Counting Classes: Thresholds, Parity, Mods, and Fewness*

Richard Beigel† John Gill‡

June 10, 1996

Abstract

Counting classes consist of languages defined in terms of the number of accepting computations of nondeterministic polynomial-time Turing machines. Well known examples of counting classes are NP, co-NP, $\oplus$P, and PP. Every counting class is a subset of $P^{\#P[1]}$, the class of languages computable in polynomial time using a single call to an oracle capable of determining the number of accepting paths of an NP machine.

Using closure properties of #P, we systematically develop a complexity theory for counting classes defined in terms of thresholds and moduli. An unexpected result is that $\mathrm{MOD}_{k^i}\mathrm{P} = \mathrm{MOD}_k\mathrm{P}$ for prime k.

Finally, we improve a result of Cai and Hemachandra by showing that recognizing languages in the class Few is as easy as distinguishing uniquely satisfiable formulas from unsatisfiable formulas (or detecting unique solutions, as in [28]).

1. Introduction

Valiant [27] defined the class #P of functions whose values equal the number of accepting paths of polynomial-time bounded nondeterministic Turing machines. Many interesting classes, such as PP [9, 21] and $\oplus$P [17, 10], are subclasses of $P^{\#P[1]}$, the class of languages computable in polynomial-time with one query to a function in #P.

Since a PP machine accepts when more than half of its paths accept, PP can be considered equivalent to computing the high-order bit of a #P function. Since a $\oplus$P machine accepts when an odd number of its paths accept, $\oplus$P can be considered equivalent to computing the low-order bit of a #P function.

It is natural to consider the relative difficulty of computing different bits of information about a #P function. In this paper, we look at subclasses of $P^{\#P[1]}$ that are defined in terms of thresholds and in terms of moduli. We obtain several relationships among these classes. (Some separations relative to an oracle appear in [4].) Next, we show that the class FewP is a subset of $\mathrm{MODZ}_k\mathrm{P}$. Finally we show that every language in the class Few is as

*A preliminary version of this paper appeared in the Proceedings of the 7th Annual Symposium on Theoretical Aspects of Computer Science, 1990.

†Dept. of Computer Science, P.O. Box 2158, Yale Station, New Haven, CT 06520-2158, USA. beigel-richard@cs.yale.edu. Supported in part by the National Science Foundation under grants CCR-8808949 and CCR-8958528. Some of this research was performed while this author was at the Johns Hopkins University.

‡Dept. of Electrical Engineering, Stanford University, Stanford, CA 94305-4055, USA. gill@isl.stanford.edu. Supported in part by the Defense Advanced Research Projects Agency under contract J-FBI-89-101.

easy as distinguishing uniquely satisfiable formulas from unsatisfiable formulas; along with our closure properties, this yields a new proof of Cai and Hemachandra's theorem that the class Few is a subset of $\mathrm{MOD}_k\mathrm{P}$.

Notation 1.

- PF denotes the class of polynomial-time computable functions.
- NPM denotes the class of all nondeterministic polynomial-time bounded Turing machines.
- N denotes an element of NPM.
- x denotes an input string.
- n or $|x|$ denote the length of the input x.
- $\textsc{paths}(N, x)$ denotes the set of computation paths of N on x.
- $\textsc{accept}(N, x)$ denotes the set of accepting paths in $\textsc{paths}(N, x)$.
- ρ denotes a computation path.
- $|S|$ denotes the cardinality of the set S.
- $\langle \cdot, \cdot \rangle$ denotes a pairing function that is computable and invertible in polynomial time.

Note that for fixed N, the predicates

$$\rho \in \textsc{paths}(N, x) \quad \text{and} \quad \rho \in \textsc{accept}(N, x)$$

are polynomial-time computable.

2. Closure properties of #P

The class #P was defined by Valiant [27].

Definition 2. [Valiant] A function f belongs to #P if there is a nondeterministic polynomial-time machine N such that $f(x)$ is the number of accepting computations of N with input x.

Because we will be defining new classes in terms of #P, it is valuable to obtain some closure properties of #P.

Property 3.

(i) #P *is closed under addition.*

(ii) #P *is closed under multiplication.*

Proof: Let f_1 and f_2 be functions in #P via N_1 and N_2, respectively.

(i) Define N to behave as follows on input x:

Step 1: Guess $c \in \{1, 2\}$;

Step 2: If $c = 1$ then guess a path $\rho \in \text{PATHS}(N_1, x)$;

Step 3: If $c = 2$ then guess a path $\rho \in \text{PATHS}(N_2, x)$;

Step 4: Accept if ρ is an accepting path.

Then $f_1 + f_2$ is in #P via N.

(ii) Define N to behave as follows on input x:

Step 1: Guess $(\rho_1, \rho_2) \in \text{PATHS}(N_1, x) \times \text{PATHS}(N_2, x)$;

Step 2: Accept if both ρ_1 and ρ_2 are accepting paths.

Then $f_1 \cdot f_2$ is in #P via N.

∎

We will see below that this property is true of more general sums and products.

Valiant [26] defined the class UP of languages accepted by NP machines that have exactly one accepting path on strings in the language (and of course no accepting paths on strings not in the language). Machines with at most one accepting path are called *categorical*. By definition, UP $\subseteq$ NP.

Definition 4. [Valiant] UP consists of those languages L for which there exists a nondeterministic polynomial-time machine N such that for all x

$$|\text{ACCEPT}(N, x)| = \begin{cases} 1 & \text{if } x \in L \\ 0 & \text{if } x \notin L. \end{cases}$$

The class UPF is defined by analogy with the class UP. UPF is the class of functions computed by an NP machine with exactly one accepting computation.

Definition 5. A function f belongs to UPF if there exists an NPM N such that for all x, $|\text{ACCEPT}(N, x)| = 1$ and $f(x)$ is equal to the output produced on the unique path in $\text{ACCEPT}(N, x)$.

The following result generalizes the closure of #P under addition and multiplication.

Property 6. *Let f be a function in* #P.

(i) *For any $L \in$ UP and any integer k, the following function of x belongs to* #P*:*

$$\sum_{|y|=|x|^k,\ \langle x,y\rangle \in L} f(\langle x, y\rangle).$$

(ii) *Let q be a function in* UPF *bounded by a polynomial in $|x|$. Then the following function of x belongs to* #P*:*

$$\prod_{1 \le y \le q(x)} f(\langle x, y\rangle).$$

Proof: Suppose that $f \in$ #P via N.

(i) Define N' to behave as follows on input x:

Step 1: Guess y such that $|y| = |x|^k$;

Step 2: Categorically check that $\langle x, y\rangle \in L$;[1]

Step 3: Guess $\rho \in \text{PATHS}(N, \langle x, y\rangle)$;

Step 4: Accept if $\rho \in \text{ACCEPT}(N, \langle x, y\rangle)$.

(ii) Define N' to behave as follows on input x:

Step 1: Categorically compute $q(|x|)$.

Step 2: Guess $\rho_1 \in \text{PATHS}(N, \langle x, 1\rangle)$, ..., $\rho_{q(x)} \in \text{PATHS}(N, \langle x, q(x)\rangle)$.

Step 3: Accept if $\rho_1 \in \text{ACCEPT}(N, \langle x, 1\rangle)$, ..., $\rho_{q(x)} \in \text{ACCEPT}(N, \langle x, q(x)\rangle)$.

▮

Note that part (ii) of Property 3 also follows from part (i) of this result, because multiplication is the same as repeated addition.

Definition 7. Binomial and multinomial coefficients are defined as follows.

(i) $\binom{n}{k}$ is the coefficient of x^k in the expansion of $(x+1)^n$.

