Restricted Information from Nonadaptive Queries to NP *

Yenjo Han †
University of Rochester

Thomas Thierauf ‡
Universität Ulm

Abstract

We investigate classes of sets that can be decided by bounded truth-table reductions to an NP set in which evaluators do *not* have full access to the answers to the queries but get only restricted information such as the number of queries that are in the oracle set or even just this number modulo m, for some $m \geq 2$. We also investigate the case in which evaluators are nondeterministic.

We show that when we vary the information that the evaluators get, this can change the resulting power of the evaluators. We locate all these classes within levels of the Boolean hierarchy which allows us to compare the complexity of such classes.

1 Introduction

Truth-table reductions were introduced in recursion theory as a type of reduction that is more flexible than the many-one reducibility, yet more restrictive than the Turing reducibility. Ladner, Lynch, and Selman [LLS75] introduced and investigated the polynomial-time analog of the truth-table reductions. We give an informal definition: a set L is *truth-table reducible to set* A, if there exist two polynomial-time bounded Turing machines, the

*Supported in part by the National Science Foundation under grants CCR-8957604 and NSF-INT-9116781/JSPS-ENG-207.

†Department of Computer Science, University of Rochester, Rochester, NY 14627, USA.

‡Abteilung Theoretische Informatik, Universität Ulm, Oberer Eselsberg, 89069 Ulm, Germany. Email: thierauf@informatik.uni-ulm.de. Part of the work done while visiting the University of Rochester. Supported in part by DFG Postdoctoral Stipend Th 472/1-1.

generator and the *evaluator*, such that for any input string x, the membership of x in L can be determined as follows. First, the generator generates a list of strings which are then asked to oracle A. (This type of querying is called *nonadaptive* because all query strings are produced before any answer is given by the oracle.) Now, the evaluator, getting x and the answers of A to the queries as input, decides the membership of x in L. A truth-table reduction is called *bounded* if the number of queries produced by the generator is bounded by a constant. For example, every set is reducible to its complement via a bounded truth-table reduction that asks only one query (namely, the input itself), but this does not hold in general with respect to many-one reductions.

In this paper, we are interested in the classes $\mathrm{P}_{tt}^{\mathrm{NP}[k]}$ of sets that are bounded truth-table reducible to some NP set, where the generator produces at most k queries, for some $k \geq 0$. These *bounded query classes for NP* are central topics of investigations in computational complexity theory [ABG90, Bei91, BH88, CGH+88, Ka88a, Ka88b, KT94, W90, Wec85]. Especially the (extended version of the) paper by Amir, Beigel, and Gasarch [ABG90] gives a very broad overview on this topic and also provides an extensive list of references. Let us point out the following, rather obvious, property of truth-table reductions: the evaluator, by getting all the answers to the queries produced by the generator, gets *full information* about the queries with respect to the oracle. However, Wagner and Wechsung [Wec85] obtained a remarkable result that the information an evaluator needs can be dramatically reduced without changing the classes $\mathrm{P}_{tt}^{\mathrm{NP}[k]}$: it suffices to give an evaluator just *the parity of the number of queries that are in the oracle*, i.e., one bit of information! To describe this result more formally, we introduce some notation.

We modify the truth-table reduction explained above in such a way that, for a given function f, instead of the list of answers to the queries, the evaluator gets the outcome of f when applied to this list. This is kind of a function composition. We use a notation introduced by Köbler and Thierauf [KT94] to express this.

Definition 1.1 [KT94] *Let $\mathcal{C}$ be a class of languages and let $\mathcal{F}$ be a class of functions from Σ^* to Σ^*. A set L is in the class $\mathcal{C}//\mathcal{F}$ if and only if there exist a set $A \in \mathcal{C}$ and a function $f \in \mathcal{F}$ such that for all $x \in \Sigma^*$, it holds that $x \in L \iff \langle x, f(x)\rangle \in A$.* [1]

[1] The definition is motivated by the advice classes originally introduced by Karp and

We consider the following function classes. Let A be a set, $k \geq 0$, and $m \geq 2$.

$$\begin{aligned} f \in \chi^{A[k]} \quad &\Longleftrightarrow \quad \exists g \in \mathrm{FP}\ \forall x: \quad g(x) = \langle x_1, \ldots, x_k \rangle \text{ and} \\ &\qquad\quad f(x) = A(x_1) \cdots A(x_k), \\ f \in \#^{A[k]} \quad &\Longleftrightarrow \quad \exists g \in \mathrm{FP}\ \forall x: \quad g(x) = \langle x_1, \ldots, x_k \rangle \text{ and} \\ &\qquad\quad f(x) = \sum_{i=1}^{k} A(x_i), \\ f \in \mathrm{Mod}_m^{A[k]} \quad &\Longleftrightarrow \quad \exists h \in \#^{A[k]}\ \forall x: \quad f(x) = h(x) \bmod m, \end{aligned}$$

where $A(\cdot)$ denotes the characteristic function of A. For $m = 2$ we write $\oplus^{A[k]}$ instead of $\mathrm{Mod}_2^{A[k]}$. By $\chi^{\mathrm{NP}[k]}$, we denote $\bigcup_{A \in \mathrm{NP}} \chi^{A[k]}$, and analogously for the other two classes.

