. The modular polynomial time hierarchy ModPH is equivalent to expressibility in the language Δ_0 of bounded arithmetic augmented with modular counting quantifiers and the smash function #. However, we do not use any constructions from bounded arithmetic. Provability in bounded arithmetic is the topic of another paper, joint by the present author, Leszek Kołodziejczyk, and Konrad Zdanowski [10], in which Toda's theorem is proved within a fragment of bounded arithmetic.

We thank Eric Allender, Leszek Kołodziejczyk, and Ryan Williams for helpful comments and discussions. We also thank the anonymous referee for substantial helpful comments and especially for finding mistakes in the proofs in the original draft of the paper.

2 Modular counting and probabilistic classes

Section 2.1 defines a version of the polynomial hierarchy augmented with modular counting quantifiers, called the "modular polynomial time hierarchy" ModPH. This hierarchy is well-known, see [2,14] for instance. Section 2.1 also defines two prenex subclasses of the ModPH predicates called the $\oplus$-ptime predicates and the $1\oplus_m$-ptime predicates. It then proves some initial closure properties for the $\oplus$-ptime and $1\oplus_m$-ptime predicates. Section 2.2 shows how numbers of satisfying assignments can be amplified by polynomials. Section 2.3 defines general symmetric quantifiers, in particular, quantifiers which combine iterated modular counting with approximate majority. Section 2.4 defines probabilistic versions of the classes of $\oplus$-ptime and $1\oplus_m$-ptime predicates.

2.1 The modular polynomial time hierarchy

The predicates in ModPH are defined from polynomial time predicates using bounded existential and universal quantifiers, and bounded modular counting quantifiers. The bounded modular counting quantifiers have the form

$$(\bigoplus_m x, |x|{=}p(|\mathbf{z}|))\varphi(x,\mathbf{z})$$

where $m \geq 2$, and $p(\mathbf{n})$ is a polynomial. By "polynomial", we mean a polynomial with non-negative integer coefficients. Variables such as $x, z_1, \ldots, z_\ell$ range over $\{0,1\}^*$; we write $|x|$ for the length of x, and $\mathbf{z}$ for $z_1, \ldots, z_\ell$ and $|\mathbf{z}|$ for $|z_1|, \ldots, |z_\ell|$. The intended meaning of the quantifier is that the number of values x such that $|x| = p(|\mathbf{z}|)$ and $\varphi(x)$ holds is congruent to zero modulo m. The *relativized* ModPH hierarchy, denoted ModPH^Ω, also allows a unary predicate Ω to be used as part of φ. The predicate Ω is the same thing as an oracle, $\Omega \subseteq \{0,1\}^*$.

Definition 3 The *(relativized) modular polynomial time hierarchy* is defined as the following set of predicates:

a. Every polynomial time predicate $\varphi(\mathbf{x})$ is in ModPH. Every predicate in P^Ω, namely every predicate $\varphi(\mathbf{x})$ which is polynomial time relative to Ω, is in ModPH^Ω.
b. If $\varphi(x,\mathbf{z})$ is in ModPH or ModPH^Ω, then so are $(\exists x, |x|{=}p(|\mathbf{z}|))\varphi(x,\mathbf{z})$ and $(\forall x, |x|{=}p(|\mathbf{z}|))\varphi(x,\mathbf{z})$.
c. Let m be prime. If $\varphi(x,\mathbf{z})$ is in ModPH or ModPH^Ω, then so is the predicate $(\bigoplus_m x, |x|{=}p(|\mathbf{z}|))\varphi(x,\mathbf{z})$.

Now fix a prime m. ModPH_m and ModPH_m^Ω are the predicates in ModPH and ModPH^Ω (respectively) which are defined as above, but requiring the fixed value m as the modulus for the counting quantifiers.

Definition 3 requires all ModPH formulas to be expressed in prenex form. Nonetheless, ModPH is closed under conjunction, disjunction and negation (that is, under intersection, union, and complementation); for this see Theorem 17. Note that a ModPH formula may use multiple values for m in modular counting quantifiers; however, we require the moduli m be prime for convenience in our normal forms for ModPH. Nonetheless, as is well-known, ModPH can simulate modular counting over composite moduli m: for this, see Theorem 8(iii) and Lemma 16.

Since all quantifiers have the form $(Qx, |x|{=}p(|\mathbf{z}|))$, every variable appearing in a ModPH or ModPH_m has a known length. Indeed, we assume that in a formula $\varphi(\mathbf{z})$, any quantifier has the form $(Qx, |x|{=}p(|\mathbf{z}|))$. That is, the quantifier bounds may w.l.o.g. depend on only the variables that occur free in the formula and not on any of the other quantified variables. (For more on this, see Lemma 11.)

Working with fixed length strings makes Gödel sequence coding very simple; namely, a sequence of binary strings is encoded by concatenating them into single string. Specifically, if $y_1, \ldots, y_\ell$ are binary strings of length $p(\mathbf{n})$, then $y = \langle y_1, \ldots, y_\ell \rangle$ denotes the string of length $\ell \cdot p(\mathbf{n})$ obtained by concatenating $y_1, \ldots, y_\ell$. We write $(y)_i$ for y_i. In addition, for $x \in \{0,1\}^*$, we write $x[0]$ for the first bit of x, and $x[i{:}j]$ for the i-th through the j-th bits of x, a substring of length $j - i + 1$. Finally, we write $x[i{:}]$ for $x[i{:}|x|{-}1]$.

The next definitions give restricted subclasses of prenex formulas in which there are no bounded existential or universal quantifiers at all. All our definitions and results relativize, so we suppress mention of the Ω in the notation.

Definition 4 The *⊕-ptime* predicates are a subset of the ModPH (respectively, ModPH^{Ω}) predicates, and are defined inductively by

a. Any polynomial time predicate (respectively, predicate in P^{Ω}) is a ⊕-ptime predicate.
b. If $\varphi(x, \mathbf{z})$ is a ⊕-ptime formula, p is a polynomial, and $m \geq 2$ is a prime, then $(\bigoplus_m x, |x| {=} p(|\mathbf{z}|))\varphi(x, \mathbf{z})$ is also a ⊕-ptime predicate.

A ⊕-ptime predicate must be in prenex form, and may have multiple modular counting quantifiers; the quantifiers may use different (prime) moduli m. The *rank* of a ⊕-ptime predicate φ is the number of modular quantifiers in the expression defining the predicate. The *matrix* of φ is the subformula which is inside the modular counting quantifiers. Thus, a rank r ⊕-ptime predicate consists of the r modular quantifiers in front of the matrix.

The "$1\oplus_m$" subclass is restricted to have a single modular quantifier, that is, to have rank ≤ 1. The class $1\oplus_2$-ptime is the same class as ⊕P, defined by [22]:

Definition 5 Let m be prime. The $1\oplus_m$-ptime predicates are the ModPH_m (respectively, $\mathrm{ModPH}_m^{\Omega}$) predicates which can be expressed in the form

$$(\bigoplus_m x, |x| {=} p(|\mathbf{z}|))\varphi(x, \mathbf{z})$$

where p is a polynomial, and $\varphi(x, \mathbf{z})$ is polynomial time computable (respectively, polynomial time relative to Ω).

The next two theorems state closure properties for the classes of ⊕-ptime predicates and $1\oplus_m$-ptime formulas.

Definition 6 A *generalized modular counting quantifier* is a quantifier of the form $(\bigoplus_m^i x, |x| {=} p(|\mathbf{z}|))$. The intended meaning of

$$(\bigoplus_m^i x, |x| {=} p(|\mathbf{z}|))\varphi(x, \mathbf{z})$$

is that the number of x's such that $|x| = p(|\mathbf{z}|)$ and $\varphi(x, \mathbf{z})$ holds is congruent to i mod m. Thus $\bigoplus_m^0$ is the same as $\bigoplus_m$.

The next definition is essentially equivalent to the notion of sharply bounded quantifiers used in bounded arithmetic [9].

Definition 7 Let $x \in \{0,1\}^*$ and $i \in \mathbb{N}$. We write, respectively, $x \prec i$, $x \preccurlyeq i$ or $x \approx i$ to mean that x is the binary representation (possibly with leading zeros) of an integer j_x such that, $j_x < i$, $j_x \le i$, or $j_x = i$. A *sharply bounded* quantifier is a quantifier of the form $(Qx{\prec}p(|\mathbf{z}|))$ for p a polynomial. The intended meaning of $(\exists x{\prec}p(|\mathbf{z}|))\varphi(x,\mathbf{z})$ is the same as

$$(\exists x, |x|{=}p(|\mathbf{z}|))(x{\prec}p(|\mathbf{z}|) \land \varphi(x,\mathbf{z})).$$

The meanings of other bounded quantifiers are defined similarly.

It is somewhat wasteful of bits that we use $p(\mathbf{n})$ bits for x to represent a number $< p(\mathbf{n})$; but there is no harm is doing this, and it is convenient to have the convention that variables range over binary strings of known polynomial length.

We will not continue to mention it explicitly, but all the results below and throughout Section 3 relativize to the presence of an oracle Ω.

Theorem 8 *The class of $\oplus$-ptime predicates is closed under:*

(i) *Conjunction, disjunction, and negation.*
(ii) *Sharply bounded existential and universal quantification.*
(iii) *Generalized modular counting quantification of the form $(\bigoplus_m^i x, |x|{=}p(|\mathbf{z}|))$, for arbitrary $m \ge 2$ (possibly m is composite).*

Theorem 9 *Let m be a fixed prime. The class of $1{\oplus_m}$-ptime predicates is closed under:*

(i) *Conjunction, disjunction, and negation.*
(ii) *Sharply bounded existential and universal quantification.*
(iii) *Generalized modular counting quantification of the form $(\bigoplus_m^i x, |x|{=}p(|\mathbf{z}|))$.*

Theorems 8 and 9 are established by Lemmas 10-16. See Theorem 18 for another closure property.

Lemma 10 *The classes $\oplus$-ptime and $1{\oplus_m}$-ptime are closed under taking a disjunction or conjunction with a polynomial time predicate. These operations preserve rank: Suppose $\chi(\mathbf{z})$ is polynomial time and $\varphi(\mathbf{z})$ is a $\oplus$-ptime or $1{\oplus_m}$-ptime predicate of rank r and let $\gamma(\mathbf{y},\mathbf{z})$ be the matrix of $\varphi(\mathbf{z})$. Then $\varphi(\mathbf{z}) \lor \chi(\mathbf{z})$ and $\varphi(\mathbf{z}) \land \chi(\mathbf{z})$ are expressible with formulas of rank r. Indeed these formulas have exactly the same modular quantifiers as $\varphi(\mathbf{z})$; and they have a matrix which is expressed as a Boolean combination of $\chi(\mathbf{z})$, of $\gamma(\mathbf{y},\mathbf{z})$ and of polynomial time predicates.*

The lemma also holds in relativized form; for instance if $\chi(\mathbf{z})$ is polynomial time relative to an oracle.

Proof We use induction on the rank of formulas. The base case of rank zero (no modular quantifiers) is trivial. For the induction step, suppose, a $\oplus$-ptime (respectively, $1\oplus_m$-ptime) predicate $\varphi(\mathbf{z})$ is defined by the formula

$$(\textstyle\bigoplus_m y, |y|{=}p(|\mathbf{z}|))\psi(y, \mathbf{z}), \tag{1}$$

where m is prime. Let $\chi(\mathbf{z})$ be a polynomial time predicate. The disjunction $\varphi(\mathbf{z}) \vee \chi(\mathbf{z})$ is equivalent to

$$(\textstyle\bigoplus_m y, |y|{=}p(|\mathbf{z}|))[\psi(y, \mathbf{z}) \wedge \neg\chi(\mathbf{z})],$$

and $\varphi(\mathbf{z}) \wedge \chi(\mathbf{z})$ is equivalent to

$$(\textstyle\bigoplus_m y, |y|{=}p(|\mathbf{z}|))[(\psi(y, \mathbf{z}) \wedge \chi(\mathbf{z})) \vee (y{\approx}0 \wedge \neg\chi(\mathbf{z}))].$$

The induction hypothesis applied to the rank $r{-}1$ formula ψ shows that these two formulas are both rank r. Note that the construction does not change the modular quantifiers. Likewise, the assertions about the matrices follow immediately from the inductive construction. □

Lemma 11 *The classes $\oplus$-ptime and $1\oplus_m$-ptime are closed under substitution by polynomial time functions. If $\varphi(z_0, \mathbf{z})$ is a $\oplus$-ptime or $1\oplus_m$-ptime predicate of rank r and $f(\mathbf{x})$ is polynomial time, then $\varphi(f(\mathbf{z}), \mathbf{z})$ is also expressible with a formula of rank r.*

Proof The only reason that $\varphi(f(\mathbf{z}), \mathbf{z})$ may fail to be a $1\oplus_m$-ptime predicate of rank r is that it may contain quantifiers of the form $(\bigoplus_m x, |x|{=}p(|f(\mathbf{z})|, |\mathbf{z}|))$, which is not permitted by our syntax. Let $p_f(\mathbf{n})$ be a polynomial such that $|f(\mathbf{z})| < p_f(|\mathbf{z}|)$ always holds. Then, any subformula of the form

$$(\textstyle\bigoplus_m x, |x|{=}p(|f(\mathbf{z})|, |\mathbf{z}|))\psi(x, \mathbf{z}, \ldots)$$

can be equivalently expressed as

$$(\textstyle\bigoplus_m x, |x|{=}p(p_f(|\mathbf{z}|), |\mathbf{z}|))[\, x[|f(\mathbf{z})|{:}]{\approx}0 \wedge \psi(x[0{:}|f(\mathbf{z})|{-}1], \mathbf{z}, \ldots)\,].$$

Since the predicate "$x[|f(\mathbf{z})|{:}]{\approx}0$" is polynomial time, Lemma 11 follows from Lemma 10. □