(ii) $\binom{n}{k_1, \ldots, k_m}$ is the coefficient of $x_1^{k_1} \cdots x_m^{k_m}$ in the expansion of $(x_1 + \cdots + x_m)^n$.

In parts (ii) and (iii) of the next result we obtain closure properties of #P that do not follow from closure under addition and multiplication. We will see later, in Theorems 30 and 39, that these closure properties have surprising consequences.

Property 8. *If f belongs to #P and if $g, g_1, \ldots, g_k$ are functions in* UPF *bounded by a polynomial in $|x|$, then the following functions of x are in #P:*

(i) $f(x)^{g(x)}$.

(ii) $\binom{f(x)}{g(x)}$.

(iii) $\binom{f(x)}{g_1(x), \ldots, g_k(x), f(x) - \sum_{1\le i\le k} g_i(x)}$.

Proof: Let $f \in$ #P via N. We show that the functions of each part of this result are in #P via the following nondeterministic machines:

(i) Define N' to behave as follows on input x:

[1] More formally, we nondeterministically guess a potential witness to $\langle x, y\rangle \in L$ and we test the witness nondeterministically. If we have guessed an invalid witness then we reject. Otherwise we continue the algorithm. If a valid witness exists, then it will be unique so this step will not affect the number of accepting paths. We use categorical checking and computing in this way in the remainder of this section.

Step 1: Categorically compute $g(x)$;

Step 2: Guess an ordered $g(x)$-tuple (allowing duplicates) of paths in $\text{PATHS}(N,x)$;

Step 3: Accept if all $g(x)$ paths accept.

(This also follows from Property 6(ii).)

(ii) Define N' to behave as follows on input x:

Step 1: Categorically compute $g(x)$;

Step 2: Guess a set containing $g(x)$ (distinct) paths in $\text{PATHS}(N,x)$;

Step 3: Accept if all $g(x)$ paths accept.

(iii) Define N' to behave as follows on input x:

Step 1: Categorically compute $g_1(x), \ldots, g_k(x)$;

Step 2: Guess an ordered k-tuple of pairwise-disjoint subsets $S_1, \ldots, S_k$ of $\text{PATHS}(N,x)$ such that $|S_i| = g_i(x)$ for each i.

Step 3: Accept if all paths in $S_1 \cup \cdots \cup S_k$ accept.

∎

Property 9. *Let f be a function in #P bounded by a polynomial in $|x|$.*

(i) *The function $2^{f(x)} - 1$ belongs to #P.*

(ii) *More generally, for every integer $k \geq 2$ the function $k^{f(x)} - 1$ belongs to #P.*

Proof: Let $f \in \#\text{P}$ via N and let q be a polynomial such that $f(x) \leq q(|x|)$ for all x.

(i) Let N' behave as follows on input x:

Step 1: Guess a nonempty set $S \subseteq \text{PATHS}(N,x)$ such that $|S| \leq q(|x|)$.

Step 2: Accept if $S \subseteq \text{ACCEPT}(N,x)$.

(This also follows from Property 8(ii) and Property 6(i) by the binomial theorem.)

(ii) A *multiset* is a "set" that may contain duplicates; that is, a multiset is a function from a universe to the nonnegative integers. Let N' behave as follows on input x:

Step 1: Guess a nonempty multiset S of paths belonging to $\text{PATHS}(N,x)$ such that $|S| \leq (k-1)q(|x|)$ and such that no element appears in S more than $k-1$ times.

Step 2: Accept if every element of S belongs to $\text{ACCEPT}(N,x)$.

(This also follows from Property 8(iii) and Property 6(i) by the multinomial theorem.)

∎

3. Counting Classes

Many well known complexity classes can be defined in terms of nondeterministic polynomial-time machines by appropriate interpretation of the results of all possible computation paths. NP, PP, US [6], and members of the polynomial-time hierarchy [16, 23] are examples of such complexity classes. Counting classes consist of languages in which acceptance is determined by the *number* of accepting computations.

Definition 10. For a relation $R(x, \alpha, \pi)$ we define $\mathrm{CP}_{R(x,\alpha,\pi)}$ to be the class of languages L for which

$$(\exists N)(\forall x)[x \in L \iff R(x, |\textsc{accept}(N, x)|, |\textsc{paths}(N, x)|)].$$

For example, NP is the counting class defined by $R(x, \alpha, \pi) \equiv \alpha > 0$, and PP is the counting class defined by $R(x, \alpha, \pi) \equiv \alpha > \frac{1}{2}\pi$.

Some researchers have considered machines that are defined to accept when exactly $t(x)$ paths accept; others have considered machines that accept when at least $t(x)$ paths accept. Still others take $t(x)$ to be a fraction of all paths, rather than an absolute number. Classes that arise from such considerations include US [6] and PP [9, 21]. The following definition introduces notation for some of these counting classes.

Definition 11.

- $\mathrm{CP}_{=f(x)} = \mathrm{CP}_{\alpha=f(x)}$.
- $\mathrm{CP}_{=f(x,\pi)} = \mathrm{CP}_{\alpha=f(x,\pi)}$.
- $\mathrm{CP}_{\geq f(x)} = \mathrm{CP}_{\alpha\geq f(x)}$.
- $\mathrm{CP}_{\geq f(x,\pi)} = \mathrm{CP}_{\alpha\geq f(x,\pi)}$.
- $\mathrm{CP}_{=} = \bigcup_{f\in\mathrm{PF}} \mathrm{CP}_{=f(x)}$.
- $\mathrm{CP}_{\geq} = \bigcup_{f\in\mathrm{PF}} \mathrm{CP}_{\geq f(x)}$.
- $\oplus\mathrm{P} = \mathrm{CP}_{\alpha\equiv 1 \pmod 2}$.

In the notation of the above definition, NP is $\mathrm{CP}_{\geq 1}$, co-NP is $\mathrm{CP}_{=\pi}$, US is $\mathrm{CP}_{=1}$, and PP is $\mathrm{CP}_{\geq\lfloor\pi/2\rfloor+1}$.

The class $\mathrm{CP}_{=}$ was first studied by Russo [19] and by Wagner [29]. The class $\mathrm{CP}_{\geq}$ was defined by Wagner [29], who showed it is equal to PP. The classes $\mathrm{CP}_{=f(x)}$ and $\mathrm{CP}_{\geq f(x)}$ were studied by Torán [25]. The class $\oplus$P was defined by Papadimitriou and Zachos [17] and by Goldschlager and Parberry [10].

4. Thresholds

In this section, we consider machines whose acceptance is based on the number of accepting paths reaching some threshold. Thresholds were studied in [21]. We present one framework in which computational power is a monotone function of the threshold, and another framework in which computational power is independent of the threshold.

Theorem 12. *If f and g are polynomial-time computable functions such that $f(x) \leq g(x)$ for every x, then*

(i) $\mathrm{CP}_{=f(x)} \subseteq \mathrm{CP}_{=g(x)}$.

(ii) $\mathrm{CP}_{\geq f(x)} \subseteq \mathrm{CP}_{\geq g(x)}$.

Proof: Let N be a nondeterministic polynomial-time machine. Since #P is closed under addition, there is a machine N' such that

$$|\textsc{accept}(N', x)| = |\textsc{accept}(N, x)| + (g(x) - f(x)).$$

Then

(i) $|\textsc{accept}(N, x)| = f(x)$ iff $|\textsc{accept}(N', x)| = g(x)$, and

(ii) $|\textsc{accept}(N, x)| \geq f(x)$ iff $|\textsc{accept}(N', x)| \geq g(x)$.