In other words, the outcome of a function in $\chi^{A[k]}$ is the sequence of answers to the queries to A produced by some generator g. A function in $\#^{A[k]}$ counts the number of queries that are in A, and a function in $\mathrm{Mod}_m^{A[k]}$ gives this number modulo m. As an example, we have $\mathrm{P}//\chi^{\mathrm{NP}[k]} = \mathrm{P}_{tt}^{\mathrm{NP}[k]}$.

The result of Wagner and Wechsung can now be stated as follows.

Theorem 1.2 [Wec85] *For all $k \geq 0$, we have*

$$\mathrm{P}//\chi^{\mathrm{NP}[k]} = \mathrm{P}//\#^{\mathrm{NP}[k]} = \mathrm{P}//\oplus^{\mathrm{NP}[k]}.$$

In other words, given a $\oplus^{\mathrm{NP}[k]}$ function value of k (appropriately chosen) strings, a P evaluator can recover the result of the computation of another P evaluator which gets full information about k queries, i.e., a $\chi^{\mathrm{NP}[k]}$ function value. Beigel [Bei91] and Wagner [W95] give a very elegant proof of Theorem 1.2 using the so-called *mind change technique*. Essentially they show that any set in $\mathrm{P}//\chi^{\mathrm{NP}[k]}$ can be expressed as the symmetric difference of k NP sets and one P set. An immediate consequence of this representation is that $\mathrm{P}//\chi^{\mathrm{NP}[k]}$ is contained in consecutive levels of the Boolean hierarchy (see next section for definitions), namely

$$\mathrm{NP}(k) \subseteq \mathrm{P}//\chi^{\mathrm{NP}[k]} \subseteq \mathrm{NP}(k+1),$$

Lipton [KL82] and denoted with a single slash, i.e., $\mathcal{C}/\mathcal{F}$. Note that the advice functions of Karp and Lipton depend on *the length* of the input, whereas in this paper, the functions depend on the input itself.

Note also that the classes introduced by Karp and Lipton are *nonuniform*, because of the use of noncomputable functions as advice. Here, we use computable functions, and therefore the resulting classes are uniform.

for all $k \geq 1$ [KSW87].

Considering Theorem 1.2, one might ask whether one can replace the parity functions in $\mathrm{P}//\oplus^{\mathrm{NP}[k]}$ by $\mathrm{Mod}_m^{\mathrm{NP}[k]}$, for values of m other than 2, and still maintain the equivalence to the class $\mathrm{P}//\chi^{\mathrm{NP}[k]}$. By Theorem 1.2, we have $\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]} \subseteq \mathrm{P}//\oplus^{\mathrm{NP}[k]}$, for all $m \geq 2$, since a $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ function cannot give more information to the evaluator than a $\#^{\mathrm{NP}[k]}$ function. On the other hand, when the modulus m is even, an evaluator can easily extract the parity bit from any $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ function. Hence, for even m, we have $\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]} = \mathrm{P}//\oplus^{\mathrm{NP}[k]}$. However, the case when m is odd is not so clear. The various proofs for Theorem 1.2 all rely heavily on properties of the parity function and do not seem to be extendable to an odd modulus. We show in Section 4 that in fact, for odd m, $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ provides less information to P evaluators than $\mathrm{Mod}_2^{\mathrm{NP}[k]}$ (unless the Boolean hierarchy collapses). Namely, we show for all $k \geq 0$,

$$\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]} \;=\; \mathrm{P}//\oplus^{\mathrm{NP}[k-\lfloor k/m \rfloor]}, \quad \text{for } m \text{ odd.} \tag{1}$$

In other words, a parity function can ask $\lfloor k/m \rfloor$ fewer queries to an oracle than a $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ function and still give the same amount of information to a P evaluator.

Motivated by Theorem 1.2, Köbler and Thierauf [KT94] studied the case when functions in $\chi^{\mathrm{NP}[k]}$ or $\#^{\mathrm{NP}[k]}$ are given to *nondeterministic* polynomial-time evaluators instead of *deterministic* polynomial-time evaluators. They showed that the counterpart of the first equality of Theorem 1.2 holds, and furthermore, that the resulting class coincides with the $(2k+1)$-th level of the Boolean hierarchy.