Lemma 12 *The classes $\oplus$-ptime and $1\oplus_m$-ptime are closed under negation and sharply bounded quantification, as well as biimplication (equivalence) with a polynomial time predicate. Furthermore, these operations preserve rank:*

(i) *If $\varphi(\mathbf{z})$ is a $\oplus$-ptime or $1\oplus_m$-ptime predicate of rank r, then $\neg\varphi(\mathbf{z})$ is expressible with a formula of rank r.*
(ii) *If in addition, $\chi(\mathbf{z})$ is a polynomial time predicate, then $\varphi(\mathbf{z}) \leftrightarrow \chi(\mathbf{z})$ is expressible with a formula of rank r.*
(iii) *If $\varphi(x, \mathbf{z})$ is a $\oplus$-ptime or $1\oplus_m$-ptime predicate of rank r, then the predicates $(\exists x{\prec}q(|\mathbf{z}|))\varphi(x, \mathbf{z})$ and $(\forall x{\prec}q(|\mathbf{z}|))\varphi(x, \mathbf{z})$ are expressible with formulas of rank r.*

Proof We prove (i)-(iii) simultaneously by induction on rank. The base case is clear. For closure under negation, suppose $\varphi(\mathbf{z})$ is $(\bigoplus_m y, |y|{=}p(|\mathbf{z}|))\psi(y,\mathbf{z})$. We use the variable w to quantify over binary strings of length $(m{-}1)p(|\mathbf{z}|)$ which encode a sequence of $m{-}1$ values of y, and let $\delta(w,\mathbf{z})$ be the formula

$$(\forall x{\prec}m{-}1)\psi((w)_{j_x},\mathbf{z}),$$

where j_x denotes the integer with binary representation given by x. If there are j many y's satisfying $\psi(y,\mathbf{z})$, then there are j^{m-1} many w's satisfying δ. By Fermat's little theorem, $j^{m-1} \bmod m \in \{0,1\}$, and is congruent to zero iff $j \equiv 0 \pmod{m}$. Therefore, $\varphi(\mathbf{z})$ is equivalent to

$$(\textstyle\bigoplus_m w, |w|{=}(m{-}1)p(|\mathbf{z}|))\delta(w,\mathbf{z}).$$

By adding $m{-}1$ many satisfying values to $\delta(-,-)$ we can make the number of satisfying values be $j^{m-1}{+}m{-}1$. This is done by letting w encode a one bit flag, followed by $m{-}1$ values for y as follows:

$$(\textstyle\bigoplus_m w, |w|{=}1{+}(m{-}1)p(|\mathbf{z}|))[(w[0]{=}1 \wedge \delta(w[1{:}],\mathbf{z})) \vee w{\prec}m{-}1]. \tag{2}$$

By inspection, (2) is equivalent to $\neg\varphi(\mathbf{z})$. Since the rank of ψ is less than the rank r of φ, the induction hypothesis and Lemma 11 imply that δ is also expressible with rank $< r$. Lemma 10 then implies that (2) can be expressed as a formula of rank r. This establishes closure under negation.

Closure under biimplication can be viewed as a closure under "conditional negation"; namely, the predicate is negated iff $\chi(\mathbf{z})$ is false. The proof for (ii) is identical to the proof of (i), except that the formula (2) is replaced with

$$(\textstyle\bigoplus_m w, |w|{=}1{+}(m{-}1)p(|\mathbf{z}|))[(w[0]{=}1 \wedge \delta(w[1{:}],\mathbf{z})) \vee (w{\prec}m{-}1 \wedge \neg\chi(\mathbf{z}))].$$

To handle sharply bounded existential quantification, suppose $\varphi(x,\mathbf{z})$ is $(\bigoplus_m y, |y|{=}p(|\mathbf{z}|))\psi(x,y,\mathbf{z})$. Then, using w to encode $q(|\mathbf{z}|)$ many values for y, we claim that $(\exists x{\prec}q(|\mathbf{z}|))\varphi(x,y,\mathbf{z})$ is equivalent to

$$(\textstyle\bigoplus_m w, |w|{=}p(|\mathbf{z}|)q(|\mathbf{z}|))(\forall x{\prec}q(|\mathbf{z}|))\psi(x,(w)_{j_x},\mathbf{z}). \tag{3}$$

To see this, note that the number of w's satisfying the formula is the product of the numbers of y's satisfying $\psi(x,y,\mathbf{z})$ for $x \prec q(|\mathbf{z}|)$. Since m is prime, this product is congruent to zero mod m iff one of its factors is. Since the rank of ψ is $< r$ by Lemma 11, the induction hypothesis applies to $(\forall x{\prec}q(|\mathbf{z}|))\psi$, and thus (3) is equivalent to a rank r formula. This proves the closure under sharply bounded existential quantification.

For sharply bounded universal quantification, note that $(\forall x{\prec}q(|\mathbf{z}|))\varphi(x,\mathbf{z})$ is equivalent to $\neg(\exists x{\prec}q(|\mathbf{z}|))\neg\varphi(x,\mathbf{z})$. Since we already proved the rank r formulas are closed under negation and sharply bounded existential quantification, it follows that $(\forall x{\prec}q(|\mathbf{z}|)|)\varphi(x,\mathbf{z})$ is equivalent to a rank r formula. □

Lemma 13 *Let $m > 1$ and let Γ be a class closed under conjunction and disjunction with polynomial time predicates and under substitution with polynomial time functions. (E.g., Γ is the rank r $\oplus$-ptime or the $1{\oplus_m}$-ptime predicates.) Suppose also that for all $\psi(y,\mathbf{z})$ in Γ, $(\bigoplus_m y, |y|{=}p(|\mathbf{z}|))\psi(y,\mathbf{z})$ is in Γ. Then, for all $\psi(y,\mathbf{z})$ in Γ and all $i < m$, $(\bigoplus_m^i y, |y|{=}p(|\mathbf{z}|))\psi(y,\mathbf{z})$ is in Γ.*

Proof The idea is to add $m{-}i$ many satisfying values for the modular quantifier, similar to the construction used for (2). Namely, $(\bigoplus_m^i y, |y|{=}p(|\mathbf{z}|))\psi(y,\mathbf{z})$ is equivalent to

$$(\bigoplus\nolimits_m w, |w|{=}1{+}p(|\mathbf{z}|))[(w[0]{=}1 \wedge \psi(w[1{:}],\mathbf{z})) \vee w{\prec}m{-}i].$$

By the hypotheses, this is a Γ predicate. □

Lemma 14 *If $\varphi(x,\mathbf{z})$ is an $1{\oplus_m}$-ptime predicate, then so is the predicate $(\bigoplus_m^i x{\prec}q(|\mathbf{z}|))\varphi(x,\mathbf{z})$. When $i = 0$, this means $1{\oplus_m}$-ptime is closed under $\oplus_m$ bounded quantification.*

Proof Suppose $\varphi(x,\mathbf{z})$ is $(\bigoplus_m y, |y|{=}p(|\mathbf{z}|))\psi(x,y,\mathbf{z})$, where ψ is polynomial time. By Lemma 13, it suffices to show

$$(\bigoplus\nolimits_m x, |x|{=}q(|\mathbf{z}|))(\bigoplus\nolimits_m y, |y|{=}p(|\mathbf{z}|))\psi(x,y,\mathbf{z}) \tag{4}$$

is in $1{\oplus_m}$-ptime (that is, with $i = 0$). Arguing similarly to the proof of Lemma 12, let $\delta(x,w,\mathbf{z})$ be $(\forall u{\prec}m{-}1)\psi(x,(w)_{j_u},\mathbf{z})$. Consider the formula $\delta^*(x,w,\mathbf{z})$ defined as

$$(w[0]{=}1 \wedge \delta(x,w[1{:}],\mathbf{z})) \vee w{\prec}m{-}1. \tag{5}$$

By Fermat's little theorem, the number of w's of length $1{+}(m{-}1)p(|\mathbf{z}|)$ which satisfy (5) equals either -1 or 0 mod m depending on whether $\varphi(x,\mathbf{z})$ is true or false, respectively. Let $r(\mathbf{n})$ equal $q(\mathbf{n}){+}1{+}(m{-}1)p(\mathbf{n})$, and let a binary string v of length $r(|\mathbf{z}|)$ encode a value for x with its first $q(|\mathbf{z}|)$ bits, and a value for w with its remaining bits. Then (4) is equivalent to

$$(\bigoplus\nolimits_m v, |v|{=}r(|\mathbf{z}|))\delta^*(v[0{:}q(|\mathbf{z}|){-}1], v[q(|\mathbf{z}|){:}],\mathbf{z}). \tag{6}$$

Note that δ and δ^* are polynomial time, since ψ is. Therefore, (6) is in $1{\oplus_m}$-ptime as desired. □

Lemma 15 *The classes $\oplus$-ptime and $1{\oplus_m}$-ptime are closed under disjunction and conjunction.*

Proof With closure under negation (Lemma 12), it suffices to show the classes are closed under disjunction. Let $\varphi(\mathbf{z})$ and $\psi(\mathbf{z})$ both be $\oplus$-ptime predicates or $1{\oplus_m}$-ptime predicates. Let $\gamma(\mathbf{y},\mathbf{z})$ be the (polynomial time) matrix of φ. The (relativized version of) Lemma 10 implies that $\varphi(\mathbf{z}) \vee \psi(\mathbf{z})$ is equivalent to a formula $\chi(\mathbf{z})$ which consists of the modular quantifiers of φ in front of a formula $\gamma^*(\mathbf{y},\mathbf{z})$, where $\gamma^*(\mathbf{y},\mathbf{z})$ is a Boolean combination of $\psi(\mathbf{z})$ and polynomial time predicates (since γ is polynomial time). Therefore, γ^* can be written in the form

$$[\gamma_1(\mathbf{y},\mathbf{z}) \wedge \psi(\mathbf{z})] \vee [\gamma_2(\mathbf{y},\mathbf{z}) \wedge \neg\psi(\mathbf{z})]$$

where γ_1 and γ_2 are polynomial time, and then equivalently in the form

$$(\gamma_1(\mathbf{y},\mathbf{z}) \wedge \gamma_2(\mathbf{y},\mathbf{z})) \vee [(\gamma_1(\mathbf{y},\mathbf{z}) \vee \gamma_2(\mathbf{y},\mathbf{z})) \wedge (\psi(\mathbf{z}) \leftrightarrow \gamma_1(\mathbf{y},\mathbf{z}))].$$

By Lemmas 10 and 12, this last formula is a $\oplus$-ptime predicate or a $1\oplus_m$-ptime predicate since $\psi(\mathbf{z})$ is (respectively).

For $\varphi(\mathbf{z})$ and $\psi(\mathbf{z})$ $\oplus$-ptime predicates, it follows from the definition of $\oplus$-ptime that $\chi(\mathbf{z})$ is expressible as a $\oplus$-ptime predicate. For $\varphi(\mathbf{z})$ and $\psi(\mathbf{z})$ $1\oplus_m$-ptime predicates, Lemma 14 implies that $\chi(\mathbf{z})$ is also equivalent to a $1\oplus_m$-ptime predicate. □

Lemma 16 *Let $m \geq 2$ and $0 \leq i < m$, where m is not necessarily prime. If $\varphi(y,\mathbf{z})$ is a $\oplus$-ptime predicate, then so is $(\bigoplus_m^i y, |y|{=}q(|\mathbf{z}|))\psi(y,\mathbf{z})$.*

Proof The proof of Lemma 13 still applies for composite m, so it suffices to assume $i = 0$. Thus, we need to show that $(\bigoplus_m y, |y|{=}q(|\mathbf{z}|))\psi(y,\mathbf{z})$ is in $\oplus$-ptime. If m is prime, this is already in $\oplus$-ptime. Otherwise, we use induction on m. Let $m = m'p$ for p a prime. Consider the predicate

$$\begin{aligned}&(\bigoplus_{m'} u, |u|{=}q(|\mathbf{z}|))[\varphi(u,\mathbf{z}) \wedge (\bigoplus_p x, |x|{=}q(|\mathbf{z}|))(x{\preceq}u \wedge \varphi(x,\mathbf{z}))]\\ &\quad \wedge (\bigoplus_p x, |x|{=}q(|\mathbf{z}|))\varphi(x,\mathbf{z}).\end{aligned} \tag{7}$$

The subformula in square brackets is a $\oplus$-ptime predicate by closure under conjuction (Lemmas 10 and especially 15). Thus, the whole formula is also in $\oplus$-ptime by the induction hypothesis (since $m' < m$) and by Lemma 15 again. The subformula in square brackets picks out every p-th x satisfying ψ; thus (7) is equivalent to $(\bigoplus_m y, |y|{=}q(|\mathbf{z}|))\psi(y,\mathbf{z})$ as desired. □

Theorems 8 and 9 follow immediately from Lemmas 10-16.

Theorem 17 *The classes* ModPH *and* ModPH$_m$ *are closed under negation, conjunction and disjunction.*

Proof (Sketch.) The proof is based on the methods of Lemmas 10, 12 and 15. The proofs are identical for ModPH and ModPH$_m$, so we only discuss ModPH. For this proof, redefine "rank" to mean the total number of modular, existential and universal quantifiers (not counting sharply bounded quantifiers appearing in the polynomial time matrix). The proof starts by establishing that ModPH is closed under disjunction and conjunction with a polynomial time predicate, and more generally, that the complete statement of Lemma 10 holds for ModPH (now using the generalized notion of rank). The proof of Lemma 10 still applies: the cases of universal and existential quantifiers are handled by prenex operations. Second, the proof shows that parts (i) and (iii) of Lemma 12 holds for ModPH. The proof of this again proceeds by induction on rank. For closure under negation, universal and existential quantifiers are handled by moving the negation sign past the quantifier. For closure under sharply bounded quantification, modular bounded quantifiers are handled as before, and the universal and existential bounded quantifiers are handled using the "replacement" principle from bounded arithmetic, namely that

$$(\forall x {\prec} q(|\mathbf{z}|))(\exists y, |y| {=} p(|\mathbf{z}|))\psi(x,y,\mathbf{z})$$
$$\leftrightarrow (\exists w, |w| {=} p(|\mathbf{z}|)q(|\mathbf{z}|))(\forall x {\prec} q(|\mathbf{z}|))\psi(x,(w)_{j_x},\mathbf{z}).$$

Third, the proof shows that if a ModPH predicate $\varphi(\mathbf{z})$ starts with a $\oplus_m$ bounded quantifier, and if $\chi(\mathbf{z})$ is a polynomial time predicate, then $\varphi(\mathbf{z}) \leftrightarrow \chi(\mathbf{z})$ is also a ModPH predicate. This is proved by exactly the same argument as was used for part (ii) of Lemma 12. (Note that there seems to be no such direct proof when φ starts with an existential or universal bounded quantifier.)