∎

The containments above need not be proper. For example, if $g(x)$ is a polynomial in $|x|$ then $\mathrm{CP}_{=1} = \mathrm{CP}_{=g(x)}$ and $\mathrm{CP}_{\geq 1} = \mathrm{CP}_{\geq g(x)}$. On the other hand, if f is much less than g then there is an oracle that makes the containments above proper.

Proposition 13. *Suppose that for all strings x, $f(x) \leq g(x)$. Suppose further that for every polynomial p there exist infinitely many strings x such that $(f(x)+1)p(|x|) < g(x)$ and $(f(x)+1)p(|x|) + g(x) \leq 2^{|x|}$. Then there exists an oracle A such that*

(i) $\mathrm{CP}^A_{=f(x)} \subset \mathrm{CP}^A_{=g(x)}$.

(ii) $\mathrm{CP}^A_{\geq f(x)} \subset \mathrm{CP}^A_{\geq g(x)}$.

Proof: The containments follow because the proof of Theorem 12 relativizes. We construct oracles relative to which they are proper.

Let N_e^B denote the eth nondeterministic polynomial-time Turing machine with oracle B (assumed to run in time $p_e(|x|) = |x|^e + e$ on input x). Fix a two-character alphabet Σ. Let $A^{=n}$ denote the set of strings in A having length n.

(i) We use a simple forcing argument in order to construct A such that for every e there exists x such that

$$|\textsc{accept}(N_e^A, x)| \neq f(x) \iff |A^{=|x|}| = g(x).$$

Thus $\{x : |A^{=|x|}| = g(x)\} \notin \mathrm{CP}^A_{=f(x)}$, implying the desired separation.

Stage 0: Let $A = \emptyset$ and $n = 0$.

Stage e: Find a string x such that $|x| > n$, $(f(x)+1)p_e(|x|) < g(x)$, and $(f(x)+1)p_e(|x|) + g(x) \leq 2^{|x|}$.

step 1: If there exists a set $B \subseteq \Sigma^{|x|}$ such that

$$|\text{ACCEPT}(N_e^{A \cup B}, x)| \geq f(x) + 1,$$

then let B be such a set having minimum cardinality. In particular, $|B| \leq f(x+1)p_e(|x|)$. Let C contain $g(x) - |B|$ strings of length $|x|$ that are not queried by the lexically first $f(x)+1$ paths in $|\text{ACCEPT}(N_e^{A \cup B}, x)|$. (This set C exists because $(f(x)+1)p_e(|x|) \leq g(x)$ and $(f(x)+1)p_e(|x|)+g(x) \leq 2^{|x|}$.) Let $A = A \cup B \cup C$, let $n = p_e(|x|)$, and go to Stage $e+1$. Otherwise go to step 2.

step 2: If there exists a set $B \subseteq \Sigma^{|x|}$ such that

$$|\text{ACCEPT}(N_e^{A \cup B}, x)| \geq f(x),$$

then let B be the smallest such set. Note that $|B| < g(x)$ because $f(x)p_e(|x|) < g(x)$. Let $A = A \cup B$, let $n = p_e(|x|)$, and go to Stage $e+1$. Otherwise, just go to step 3.

step 3: Let C contain exactly $g(x)$ strings of length $|x|$, let $A = A \cup C$, let $n = p_e(|x|)$, and go to Stage $e+1$.

If the condition of step 1 is met then $|\text{ACCEPT}(N_e^A, x)| > f(x)$, but $|A^{=|x|}| = g(x)$. Otherwise, if the condition of step 2 is met then $|\text{ACCEPT}(N_e^A, x)| = f(x)$, but $|A^{=|x|}| < g(x)$. Otherwise, step 3 ensures that $|\text{ACCEPT}(N_e^A, x)| < f(x)$, but $|A^{=|x|}| = g(x)$.

(ii) Similar to part (i). Just skip step 1 of the construction.

∎

It would be interesting to know when $\text{CP}_{=f(x)} = \text{CP}_{=f(x)+1}$ and when $\text{CP}_{\geq f(x)} = \text{CP}_{\geq f(x)+1}$.

In the remainder of this section, we consider machines that are defined to accept when the number of accepting paths exceeds some function of the total number of paths. In this framework, we show that computational power does not depend on the threshold.

Lemma 14. [2] *Let r and s be polynomial-time computable, rational-valued functions such that $0 < r(x) < 1$ and $0 < s(x) < 1$ for all x.*

(i) $\text{CP}_{=r(x)\pi} = \text{CP}_{=s(x)\pi}$.

(ii) $\text{CP}_{\geq r(x)\pi} = \text{CP}_{\geq s(x)\pi}$.

(Since $r(x)\pi$ and $s(x)\pi$ need not be integers, part (i) may not be entirely satisfying.)

[2] In the STACS '90 proceedings, we noted in passing that if two threshold functions are not polynomially related then the corresponding classes are distinct; this claim is not true in the current framework. We also remarked that if two threshold functions are polynomially related then the corresponding classes are equal; we have established this claim only for the current framework, where it is clearly an understatement. We regret any confusion that we may have caused.

Proof: We show that $\mathrm{CP}_{=r(x)\pi} \subseteq \mathrm{CP}_{=s(x)\pi}$ and $\mathrm{CP}_{\geq r(x)\pi} \subseteq \mathrm{CP}_{\geq s(x)\pi}$. The equalities then follow by symmetry. Let $\mu(x)$ be the least common denominator of $s(x)$ and $r(x)$. By interchanging accepting and rejecting paths if necessary, we may assume without loss of generality that $0 < s(x) \leq r(x) < 1$. Let N be a nondeterministic polynomial-time machine. Let N' behave as follows on input x:

Step 1: Guess t such that $1 \leq t \leq \mu(x)r(x)$;

Step 2: Guess $\rho \in \textsc{paths}(N, x)$;

Step 3: If $t \leq \mu(x)s(x)$ and $\rho \in \textsc{accept}(N, x)$ then accept, else reject;

Then

$$\begin{aligned} |\textsc{paths}(N', x)| &= \mu(x)r(x)|\textsc{paths}(N, x)|, \\ |\textsc{accept}(N', x)| &= \mu(x)s(x)|\textsc{accept}(N, x)|. \end{aligned}$$

Hence

$$\frac{|\textsc{accept}(N', x)|}{|\textsc{paths}(N', x)|} = \frac{s(x)}{r(x)} \cdot \frac{|\textsc{accept}(N, x)|}{|\textsc{paths}(N, x)|}.$$

Therefore,

$$\begin{aligned} |\textsc{accept}(N, x)| = r(x)|\textsc{paths}(N, x)| &\iff |\textsc{accept}(N', x)| = s(x)|\textsc{accept}(N', x)|, \\ |\textsc{accept}(N, x)| \geq r(x)|\textsc{paths}(N, x)| &\iff |\textsc{accept}(N', x)| \geq s(x)|\textsc{accept}(N', x)|. \end{aligned}$$

Thus $\mathrm{CP}_{=r(x)\pi} \subseteq \mathrm{CP}_{=s(x)\pi}$ and $\mathrm{CP}_{\geq r(x)\pi} \subseteq \mathrm{CP}_{\geq s(x)\pi}$. ∎

Hence we have the following partial characterization:

Theorem 15. *Let r be a polynomial-time computable rational-valued function. Then*

(i) $$\mathrm{CP}_{=r(x)\pi} = \begin{cases} \text{co-NP} & \text{if } r(x) \text{ is identically } 0 \text{ or } r(x) \text{ is identically } 1, \\ \mathrm{CP}_{=} & \text{if } 0 < r(x) < 1 \text{ for all } x. \end{cases}$$

(ii) $$\mathrm{CP}_{\geq r(x)\pi} = \begin{cases} \{\Sigma^*\} & \text{if } r(x) \text{ is identically } 0, \\ \text{co-NP} & \text{if } r(x) \text{ is identically } 1, \\ \mathrm{PP} & \text{if } 0 < r(x) < 1 \text{ for all } x. \end{cases}$$

5. Number Theory

In studying $\oplus\mathrm{P}$ and other subclasses of $\mathrm{P}^{\#\mathrm{P}[1]}$ defined in terms of moduli, we will need to use some facts from number theory.