Theorem 1.3 [KT94] *For all $k \geq 0$, we have*

$$\mathrm{NP}//\chi^{\mathrm{NP}[k]} = \mathrm{NP}//\#^{\mathrm{NP}[k]} = \mathrm{NP}(2k+1).$$

As already mentioned, the class $\mathrm{P}//\chi^{\mathrm{NP}[k]}$ is located between the kth and $(k+1)$th level of the Boolean hierarchy. Therefore, when $\chi^{\mathrm{NP}[k]}$ or $\#^{\mathrm{NP}[k]}$ functions are given to NP evaluators instead of P evaluators, this roughly doubles the level of the Boolean hierarchy where the resulting classes are located.

What happens when parity information is given to NP evaluators? It is easy to see that the second equation in Theorem 1.2 cannot carry over to NP evaluators, unless the Boolean hierarchy collapses. It has been asked [KT94] whether the $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$ classes also coincide with levels of the Boolean

hierarchy. In Section 3, we answer this question affirmatively. We show that when k is odd, both $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$ and $\mathrm{NP}//\oplus^{\mathrm{NP}[k+1]}$ coincide with the $(k+2)$-th level of the Boolean hierarchy; i.e., for all $k \geq 0$,

$$\mathrm{NP}//\oplus^{\mathrm{NP}[2k+1]} \;=\; \mathrm{NP}//\oplus^{\mathrm{NP}[2k+2]} \;=\; \mathrm{NP}(2k+3). \tag{2}$$

As in the case of P evaluators, it is interesting to investigate the case when $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ functions are given to NP evaluators for values of m other than 2. We have already seen that $\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ coincides with $\mathrm{P}//\oplus^{\mathrm{NP}[k]}$ for even m. The nontrivial inclusion here, that $\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ is contained in $\mathrm{P}//\oplus^{\mathrm{NP}[k]}$, was given by Theorem 1.2. Since we don't have an analogous theorem for NP evaluators, we cannot argue so easily in this case. However, we show that the equation indeed carries over to NP evaluators. Namely, we have for all $k \geq 2m-2$,

$$\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]} \;=\; \mathrm{NP}//\oplus^{\mathrm{NP}[k]}, \quad \text{for } m \text{ even}. \tag{3}$$

Again, there seems to be a difference depending on whether the modulus m is odd or even. In case m is odd, we will give a lower and an upper bound as follows. For all $k \geq 2m-2$,

$$\mathrm{NP}//\oplus^{\mathrm{NP}[k-\lfloor k/m \rfloor]} \;\subseteq\; \mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]} \;\subseteq\; \mathrm{NP}//\oplus^{\mathrm{NP}[k]}, \quad \text{for } m \text{ odd}. \tag{4}$$

We note that the exact location of $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$, for odd m, in the Boolean hierarchy has been settled very recently by Agrawal, Beigel, and Thierauf [ABT96].

We want to point out one interesting consequence of our results. By Theorem 1.2, $\#^{\mathrm{NP}[k]}$ functions contain the same amount of information for P evaluators as $\mathrm{Mod}_2^{\mathrm{NP}[k]}$ functions, and we have already argued that this does not carry over to NP evaluators unless the Boolean hierarchy collapses. However, the following weaker version holds: let $m = 2^l$ for some $l \geq 1$ and $k \geq 2m-2 = 2^{l+1}-2$. Then, by equation (3), we have

$$\mathrm{NP}//\mathrm{Mod}_{2^l}^{\mathrm{NP}[k]} = \mathrm{NP}//\oplus^{\mathrm{NP}[k]}.$$

Note that a $\mathrm{Mod}_{2^l}^{\mathrm{NP}[k]}$ function consists exactly of the l least significant bits in the binary representation of a $\#^{\mathrm{NP}[k]}$ function. It follows that if as few as the two most significant bits are discarded from the binary representation of a $\#^{\mathrm{NP}[k]}$ function value, their information content for NP evaluators abruptly drops down to the level of a parity function. Indeed, when $2^{l+1}-2 \leq k \leq$

$2^{l+1} - 1$, even omitting only the most significant bit from a $\#^{\mathrm{NP}[k]}$ function leaves NP evaluators essentially with parity information only.

The paper is organized as follows. In Section 3, we start by considering NP evaluators that get parity information and show equation (2). Parity functions turn out to be technically simpler to handle than $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ functions for values m larger than 2. In Section 4, we extend the techniques from Section 3 to study the classes $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$. We also consider P evaluators and show equation (1).

2 Preliminaries

We follow standard definitions and notations in computational complexity theory. Readers are referred to a standard reference (see, e.g., [HU79] or [BDG88]) for the definitions of common notations and concepts such as alphabets, strings, languages, Turing machines, polynomial-time bounded computation, and nondeterminism. Throughout this paper, we use the alphabet $\Sigma = \{0, 1\}$. If A is a set, we use $A(\cdot)$ to denote the characteristic function of A. $\langle \cdot, \cdot \rangle$ is a one-to-one pairing function from $\Sigma^* \times \Sigma^*$ to Σ^* that is computable and invertible in polynomial time.