Finally, the proof shows that if $\varphi(\mathbf{z})$ and $\psi(\mathbf{z})$ are both in ModPH, then $\varphi(\mathbf{z}) \vee \psi(\mathbf{z})$ is in ModPH. This is was already proved if either formula is polynomial time. Otherwise, we use induction on rank. If either formula starts with an existential or universal quantifier, then a prenex operation can move that quantifier to the front. Thus, w.l.o.g., both $\varphi(\mathbf{z})$ and $\psi(\mathbf{z})$ start with $\oplus$ bounded quantifiers. But now, the proof of Lemma 15 applies, since the biimplication of a polynomial time predicate and the ModPH predicate $\psi(\mathbf{z})$ is a ModPH predicate. □

In the next theorem, $Q(\mathbf{z},\Omega)$ is a predicate computed by some Turing machine which has ordinary inputs $\mathbf{z}$ and oracle access to Ω, such that there is a polynomial $p(|\mathbf{z}|)$ which bounds the runtime of $Q(\mathbf{z},\Omega)$ for all $\mathbf{z}$ and Ω. Also, $\lambda x.\varphi$ denotes the oracle Ω such that $\Omega(x)$ returns the truth value of $\varphi(x)$.

Theorem 18 *Suppose the predicate $Q(\mathbf{z},\Phi)$ is a polynomial time relative to the oracle Φ, and that $\varphi(x)$ is a rank r $\oplus$-ptime or $1{\oplus_m}$-ptime predicate. Then $Q(\mathbf{z},\lambda x.\varphi)$ is also a rank r $\oplus$-ptime or $1{\oplus_m}$-ptime predicate, respectively.*

Proof We claim that we may assume w.l.o.g. that $Q(\mathbf{z},\Phi)$ makes only fixed length queries to the oracle Φ, namely that there is a polynomial $q(\mathbf{n})$ so that $Q(\mathbf{z},\Phi)$ only queries Φ about strings of length $q(|\mathbf{z}|)$. The reason this can be assumed w.l.o.g. is that otherwise, letting $q(|\mathbf{z}|)$ be be greater than the length of any query made to Φ, we can define a new oracle $\Psi(w)$ so that $\Psi(1^i0w)$ is equal to $\Phi(w)$ for all i and w. Oracle queries to Φ can be replaced with queries to Ψ; and if φ is $1{\oplus_m}$-ptime or rank r $\oplus$-ptime, so is a predicate ψ such that $\psi(1^i0w)$ equals $\varphi(w)$ for all w.

So assume $Q(\mathbf{z},\Phi)$ has runtime $< p(|\mathbf{z}|)$ and only makes queries of length $q(|\mathbf{z}|)$ to Φ. Using standard methods of encoding computations of $Q(\mathbf{z},\Phi)$, there is a polynomial $r(\mathbf{n})$ so that any computation of $Q(\mathbf{z},\Phi)$ can be coded uniquely by a binary string w of length $r(|\mathbf{z}|)$. Furthermore, there are polynomial time functions $\textsc{Accept}(\mathbf{z},w)$, $\textsc{Query}(i,\mathbf{z},w)$ and $\textsc{Answer}(i,\mathbf{z},w)$ such that:

(1) $\textsc{Accept}(\mathbf{z}, w)$ determines whether w correctly encodes an accepting computation of $Q(\mathbf{z}, \Phi)$ assuming the oracle answers as encoded in w are correct.
(2) $\textsc{Query}(i, \mathbf{z}, w)$ is the query to Φ made in the i-th step of the computation coded by w, if any. Thus $\textsc{Query}(i, \mathbf{z}, w)$ is a string of length $q(|\mathbf{z}|)$.
(3) $\textsc{Answer}(i, \mathbf{z}, w)$ is equal to 0 if no query was made in the i-th step, and otherwise is equal to 1 or 2 depending on whether the query answer was "Yes" or "No" (respectively).

Fix m such that the outermost quantifier of φ is a $\oplus_m$-quantifier. Since w is unique, the condition $Q(\mathbf{z}, \lambda x.\varphi)$ can now be expressed as

$$\begin{aligned}(\textstyle\bigoplus^1_m w, |w|{=}r(|\mathbf{z}|))[\textsc{Accept}(\mathbf{z}, w)\wedge \\ (\forall i{\prec}p(|\mathbf{z}|))(\textsc{Answer}(i, \mathbf{z}, w){=}0 \vee \\ (\textsc{Answer}(i, \mathbf{z}, w){=}1 \leftrightarrow \varphi(\textsc{Query}(i, \mathbf{z}, w))))\,].\end{aligned} \tag{8}$$

By Lemmas 10-12, the subformula in square brackets is a rank r $\oplus$-ptime or $1\oplus_m$-ptime predicate, the same as φ. Thus, by (the proof of) Lemma 14, (8) is also a rank r $\oplus$-ptime or $1\oplus_m$-ptime predicate, respectively. □

2.2 Closing satisfiability counts under polynomials

We write $(\#x, |x|{=}p(|\mathbf{z}|))\varphi(x, \mathbf{z})$ for the number of x's such that $|x| = p(|\mathbf{z}|)$ and $\varphi(x, \mathbf{z})$ holds. This section shows how to manipulate the number of satisfying assignments by transforming them by a polynomial. Proposition 19 does this for a fixed polynomial r. Proposition 20 does this for a varying polynomial r, which is given as part of the input. These constructions will be applied to modulus amplifying polynomials.

We have already a couple basic techniques for manipulating numbers of satisfying assignments. The proofs of Lemmas 12 and 14 showed how to change the number of satisfying assignments from j to j^{m-1} so as to apply Fermat's little theorem. Exponents other than $m{-}1$, can used; in fact, even changing j satisfying assignments to $j^{q(|\mathbf{z}|)}$ can be done, for any polynomial $q(\mathbf{n})$. Lemmas 12-14 showed how to introduce a fixed number of satisfying assignments: those lemmas added either $m{-}1$ or $m{-}i$ many satisfying assignments, but the same technique works to add an arbitrary number of satisfying assignments. As will be seen in the proof of Proposition 19 it is not hard to extend these techniques to take sums or products of numbers of satisfying assignments.

Proposition 19 *Let Γ be the class of $\oplus$-ptime predicates, the class of $1\oplus_m$-ptime predicates, or the class of polynomial time predicates. Let $\varphi(x, \mathbf{z})$ be a Γ predicate, and $p(|\mathbf{z}|)$ be a polynomial. Further let $r(n)$ be a polynomial (as always, with non-negative integer coefficients). Then there is a Γ predicate $\chi(w, \mathbf{z})$ and a polynomial $q(|\mathbf{z}|)$ so that*

$$(\#w, |w|{=}q(|\mathbf{z}|))\chi(w, \mathbf{z}) \;=\; r(\,(\#x, |x|{=}p(|\mathbf{z}|))\varphi(x, \mathbf{z})\,).$$

Proof It suffices to show that it is possible to take the sum or the product of numbers of satisfying values. Let N_i equal $(\#x{,}|x|{=}p_i(|\mathbf{z}|))\varphi_i(x,\mathbf{z})$, for $i = 1, 2$. To handle summation, define $\chi(w,\mathbf{z})$ to be

$$\begin{aligned}&[w[0] = 0 \land w[1{:}p_2(|\mathbf{z}|)]{\approx}0 \land \varphi_1(w[p_2(|\mathbf{z}|){+}1{:}], \mathbf{z})]\\ &\lor\ [w[0] = 1 \land w[1{:}p_1(|\mathbf{z}|)]{\approx}0 \land \varphi_2(w[p_1(|\mathbf{z}|){+}1{:}], \mathbf{z})].\end{aligned}$$

Then $(\#w{,}|w|{=}p_1(|\mathbf{z}|){+}p_2(|\mathbf{z}|){+}1)\chi(w,\mathbf{z}) = N_1 + N_2$. To handle products, define $\gamma(w,\mathbf{z})$ to be

$$\varphi_1(w[0{:}p_1(|\mathbf{z}|){-}1], \mathbf{z}) \land \varphi_2(w[p_1(|\mathbf{z}|){:}], \mathbf{z}).$$

Then $(\#w{,}|w|{=}p_1(|\mathbf{z}|){+}p_2(|\mathbf{z}|))\gamma(w,\mathbf{z}) = N_1 \cdot N_2$. Using repeated summation and products, as well as the addition of a fixed number of satisfying assignments, proves the proposition for an arbitrary fixed polynomial r. □

When working with modulus amplifying polynomials for the proof of the Toda and Beigel-Tarui theorems, it will be convenient to let the polynomial r be specified as part of the input. For this, a degree d univariate polynomial $r(t) = a_d t^d + \cdots + a_1 t + a_0$ is represented by a binary string $\mathrm{gn}(r)$, called the *Gödel number* of r, which encodes the sequence $\langle a_0, a_1, \ldots, a_d\rangle$. The Gödel number $\mathrm{gn}(r)$ should have its length $|\mathrm{gn}(r)|$ polynomially bounded by d and $\max_i |a_i|$; in fact, using an efficient encoding, $|\mathrm{gn}(r)|$ can be linearly bounded by $d + \sum_i |a_i|$. We require that $|\mathrm{gn}(r)| > d$ and $|\mathrm{gn}(r)| \geq |a_i|$: this holds naturally, since $\mathrm{gn}(r)$ codes a sequence of length $d + 1$.

Proposition 20 *Let Γ, $\varphi(x,\mathbf{z})$ and $p(|\mathbf{z}|)$ be as in Proposition 19. Then there is a Γ predicate $\chi(x,\mathbf{z},g)$ and a polynomial $q(\mathbf{n},n')$ so that, for all polynomials $r(t)$,*

$$(\#w{,}|w|{=}q(|\mathbf{z}|,|\mathrm{gn}(r)|))\chi(w,\mathbf{z},\mathrm{gn}(r)) \;=\; r(\,(\#x{,}|x|{=}p(|\mathbf{z}|))\varphi(x,\mathbf{z})\,). \qquad (9)$$

Proof (Sketch) Let $\chi(w,\mathbf{z},g)$ be a predicate expressing:

> "g is the Gödel number $\langle a_0, a_1, \ldots, a_d\rangle$ of a degree d polynomial r, and w encodes a sequence $\langle i, b, x_0, \ldots, x_{i-1}\rangle$ such that $i \leq d$ and $b < a_i$ and $(\forall j{<}|w|)[j < i \to |x_j|{=}p(|\mathbf{z}|) \land \varphi(x_j, \mathbf{z})]$."

Choose an appropriate Gödel encoding method and a polynomial $q(\mathbf{n},n')$ large enough so that every sequence $\langle i, b, x_0, \ldots, x_{i-1}\rangle$ satisfying $\chi(w,\mathbf{z},\mathrm{gn}(r))$ can be uniquely encoded by a w of length $|w| = q(|\mathbf{z}|, |\mathrm{gn}(r)|)$. By the construction of χ, the equality (9) holds. There are polynomial time procedures for parsing the Gödel number g and the sequence w and extracting the values x_i, and the quantifier $(\forall j{<}|w|)$ is sharply bounded. In view of Theorems 8 and 9, it follows that χ is a Γ predicate. □

Proposition 20 will be applied with *modulus amplifying polynomials P_k*. These polynomials have the property that, for all $N \geq 0$ and all $M \geq 2$ and for $i = 0, 1$,

$$N \equiv i \bmod M \quad\Rightarrow\quad P_k(N) \equiv i \bmod M^k.$$

There are various possible choices for P_k. Similarly to Toda's original construction [31], one can form P_k by composing the function $x \mapsto 4x^3 - 3x^4$ with itself $O(\log k)$ times. Yao [35] suggested composing $3x^2 - 2x^3$ with itself $O(\log k)$ times. Beigel and Tarui [7] suggested the optimal degree construction of

$$P_k(t) \;=\; 1 - (1-t)^k \sum_{j=0}^{k-1} \binom{k+j-1}{j} t^j.$$

These last polynomials have degree $2k-1$ and their Gödel numbers $\mathrm{gn}(P_k)$ can be constructed in polynomial time. (Toda's or Yao's polynomials could also be used. They have degrees k^2 and $k^{\log_2 3}$.) Since we are only interested in values modulo M^k, we adjust the coefficients of P_k so that they are non-negative and $< M^k$.

For $M = m$ a prime, we will use a variation $P'_{k,m}$ of P_k such that $P'_{k,m}(i) \equiv 1 \bmod m^k$ for all $i \equiv 0 \bmod m$ and such that $P'_{k,m}(i) \equiv 0 \bmod m^k$ for all $i \not\equiv 0 \bmod m$. Since m is prime, $R(x) = 1 - x^{m-1}$ satisfies that $R(i) \equiv 1 \bmod m$ for all $i \equiv 0 \bmod m$ and that $R(i) \equiv 0 \bmod m$ for all $i \not\equiv 0 \bmod m$. Then form P'_k as the composition $P'_k = P_k \circ R$, a polynomial of degree $(2k-1)(m-1)$, taking all coefficients to be in $\{0, \ldots, m^k - 1\}$. The Gödel number $\mathrm{gn}(P'_{k,m})$ can be constructed in time polynomial in k and m.