Theorem 16 (Fermat). *Let p be prime. Then*

$$a^{p-1} \equiv \begin{cases} 0 \pmod{p} & \text{if } a \equiv 0 \pmod{p} \\ 1 \pmod{p} & \text{otherwise.} \end{cases}$$

We obtain a useful corollary about square-free integers.

Definition 17. An integer is *square-free* if it is not divisible by the square of any positive integer larger than one.

We write $p \mid k$ to denote that p is a divisor of k.

Corollary 18. *Let k be odd and square-free. Let*

$$\lambda = \prod_{p \mid k,\ p \text{ prime}} (p-1).$$

Then $u^\lambda + v^\lambda \equiv 0 \pmod{k}$ if and only if $u \equiv v \equiv 0 \pmod{k}$.

Proof: If $u \equiv v \equiv 0 \pmod{k}$ then obviously $u^\lambda + v^\lambda \equiv 0 \pmod{k}$. For the converse, assume without loss of generality that $u \not\equiv 0 \pmod{k}$. Since k is odd and square-free, there is an odd prime p such that $p \mid k$, but $u \not\equiv 0 \pmod{p}$. Then $u^\lambda \equiv (u^{p-1})^{\lambda/(p-1)} \equiv 1^{\lambda/(p-1)} \equiv 1 \pmod{p}$, where the second-to-last equivalence is by Fermat's theorem. Similarly, v^λ is either 0 or 1 modulo p. Therefore $u^\lambda + v^\lambda$ is either 1 or 2 modulo p, and so $u^\lambda + v^\lambda$ is not divisible by p, since $p > 2$. Since p is a divisor of k, we conclude that $u^\lambda + v^\lambda$ is not divisible by k. ∎

Theorem 19 (Kummer). *Let p be prime, and let $n = a + b$. Then*

$$\binom{n}{a} \equiv 0 \pmod{p^c}$$

if and only if the number of carries when adding a to b in base p is at least c.

See [12, Exercise 1.2.6-11] for a proof. We will need only the following corollary of Kummer's theorem.

Corollary 20. *Let p be prime. Then*

$$\binom{n}{p^k} \equiv 0 \pmod{p}$$

if and only if the coefficient of p^k in the base-p expansion of n is zero.

Proof: Suppose that the coefficient of p^k in the base-p expansion of n is not zero. Using base p arithmetic, we have

$$\begin{array}{rcccccccccc}
n & = & d_m p^m & + \cdots + & d_k p^k & + & d_{k-1} p^{k-1} & + \cdots + & d_0 p^0 & , \\
p^k & = & & + & p^k & + & 0 & + \cdots + & 0 & , \\
n - p^k & = & d_m p^m & + \cdots + & (d_k - 1) p^k & + & d_{k-1} p^{k-1} & + \cdots + & d_0 p^0 & ,
\end{array}$$

so p^k can be added to $n - p^k$ in base p without carrying. On the other hand, if the coefficient of p^k in the base-p expansion of n is zero, then the coefficient of p^k in the base-p expansion of $n - p^k$ must be $p - 1$; therefore there must be a carry when adding 1 to $p - 1$ in the p^k's position. The corollary follows from Kummer's theorem with $c = 1$. ∎

6. MOD Classes

In this section, we define $\oplus$P and the related classes $\mathrm{MOD}_k\mathrm{P}$ for $k > 2$. Complete problems for $\mathrm{MOD}_k\mathrm{P}$ include the set of all formulas f such that the number of satisfying assignments of f is not congruent to zero modulo k. We prove closure properties and relations among these classes. (Similar results for circuit complexity were stated without proof in [22].) In Section 9, these closure properties will permit a simple proof that Cai and Hemachandra's class Few is a subset of $\mathrm{MOD}_k\mathrm{P}$.

The class $\oplus$P was defined by Papadimitriou and Zachos [17]. Recall that $\oplus\mathrm{P} = \mathrm{CP}_{\alpha\equiv 1 \pmod 2}$. We will consider the analogous classes defined in terms of arbitrary moduli.

Definition 21. For every positive integer k,

$$\mathrm{MOD}_k\mathrm{P} = \mathrm{CP}_{\alpha\not\equiv 0 \pmod k}.^3$$

That is, a language L belongs to $\mathrm{MOD}_k\mathrm{P}$ if there exists N such that

$$x \in L \iff |\textsc{accept}(N,x)| \not\equiv 0 \pmod k.$$

According to the above definition, if L belongs to $\mathrm{MOD}_k\mathrm{P}$ via N, then $x \notin L$ when N has zero accepting paths (in analogy to NP). We show below that there is nothing special about the residue $0 \pmod k$; any other residue gives rise to the same class $\mathrm{MOD}_k\mathrm{P}$.

Theorem 22. *For every integer j,*

$$\mathrm{MOD}_k\mathrm{P} = \mathrm{CP}_{\alpha\not\equiv j \pmod k}.$$

Proof: Without loss of generality, assume $0 \le j < k$. Let

$$\mathcal{C} = \mathrm{CP}_{\alpha\not\equiv j \pmod k}.$$

Let $L \in \mathrm{MOD}_k\mathrm{P}$ via N. Since #P is closed under addition, there is a machine $N' \in \mathrm{NPM}$ such that for all x

$$|\textsc{accept}(N',x)| = |\textsc{accept}(N,x)| + j,$$

so

$$|\textsc{accept}(N',x)| \not\equiv j \pmod k \iff |\textsc{accept}(N,x)| \not\equiv 0 \pmod k.$$

Therefore $\mathrm{MOD}_k\mathrm{P} \subseteq \mathcal{C}$.

Conversely, let $L \in \mathcal{C}$ via N. There is a machine $N' \in \mathrm{NPM}$ such that for all x

$$|\textsc{accept}(N',x)| = |\textsc{accept}(N,x)| + k - j,$$

so

$$|\textsc{accept}(N,x)| \not\equiv j \pmod k \iff |\textsc{accept}(N',x)| \not\equiv 0 \pmod k.$$

Therefore $\mathcal{C} \subseteq \mathrm{MOD}_k\mathrm{P}$. ∎

When k is prime, we have the following normal form for languages in $\mathrm{MOD}_k\mathrm{P}$:

[3] Since a MOD_k gate outputs 1 whenever the sum of its inputs is $0 \pmod k$, researchers in circuit complexity may prefer to define $\mathrm{MOD}_k\mathrm{P} = \mathrm{CP}_{\alpha\equiv 0 \pmod k}$. When k is prime we will show that $\mathrm{MOD}_k\mathrm{P}$ is closed under complement, so that the definitions are equivalent. When k is composite, we do not believe that $\mathrm{MOD}_k\mathrm{P}$ is closed under complement; and there are oracles that back up our belief [5]. We prefer our definition because we want machines to reject when all paths reject.