For any two sets A and B, $A \triangle B$ denotes the symmetric difference of A and B. For the intersection $A \cap B$, we often omit the intersection symbol and simply write $A\,B$.

P (NP) denote the classes of languages that can be recognized by a polynomial-time deterministic (nondeterministic) Turing machine. FP is the class of polynomial-time computable total functions.

The Boolean hierarchy is defined as the closure of NP under Boolean operations. There are many equivalent ways of defining the levels of the Boolean hierarchy [CGH+88]. We use the following.

Definition 2.1 *Let $k \geq 1$. A set L is in* $\mathrm{NP}(k)$*, the k-th* level of the Boolean hierarchy*, if there exist $A_1, \ldots, A_k \in \mathrm{NP}$ such that $L = A_1 \triangle \cdots \triangle A_k$.*

A set L is in $\mathrm{coNP}(k)$*, if $\overline{L} \in \mathrm{NP}(k)$. The* Boolean hierarchy*, BH, is the union of all the levels,* $\bigcup_{k \geq 1} \mathrm{NP}(k)$.

In the definition of $\mathrm{NP}(k)$, we can require in addition that the sets A_i form a decreasing chain $A_1 \supseteq \cdots \supseteq A_k$ [CGH+88]. We will often use this additional property.

The Boolean hierarchy has a downward separation property, i.e., for all $k \geq 1$, $\mathrm{NP}(k) = \mathrm{coNP}(k)$ implies $\mathrm{BH} = \mathrm{NP}(k)$. The levels of the Boolean

hierarchy interleave with the levels of the (bounded) query hierarchy of NP, that is,

$$\mathrm{NP}(k) \;\subseteq\; \mathrm{P}//\chi^{\mathrm{NP}[k]} \;\subseteq\; \mathrm{NP}(k+1)$$

for all $k \geq 1$ [KSW87] (see also [Bei91]). It follows from the downward separation property that the Boolean hierarchy collapses if any of these inclusions is an equality.

Finally, we want to derive a Boolean expression in terms of NP sets for sets in $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$, for $m \geq 2$ and $k \geq 0$. Let $L \in \mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$. By definition, there exist a set $E \in \mathrm{NP}$ and a function $f \in \mathrm{Mod}_m^{\mathrm{NP}[k]}$ such that for all $x \in \Sigma^*$, $x \in L$ if and only if $\langle x, f(x)\rangle \in E$. Let g be an FP function such that $g(x) = \langle x_1, \ldots, x_k\rangle$ and $f(x) = \sum_{i=1}^{k} \mathrm{SAT}(x_i) \pmod m$. We associate the following NP sets A_i and E_j with f, g, and E, for $i = 0, \ldots, k$ and $j = 0, \ldots, m$.

$$A_i \;=\; \{\, x \mid \text{at least } i \text{ of the strings generated by } g(x) \text{ are in SAT} \,\}.$$

Sets A_i form a decreasing chain, i.e., we have $\Sigma^* = A_0 \supseteq A_1 \supseteq \cdots \supseteq A_k$. Furthermore, for a given x, let i_0 be the maximum i such that $x \in A_i$. Note that i_0 can be expressed as the unique i such that $x \in A_i - A_{i+1}$. Clearly, we have $f(x) = i_0 \pmod m$.

There are only m possibilities for the value of $f(x)$. For each potential value j, where $0 \leq j < m$, we define NP set E_j as the set of strings that is in E assuming $f(x) = j$. That is, for $j = 0, \ldots, m-1$,

$$E_j \;=\; \{\, x \mid \langle x, j\rangle \in E \,\}.$$

Now, we can express L in terms of the sets A_i and E_j, since, by the above discussion, an x is in L if and only if there is an i such that $x \in A_i - A_{i+1}$ and $\langle x, i\rangle$ is in E_i. That is

$$L \;=\; \bigcup_{i=0}^{k-1} \Big((A_i - A_{i+1})\, E_i \Big) \;\cup\; A_k\, E_k,$$

where the indices of sets E_i are taken modulo m. (Recall that we omit the intersection symbol.) Since $A_0 \supseteq \cdots \supseteq A_k$, all the terms in the union are mutually disjoint and we can rewrite this expression in terms of symmetric differences, thereby getting an analog of the *ring sum expansion* of Boolean functions.

$$L \;=\; \mathop{\triangle}_{i=0}^{k-1} \Big(A_i\, E_i \triangle A_{i+1}\, E_i \Big) \triangle A_k\, E_k$$

$$= \quad A_0\, E_0 \triangle \mathop{\triangle}_{i=1}^{k} \Big(A_i\, E_{i-1} \triangle A_i\, E_i \Big). \tag{5}$$

The latter equation holds since symmetric difference is an associative operation. From this representation we can already conclude that L is contained in the $(2k+1)$th level of the Boolean hierarchy. As we will show in the following sections, in fact, L is located much lower in the Boolean hierarchy.