For $m^k > 2$, we will also use the polynomial $P''_{k,m}$ such that $P''_{k,m}(i) \equiv -1 \bmod m^k$ for all $i \equiv 0 \bmod m$ and such that $P''_{k,m}(i) \equiv 1 \bmod m^k$ for all $i \not\equiv 0 \bmod m$. We form $P''_{k,m}$ as the composition $Q \circ P'_{k,m}$ where $Q(t) = 1 - 2t$, again adjusting the coefficients to be in $\{0, \ldots, m^k - 1\}$. The degree of $P''_{k,m}$ is also $(2k-1)(m-1)$. The Gödel number $\mathrm{gn}(P''_{k,m})$ can also be constructed in time polynomial in k and m.

2.3 Symmetric quantifiers

A generalized bounded quantifier "$\mathcal{Q}$" is called symmetric if the truth value of the quantifier depends only on the cardinalities of the true and false instances. Specifically, $\mathcal{Q}$ is a *symmetric bounded quantifier* if the truth value of

$$(\mathcal{Q}x, |x| = p(|\mathbf{z}|))\varphi(x, \mathbf{z}) \tag{10}$$

depends only on the number of x's of length $p(|\mathbf{z}|)$ such that $\varphi(x, \mathbf{z})$ is true. More formally, there is a function $f_{\mathcal{Q}} : \mathbb{N}^2 \to \{\mathit{True}, \mathit{False}\}$ so that the truth value of (10) is equal to $f_{\mathcal{Q}}(N, 2^{p(|\mathbf{z}|)})$ where $N = (\#x, |x| = p(|\mathbf{z}|))\varphi(x, \mathbf{z})$.

Following Beigel-Tarui, we are particularly interested in symmetric quantifiers which consist of an approximate majority applied to iterated applications of modular counting. Let $M_1 < M_2 < \cdots < M_\ell$. The iterated modular counting function $\mathcal{C}_{[M_1, M_2, \ldots, M_\ell]}(N)$, or $\mathcal{C}_{[\mathbf{M}]}(N)$ for short, has the meaning

$$\mathcal{C}_{[\mathbf{M}]}(N) \;:=\; ((\cdots((N \bmod M_\ell) \bmod M_{\ell-1}) \cdots \bmod M_2) \bmod M_1). \tag{11}$$

Clearly $\mathcal{C}_{[\mathbf{M}]}(N)$ gives values in $\{0, 1, \ldots, M_1{-}1\}$. To compose this with an approximate majority function, define

$$\text{ApxMaj-}\mathcal{C}_{[\mathbf{M}]}(N) \;=\; \begin{cases} \textit{True} & \text{if } \mathcal{C}_{[\mathbf{M}]}(N) > (3/4)M_1 \\ \textit{False} & \text{if } \mathcal{C}_{[\mathbf{M}]}(N) < (1/4)M_1. \end{cases}$$

We can express this as a symmetric bounded quantifier by defining

$$(\text{ApxMaj-}\mathcal{C}_{[\mathbf{M}]}x, |x|{=}p(|\mathbf{z}|))\varphi(x, \mathbf{z})$$

to have truth value ApxMaj-$\mathcal{C}_{[\mathbf{M}]}(N)$ where $N = (\#x, |x|{=}p(|\mathbf{z}|))\varphi(x, \mathbf{z})$.

ApxMaj-$\mathcal{C}_{[\mathbf{M}]}$ may not be used unless it is impossible to have $(1/4)M_1 \le \mathcal{C}_{[\mathbf{M}]}(N) \le (3/4)M_1$. That is, we use ApxMaj-$\mathcal{C}_{[\mathbf{M}]}$ only when it is guaranteed to be well-defined.

2.4 Probabilistic classes

We now define the well-known probabilistic versions of $\oplus$-ptime predicates and $1\oplus_m$-ptime predicates. The "BP·" notation is from Schöning [26].

Definition 21 The class of probabilistic $\oplus$-ptime predicates, BP·$\oplus$-ptime, contains the predicates $A(\mathbf{z})$ for which there are a $\oplus$-ptime $\varphi(\mathbf{z}, r, u)$ and a polynomial $p(\mathbf{n}, \ell)$ so that, for all $\mathbf{z}$ and u,

$$\begin{aligned} A(\mathbf{z}) \text{ is true} &\Longrightarrow (\#r, |r|{=}p(|\mathbf{z}|, |u|))\varphi(\mathbf{z}, r, u) > 2^{p(|\mathbf{z}|,|u|)} \cdot (1 - 2^{-|u|}) \\ A(\mathbf{z}) \text{ is false} &\Longrightarrow (\#r, |r|{=}p(|\mathbf{z}|, |u|))\varphi(\mathbf{z}, r, u) < 2^{p(|\mathbf{z}|,|u|)} \cdot 2^{-|u|}. \end{aligned}$$

The point of Definition 21 is that when r is chosen at random, then the value of $\varphi(\mathbf{z}, r, u)$ gives, with high probability, the correct value for the truth or falsity of $A(\mathbf{z})$. The parameter u is used to control the probabilities. Only the length of u is important, and u can w.l.o.g. be set equal to 0^ℓ.

Definition 22 Fix a prime $m \ge 2$. The class of probabilistic $1\oplus_m$-ptime predicates, BP·$1\oplus_m$-ptime, is defined exactly as in Definition 21 except that φ is a $1\oplus_m$-ptime predicate.

The classes BP·ModPH and BP·ModPH$_m$ are defined similarly, but with φ a ModPH or ModPH$_m$ predicate, respectively. By Sipser [27], an approximate counting quantifier can be replaced with bounded quantifiers. This gives the following inclusions:

Theorem 23 *The* BP·ModPH *predicates are contained in the* ModPH *predicates. For fixed $m > 1$, the* BP·ModPH$_m$ *predicates are contained in the* ModPH$_m$ *predicates.*

As a corollary, BP·$\oplus$-ptime and BP·$1\oplus_m$-ptime are contained in ModPH and ModPH$_m$, respectively. The converse of this holds also; see Theorem 27 and Corollaries 28 and 29 below.

Proof (Sketch) We use the Lautemann [20] approximate counting construction. Suppose the predicate $A(\mathbf{z})$ satisfies the equations of Definition 21, but with $\varphi(|\mathbf{z}|, r, u)$ a ModPH or ModPH$_m$ predicate. Choose a polynomial $\ell(\mathbf{n})$ so that $p(|\mathbf{z}|, \ell(\mathbf{n})) < 2^{\ell(\mathbf{n})}$ always holds. We will use $u = 0^{\ell(|\mathbf{z}|)}$. Let a string w of length $(p(|\mathbf{z}|, \ell(|\mathbf{z}|)))^2$ encode $p(|\mathbf{z}|, \ell(|\mathbf{z}|))$ many strings of length $p(|\mathbf{z}|, \ell(|\mathbf{z}|))$ each. Let $(w)_i$ denote the i-th string encoded by w. For $(w)_i$ and r both of length $p(|\mathbf{z}|, \ell(|\mathbf{z}|))$, write $r \oplus (w)_i$ for their bitwise exclusive-or. Then $A(\mathbf{z})$ is true precisely when

$$(\exists w, |w|{=}(p(|\mathbf{z}|, \ell(\mathbf{z})))^2)(\forall r, |r|{=}p(|\mathbf{z}|, \ell(\mathbf{z}))) \\ (\exists i{\prec}p(|\mathbf{z}|, \ell(\mathbf{z})))[\varphi(\mathbf{z}, r{\oplus}(w)_i, 0^{\ell(|\mathbf{z}|)})] \tag{12}$$

holds. This equivalence is proved by the argument of [20]: When $A(\mathbf{z})$ is true, most r's make $\varphi(\mathbf{z}, r, 0^{\ell(|\mathbf{z}|)})$ true, and a probabilistic argument shows there is a w so that every r has at least one translation $r \oplus (w)_i$ that satisfies φ. On the other hand, when A is false, hardly any r's make φ true, and a cardinality argument, using $p(|\mathbf{z}|, \ell(\mathbf{n})) < 2^{\ell(\mathbf{n})}$, shows that no such w can exist.

By definition, (12) expresses $A(\mathbf{z})$ as a ModPH or ModPH$_m$ predicate, respectively. □

Remark on using probabilities of 1/4 *and* 3/4.As is usual with probabilistic classes (e.g., [26,5]), Definitions 21 and 22 could equivalently be stated without the parameter u and using the fractions 3/4 and 1/4 in place of $1 - 2^{-|u|}$ and $2^{-|u|}$. The general idea is that probabilities can be amplified from 3/4 and 1/4 to $1{-}2^{-|u|}$ and $2^{-|u|}$ by working with a property $\varphi(\mathbf{z}, r)$ and taking $c \cdot |u|$ many random samples of r for some constant c, and accepting if a majority of them satisfy $\varphi(\mathbf{z}, r)$. The amplification of probabilities follows from Chernoff bounds.

In more detail, suppose that

$$\begin{aligned} A(\mathbf{z}) \text{ is true} &\Longrightarrow (\#r, |r|{=}p(|\mathbf{z}|))\varphi(\mathbf{z}, r) > 2^{p(|\mathbf{z}|)} \cdot (3/4) \\ A(\mathbf{z}) \text{ is false} &\Longrightarrow (\#r, |r|{=}p(|\mathbf{z}|))\varphi(\mathbf{z}, r) < 2^{p(|\mathbf{z}|)} \cdot (1/4). \end{aligned}$$

Set $p'(|\mathbf{z}|, |u|) = 17|u| \cdot p(|\mathbf{z}|)$, so a string w with $|w| = p'(|\mathbf{z}|, |u|)$ codes $17|u|$ many strings r of length $p(|\mathbf{z}|)$; we write $(w)_i$ for the i-th such string r. If $A(\mathbf{z})$ is true (respectively, false), then for randomly chosen w, Chernoff bounds imply that with probability $1 - 2^{-|u|}$ more than one half (respectively, less than one half) of the $(w)_i$'s satisfy $\varphi(\mathbf{z}, (w)_i)$. Let $\psi(\mathbf{z}, w)$ express that at least one half of the $(w)_i$'s satisfy $\varphi(\mathbf{z}, (w)_i)$. By Lemma 18, ψ is a $\oplus$-ptime or $1\oplus_m$-ptime predicate if φ is. Thus ψ with the polynomial $p'(|\mathbf{z}|, |u|)$ gives a BP$\cdot\oplus$-ptime or BP$\cdot 1\oplus_m$-ptime (respectively) definition for $A(\mathbf{z})$.

3 Closure under bounded quantification for probabilistic classes

Theorems 8 and 9 gave basic closure properties for the $\oplus$-ptime and the $1\oplus_m$-ptime predicates. It is open whether these two classes are also closed under (non-sharply) bounded existential and universal quantification. However, Theorem 27 and Corollaries 28 and 29 below show that the probabilistic versions BP$\cdot\oplus$-ptime and BP$\cdot 1\oplus_m$-ptime of these classes are closed under bounded existential and universal quantification as well as bounded modular counting quantification. These results — in particular, Corollary 29 — are the "uniform version" of Theorem 1. Indeed, Section 4 will show that Theorem 1 follows from Corollary 29 by the Furst-Saxe-Sisper and Paris-Wilkie translations. Theorems 30 and 31 of Section 3.3 are related in the same way to the uniform version of Theorem 2.

A primary proof technique for Theorem 27 is the Valiant-Vazirani theorem, which we next state without proof.

3.1 The Valiant-Vazirani theorem

If $x, y \in \{0,1\}^n$ are binary strings $x = x_1x_2\cdots x_n$ and $y = y_1y_2\cdots y_n$, the inner product mod 2 of x and y, denoted $\langle x, y\rangle$ is equal to $\sum_i x_iy_i \bmod 2$. In this section, we let w range over strings of length $|w| = tn$, where $t > 0$, and write $(w)_i$ to denote $w[in{:}(i{+}1)n{-}1]$. That is, w codes t many strings of length n, and $(w)_i$ denotes the i-th one. The next theorem is due to Valiant and Vazirani, and is stated in the way used by Toda [31].

Theorem 24 (Valiant-Vazirani [33]) *Let $S \subseteq \{0,1\}^n$ be nonempty. Then, for some $i < n$,*

$$\Pr_{w\in\{0,1\}^{n^2}}\Big[(\#x, |x|{=}n)((\forall j{\le}i)(\langle x, (w)_j\rangle{=}0) \wedge x \in S) = 1\Big] \;\ge\; \frac{1}{4}.$$

Furthermore, there is an $i < n$ so that for all $\ell \ge 1$,

$$\Pr_{w\in\{0,1\}^{3\ell n^2}}\Big[(\exists k{<}3\ell)[(\#x, |x|{=}n)((\forall j{\le}i)(\langle x, (w)_{kn+j}\rangle{=}0) \wedge x{\in}S) = 1]\Big]$$
$$\ge\; 1 - \left(\frac{3}{4}\right)^{3\ell} \;>\; 1 - 2^{-\ell}.$$

For a proof, see for instance [5]. The second part of the theorem follows by iterating the first part 3ℓ times. Expressing the probability in terms of cardinality, and replacing the exact counting of x's with modular counting, gives:

Corollary 25 *Let $S \subseteq \{0,1\}^n$ and let $m \ge 2$. Let N equal*

$$(\#w, |w|{=}3\ell n^2)(\exists i{<}n)(\exists k{<}3\ell)(\textstyle\bigoplus_m^1 x, |x|{=}n)((\forall j{\le}i)(\langle x, (w)_{kn+j}\rangle{=}0) \wedge x{\in}S).$$

If $S \neq \emptyset$, then $N \ge 2^{3\ell n^2}(1 - 2^{-\ell})$. If $S = \emptyset$, then $N = 0$.