Theorem 23. *Let k be prime.*

(i) *A language L is in* $\mathrm{MOD}_k\mathrm{P}$ *iff there there exists N such that for every x*

$$|\textsc{accept}(N,x)| \equiv \begin{cases} 1 \pmod{k} & \text{if } x \in L \\ 0 \pmod{k} & \text{if } x \notin L. \end{cases}$$

(ii) *More generally, if $i \not\equiv j \pmod{k}$, then $L \in \mathrm{MOD}_k\mathrm{P}$ iff there there exists N such that for every x*

$$|\textsc{accept}(N,x)| \equiv \begin{cases} i \pmod{k} & \text{if } x \in L \\ j \pmod{k} & \text{if } x \notin L. \end{cases}$$

Proof:

(i) If such a machine N exists then $L \in \mathrm{MOD}_k\mathrm{P}$ by definition. Conversely, assume that $L \in \mathrm{MOD}_k\mathrm{P}$ via a machine N. By Property 8(i), there is a machine N' such that for all x

$$\begin{aligned} |\textsc{accept}(N',x)| &= |\textsc{accept}(N,x)|^{k-1} \\ &\equiv \begin{cases} 1 \pmod{k} & \text{if } |\textsc{accept}(N,x)| \not\equiv 0 \pmod{k} \\ 0 \pmod{k} & \text{if } |\textsc{accept}(N,x)| \equiv 0 \pmod{k}, \end{cases} \end{aligned}$$

by Fermat's theorem.

(ii) If such a machine N exists then we can add $(k-j) \bmod k$ to the number of accepting paths to obtain $L \in \mathrm{MOD}_k\mathrm{P}$. Conversely, assume that $L \in \mathrm{MOD}_k\mathrm{P}$ via the machine N' constructed in part (i). Since #P is closed under addition and multiplication, there is a machine N'' such that for all x

$$\begin{aligned} |\textsc{accept}(N'',x)| &= (i-j)\cdot|\textsc{accept}(N',x)| + j \\ &\equiv \begin{cases} i \pmod{k} & \text{if } |\textsc{accept}(N',x)| \equiv 1 \pmod{k} \\ j \pmod{k} & \text{if } |\textsc{accept}(N',x)| \equiv 0 \pmod{k}. \end{cases} \end{aligned}$$

∎

In [17], Papadimitriou and Zachos proved that the class $\oplus\mathrm{P}$ is equal to $\oplus\mathrm{P}^{\oplus\mathrm{P}}$. We will prove the same for $\mathrm{MOD}_k\mathrm{P}$ whenever k is prime. First, it will be helpful to define a generalization of Long's [15] nondeterministic many-one reducibility.

Definition 24. Let Q be any predicate. $A \leq_{\mathrm{m}}^{\mathrm{CP}_{Q(\alpha)}} B$ if there exists $f \in \mathrm{PF}$ and a constant k such that

$$x \in A \iff Q(|\{y : |y| = |x|^k \text{ and } f(x,y) \in B\}|).$$

For the sake of intuition, consider a machine $N \in \mathrm{NPM}^B$ that guesses a string y of length $|x|^k$ and accepts if $f(x,y) \in B$. The machine N $\leq_{\mathrm{m}}^{\mathrm{CP}_{Q(\alpha)}}$-reduces A to B if $x \in A$ whenever the number of accepting paths of N on input x satisfies Q.

Proposition 25. *If $\mathcal{C}$ is closed under complement, join, polynomial-time conjunctive reducibility, and $\leq_{\mathrm{m}}^{\mathrm{CP}_{Q(\alpha)}}$-reducibility, then $\mathcal{C} = \mathrm{CP}_{Q(\alpha)}^{\mathcal{C}}$.*

Proof: Let $B \in \mathcal{C}$ and let $A \in \mathrm{CP}^{B}_{Q(\alpha)}$ via N. Consider the following algorithm: On input x, (1) guess a computation ρ (including oracle answers) of $N^B(x)$; (2) verify that ρ accepts; (3) verify that all oracle answers in ρ are correct (the same as those given by oracle B).

Since $\mathcal{C}$ is closed under complement, join, and polynomial-time conjunctive reducibility, step (3) can be accomplished via a single query to a set B' in $\mathcal{C}$. Furthermore, the number of accepting paths for this algorithm is equal to the number of accepting computations of N. Let $f(x, \rho)$ produce the query from (3). Then $A \leq_{\mathrm{m}}^{\mathrm{CP}_{Q(\alpha)}} B'$ via f, so $A \in \mathcal{C}$. ∎

Lemma 26. *Let k be prime.*

(i) $\mathrm{MOD}_k\mathrm{P}$ *is closed under intersection.*

(ii) $\mathrm{MOD}_k\mathrm{P}$ *is closed under polynomial-time conjunctive reducibility.*

(iii) $\mathrm{MOD}_k\mathrm{P}$ *is closed under complement.*

(iv) $\mathrm{MOD}_k\mathrm{P}$ *is closed under union.*

(v) $\mathrm{MOD}_k\mathrm{P}$ *is closed under polynomial-time disjunctive reducibility.*

(vi) $\mathrm{MOD}_k\mathrm{P}$ *is closed under join.*

(vii) $\mathrm{MOD}_k\mathrm{P}$ *is closed under* $\leq_{\mathrm{m}}^{\mathrm{MOD}_k\mathrm{P}}$*-reducibility.*

Proof:

(i) Let L_1 and L_2 be languages in $\mathrm{MOD}_k\mathrm{P}$ via machines N_1 and N_2, respectively. Since $\#\mathrm{P}$ is closed under multiplication, there is a machine N_3 such that for every x

$$|\mathrm{ACCEPT}(N_3, x)| = |\mathrm{ACCEPT}(N_1, x)| \cdot |\mathrm{ACCEPT}(N_2, x)|.$$

Since k is prime, $|\mathrm{ACCEPT}(N_3, x)| \not\equiv 0 \pmod{k}$ if and only if

$$|\mathrm{ACCEPT}(N_1, x)| \not\equiv 0 \pmod{k} \text{ and } |\mathrm{ACCEPT}(N_2, x)| \not\equiv 0 \pmod{k}.$$

Therefore $L_1 \cap L_2 \in \mathrm{MOD}_k\mathrm{P}$.

(ii) This is similar to (i). A conjunctive reduction computes a truth-table reduction using a polynomial-size AND function. We can simulate the polynomial-size AND function by forming a polynomial-size product (using Property 6(ii)).

(iii) Let $L \in \mathrm{MOD}_k\mathrm{P}$ via N. For any integer j let

$$L_j = \{x : |\mathrm{ACCEPT}(N, x)| \not\equiv j \pmod{k}\}.$$

Then

$$\overline{L} = \bigcap_{1 \leq j < k} L_j.$$

Since each L_j belongs to $\mathrm{MOD}_k\mathrm{P}$ and $\mathrm{MOD}_k\mathrm{P}$ is closed under intersection, $\overline{L} \in \mathrm{MOD}_k\mathrm{P}$. (This property also follows from Theorem 23(ii).)

(iv) This follows from (i) and (iii) by DeMorgan's laws.

(v) This follows from (ii) and (iii) by DeMorgan's laws.

(vi) Obvious.

(vii) Let $A \leq_{\mathrm{m}}^{\mathrm{MOD}_k\mathrm{P}} B$ via f where $B \in \mathrm{MOD}_k\mathrm{P}$. By Theorem 23(i) there exists an NPM M_B such that for every x

$$|\textsc{accept}(M_B, x)| \equiv \begin{cases} 1 \pmod{k} & \text{if } x \in B \\ 0 \pmod{k} & \text{if } x \notin B. \end{cases}$$

Construct a machine M that behaves as follows on input x:

Step 1: Guess y such that $|y| = |x|^k$;

Step 2: Nondeterministically simulate M_B on input $f(x, y)$.