3 Parity Functions

In this section, we consider NP evaluators that get parity information. Our goal is to locate the classes $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$, for all $k \geq 0$, in the Boolean hierarchy which is posed as an open problem in [KT94]. Before stating our result, we will argue that for each class $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$, one can easily exclude all except one level of the Boolean hierarchy as a possible candidate it can coincide with. Note first that we have

$$\mathrm{P}//\chi^{\mathrm{NP}[k]} \;\subseteq\; \mathrm{NP}//\oplus^{\mathrm{NP}[k]} \;\subseteq\; \mathrm{P}//\chi^{\mathrm{NP}[k+2]}.$$

The first inclusion follows from Theorem 1.2. To show the second inclusion, let L be a language in $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$. Given an input string x, its membership in L is decided by an NP evaluator E that has access to a parity bit that is computed from the result of k queries, say, $y_1, \ldots, y_k$, to SAT. Let furthermore z_0 and z_1 be two strings such that $z_j \in \mathrm{SAT} \iff E$ accepts input $\langle x, j\rangle$, for $j = 0, 1$. Since parity has a value of either 0 or 1, a P evaluator that gets the list of answers of SAT to the $k+2$ queries $y_1, \ldots, y_k, z_0, z_1$, can decide the membership of x in L.

Since the levels of the query hierarchy to NP and the Boolean hierarchy interleave, there remain only $\mathrm{NP}(k+1)$ and $\mathrm{NP}(k+2)$ as possible candidates for $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$ to coincide with. Observe furthermore that $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$, like the odd levels of the Boolean hierarchy [CGH+88], is closed under union with NP sets. That is, for $L_0 \in \mathrm{NP}//\oplus^{\mathrm{NP}[k]}$ and $L_1 \in \mathrm{NP}$, we have $L_0 \;\cup\; L_1 \in \mathrm{NP}//\oplus^{\mathrm{NP}[k]}$. On the other hand, even-numbered levels of the Boolean hierarchy are closed under union with NP sets only if the Boolean hierarchy collapses [CGH+88]. Hence, if $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$ coincides with a level of the Boolean hierarchy, we expect the level to be odd. Therefore, from the above two candidates just one remains and we show in the next theorem that indeed each class $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$ coincides with the next odd level of the Boolean hierarchy, that is $\mathrm{NP}(k+1)$, if k is even and $\mathrm{NP}(k+2)$, if k is odd.

Theorem 3.1 *For all $k \geq 0$, we have*

$$\mathrm{NP}//\oplus^{\mathrm{NP}[2k+1]} \;=\; \mathrm{NP}//\oplus^{\mathrm{NP}[2k+2]} \;=\; \mathrm{NP}(2k+3).$$

Proof. Clearly $\mathrm{NP}//\oplus^{\mathrm{NP}[2k+1]}$ is contained in $\mathrm{NP}//\oplus^{\mathrm{NP}[2k+2]}$. To show that $\mathrm{NP}//\oplus^{\mathrm{NP}[2k+2]} \subseteq \mathrm{NP}(2k+3)$, let $L \in \mathrm{NP}//\oplus^{\mathrm{NP}[2k+2]}$. By equation (5), we can express L as

$$L \;=\; A_0\, E_0 \triangle \mathop{\triangle}_{i=1}^{2k+2} \Big(A_i\, E_{(i-1)\mathrm{mod}2} \triangle A_i\, E_{i\mathrm{mod}2} \Big),$$

for NP sets A_i, for $i = 0, \ldots, 2k+2$, E_0, and E_1 as defined in Section 2. The crucial observation now is that we can somehow *fold* any two consecutive terms of the big symmetric difference in the way stated explicitly in the following lemma. The proof is elementary and thus omitted.

Lemma 3.2 (Folding Lemma) *For all sets B_0, B_1, F_0, and F_1 such that $B_0 \supseteq B_1$, we have*

$$B_0\, F_0 \triangle B_0\, F_1 \triangle B_1\, F_0 \triangle B_1\, F_1 \quad = \quad (B_0\, F_0 \cup B_1\, F_1) \;\triangle\; (B_0\, F_1 \cup B_1\, F_0).$$

Note that while the left part of this equation has the form of a set in NP(4), this set is in fact in NP(2) by the right part of the equation.

We apply the Folding Lemma as described above and get

$$L \;=\; A_0\, E_0 \;\triangle\; \mathop{\triangle}_{i=1}^{k+1} \Big(A_{2i-1}\, E_0 \cup A_{2i}\, E_1 \Big) \triangle \Big(A_{2i-1}\, E_1 \cup A_{2i}\, E_0 \Big).$$

Hence, we have $L \in \mathrm{NP}(2k+3)$.