From this, and Theorems 8 and 9, we get immediately:

Corollary 26 *Let* Γ *be either the set of* $\oplus$*-ptime predicates or the set of* $1\oplus_m$*-ptime predicates. Let* $\varphi(x,\mathbf{z})$ *be a* Γ *predicate and* $p(\mathbf{n})$ *a polynomial. Then there is a* Γ *predicate* $\psi(\mathbf{z},w,u)$ *and a polynomial* $q(\mathbf{n},\ell)$ *such that the following holds. Let* $N(\mathbf{z},u)=(\#w,|w|{=}q(|\mathbf{z}|,|u|))\psi(\mathbf{z},w,u)$. *For all values of* $\mathbf{z}$ *and* u,

$$(\exists x,|x|{=}p(|\mathbf{z}|))\varphi(x,\mathbf{z}) \Longrightarrow N(\mathbf{z},u)\geq 2^{q(|\mathbf{z}|,|u|)}\cdot(1-2^{-|u|})$$
$$\neg(\exists x,|x|{=}p(|\mathbf{z}|))\varphi(x,\mathbf{z}) \Longrightarrow N(\mathbf{z},u)=0.$$

Corollary 26 is proved by letting S be the set of $x\in\{0,1\}^{p(|\mathbf{z}|)}$ satisfying $\varphi(x,\mathbf{z})$, and taking $q(|\mathbf{z}|,|u|)=3|u|\cdot(p(|\mathbf{z}|))^2$.

3.2 Toda's theorem

The next theorem, especially Corollary 29, will be used for the proof of Theorem 1. (Theorem 2 will follow from Theorems 30 and 31.)

Theorem 27 (a) BP$\cdot\oplus$-ptime *is closed under disjunction, conjunction, negation, bounded existential quantification, bounded universal quantification, and bounded* $\bigoplus_m$ *quantification for arbitrary* $m\geq 2$.

(b) *Fix a prime* $m\geq 2$. BP$\cdot 1\oplus_m$-ptime *is closed under disjunction, conjunction, negation, bounded existential quantification, bounded universal quantification, and bounded* $\bigoplus_m$ *quantification.*

Theorem 27 applies also to computation relative to an oracle Ω. As immediate corollaries to Theorems 23 and 27, we obtain:

Corollary 28 ModPH *is equal to* BP$\cdot\oplus$-ptime. *Likewise,* ModPH$^{\Omega}$ *is equal to* BP$\cdot\oplus$-ptime *relative to* Ω.

Corollary 29 *Fix a prime* $m\geq 2$. ModPH$_m$ *is equal to* BP$\cdot 1\oplus_m$-ptime. *Likewise,* ModPH$_m^{\Omega}$ *is equal to* BP$\cdot 1\oplus_m$-ptime *relative to* Ω.

Proof (Proof of Theorem 27) We prove (a) and (b) simultaneously. Let Γ be either the class of $1\oplus_m$-ptime predicates or the class of $\oplus$-ptime predicates, so BP$\cdot\Gamma$ denotes one of the classes BP$\cdot 1\oplus_m$-ptime or BP$\cdot\oplus$-ptime. Closure of BP$\cdot\Gamma$ under disjunction, conjunction and negation is straightforward with the aid of Theorems 8 and 9, and we omit their proofs. Instead, we prove closure under bounded existential quantification and $\bigoplus_m$ quantification.

Let $A(x,\mathbf{z})$ be a BP$\cdot\Gamma$ predicate. By Definition 21 or 22, there are a polynomial $p_1(n',\mathbf{n},\ell)$ and a Γ predicate $\varphi_1(x,\mathbf{z},r,u_1)$ such that, letting $N_1(x,\mathbf{z},u_1)$ equal

$$(\#r,|r|{=}p_1(|x|,|\mathbf{z}|,|u_1|))\varphi_1(x,\mathbf{z},r,u_1),$$

we have for all $x,\mathbf{z},u_1$ that

$$A(x,\mathbf{z})\text{ is true}\Longrightarrow N_1(x,\mathbf{z},u_1)>2^{p_1(|x|,|\mathbf{z}|,|u_1|)}\cdot(1-2^{-|u_1|})$$
$$A(x,\mathbf{z})\text{ is false}\Longrightarrow N_1(x,\mathbf{z},u_1)<2^{p_1(|x|,|\mathbf{z}|,|u_1|)}\cdot 2^{-|u_1|}.$$

It is helpful to first show closure of BP$\cdot\Gamma$ under sharply bounded universal quantification. With negation, this also implies closure under sharply bounded existential quantification. Let $B(\mathbf{z})$ be the predicate $(\forall x{\prec}q(|\mathbf{z}|))A(x,\mathbf{z})$.

We use $|u_1| = |u| + q(|\mathbf{z}|)$. Define $p(|\mathbf{z}|,|u|) = p_1(q(|\mathbf{z}|),|\mathbf{z}|,|u|{+}q(|\mathbf{z}|))$, and define $\varphi(\mathbf{z},r,u)$ as

$$(\forall x{\prec}q(|\mathbf{z}|))\varphi_1(x,\mathbf{z},r,u0^{q(|\mathbf{z}|)}).$$

By Theorem 8 or 9, φ is in Γ. Let $N(\mathbf{z},u)$ equal

$$(\#r,|r|{=}p(|\mathbf{z}|,|u|))\varphi(\mathbf{z},r,u).$$

We claim that p, φ and N define $B(\mathbf{z})$ as a BP$\cdot\Gamma$ predicate; that is:

$$\begin{aligned} B(\mathbf{z}) \text{ is true} &\Longrightarrow N(\mathbf{z},u) > 2^{p(|\mathbf{z}|,|u|)}\cdot(1-2^{-|u|}) \\ B(\mathbf{z}) \text{ is false} &\Longrightarrow N(\mathbf{z},u) < 2^{p(|\mathbf{z}|,|u|)}\cdot 2^{-|u|}. \end{aligned}$$

To prove this, first suppose $B(\mathbf{z})$ is true and fix u. By the union bound, the fraction of r's falsifying $\varphi(\mathbf{z},w,u)$ is $< q(|\mathbf{z}|)2^{-(|u|+q(|\mathbf{z}|))} < 2^{-|u|}$, as desired. Similarly, if $B(\mathbf{z})$ is false, the fraction of r's satisfying $\varphi(\mathbf{z},w,u)$ is $< 2^{-|u|}$.

We next show closure of BP$\cdot\Gamma$ under bounded modular counting quantification. Suppose $B(\mathbf{z})$ is $(\bigoplus_m x,|x|{=}q(|\mathbf{z}|))A(x,\mathbf{z})$. Let the polynomial $p(|\mathbf{z}|,|u|)$ again be $p_1(q(|\mathbf{z}|),|\mathbf{z}|,|u|{+}q(|\mathbf{z}|))$, and let $\varphi(\mathbf{z},r,u)$ be the predicate

$$(\bigoplus\nolimits_m x,|x|{=}q(|\mathbf{z}|))\varphi_1(x,\mathbf{z},r,u0^{q(|\mathbf{z}|)}).$$

Also let $N(\mathbf{z},u)$ equal

$$(\#r,|r|{=}p(|\mathbf{z}|,|u|))\varphi(\mathbf{z},r,u).$$

Theorem 8 or 9 shows that φ is a Γ predicate. We claim that these p, φ, and N define $B(\mathbf{z})$ as a BP$\cdot\Gamma$ predicate. Fix lengths for the $\mathbf{z}$'s and u. Let u_1 denote $u0^{q(|\mathbf{z}|)}$. By choice of p_1, N_1 and φ, if r of length $p(|\mathbf{z}|,|u|)$ is chosen at random, then $A(x,\mathbf{z})$ and $\varphi_1(x,\mathbf{z},r,u_1)$ have equal truth values with probability greater than $1-2^{-|u_1|}$. There are $2^{q(|\mathbf{z}|)}$ many values for x, and $|u_1| = q(|\mathbf{z}|){+}|u|$. Thus by the union bound, for a randomly chosen value for r, the truth values of $A(x,\mathbf{z})$ and $\varphi_1(x,\mathbf{z},r,u_1)$ are equal for all values x with probability at least $1-2^{-|u|}$. Consequently, if $B(\mathbf{z})$ is true, then $N(\mathbf{z},r,u)$ is greater than $2^{p(|\mathbf{z}|,|u|)}\cdot(1-2^{-|u|})$. Similarly, if $B(\mathbf{z})$ is false, then $N(\mathbf{z},r,u)$ is less than $2^{p(|\mathbf{z}|,|u|)}\cdot 2^{-|u|}$. This proves the claim, and shows that $B(\mathbf{z})$ is in BP$\cdot\Gamma$.

We finally show closure under bounded existential quantification. Consider the predicate $B(\mathbf{z})$ defined by $(\exists x,|x|{=}q(|\mathbf{z}|))A(x,\mathbf{z})$. By Corollary 25 to the Valiant-Vazirani theorem, there is a polynomial $p_2(|u_2|)$ so that, letting $N_2(\mathbf{z},u_2)$ equal

$$\begin{aligned} &(\#w,|w|{=}p_2(|u_2|)q(|\mathbf{z}|)^2)[(\exists i{\prec}q(|\mathbf{z}|))(\exists k{\prec}p_2(|u_2|)) \\ &\qquad(\bigoplus\nolimits_m^1 x,|x|{=}q(|\mathbf{z}|))((\forall j{\le}i)(\langle x,(w)_{kq(|\mathbf{z}|)+j}\rangle{=}0)\wedge A(x,\mathbf{z}))], \end{aligned} \tag{13}$$

we have

$$\begin{aligned} B(\mathbf{z}) \text{ is true} &\Longrightarrow N_2(\mathbf{z},u_2) > 2^{p_2(|u_2|)q(|\mathbf{z}|)^2}\cdot(1-2^{-|u_2|}) \\ B(\mathbf{z}) \text{ is false} &\Longrightarrow N_2(\mathbf{z},u_2) < 2^{p_2(|u_2|)q(|\mathbf{z}|)^2}\cdot 2^{-|u_2|}. \end{aligned}$$

Let $A_2(\mathbf{z},w)$ be the subformula of (13) in square brackets. By the closure of BP·Γ under conjunction, sharply bounded quantification and bounded modular counting, there are a Γ predicate $\varphi_3(\mathbf{z},w,r,u_3)$ and a polynomial $p_3(|\mathbf{z}|,|w|,|u_3|)$ so that, letting $N_3(\mathbf{z},w,u_3)$ equal

$$(\#r,|r|{=}p_3(|\mathbf{z}|,|w|,|u_3|))\varphi_3(\mathbf{z},w,r,u_3),$$

we have for all $\mathbf{z},w,u_3$ that

$$\begin{aligned} A_2(\mathbf{z},w) \text{ is true} &\Longrightarrow N_3(\mathbf{z},w,u_3) > 2^{p_3(|\mathbf{z}|,|w|,|u_3|)}\cdot(1-2^{-|u_3|}) \\ A_2(\mathbf{z},w) \text{ is false} &\Longrightarrow N_3(\mathbf{z},w,u_3) < 2^{p_3(|\mathbf{z}|,|w|,|u_3|)}\cdot 2^{-|u_3|} \end{aligned}$$

We can now define $p(|\mathbf{z}|,|u|)$ and $\varphi(\mathbf{z},v,u)$. We let $u_2 = u_3 = u0$ so that $|u_2| = |u_3| = |u|+1$. The polynomial p is defined to equal

$$p(|\mathbf{z}|,|u|) \;=\; p_2(|u|{+}1)q(|\mathbf{z}|)^2 + p_3(|\mathbf{z}|,|u|{+}1).$$

A string $v\in\{0,1\}^{p(|\mathbf{z}|,|u|)}$ codes a string w of length $p_2(|u|{+}1)q(|\mathbf{z}|)^2$ concatenated with a string r of length $p_3(|\mathbf{z}|,|u|{+}1)$, and we write $(v)_w$ and $(v)_r$ for these two substrings of v. Let $\varphi(\mathbf{z},v,u)$ be equal to $\varphi_3(\mathbf{z},(v)_w,(v)_r,u)$. Define $N(\mathbf{z},u)$ to equal

$$(\#v,|v|{=}p(|\mathbf{z}|,|u|))\varphi(\mathbf{z},v,u).$$

We claim that

$$\begin{aligned} B(\mathbf{z}) \text{ is true} &\Longrightarrow N(\mathbf{z},u) > 2^{p(|\mathbf{z}|,|u|)}\cdot(1-2^{-|u|}) \\ B(\mathbf{z}) \text{ is false} &\Longrightarrow N(\mathbf{z},u) < 2^{p(|\mathbf{z}|,|u|)}\cdot 2^{-|u|}. \end{aligned}$$

To prove the first assertion, note that if $B(\mathbf{z})$ is true, then for all but fraction $2^{-|u|-1}$ of the values of $w=(v)_w$, $A_2(\mathbf{z},w)$ is true. And then, for these good w's, all but a fraction $2^{-|u|-1}$ of the values for $r=(v)_r$ make $\varphi_3(\mathbf{z},w,r,u)$ true. Thus by the union bound, less than a fraction $2\cdot 2^{-|u|-1} = 2^{-|u|}$ of the values for v make $\varphi(\mathbf{z},v,u)$ false. A similar argument works when $B(\mathbf{z})$ is false.

Since φ_3 is a Γ predicate, φ is also in Γ. This shows that B is a BP·Γ predicate, and completes the proof of Theorem 27. □

3.3 A single symmetric quantifier suffices

An important component of the Beigel-Tarui theorem is that multiple modular quantifiers can be replaced by a single symmetric quantifier. For this, we now prove that BP·⊕-ptime and BP·$1\oplus_m$-ptime predicates can be expressed with a single symmetric quantifier applied to a polynomial time predicate. Theorem 31 is the uniform version of Theorem 2. Theorem 30 is the special case of BP·$1\oplus_m$-ptime predicates:

Theorem 30 *Let $m \ge 2$ be a prime and $A(\mathbf{z})$ be a* $\mathrm{BP}{\cdot}1{\oplus}_m$-ptime *predicate (i.e., a* ModPH_m*-predicate). Then $A(\mathbf{z})$ can be expressed in the form*

$$(\mathrm{ApxMaj}\text{-}\mathcal{C}_{[M]}x, |x|{=}q(|\mathbf{z}|))\chi(x,\mathbf{z})$$

where $[M]$ denotes a sequence of length one with $M = m^{s(|\mathbf{z}|)}$ for some polynomial $s(|\mathbf{z}|)$, where q is a polynomial, and where χ is a polynomial time predicate.