First suppose that $x \notin A$. Then the number of y's such that $f(x, y) \in B$ is congruent to $0 \pmod{k}$. For each such y, $|\textsc{accept}(M_B, f(x, y))| \equiv 1 \pmod{k}$. For the other y's, $|\textsc{accept}(M_B, f(x, y))| \equiv 0 \pmod{k}$. Therefore $|\textsc{accept}(M, x)|$ is congruent to $0 \cdot 1 + 0 \equiv 0 \pmod{k}$.

Now suppose that $x \in A$. Then the number of y's such that $f(x, y) \in B$ is congruent to $1 \pmod{k}$. Therefore $|\textsc{accept}(M, x)|$ is congruent to $1 \cdot 1 + 0 \equiv 1 \pmod{k}$. Thus $A \in \mathrm{MOD}_k\mathrm{P}$ via M.

∎

By combining the preceding results we have the following theorem and corollary.

Theorem 27. *If k is prime then* $\mathrm{MOD}_k\mathrm{P} = \mathrm{MOD}_k\mathrm{P}^{\mathrm{MOD}_k\mathrm{P}}$.

Corollary 28. *If k is prime, then* $\mathrm{MOD}_k\mathrm{P}$ *is closed under polynomial-time Turing reductions.*

7. Closure under Union and a Characterization of $\mathrm{MOD}_k\mathrm{P}$

Building on some of our results, Hertrampf [11] has discovered a very nice characterization of $\mathrm{MOD}_k\mathrm{P}$ in terms of the prime factorization of k. As a corollary, he has proved that $\mathrm{MOD}_k\mathrm{P}$ is closed under union for every k.[4] On the other hand there is an oracle relative to which $\mathrm{MOD}_k\mathrm{P}$ is not closed under intersection [5].

A key step towards Hertrampf's characterization is our result that $\mathrm{MOD}_{k^i}\mathrm{P} = \mathrm{MOD}_k\mathrm{P}$ for prime k (Theorem 30). For the sake of completeness, we follow our theorem with a proof of Hertrampf's main results (Theorem 32 and Corollaries 33-35).

Proposition 29. *If $j \mid k$ then* $\mathrm{MOD}_j\mathrm{P} \subseteq \mathrm{MOD}_k\mathrm{P}$.

[4] As mentioned in an earlier footnote, researchers in circuit complexity may find it more natural to work with the class co-$\mathrm{MOD}_k\mathrm{P}$, which is closed under intersection, but may not be closed under union.

Proof: Let $k = hj$. If $L \in \mathrm{MOD}_j\mathrm{P}$ via N, then since #P is closed under multiplication, there is a machine N' such that $|\text{ACCEPT}(N', x)| = h \cdot |\text{ACCEPT}(N, x)|$ for every x. Therefore $|\text{ACCEPT}(N, x)| \equiv 0 \pmod{j} \iff |\text{ACCEPT}(N', x)| \equiv 0 \pmod{k}$. ▮

If p is a prime that is not a divisor of k, then we expect that $\mathrm{MOD}_p\mathrm{P} \not\subseteq \mathrm{MOD}_k\mathrm{P}$; this belief is supported by oracle constructions [5].

Next we show that prime powers and primes are equally useful as moduli. Similar results seem to have been known for some time by some circuit researchers. Chandra, Stockmeyer, and Vishkin [8] showed that MOD_{2^i} gates are AC^0 truth-table reducible to MOD_2 gates. Smolensky [22] noted without proof that MOD_{k^i} gates are AC^0 reducible to MOD_k gates for every prime k. Recently, Barrington [3] noted that the reduction in [8] could be made to work for every prime k, and that a single MOD_k gate suffices for the reduction.

Theorem 30. *If k is prime then $\mathrm{MOD}_{k^i}\mathrm{P} = \mathrm{MOD}_k\mathrm{P}$ for every $i \geq 1$.*

Proof: By Proposition 29, $\mathrm{MOD}_k\mathrm{P} \subseteq \mathrm{MOD}_{k^i}\mathrm{P}$. We prove the reverse inclusion by induction on i. Assume that the theorem is true for some $i \geq 1$. Let $L \in \mathrm{MOD}_{k^{i+1}}\mathrm{P}$ via N. A number m is divisible by k^{i+1} if and only if

(1) m is divisible by k^i, and

(2) the coefficient of k^i in the base-k expansion of m is 0.

Condition (2) is equivalent to $\binom{m}{k^i} \equiv 0 \pmod{k}$ by Corollary 20. Therefore $|\text{ACCEPT}(N, x)| \not\equiv 0 \pmod{k^{i+1}} \iff$

$$|\text{ACCEPT}(N, x)| \not\equiv 0 \pmod{k^i} \text{ or } \binom{|\text{ACCEPT}(N, x)|}{k^i} \not\equiv 0 \pmod{k}.$$

Let

$$L_1 = \{x : |\text{ACCEPT}(N, x)| \not\equiv 0 \pmod{k^i}\},$$

and

$$L_2 = \left\{x : \binom{|\text{ACCEPT}(N, x)|}{k^i} \not\equiv 0 \pmod{k}\right\}.$$

Then $L = L_1 \cup L_2$. By the induction hypothesis, L_1 is in $\mathrm{MOD}_k\mathrm{P}$. By Property 8(ii), there is a machine N_2 such that for all x

$$|\text{ACCEPT}(N_2, x)| = \binom{|\text{ACCEPT}(N, x)|}{k^i}.$$

Therefore $L_2 \in \mathrm{MOD}_k\mathrm{P}$ via N_2. Since $\mathrm{MOD}_k\mathrm{P}$ is closed under union, $L \in \mathrm{MOD}_k\mathrm{P}$. ▮

Because we expect PP to be incomparable to $\oplus$P, we expect that computing the highest order bit of #P functions is incomparable in difficulty to computing the lowest order bit of #P functions. However, the proof technique used above allows us to show that computing the $b(n)$-th low-order bit of a #P function, for small b, is exactly as difficult as computing the lowest-order bit of a #P function.

Theorem 31. *Let $\mathrm{bit}_b(i)$ denote the b-th low-order bit of i (more precisely, $\lfloor i/2^b \rfloor \bmod 2$). Let b be a function in UPF such that $b(n) = O(\log n)$. Then*

$$\mathrm{CP}_{\mathrm{bit}_{b(n)}(\alpha)=1} = \oplus\mathrm{P}.$$

Proof: Computing the $b(n)$-th low-order bit of a #P function is the same as checking condition (2) in the preceding proof with $i = b(n)$ and $k = 2$. Since 2^i is bounded by a polynomial, that check is in $\mathrm{MOD}_2\mathrm{P}$ by Property 8(ii). For the converse, we can use Property 3 to multiply the number of accepting paths by $2^{b(n)}$. ∎

The next theorem brings us closer to Hertrampf's characterization.

Theorem 32 ([11]). *Let j and k be relatively prime. Then $L \in \mathrm{MOD}_{jk}\mathrm{P}$ if and only if there exist $L_j \in \mathrm{MOD}_j\mathrm{P}$ and $L_k \in \mathrm{MOD}_k\mathrm{P}$ such that $L = L_j \cup L_k$.*

Proof: Assume that L belongs to $\mathrm{MOD}_{jk}\mathrm{P}$ via N. Then let

$$L_i = \{x : |\textsc{accept}(N, x)| \not\equiv 0 \pmod{i}\},$$

for $i = j, k$. Clearly $L = L_j \cup L_k$.