To show $\mathrm{NP}(2k+3) \subseteq \mathrm{NP}//\oplus^{\mathrm{NP}[2k+1]}$, let $L \in \mathrm{NP}(2k+3)$. Then, there exist sets $A_1, \ldots, A_{2k+3}$ in NP such that $A_1 \supseteq \cdots \supseteq A_{2k+3}$ and $L = A_1 \triangle \cdots \triangle A_{2k+3}$. Because of the inclusion structure of the sets A_i,

$$L \;=\; \Big(A_1 - (A_2 \triangle \cdots \triangle A_{2k+2}) \Big) \cup A_{2k+3}.$$

Let us define f as

$$f(x) \;=\; (A_2(x) \;+\; \cdots \;+\; A_{2k+2}(x)) \bmod 2.$$

Clearly, $f \in \oplus^{\mathrm{NP}[2k+1]}$ and we have

- if $f(x) = 0$ then $x \in L \iff x \in A_1$, and

- if $f(x) = 1$ then $x \in L \iff x \in A_{2k+3}$.

Therefore, given $f(x)$, an NP machine can decide membership of x in L. Hence, $L \in \mathrm{NP}//\oplus^{\mathrm{NP}[2k+1]}$. $\square$

From Theorems 1.2 and 3.1, we get

Corollary 3.3 *For all $k \geq 0$, we have*

$$\mathrm{NP}//\mathrm{P}_{tt}^{\mathrm{NP}[2k+1]} = \mathrm{NP}//\mathrm{P}_{tt}^{\mathrm{NP}[2k+2]} = \mathrm{NP}(2k+3).$$

Here, a class of sets (as $\mathrm{P}_{tt}^{\mathrm{NP}[2k+1]}$) has to be read as a class of zero-one valued functions.

4 Modulo Functions

In this section, we study the classes $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ and $\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ with arbitrary values of $m \geq 2$.

First of all, note that if the number of queries, k, is smaller than the modulus m, then a Mod function is in fact a # function; i.e., $\mathrm{Mod}_m^{\mathrm{NP}[k]} = \#^{\mathrm{NP}[k]}$ for $1 \leq k < m$. It follows from Theorems 1.3 and 3.1 that

$$\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]} \;=\; \mathrm{NP}//\oplus^{\mathrm{NP}[2k]}, \quad \text{for } 1 \leq k < m.$$

As a consequence of the next theorem, it follows that $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ remains unchanged for all $k = m-1, \ldots, 2m-2$; i.e., $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]} = \mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[m-1]} = \mathrm{NP}//\oplus^{\mathrm{NP}[m-1]}$ for $m-1 \leq k \leq 2m-2$. For larger values of k, the classes $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ show their "normal" behavior. Our first result states that no $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ function class gives more information to NP evaluators than $\oplus^{\mathrm{NP}[k]}$.

Theorem 4.1 *For all $m \geq 2$ and $k \geq 2m-2$, we have*

$$\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]} \subseteq \mathrm{NP}//\oplus^{\mathrm{NP}[k]}.$$

Proof. By Theorem 3.1, it suffices to show this claim for even k. Assume that k is even. Let $L \in \mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$. By equation (5), we can write L as

$$L \;=\; A_0\, E_0 \;\triangle\; \mathop{\triangle}_{i=1}^{k} \Big(A_i\, E_{i-1} \triangle A_i\, E_i \Big),$$

for NP sets A_i forming a decreasing chain, for $i = 1, \ldots, k$, and E_j, for $j = 0, \ldots, m-1$, where indices of sets E_i are taken modulo m.

We will show by induction on k that, by appropriately applying the Folding Lemma, we can cut down to half the number of symmetric differences needed to express L, thereby getting $L \in \mathrm{NP}(k+1) = \mathrm{NP}/\!/\oplus^{\mathrm{NP}[k]}$. For the inductive argument, we slightly weaken our assumption on the sets A_i as done in the following lemma. This will complete the proof. $\square$

Lemma 4.2 *Let L be a set that can be written as*

$$L \;=\; A_0\, E_0 \;\triangle\; \mathop{\triangle}_{i=1}^{k} \Big(A_i\, E_{i-1} \triangle A_i\, E_i \Big),$$

for NP sets A_i, for $i = 1, \ldots, k$, and E_j, for $j = 0, \ldots, m-1$, $k \geq 2m-2$ is even, and $(A_0 \cap A_1 \cap \cdots \cap A_{m-1}) \supseteq A_m \supseteq A_{m+1} \supseteq \cdots \supseteq A_k$. Then $L \in \mathrm{NP}(k+1)$.

Proof. Let $k = 2m-2$ for the base case. We can apply the Folding Lemma as follows. For $i = 1, \ldots, m-2$, we fold $A_i\, E_{i-1} \triangle A_i\, E_i$ and $A_{i+m}\, E_{i-1} \triangle A_{i+m}\, E_i$.