Theorem 30 is the $\ell = 1$ case of the next theorem. Assume A is a $\mathrm{BP}{\cdot}{\oplus}$-ptime predicate, so there are a $\oplus$-ptime predicate φ and a polynomial $p(|\mathbf{z}|)$ such that for all $\mathbf{z}$,

$$(\#r, |r|{=}p(|\mathbf{z}|))\varphi(\mathbf{z},r) \tag{14}$$

is either $> \frac{3}{4}2^{p(|\mathbf{z}|)}$ or $< \frac{1}{4}2^{p(|\mathbf{z}|)}$ depending on whether $A(\mathbf{z})$ is true or false. The predicate φ can be expressed in the form

$$(\textstyle\bigoplus_{m_1} x_1, |x_1|{=}p_1(|\mathbf{z}|)) \cdots (\textstyle\bigoplus_{m_\ell} x_\ell, |x_\ell|{=}p_\ell(|\mathbf{z}|))\psi(\mathbf{z},r,x_1,\ldots,x_\ell) \tag{15}$$

for some polynomial time $\psi(\mathbf{z},r,\mathbf{x})$, some polynomials $p_1(|\mathbf{z}|),\ldots,p_\ell(|\mathbf{z}|)$, and some primes $m_1,\ldots,m_\ell$.

Theorem 31 *Let $A(\mathbf{z})$ be a* $\mathrm{BP}{\cdot}{\oplus}$-ptime *predicate (i.e., a* ModPH*-predicate) defined as above by* (14) *and* (15)*. Then $A(\mathbf{z})$ can be expressed in the form*

$$(\mathrm{ApxMaj}\text{-}\mathcal{C}_{[\mathbf{M}]}y, |y|{=}q(|\mathbf{z}|))\chi(\mathbf{z},y) \tag{16}$$

where $[\mathbf{M}]$ denotes a sequence $M_1,\ldots,M_\ell$, with each $M_j = m_j^{s_j(|\mathbf{z}|)}$ where the m_j's are the primes in the quantifier prefix of $\varphi(\mathbf{z},r)$, where q and the s_j's are polynomials, and where χ is a polynomial time predicate.

Proof We first give the construction of q, the s_j's and χ, and then work on proving their properties. Define the polynomials $s_j = s_j(|\mathbf{z}|)$ by letting $s_1(|\mathbf{z}|) = p(|\mathbf{z}|) + 3$ and $s_2(|\mathbf{z}|) = 2s_1(|\mathbf{z}|)m_1(p_1(|\mathbf{z}|){+}1)$ and, for $j \ge 2$,

$$s_{j+1}(|\mathbf{z}|) \;=\; 2^j s_j(|\mathbf{z}|)^2 m_j^2(p_j(|\mathbf{z}|){+}1)\prod_{i=1}^{j-1} s_i(|\mathbf{z}|)m_i.$$

Then $M_j = M_j(|\mathbf{z}|) = m_j^{s_j(|\mathbf{z}|)}$. Let $\psi_j(\mathbf{z},r,x_1,\ldots,x_j)$ be the subformula

$$(\textstyle\bigoplus_{m_{j+1}} x_{j+1}, |x_{j+1}|{=}p_{j+1}(|\mathbf{z}|)) \cdots (\textstyle\bigoplus_{m_\ell} x_\ell, |x_\ell|{=}p_\ell(|\mathbf{z}|))\psi(\mathbf{z},r,x_1,\ldots,x_\ell).$$

of (15).

We inductively define polynomial time predicates $\chi_j(\mathbf{z},r,x_1,\ldots,x_j,y)$ and polynomials $q_j(|\mathbf{z}|)$ for $j = \ell,\ldots,1,0$. For χ_j, we always enforce $|y| = q_j(|\mathbf{z}|)$. To start the inductive definition, χ_ℓ is the same as $\psi(\mathbf{z},r,x_1,\ldots,x_\ell)$, and $q_\ell = 0$ (that is, y is the empty string). Now let $1 \le j < \ell$, and recall the definition of $P'_{s_{j+1},m_{j+1}}$. Let $a_{j+1,i}$ denote the coefficient of x^i in $P'_{s_{j+1},m_{j+1}}$. Recalling the proof of Proposition 20, define $\chi_j(\mathbf{z},r,x_1,\ldots,x_j,y)$ to be:

"y has the form $\langle i, b, y'_0, \ldots, y'_{i-1}\rangle$, where $i \le \text{degree}(P'_{s_{j+1},m_{j+1}})$, and $b < a_{j+1,i}$, and each y'_t is a pair $y'_t = \langle x'_t, y''_t\rangle$ with $|x'_t| = p_{j+1}(|\mathbf{z}|)$ and $|y''_t| = q_{j+1}(|\mathbf{z}|)$ so that

$$\chi_{j+1}(\mathbf{z}, r, x_1, \ldots, x_j, x'_t, y''_t)$$

is true."

Using a suitable method of encoding sequences, we define $q_j = q_j(|\mathbf{z}|)$ to be a sufficiently large polynomial such that every y satisfying χ_j has length exactly q_j (and so that each sequence y has a unique encoding).

The predicate $\chi_0(\mathbf{z}, r, y)$ is defined similarly, but using P''_{s_1,m_1} instead of P'_{s_1,m_1}, namely $\chi_0(\mathbf{z}, r, y)$ is defined as:

"y has the form $\langle i, b, y'_0, \ldots, y'_{i-1}\rangle$, where $i \le \text{degree}(P''_{s_1,m_1})$, $b < a_{1,i}$, and each y'_t is a pair $y'_t = \langle x'_t, y''_t\rangle$ with $|x'_t| = p_1(|\mathbf{z}|)$ and $|y''_t| = q_1(|\mathbf{z}|)$ so that

$$\chi_1(\mathbf{z}, r, x'_t, y''_t)$$

is true."

Again, define $q_0 = q_0(|\mathbf{z}|)$ to be a sufficiently large polynomial such that every y satisfying χ_0 has a unique encoding of length exactly q_0.

Finally, define $q(|\mathbf{z}|) = p(|\mathbf{z}|) + q_0(|\mathbf{z}|)$, and define the polynomial time predicate $\chi(\mathbf{z}, y)$ by

$$|y| = q(|\mathbf{z}|) \ \wedge\ \chi_0(\mathbf{z}, y[0{:}p(|\mathbf{z}|){-}1], y[p(|\mathbf{z}|){:}]).$$

That is, y encodes a pair of strings r, y' which satisfy $\chi_0(\mathbf{z}, r, y')$.

We claim that this χ and q satisfy the property (16) of the theorem. To prove this, we first need to analyze how many y's satisfy the formulas χ_j and χ. Define values $N_{r,x_1,\ldots,x_j}$ as follows. (We suppress the dependence of $N_{r,x_1,\ldots,x_j}$ on $\mathbf{z}$ in the notation.) We implicitly require always that the values x_i denote strings of length $p_i(|\mathbf{z}|)$ and the value r is a string of length $p(|\mathbf{z}|)$; e.g., the summation over x_{j+1} means over x_{j+1}'s of length $p_{j+1}(|\mathbf{z}|)$.

$$N_{r,x_1,\ldots,x_\ell} = \begin{cases} 1 & \text{if } \psi(\mathbf{z}, r, x_1, \ldots, x_\ell) \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{r,x_1,\ldots,x_j} = P'_{s_{j+1},m_{j+1}}\Big(\sum_{x_{j+1}} N_{r,x_1,\ldots,x_j,x_{j+1}}\Big) \qquad \text{for } 1 \le j < \ell$$

$$N_r = P''_{s_1,m_1}\Big(\sum_{x_1} N_{r,x_1}\Big)$$

$$N = \sum_r N_r.$$

Claim 32 *Let $0 \le j \le \ell$ and let $r, x_1, \ldots, x_j$ be strings of the appropriate lengths $|r| = p(|\mathbf{z}|)$ and $|x_i| = p_i(|\mathbf{z}|)$. The number of y's which satisfy $\chi_j(\mathbf{z}, r, x_1, \ldots, x_j, y)$ is equal to $N_{r,x_1,\ldots,x_j}$. The number of y's which satisfy $\chi(\mathbf{z}, y)$ is equal to N.*

The claim is proved by induction on j, descending from ℓ to 0. The base case, $j = \ell$ is trivial from the definitions. For the induction step, fix $0 \leq j < \ell$ and strings $r, x_1, \ldots, x_j$. The induction hypothesis states that, for each x' of length $p_{j+1}(|\mathbf{z}|)$, there are $N_{r,x_1,\ldots,x_j,x'}$ many strings y'' such that $\chi_{j+1}(\mathbf{z}, r, x_1, \ldots, x_j, x', y'')$. From this, there are $\sum_{x'} N_{r,x_1,\ldots,x_j,x'}$ many pairs $y' = \langle x', y'' \rangle$ satisfying this condition. From the definition of χ_j and Proposition 20, the number y's satisfying $\chi_j(\mathbf{z}, r, x_1, \ldots, x_j, y)$ is equal to $N_{r,x_1,\ldots,x_j}$. The fact that N is the number of y's which satisfy $\chi(\mathbf{z}, y)$ is now immediate from the definitions. This proves the claim. □

Recall the iterated modular counting quantity $\mathcal{C}_{[\mathbf{M}]}(\cdots)$ of (11).

Claim 33 *The value $\mathcal{C}_{[\mathbf{M}]}(N_r)$ is equal to either $M_1{-}1$ or 1, depending on whether $\varphi(\mathbf{z}, r)$ is true or false, respectively.*

Before embarking on the proof of this claim, we show that Claims 32 and 33 together imply Theorem 31. From Claim 33 and the definition of N, we have that N is congruent modulo M_1 to the number of r's such that $\varphi(\mathbf{z}, r)$ is false, minus the number of r's such that $\varphi(\mathbf{z}, r)$ is true. Since $s_1(|\mathbf{z}|) = p(|\mathbf{z}|) + 3$, we have $M_1 > 4 \cdot 2^{p(|\mathbf{z}|)}$. From these facts, it follows that, if $A(x)$ is true then $\mathcal{C}_{[\mathbf{M}]}(N)$ is $> \frac{3}{4}2^{p(|\mathbf{z}|)}$, since more r's make $\varphi(\mathbf{z}, r)$ true than make it false. Likewise, if $A(x)$ is false, then $\mathcal{C}_{[\mathbf{M}]}(N)$ is $< \frac{1}{4}2^{p(|\mathbf{z}|)}$. By Claim 32, N is the number of y's such that $\chi(\mathbf{z}, y)$. In addition, since ψ is polynomial time computable, each χ_j and hence χ is in polynomial time. This satisfies all the conditions of Theorem 31.

As the first step of proving Claim 33, define quantities $N^*_{r,x_1,\ldots,x_j}$ by:

$$\begin{aligned} N^*_{r,x_1,\ldots,x_\ell} &= N_{r,x_1,\ldots,x_\ell} \\ N^*_{r,x_1,\ldots,x_j} &= P'_{s_{j+1},m_{j+1}}\Big(\sum_{x_{j+1}} N^*_{r,x_1,\ldots,x_j,x_{j+1}}\Big) \bmod M_{j+1} \qquad \text{for } 1 \leq j < \ell \\ N^*_r &= P''_{s_1,m_1}\Big(\sum_{x_1} N^*_{r,x_1}\Big) \bmod M_1 \end{aligned}$$

We claim that, for each $r, x_1, \ldots, x_j$, where $1 \leq j \leq \ell$, the value $N^*_{r,x_1,\ldots,x_j}$ is equal to either 1 or 0 depending on whether $\psi_j(\mathbf{z}, r, x_1, \ldots, x_j)$ is true or false (respectively). This is proved by on induction on j, descending from ℓ to 1. The base case $j = \ell$ is true by definition. The induction step is immediate from the definitions of ψ_j and $P'_{s_{j+1},m_{j+1}}$. Similarly, N^*_r is equal to either $M_1{-}1$ or 1, depending on whether $\varphi(\mathbf{z}, r)$ (equivalently, $\psi_0(\mathbf{z}, r)$) is true or false, respectively.

Therefore, to prove Claim 33, we must prove that $\mathcal{C}_{[\mathbf{M}]}(N_r) = N^*_r$ for all r. The difference between the definitions of $\mathcal{C}_{[\mathbf{M}]}(N_r)$ and N^*_r is that modular operations are used in intermediate steps when computing N^*_r, but are used only at the end of the computation when computing $\mathcal{C}_{[\mathbf{M}]}(N_r)$. To prove the quantities are equal, we need to "pull out" the modular operators in the definition of N^*_r.