Conversely, let $L_i \in \mathrm{MOD}_i\mathrm{P}$ via N_i for $i = j, k$. Since #P is closed under addition and multiplication, there is a machine N such that

$$|\textsc{accept}(N, x)| = k \cdot |\textsc{accept}(N_j, x)| + j \cdot |\textsc{accept}(N_k, x)|.$$

Now, $L_j \cup L_k \in \mathrm{MOD}_{jk}\mathrm{P}$ via N. ∎

Corollary 33 (Characterization [11]). *Let the prime factorization of k be $p_1^{e_1} \cdots p_m^{e_m}$. Then $L \in \mathrm{MOD}_k\mathrm{P}$ if and only if there exist $L_1 \in \mathrm{MOD}_{p_1}\mathrm{P}$, …, $L_m \in \mathrm{MOD}_{p_m}\mathrm{P}$ such that $L = L_1 \cup \cdots \cup L_m$.*

Proof: By Theorem 30, $\mathrm{MOD}_{p_i^{e_i}}\mathrm{P} = \mathrm{MOD}_{p_i}\mathrm{P}$. Now the corollary follows from the preceding theorem by induction. ∎

Corollary 34 ([11]). *Let the prime factorization of k be $p_1^{e_1} \cdots p_m^{e_m}$. Then $\mathrm{MOD}_k\mathrm{P} = \mathrm{MOD}_{p_1 \cdots p_m}\mathrm{P}$.*

Corollary 35 ([11]). *For every k, $\mathrm{MOD}_k\mathrm{P}$ is closed under union.*

Proof: Recall that $\mathrm{MOD}_p\mathrm{P}$ is closed under union for each prime p, and apply Corollary 33. ∎

Hertrampf [11] has a simple proof that $\mathrm{MOD}_k\mathrm{P}$ is closed under intersection if and only if it is closed under complement, and his proof relativizes. There is an oracle A such that $\mathrm{MOD}_k\mathrm{P}^A$ is not closed under complement, and hence not closed under intersection [5].

8. Subclasses of NP

We can define "counting subclasses" of NP by placing a restriction on the number of accepting paths for strings in the language. Classes defined in this way include R [1, 18], UP, and Allender's class FewP [2] defined by analogy to UP.

Definition 36. A language L is in FewP if there exist N and a polynomial q such that L is accepted by N and N has at most $q(|x|)$ accepting paths for every x.

We define the class $\mathrm{MODZ}_k\mathrm{P}$, which is a subclass of NP corresponding to $\mathrm{MOD}_k\mathrm{P}$.

Definition 37. $\mathrm{MODZ}_k\mathrm{P}$ consists of those languages L for which there exists a nondeterministic polynomial-time machine N such that for all x

$$\begin{array}{ll} |\textsc{accept}(N,x)| \not\equiv 0 \pmod{k} & \text{if } x \in L \\ |\textsc{accept}(N,x)| = 0 & \text{if } x \notin L. \end{array}$$

Note that $\mathrm{MODZ}_k\mathrm{P} \subseteq \mathrm{MOD}_k\mathrm{P} \cap \mathrm{NP}$. We do not know whether the inclusion is proper. However, we have constructed oracles relative to which this is the case [4]. Most of the closure properties we obtained for $\mathrm{MOD}_k\mathrm{P}$ go through for $\mathrm{MODZ}_k\mathrm{P}$ with essentially the same proof. The only major difference is in proving closure under union; here we use Fermat's theorem in a novel way.

Theorem 38.

(i) *If k is prime then* $\mathrm{MODZ}_k\mathrm{P}$ *is closed under intersection.*

(ii) *If k is prime then* $\mathrm{MODZ}_k\mathrm{P}$ *is closed under polynomial-time conjunctive reducibility.*

(iii) *If k is prime and $i \not\equiv 0 \pmod{k}$, then $L \in \mathrm{MODZ}_k\mathrm{P}$ iff there there exists N such that for every x*

$$\begin{array}{ll} |\textsc{accept}(N,x)| \equiv i \pmod{k} & \textit{if } x \in L \\ |\textsc{accept}(N,x)| = 0 & \textit{if } x \notin L. \end{array}$$

(iv) *If k is odd and square-free, then* $\mathrm{MODZ}_k\mathrm{P}$ *is closed under union.*

(v) *If $j \mid k$ then* $\mathrm{MODZ}_j\mathrm{P} \subseteq \mathrm{MODZ}_k\mathrm{P}$.

(vi) *If k is an odd prime then* $\mathrm{MODZ}_{k^i}\mathrm{P} = \mathrm{MODZ}_k\mathrm{P}$ *for every $i \geq 1$.*

Proof:

(i) Same as the proof of Lemma 26(i).

(ii) Same as the proof of Lemma 26(ii).

(iii) Same as the proof of Theorem 23 with $j = 0$.

(iv) Let

$$\lambda = \prod_{p|k,\ p \text{ prime}} (p-1).$$

Let L_1 and L_2 be in $\mathrm{MOD}_k\mathrm{P}$ via N_1 and N_2, respectively. Because #P is closed under multiplication and addition, there is a machine N_3 such that

$$|\textsc{accept}(N_3,x)| = |\textsc{accept}(N_1,x)|^\lambda + |\textsc{accept}(N_2,x)|^\lambda.$$

By Corollary 18, the only solution of $u^\lambda + v^\lambda \equiv 0 \pmod{k}$ is $u \equiv v \equiv 0 \pmod{k}$. Therefore, $|\textsc{accept}(N_3,x)| \not\equiv 0 \pmod{k}$ iff $x \in L_1 \cup L_2$.

(v) Same as the proof of Proposition 29.

(vi) We use (iv) to obtain closure under union. The remainder of the proof is the same as the proof of Theorem 30.

∎

9. Fewness

In [7] Cai and Hemachandra proved a surprising result: the class Few (defined below) is a subset of $\mathrm{MOD}_k\mathrm{P}$ for every $k \geq 2$ (that result has been generalized by Köbler, Schöning, Toda, and Torán [14]). In this section, we prove a similar result: $\mathrm{FewP} \subseteq \mathrm{MODZ}_k\mathrm{P}$ for every $k \geq 2$. Though this result can also be obtained by a close inspection of Cai and Hemachandra's proof, our proof will be simpler. Finally we show that every language in the class Few is as easy as distinguishing categorical acceptance from nonacceptance. This generalizes Cai and Hemachandra's result.

Theorem 39. *If* $k \geq 2$ *then* $\mathrm{FewP} \subseteq \mathrm{MODZ}_k\mathrm{P}$.

Proof: Let $\mathrm{L} \in \mathrm{FewP}$ via a machine N. By Property 9(ii), there is a machine N' such that for every x

$$\begin{aligned} |\textsc{accept}(N', x)| &= k^{|\textsc{accept}(N,x)|} - 1 \\ &\begin{cases} \not\equiv 0 \pmod{k} & \text{if } |\textsc{accept}(N, x)| \geq 1 \\ = 0 & \text{if } |\textsc{accept}(N, x)| = 0. \end{cases} \end{aligned}$$

∎

Schöning [20] applied similar techniques to the case $k = 2$.

Cai and Hemachandra [7] defined the class Few as follows.

Definition 40. A language L belongs to the class Few if there exist a nondeterministic polynomial-time machine N, a polynomial q such that $|\textsc{accept}(N, x)| \leq q(|x|)$, and a polynomial-time computable predicate R such that $x \in L$ iff $R(x, |\textsc{accept}(N, x)|)$.

Equivalently, a language L belongs to Few if $L \in \mathrm{P}^{\#\mathrm{P}[1]}$ via a #P oracle whose output *value* is bounded by a polynomial in the input length. Now we show that every language in Few is as easy as distinguishing categorical acceptance from nonacceptance.