But there remain now five terms where the Folding Lemma doesn't apply to, namely $A_0\, E_0$, $A_{m-1}\, E_{m-1}$, $A_m\, E_0$, $A_m\, E_{m-1}$, and $A_{m-1}\, E_{m-2}$. However, with the following generalized version, we can fold the first four terms, so that there remains only $A_{m-1}\, E_{m-2}$ unfolded.

Lemma 4.3 (Generalized Folding Lemma) *For all sets B_0, B_1, B_2, F_0, and F_1 such that $B_0 \cap B_1 \supseteq B_2$, we have*

$$B_0\, F_0 \triangle B_1\, F_1 \triangle B_2\, F_0 \triangle B_2\, F_1 \;=\; (B_0\, F_0 \cup B_2\, F_1) \;\triangle\; (B_1\, F_1 \cup B_2\, F_0).$$

Therefore, we have

$$\begin{aligned} L \;=\;& (A_0\, E_0 \cup A_m\, E_{m-1}) \triangle (A_{m-1} E_{m-1} \cup A_m\, E_0) \\ & \triangle \mathop{\triangle}_{i=1}^{m-2} (A_i\, E_{i-1} \cup A_{m+i}\, E_i) \triangle (A_i\, E_i \cup A_{m+i}\, E_{i-1}) \\ & \triangle\; A_{m-1}\, E_{m-2}. \end{aligned}$$

Thus, $L \in \mathrm{NP}(2m-1)$.

For the induction step, let $k > 2m-2$ be even. Here, we fold $A_0\, E_0 \triangle A_2\, E_1$ with $A_{m+1}\, E_0 \triangle A_{m+1}\, E_1$, getting

$$L \;=\; (A_0\, E_0 \cup A_{m+1}\, E_1) \;\triangle\; (A_2\, E_1 \cup A_{m+1}\, E_0)$$

$$\triangle\ A_1\,E_0 \triangle A_1\,E_1$$
$$\triangle\ A_2\,E_2\ \triangle\ \mathop{\triangle}_{i=3}^{m}(A_i\,E_{i-1} \triangle A_i\,E_i)$$
$$\triangle\ \mathop{\triangle}_{i=m+2}^{k}(A_i\,E_{i-1} \triangle A_i\,E_i).$$

(Indices of sets E_i have to be taken modulo m.) Now, we only have to renumber the sets appropriately so that we can apply the induction hypothesis. That is, we define sets A'_i for $i = 0, \ldots, k-2$ as follows. For $i \neq m-1$, let $A'_i = A_{i+2}$. That is, we shift all the indices by two. Note that A_{m+1} is already folded. But this can be replaced by A_1 because $E_{(m+1)\mathrm{mod}m} = E_1$. Therefore, we define $A'_{m-1} = A_1$. Note that, by this rearrangement, we again have $(A'_0 \cap A'_1 \cap \cdots \cap A'_{m-1}) \supseteq A'_m \supseteq A'_{m+1} \supseteq \cdots \supseteq A'_{k-2}$.

Now, we define sets E'_j for $j = 0, \ldots, m-1$ by simply shifting all the indices by two, i.e., $E'_j = E_{(j+2)\mathrm{mod}m}$. Then L can be written as

$$\begin{aligned} L &= (A_0\,E_0 \cup A_{m+1}\,E_1)\ \triangle\ (A_2\,E_1 \cup A_{m+1}\,E_0) \\ &\quad \triangle\ A'_0\,E'_0\ \triangle\ \mathop{\triangle}_{i=1}^{k-2}(A'_i\,E'_{i-1} \triangle A'_i\,E'_i). \end{aligned}$$

By the induction hypothesis, the second line corresponds to a set in $\mathrm{NP}(k-1)$. Hence, L is in $\mathrm{NP}(k+1)$. □

From Theorems 1.3, 3.1, and 4.1, it follows that all the $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ classes are identical for $k = m-1, \ldots, 2m-2$.

Corollary 4.4 *For all $m \geq 2$ and $m-1 \leq k \leq 2m-2$, we have*

$$\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]} = \mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[m-1]} = \mathrm{NP}//\oplus^{\mathrm{NP}[m-1]}.$$

Clearly, for all $n, m \geq 2$ such that n divides m, we have $\mathrm{NP}//\mathrm{Mod}_n^{\mathrm{NP}[k]} \subseteq \mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ (and the same holds for the corresponding P// classes). Therefore, for even m, the inclusion relation in Theorem 4.1 becomes an equality.

Corollary 4.5 *For all even $m \geq 2$ and $k \geq 2m-2$, we have*

$$\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]} = \mathrm{NP}//\oplus^{\mathrm{NP}[k]}.$$

Corollary 4.5 provides a tight characterization of the $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ classes for even moduli. For odd moduli, we will show upper and lower bounds (Corollary 4.7). The upper bound is given by Theorem 4.1 and the lower bound follows from the next theorem.