Fix $j \geq 1$. From its definition, N_r can be expressed as a polynomial Q_j of the values $N_{r,x_1,\ldots x_j}$. To write out Q_j explicitly, let $Q'_j(\mathbf{y}_{x_j})$ be the polynomial over the variables $\{y_{x_j}\}_{|x_j|=p_j(|\mathbf{z}|)}$, defined by

$$Q'_j \;=\; P'_{s_j,m_j}(\sum_{x_j} y_{x_j})$$

if $j > 1$, and by $Q'_1 = P''_{s_1,m_1}(\sum_{x_1} y_{x_1})$ if $j = 1$. Now let $Q_j(\mathbf{y}_{x_1,\ldots,x_j})$ be the polynomial over variables $\{y_{x_1,\ldots,x_j}\}_{x_1,\ldots,x_j}$ defined by

$$Q_j(\mathbf{y}_{x_1,\ldots,x_j}) \;=\; Q'_1(Q'_2(\cdots Q'_{j-1}(Q'_j(\mathbf{y}_{x_1,\ldots,x_j}))\cdots)).$$

To be precise, Q_1 is Q'_1, and $Q_{j+1}(\mathbf{y}_{x_1,\ldots,x_{j+1}})$ is formed from $Q_j(\mathbf{y}_{x_1,\ldots,x_j})$ by replacing each $y_{x_1,\ldots,x_j}$ with $Q'_{j+1}(\{y_{x_1,\ldots,x_j,x_{j+1}}\}_{x_{j+1}})$. Thus, we have

$$N_r \;=\; Q_j(\mathbf{N}_{r,x_1,\ldots,x_j}) \;=\; Q'_1(Q'_2(\cdots Q'_{j-1}(Q'_j(\mathbf{N}_{r,x_1,\ldots,x_j}))\cdots)).$$

Claim 34 *Suppose that each* $y_{x_1,\ldots,x_j}$ *is congruent modulo* M_{j+1} *to* 0 *or* 1. *Then*

$$Q_j(\mathbf{y}_{x_1,\ldots,x_j} \bmod M_{j+1}) \;=\; Q_j(\mathbf{y}_{x_1,\ldots,x_j}) \bmod M_{j+1}. \tag{17}$$

In the lefthand side of (17), the notation $\mathbf{y}_{x_1,\ldots,x_j} \bmod M_{j+1}$ means that, for each set of values for $x_1,\ldots,x_j$, the corresponding argument of Q_j is equal to $y_{x_1,\ldots,x_j} \bmod M_j$.

Claim 33 follows easily from Claim 34. Namely, N^*_r is equal to

$$\begin{aligned}
&Q'_1(Q'_2(Q'_3(\cdots Q'_\ell(\mathbf{N}_{r,x_1,\ldots,x_\ell}) \bmod M_\ell)\cdots) \bmod M_3) \bmod M_2) \bmod M_1\\
&= Q'_1(Q'_2(Q'_3(\cdots Q'_\ell(\mathbf{N}_{r,x_1,\ldots,x_\ell}) \bmod M_\ell)\cdots) \bmod M_3)) \bmod M_2 \bmod M_1\\
&= Q'_1(Q'_2(Q'_3(\cdots Q'_\ell(\mathbf{N}_{r,x_1,\ldots,x_\ell}) \bmod M_\ell)\cdots))) \bmod M_3 \bmod M_2 \bmod M_1\\
&\quad\ldots\\
&= Q'_1(Q'_2(Q'_3(\cdots Q'_\ell(\mathbf{N}_{r,x_1,\ldots,x_\ell}))\cdots))) \bmod M_\ell \cdots \bmod M_3 \bmod M_2 \bmod M_1\\
&= N_r \bmod M_\ell \cdots \bmod M_3 \bmod M_2 \bmod M_1\\
&= \mathcal{C}_{[\mathbf{M}]}(N_r).
\end{aligned}$$

The first line is the definition of N^*_r. The final two equalities follow from the definitions of N_r and $\mathcal{C}_{[\mathbf{M}]}$. The remaining $\ell-1$ equalities follow from Claim 34. To apply Claim 34, we need $Q'_t(\cdots Q'_\ell(\mathbf{N}_{r,x_1,\ldots,x_\ell})\cdots) \in \{0,1\}$ for all t and all fixed $r, x_1,\ldots,x_t$. This fact follows from the characterization of the values $N^*_{r,x_1,\ldots,x_t}$ given after their definitions.

It therefore suffices to prove Claim 34 to finish the proof of Theorem 31. The idea behind Claim 34 is that M_j is larger than the absolute value of the lefthand side of (17). The bound on the value of lefthand side will be in terms of the “norm” of Q_j.

The *norm* $\mathcal{N}(p)$ of a polynomial p is by definition the sum of the absolute values of the coefficients of p.[1] Our polynomials have only non-negative coefficients; for these, $\mathcal{N}(p)$ is the same as the sum of the coefficients of p. The next lemma gives simple properties of the norm.

[1] The term “norm” is due to Beigel-Tarui [7]; Yao [35] used “size”.

Lemma 35 *Let $p(\mathbf{n})$ and $q(\mathbf{n})$ be polynomials.*

(a) *For $\mathbf{a} \in \{-1, 0, 1\}$, $-\mathcal{N}(p) \le p(\mathbf{a}) \le \mathcal{N}(p)$.*
(b) *$\mathcal{N}(p + q) \le \mathcal{N}(p) + \mathcal{N}(q)$.*
(c) *$\mathcal{N}(p \cdot q) \le \mathcal{N}(p) \cdot \mathcal{N}(q)$.*
(d) *Let $\mathbf{n}$ have length ℓ, and $q_i(\mathbf{m})$ be polynomials with $\mathcal{N}(q_i) \le \mathcal{N}_q$ for all $i \le \ell$. Let $h(\mathbf{m}) = f(q_1(\mathbf{m}), \ldots, q_\ell(\mathbf{m}))$. Then $\mathcal{N}(h) \le \mathcal{N}(f) \cdot (\mathcal{N}_q)^{\deg(f)}$.*

Another simple property that follows from (a) is:

Lemma 36 *Suppose $p(n_1, \ldots, n_t)$ is a polynomial (with non-negative coefficients) and that $\mathcal{N}(p) < M$. Further suppose $a_i \bmod M \in \{0, 1\}$ for all i. Then*

$$p(a_1 \bmod M,\ a_2 \bmod M, \ldots, a_t \bmod M) \ = \ p(a_1, a_2, \ldots, a_t) \bmod M.$$

In view of Lemma 36, Claim 34 follows once we prove that $\mathcal{N}(Q_j) < M_{j+1}$. First note that

$$\begin{aligned} \mathrm{degree}(Q_j) &= \mathrm{degree}(P''_{s_1,m_1}) \prod_{i=2}^{j} \mathrm{degree}(P'_{s_i,m_i}) \\ &= \prod_{i=1}^{j} (2s_i - 1)(m_i - 1) \ < \ \prod_{i=1}^{j} 2 s_i m_1. \end{aligned}$$

(We are suppressing the dependence of s_i and p_i on $|\mathbf{z}|$ in the notation.) We prove that $\mathcal{N}(Q_j) < M_{j+1}$ by induction on j. For $j = 1$, Q_1 is $P''_{s_1,m_1} \circ \sum_{x_1} y_{x_1}$. Since P''_{s_1,m_1} has degree $(2s_1{-}1)(m{-}1)$ and coefficients $< M_1 = m_1^{s_1}$, its norm $\mathcal{N}(P_{s_1,m_1})$ is $< ((2s_1{-}1)(m_1{-}1){+}1)m_1^{s_1}$. So, since the sums are taken over the 2^{p_1} many values for x_1, Lemma 35(d) gives

$$\mathcal{N}(Q_1) \ < \ ((2s_1{-}1)(m_1{-}1){+}1) m_1^{s_1} (2^{p_1})^{(2s_1-1)(m_1-1)} \ < \ 2^{2s_1 m_1 (p_1+1)}. \tag{18}$$

where the second inequality holds since s_1, m_1, p_1 are positive integers. By definition, $s_2 = 2s_1 m_1(p_1{+}1)$, so $M_2 = m_2^{2s_1 m_1 (p_1+1)}$. Thus $\mathcal{N}(Q_1) < M_2$.

For the induction step, we have $Q_j = Q_{j-1} \circ Q'_j$, and by the same computation as for (18),

$$\mathcal{N}(Q'_j) < ((2s_j{-}1)(m_j{-}1){+}1) m_j^{s_j} (2^{p_j})^{(2s_j-1)(m_j-1)} \ < \ 2^{2s_j m_j (p_j+1)}$$

The induction hypothesis gives $\mathcal{N}(Q_{j-1}) < M_j = m_j^{s_j}$. By Lemma 35(d),

$$\begin{aligned} \mathcal{N}(Q_j) &< m_j^{s_j} \cdot \mathcal{N}(Q'_j)^{\mathrm{degree}(Q_{j-1})} \ < \ 2^{s_j m_j} (2^{2s_j m_j (p_j+1)})^{\prod_{i=1}^{j-1} 2 s_i m_i} \\ &= 2^{2^j s_j^2 m_j^2 (p_j+1) \prod_{i=1}^{j-1} s_i m_i} \ = \ 2^{s_{j+1}} \le m_{j+1}^{s_{j+1}}. \end{aligned}$$

Thus, $\mathcal{N}(Q_j) < M_j$. This completes the proof of Claim 34, and thereby Theorem 31. □

4 Proofs of Uniform Beigel-Tarui Theorems

We now prove Theorems 1 and 2. Fix values for d and $m \geq 2$. Suppose a circuit C has depth d and size S and uses unbounded fanin $\wedge$, $\vee$, and $\oplus_m$ gates. We wish to constuct equivalent circuits that satisfy Theorems 1 and 2.

A high-level sketch of the proofs is as follows. We will encode the circuit C in terms of its direct connection language using an oracle Ξ. A second oracle Ω is used to encode the values of the Boolean inputs to C. The circuit value problem for C is then expressible as a $\mathrm{ModPH}_m^{\Omega,\Xi}$ predicate or a $\mathrm{ModPH}^{\Omega,\Xi}$ predicate. Corollary 29 or Theorem 31 expresses this predicate as a constant depth formula with a symmetric gate as output. The Furst-Saxe-Sipser, Paris-Wilkie translation then converts this into a constant-depth quasipolynomial size circuit satisfying Theorem 1 or 2.

We prove Theorem 1 first; the proof for Theorem 2 is almost identical. Let n be the least value such that $2^n \geq S \log m$, so $n = O(\log S)$. Let $x_1, \ldots, x_i$ be the inputs to C, and G the number of gates in C. W.l.o.g., $G + 2i < 2^n$. Thus, an n bit string z can be used to specify uniquely one of the G many gates or $2i$ many literals x_j or $\overline{x}_j$. In addition, each gate has fewer than 2^n many inputs. As we explain next, this allows the oracle Ξ to encode the direct connection language of C by using its values $\Xi(x)$ on binary strings x of length $|x| = 3n$.

The oracle Ξ encodes enough information about the circuit C so that its direct connection language [25] can be computed in deterministic polynomial time (in n) relative to Ξ on a multitape Turing machine. Specifically, the following properties of C can by computed quickly with the aid of Ξ: The gate type of the k-th gate in C, the input arity of the k-th gate, and the gate number or literal number of the ℓ-th input to to the k-th gate. One natural encoding for the direct connection language with the oracle Ξ is as follows. First, let the value $\Xi(x) = \mathit{True}$ for a triple $x = (y, 0, g)$ with $|y| = n$ and g a code for one of the finitely many gate types such that $y \in \{0,1\}^n$ is the code for a gate of type g. Likewise, $\Xi(x)$'s value for a triple $x = (y, \ell, 0)$ can be used to indicate whether y designates a gate y with $> \ell$ many inputs (allowing binary search to compute the precise arity of gate y). Finally, let values $z \in \{0,1\}^n$ designate either one of the G many gates or one of the $2i$ many literals. Then $\Xi(x)$'s value for a triple $x = (y, \ell, j)$ can indicate whether the $(\ell+1)$-st input to the gate y is designated by a z such that the j-th bit of z is a 1. This allows z to be computed from y and ℓ in polynomial time by querying its bits. Since y and ℓ can be encoded with n bits, g ranges over finitely many values, and j ranges over $1, \ldots, n$, all these triples can be uniquely encoded with binary strings of length $3n$. The truth values $\Xi(x)$ on these 2^{3n} strings fully specify the direct connection language of C.

The circuit C is, by hypothesis, of constant depth d. Without loss of generality, we may also assume that C is leveled, alternating between levels of unbounded fanin ANDs, unbounded fanin ORs, and unbounded fanin $\oplus_m$ gates for some fixed $m \geq 2$, and with all inputs at the bottom level being literals x_i or $\overline{x}_i$. By adding extra single-input gates, this can be achieved while at most tripling the depth of the circuit. Furthermore, we can assume w.l.o.g.

that the strings y and z, designating gates or gate inputs, are sufficiently uniformly assigned so that there is a polynomial time algorithm to obtain from a value y the depth of the gate y in the circuit.

The circuit C has i inputs, $x_1, \ldots, x_i$ (and their complements $\overline{x}_1, \ldots, \overline{x}_i$). We identify $1, \ldots, i$ with the first i binary strings in $\{0,1\}^n$ and encode the Boolean truth values of the x_j's with an oracle Ω. Namely, the value of $\Omega(j)$ specifies the truth value of x_j. Then, given Ξ and Ω, the *circuit value problem* for C is the problem of whether C, as described by Ξ and with inputs given by Ω outputs *True*.

We claim that the circuit value problem for C is uniformly expressible by a $\mathrm{ModPH}_m^{\Omega,\Xi}$ predicate. That is, there is a formula $\mathrm{Val}_{d,m}(0^n)$ in $\mathrm{ModPH}_m^{\Omega,\Xi}$ which is true precisely when Ξ encodes the direct connection language for a leveled circuit of depth d and size $\le 2^n$ over the unbounded fanin connectives $\wedge$, $\vee$ and $\oplus_m$ such that circuit outputs *True* when its Boolean inputs are given by Ω. The fact that the formula $\mathrm{Val}_{d,m}$ exists in $\mathrm{ModPH}_m^{\Omega,\Xi}$ follows straightforwardly the fact that the circuit is leveled and constant depth: the formula $\mathrm{Val}_{d,m}$ uses the appropriate universal, existential, and $\bigoplus_m$ quantifiers to talk about the truth of gates at each level in the circuit C. By design, $\mathrm{Val}_{d,m}(0^n)$ queries values of $\Xi(x)$ only for $|x| = 3n$ and queries values of $\Omega(x)$ only for $|x| = n$.