Theorem 41. *Let* Q *be any predicate such that* $Q(1) = 1$ *and* $Q(0) = 0$. *Then*

$$\mathrm{Few} \subseteq \mathrm{P}^{\mathrm{CP}_{Q(\alpha)}},$$

and in fact every language in Few *is polynomial-time bounded-depth Boolean-formula reducible to a language in* $\mathrm{CP}_{Q(\alpha)}$.

Proof: Let $L \in \mathrm{Few}$ via machine N and polynomial-time computable predicate R. Let

$$L' = \left\{ \langle 0^i, x \rangle : Q\left(\binom{|\textsc{accept}(N, x)|}{i} \right) \right\}.$$

Then $L' \in \mathrm{CP}_{Q(\alpha)}$ by Property 8(ii). Note that

$$\binom{|\textsc{accept}(N, x)|}{i} = \begin{cases} 1 & \text{if } i = |\textsc{accept}(N, x)| \\ 0 & \text{if } i > |\textsc{accept}(N, x)|. \end{cases}$$

Therefore,

$$|\textsc{accept}(N, x)| = \max\{i : \langle 0^i, x \rangle \in L'\}.$$

Consequently, $L(x) = R(x, \max\{i : \langle 0^i, x\rangle \in L'\}) =$

$$\bigvee_{0\le i\le q(|x|)} \left(R(x,i) \wedge L'(\langle 0^i, x\rangle) \wedge \left(\bigwedge_{i<j\le q(|x|)} \neg L'(\langle 0^j, x\rangle) \right) \right),$$

so, in fact, L is polynomial-time reducible to L' via a Boolean formula with depth 2. ▮

Valiant and Vazirani have shown that if it is possible to distinguish uniquely satisfiable formulas from unsatisfiable formulas in polynomial time, then R = NP. Toda [24] has implicitly shown that the same hypothesis implies P = FewP. Our theorem above implies a further collapse:

Corollary 42. *If it is possible to distinguish uniquely satisfiable formulas from unsatisfiable formulas in polynomial time, then* P = Few*.*

The following result was proved previously via different techniques by Cai and Hemachandra [7].

Corollary 43. *If* $k \ge 2$ *then* Few $\subseteq$ $\mathrm{MOD}_k\mathrm{P}$.

Proof: By Proposition 29, it suffices to prove this result for prime k. By the preceding theorem, Few $\subseteq \mathrm{P}^{\mathrm{MOD}_k\mathrm{P}}$, which is equal to $\mathrm{MOD}_k\mathrm{P}$ by Corollary 28. ▮

Independently of this research, Köbler [13] has discovered the lovely theorem that Few $\subseteq \mathrm{P}^{\mathrm{FewP}}$. His theorem can be used in alternate proofs of Corollaries 42 and 43.

10. Acknowledgments

Parts of this research were performed while the first author was visiting Bob Floyd at Stanford University. We are grateful to Bob Floyd and Uli Hertrampf for helpful discussions. We are also grateful for valuable suggestions made by the anonymous referees of several versions of this paper and by the STACS '90 reviewers.

References

[1] L. Adleman and K. Manders. Reducibility, randomness, and intractability. In *Proceedings of the 9th Annual ACM Symposium on Theory of Computing*, pages 151–153, 1977.

[2] E. W. Allender. The complexity of sparse sets in P. In A. L. Selman, editor, *Structure in Complexity Theory*, pages 1–11. Springer-Verlag, June 1986. Lecture Notes in Computer Science 223.

[3] D. M. Barrington. Some problems involving Razborov-Smolensky polynomials. TR 90-59, UMass COINS, 1990. A revised version will appear in a special volume of the Proceedings of the London Mathematical Society devoted to the 1990 Durham Symposium.

[4] R. Beigel. Relativized counting classes: Relations among thresholds, parity, and mods. *J. Comput. Syst. Sci.*, 42(1):76–96, Feb. 1991.

[5] D. A. M. Barrington, R. Beigel, and S. Rudich. Representing Boolean functions as polynomials modulo composite integers. In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, 1992. To appear.

[6] A. Blass and Y. Gurevich. On the unique satisfiability problem. *Information and Control*, 55:80–88, 1982.

[7] J. Cai and L. Hemachandra. On the power of parity polynomial time. *Mathematical Systems Theory*, 23:95–106, 1990.

[8] A. K. Chandra, L. Stockmeyer, and U. Vishkin. Constant depth reducibility. *SIAM J. Comput.*, 13(2):423–438, May 1984.

[9] J. Gill. Computational complexity of probabilistic Turing machines. *SIAM J. Comput.*, 6:675–695, 1977.

[10] L. M. Goldschlager and I. Parberry. On the construction of parallel computers from various bases of Boolean functions. *Theoretical Comput. Sci.*, 43:43–58, 1986.

[11] U. Hertrampf. Relations among MOD-classes. *Theoretical Comput. Sci.*, 74(3):325–328, Aug. 1990.

[12] D. E. Knuth. *Fundamental Algorithms*, volume 1 of *The Art of Computer Programming*. Addison-Wesley, Reading, MA, 1973.

[13] J. Köbler. *Strukturelle Komplexität von Anzahlproblemen*. PhD thesis, Institut für Informatik der Universität Stuttgart, Stuttgart, Germany, May 1989.

[14] J. Köbler, U. Schöning, S. Toda, and J. Torán. Turing machines with few accepting computations and low sets for PP. *J. Comput. Syst. Sci.*, 44, 1992. To appear.

[15] T. J. Long. Strong nondeterministic polynomial-time reducibilities. *Theoretical Comput. Sci.*, 21:1–25, 1982.

[16] A. Meyer and L. J. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential space. In *Proceedings of the 13th Annual IEEE Symposium on Switching and Automata Theory*, pages 125–129, 1972.

[17] C. H. Papadimitriou and S. K. Zachos. Two remarks on the complexity of counting. In *Proceedings of the 6th GI Conference on Theoretical Computer Science*, pages 269–276. Springer-Verlag, 1983. Volume 145 of *Lecture Notes in Computer Science*.

[18] C. Rackoff. Relativized questions involving probabilistic algorithms. *J. ACM*, 29(1):261–268, Jan. 1982.

[19] D. A. Russo. *Structural Properties of Complexity Classes*. PhD thesis, University of California at Santa Barbara, Mar. 1985.

[20] U. Schöning. The power of counting. In *Proceedings of the 3rd Annual Conference on Structure in Complexity Theory*, pages 2–18. IEEE Computer Society Press, June 1988.

[21] J. Simon. *On Some Central Problems In Computational Complexity.* PhD thesis, Cornell University, Ithaca, New York, 1975. Dept. of Computer Science, TR 75-224.

[22] R. Smolensky. Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing*, pages 77–82, 1987.

[23] L. J. Stockmeyer. The polynomial-time hierarchy. *Theoretical Comput. Sci.*, 3:1–22, 1977.

[24] S. Toda. On polynomial-time truth-table reducibilities of intractable sets to p-selective sets. *Mathematical Systems Theory*, 24:69–82, 1991.

[25] J. Torán. Complexity classes defined by counting quantifiers. *J. ACM*, 38(3):753–774, July 1991.

[26] L. G. Valiant. The relative complexity of checking and evaluating. *Inf. Process. Lett.*, 5:20–23, 1976.

[27] L. G. Valiant. The complexity of computing the permanent. *Theoretical Comput. Sci.*, 8:189–201, 1979.

[28] L. G. Valiant and V. V. Vazirani. NP is as easy as detecting unique solutions. *Theoretical Comput. Sci.*, 47:85–93, 1986.

[29] K. W. Wagner. The complexity of combinatorial problems with succinct input representation. *Acta Inf.*, 23:325–356, 1986.