We write "C accepts" to mean "C as encoded by Ξ with inputs as specified by Ω outputs *True*". By definition, $\mathrm{Val}_{d,m}(0^n)$ expresses that C accepts. Since $\mathrm{Val}_{d,m}$ is in $\mathrm{ModPH}_m^{\Omega,\Xi}$, Corollary 29 with $|u| = 2$, Definition 22, and the definition of $1\oplus_m$-ptime state there exists polynomials $q(n)$ and $s(n)$ and a predicate ψ which is polynomial time (relative to Ω, Ξ) so that

$$C \text{ accepts} \Leftrightarrow N(0^n) \ge \frac{3}{4}2^{q(n)}$$

$$C \text{ rejects} \Leftrightarrow N(0^n) \le \frac{1}{4}2^{q(n)}$$

where

$$N(0^n) = (\#r, |r| {=} q(n))(\textstyle\bigoplus_m y, |y| {=} s(n))\psi(0^n, r, y). \tag{19}$$

Via the Paris-Wilkie, Furst-Saxe-Sipser translation, we can convert $\mathrm{Val}_{d,m}$ into a circuit D which evaluates whether C outputs *True*: The inputs to D are Boolean inputs ω_j and $\overline{\omega}_j$ representing the values of $\Omega(j)$ for $1 \le j \le i$, and Boolean inputs ξ_j and $\overline{\xi}_j$ representing the values of $\Xi(j)$ for $j \in \{0,1\}^{3n}$. The size of D will be $2^{n^{O(1)}}$. We initially construct D as a depth four circuit, but then collapse it to have depth three. The top gate of D is a majority gate (hence, a symmetric gate) with $2^{q(n)}$ many inputs, one for each value of r in (19). The second level contains $\oplus_m$ gates, each with $2^{s(n)}$ many inputs; namely, each input corresponds to a particular value for y in (19). The final two levels initially consist of disjunctions of conjunctions, one for each pair r and y. The conjunctions all have size $n^{O(1)}$. For fixed values of r and y, these conjunctions are chosen by considering all possible computations γ of $\psi(0^n, r, y)$ over all possible oracles Ξ and Ω. (So we are thinking of Ξ and Ω as being unspecified.) Some of these computations γ are accepting; the rest are rejecting. For each computation γ which is accepting, form the conjunction A_γ of the literals ω_j, $\overline{\omega}_j$, ξ_j, $\overline{\xi}_j$ such that γ queries the value of $\Omega(j)$ or $\Xi(j)$ and receives the corresponding answer of *True* or *False*. Since Ψ has runtime polynomially bounded in terms of n, each conjunction A_γ is polynomial size, $n^{O(1)}$. The condition that $\Psi(0^n, r, y)$ accepts is equivalent to

$$\bigvee_\gamma \{A_\gamma : \gamma \text{ is an accepting computation for } \Psi(0^n, r, y)\}. \tag{20}$$

For any two distinct computations γ and γ', there must exist an oracle query for which γ and γ' receive different answers. (Otherwise, the computations would not be different.) Therefore, for any fixed setting of the ω_j's and ξ_j's at most one conjunction A_γ can be true.

The circuit D, as constructed so far, has size $2^{n^{O(1)}}$ and depth four; namely a symmetric gate at the top, then a level of $\oplus_m$ gates, then disjunctions of conjunctions of size $n^{O(1)}$. Since the disjunctions (20) of conjunctions are used as inputs to $\oplus_m$ gates, the only important things is the count (mod m) of how many are true. Since each such disjunction has either exactly zero or exactly one true input, we can "bypass" the disjunctions, and feed the conjunctions A_γ directly into the $\oplus_m$ gates without changing the value output by D. This makes D into a depth three circuit, with a majority gate, then a level of $\oplus_m$ gates, and then conjunctions of small size $O(n^{O(1)})$.

The circuit D computes the circuit value problem for a circuit as described by Ξ. To finalize D, specialize D to use the oracle Ξ which correctly encodes the direct connection language for C, by replacing each Boolean input ξ_i with the appropriate (hardwired) constant *True* or *False*. Identifying the ω_j's with the inputs x_j, D now computes the same Boolean function as C. The size of D is $2^{n^{O(1)}}$. Since $n = O(\log S)$, D has size $2^{(\log S)^{O(1)}}$ and so is quasipolynomially bounded by the size of C. The bottom of level of conjunctions in D have size $(\log S)^{O(1)}$, namely polylogarithmic in the size of C. This thus proves Theorem 1.

The proof of Theorem 2 is similar. $\mathrm{Val}_{d,m}$ is now a $\mathrm{ModPH}^{\Omega,\Xi}$ predicate. By Theorem 31, we can express "C accepts" by

$$C \text{ accepts} \Leftrightarrow N'(0^n) \geq \frac{3}{4} m^{q(n)}$$

$$C \text{ rejects} \Leftrightarrow N'(0^n) \leq \frac{1}{4} m^{q(n)}$$

where

$$N'(0^n) \;=\; (\text{ApxMaj-}\mathcal{C}_{[\mathbf{M}]} r, |r| {=} q(|\mathbf{z}|)) \psi'(0^n, r),$$

where ψ' is polynomial time (relative to Ω and Ξ), and q is a polynomial, and $\mathbf{M}$ is a fixed sequence $m_1^{s_1(n)}, \ldots, m_\ell^{s_\ell(n)}$ for primes m_i and polynomials $s_i(n)$.

We build again a circuit D which evaluates whether C accepts. The inputs to D are again the ω_j's and ξ_j's, and the size of D is $2^{n^{O(1)}}$. We form D as a depth three circuit, and subsequently we collapse it to depth two. The top gate of D has $2^{q(n)}$ inputs, one for each value of r. This gate is symmetric, computing iterated mod-counting and then taking the approximate majority. The gates at the second level are initially disjunctions of the form

$$\bigvee\{A_\gamma : \gamma \text{ is accepting}\} \tag{21}$$

where γ now ranges over accepting computations of $\Psi'(0^n, r)$. Each A_γ is a conjunction of size $n^{O(1)}$. As before, for a fixed value of r, at most one A_γ in the disjunction (21) can be true. Thus, we can replace each disjunction (21) which is an input to the top gate of D by the set of inputs A_γ. This replaces each input to the top gate by up to $2^{n^{O(1)}}$ many inputs, but does not change the value of the circuit (no matter what Ξ or Ω are). This also makes D have depth two. To finalize D we again replace the Boolean inputs ξ_i by the appropriate constants *True* or *False* which correctly encode the direct connection language for the circuit C. The resulting circuit D is equivalent to C with size quasipolynomially bounded by C. The size of D is $2^{n^{O(1)}}$; it is a depth two circuit with a symmetric gate as its output gate, and the remaining gates are conjunctions with fanin $n^{O(1)}$. This completes the proof of Theorem 2. □

In Theorem 1, the top gate is an approximate majority gate. By the fact [27, 20] that BPP $\in \Sigma^p_2 \cap \Pi^p_2$, similarly to Theorem 23, the majority gate can be replaced by either a conjunction of disjunctions, or a disjunction of conjunctions. This gives the following weaker form of the Beigel-Tarui theorem:

Theorem 37 *Fix $d \geq 1$ and prime $m \geq 2$. Suppose C is an* ACC *circuit of size S which uses* MOD *m gates. (That is, C is an* $\mathrm{AC}^0[m]$ *circuit.) Then there is an equivalent circuit C' of size $2^{(\log S)^{O(1)}}$ which has depth four. The first (input) level of C' contains $\wedge$ gates of size* $(\log S)^{O(1)}$*, the second level contains* MOD *m gates, and the top two levels can be either a conjunction of disjunctions, or a disjunction of conjunctions.*

Buss, Kołodziejczyk and Zdanowski [10] have shown that Theorem 37 can be proved in a version of bounded arithmetic that includes mod m quantifiers for m a prime, based on Jeřábek's bounded arithmetic theories for approximate counting [16,17]. They then used this to prove that constant-depth propositional proofs over the unbounded fanin connectives $\wedge$, $\vee$ and $\oplus_p$ can be simulated by quasipolynomial size propositional proofs of depth four (over the same connectives). On a related note, earlier work by Maciel and Pitassi [21] established that constant depth propositional proofs over the connectives $\wedge$, $\vee$, and $\oplus_{p^k}$ can be simulated by depth three proofs if exact counting (threshold) gates are also permitted. It is likely that, analogously to [10], the simulation of [21] can be "uniformized" by proving it in a version of bounded arithmetic augmented with exact counting quantifiers and modular counting quantifiers for prime moduli $p \geq 2$.

References

1. Allender, E.: A note on the power of threshold circuits. In: Proceedings 30th IEEE Symp. on Foundations of Computer Science (FOCS), pp. 580–584 (1989)
2. Allender, E.: The permanent requires large uniform threshold circuits. Chicago Journal of Theoretical Computer Science p. article 7 (1999)
3. Allender, E., Gore, V.: A uniform circuit lower bound for the permanent. SIAM J. on Computing **23**(5), 1026–1049 (1994)
4. Allender, E., Hertrampf, U.: Depth reduction for circuits of unbounded fan-in. Information and Computation **112**, 217–238 (1994)
5. Arora, S., Barak, B.: Computational Complexity: A Modern Approach. Cambridge University Press (2009)
6. Barrington, D.A.M.: Quasipolynomial size circuit classes. In: Proc. 7th Structure in Complexity Conference, pp. 86–93 (1992)
7. Beigel, R., Tarui, J.: On ACC. Computational Complexity **4**, 350–366 (1994)
8. Beigel, R., Tarui, J., Toda, S.: On probabilistic ACC circuits with an exact-threshold output gate. In: Algorithms and Computation: Third Intl. Symp., ISAAC'92, Lecture Notes in Computer Science 650, pp. 420–429 (1992)
9. Buss, S.R.: Bounded Arithmetic. Bibliopolis, Naples, Italy (1986). Revision of 1985 Princeton University Ph.D. thesis
10. Buss, S.R., Kołodziejczyk, L.A., Zdanowski, K.: Collapsing modular counting in bounded arithmetic and constant depth propositional proofs. Transactions of the AMS **367**, 7517–7563 (2015)
11. Chen, S., Papakonstantinou, P.A.: Depth-reduction for composites. In: Proc. of the 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS'16), pp. 99–108. IEEE Computer Society (2016)
12. Fortnow, L.: A simple proof of Toda's theorem. Theory of Computing **5**, 135–140 (2009)
13. Furst, M., Saxe, J.B., Sipser, M.: Parity, circuits and the polynomial-time hierarchy. Math. Systems Theory **17**, 13–27 (1984)
14. Green, F., Köbler, J., Regan, K.W., Schwentick, T., Torán, J.: The power of the middle bit of a #P function. Journal of Computer and System Sciences **50**, 456–467 (1995)
15. Hansen, K.A., Koucký, M.: A new characterization of ACC^0 and probabilistic CC^0. Conputational Complexity **19**, 211–234 (2010)
16. Jeřábek, E.: Approximate counting in bounded arithmetic. Journal of Symbolic Logic **72**(3), 959–993 (2007)
17. Jeřábek, E.: Approximate counting by hashing in bounded arithmetic. Journal of Symbolic Logic **74**(3), 829–860 (2009)
18. Kannan, R., Venkateswaran, H., Vinay, V., Yao, A.C.: A circuit-based proof of Toda's theorem. Information and Computation **104**(2), 271–276 (1993)
19. Krajíček, J.: Bounded Arithmetic, Propositional Calculus and Complexity Theory. Cambridge University Press, Heidelberg (1995)
20. Lautemann, C.: BPP and the polynomial hierarchy. Information Processing Letters **17**(4), 215–217 (1983)
21. Maciel, A., Pitassi, T.: Towards lower bounds for bounded-depth Frege proofs with modular connectives. In: P.W. Beame, S.R. Buss (eds.) Proof Complexity and Feasible Arithmetics, pp. 195–227. American Mathematical Society (1998)
22. Papadimitriou, C.H., Zachos, S.K.: Two remarks on the power of counting. In: Proc. 6th Gesellschaft für Informatik Conference on Theoretical Computer Science, Lecture Notes in Computer Science 145, pp. 269–276. Springer-Verlag, Berlin (1983)
23. Paris, J.B., Wilkie, A.J.: Δ_0 sets and induction. In: W. Guzicki, W. Marek, A. Pelc, C. Rauszer (eds.) Open Days in Model Theory and Set Theory, pp. 237–248 (1981)
24. Razborov, A.A.: Lower bounds on the size of bounded depth networks over a complete basis with logical addition. Matematicheskie Zametki **41**, 598–607 (1987). English translation in *Mathematical Notes of the Academy of Sciences of the USSR* 41 (1987) 333-338
25. Ruzzo, W.L.: On uniform circuit complexity. J. Comput. Syst. Sci. **22**, 365–383 (1981)
26. Schöning, U.: Probabilistic complexity classes and lowness. Journal of Computer and System Sciences **39**, 84–100 (1989)

27. Sipser, M.: A complexity theoretic approach to randomness. In: Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing, pp. 330–335. ACM Press (1983)
28. Smolensky, R.: Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In: Proceedings of the Nineteenth Annual ACM Symposium on the Theory of Computing, pp. 77–82. ACM Press (1987)
29. Straubing, H., Thérien, D.: A note on MOD_p-MOD_m circuits. Theory of Computing Systems **39**(5), 699–706 (2006)
30. Tarui, J.: Probabilistic polynomials, AC^0 functions and the polynomial-time hierarchy. Theoretical Computer Science **113**, 167–183 (1993)
31. Toda, S.: PP is as hard as the polynomial-time hierarchy. SIAM Journal on Computing **20**(5), 865–877 (1991)
32. Toda, S., Ogiwara, M.: Counting classes are at least as hard as the polynomial-time hierarchy. In: Proc. 6th Structure in Complexity Theory Conference, pp. 2–12 (1991)
33. Valiant, L.G., Vazirani, V.V.: NP is a easy as detecting unique solutions. Theoretical Computer Science **47**, 85–93 (1986)
34. Williams, R.: Non-uniform ACC circuit lower bounds (2011). To appear in *Journal of the ACM.* Shorter version appeared in *26th IEEE Conference on Computational Complexity (CCC), pp. 115-125, 2011*
35. Yao, A.C.C.: On ACC and threshold circuits. In: Proc. 31st IEEE Symp. on Foundations of Computer Science (FOCS), pp. 619–627 (1990)