Theorem 4.6 *For all odd $m > 2$ and $k \geq 0$, (slightly abusing notation) we have*

$$\oplus^{\mathrm{NP}[k-\lfloor k/m \rfloor]} \subseteq \mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]}.$$

Proof. Let $l = k - \lfloor k/m \rfloor$. Let $f \in \oplus^{\mathrm{NP}[l]}$ and let $A_1, \ldots, A_l$ be the NP sets associated with f. For any x, if i_0 is the maximal i such that $x \in A_i$, then $f(x) = i_0 \bmod 2$. Let h_i be a many-one reduction from A_i to SAT, for $i = 1, \ldots, l$. Then i_0 is the maximal i such that $h_i(x) \in \mathrm{SAT}$.

We construct a $\#^{\mathrm{NP}[k]}$ function f' such that $f(x) = (f'(x) \bmod m) \bmod 2$. Since m is odd, we cannot just ask $h_i(x)$, for $i = 1, \ldots, l$, because, for example, $(i_0 \bmod m) \bmod 2 \neq i_0 \bmod 2$ for $m \leq i_0 \leq 2m - 1$. The idea now is to introduce an extra query per every m queries, thereby correcting the parity. That is, $f'(x)$ asks all the queries $h_i(x)$, for $i = 1, \ldots, l$, and, in addition, it asks the queries $h_{j(m-1)+1}(x)$ once more, for $j = 1, 2, \ldots$, as long as $j(m-1) + 1 \leq l$. That is, the queries of f' are as follows.

$$\begin{array}{lllll}
h_1, & h_2, & \ldots, & h_{m-1}, & h_m, \\
h_m, & h_{m+1}, & \ldots, & h_{2m-2}, & h_{2m-1}, \\
h_{2m-1}, & h_{2m}, & \ldots, & h_{3m-3}, & h_{3m-2} \\
 & & \vdots & & \\
 & & \ldots, & h_l &
\end{array}$$

Then the total number of queries is k and we have $f(x) = (f'(x) \bmod m) \bmod 2$. Hence, f is in $\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$. □

Corollary 4.7 *For all odd $m > 2$ and $k \geq 2m - 2$, we have*

$$\mathrm{NP}//\oplus^{\mathrm{NP}[k-\lfloor k/m \rfloor]} \subseteq \mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]} \subseteq \mathrm{NP}//\oplus^{\mathrm{NP}[k]}.$$

Very recently, classes $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ have been characterized in terms of $\mathrm{NP}//\oplus^{\mathrm{NP}[k]}$, for odd $m > 2$ [ABT96].

An analog of Corollary 4.7 clearly holds for classes $\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$, for odd m. However, in this case we can even show that the lower bound is indeed a tight characterization. Thus, for odd m, $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ functions provide less information to P evaluators than $\oplus^{\mathrm{NP}[k]}$ functions, unless the Boolean hierarchy collapses.

Theorem 4.8 *For all odd $m > 2$ and $k \geq 0$, we have*

$$\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]} = \mathrm{P}//\oplus^{\mathrm{NP}[k-\lfloor k/m \rfloor]}.$$

Proof. Given Theorems 4.6 and 1.2, it suffices to prove $\mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]} \subseteq \mathrm{P}//\#^{\mathrm{NP}[k-\lfloor k/m \rfloor]}$. Let $L \in \mathrm{P}//\mathrm{Mod}_m^{\mathrm{NP}[k]}$ via a function $f \in \mathrm{Mod}_m^{\mathrm{NP}[k]}$ and a set $E \in \mathrm{P}$. Let furthermore $A_1, \ldots, A_k$ be the NP sets associated with f, and let h_i be a many-one reduction from A_i to SAT. Then $f(x) = \mathrm{SAT}(h_1(x)) + \cdots + \mathrm{SAT}(h_k(x)) \pmod m$. Since the sets A_i form a decreasing chain, we have for all x and for all i such that $1 \leq i < k$, $h_{i+1}(x) \in \mathrm{SAT}$ implies $h_i(x) \in \mathrm{SAT}$.

The key point to observe is that, since m is odd, for any x there must be an index $j_0 < m$ such that $\langle x, j_0 \rangle \in E \iff \langle x, (j_0 + 1) \bmod m \rangle \in E$. Moreover, since E is in P, we can compute j_0 in polynomial time in $|x|$. In other words, to decide x, we don't need to distinguish between values j_0 and $j_0 + 1$, because the result with respect to E is the same for these values. Therefore, when asking the oracle, we can skip one of them. That is, we ask all the queries $h_i(x)$ to SAT for $i = 1, \ldots, k$, except when $i \equiv j_0 + 1 \pmod m$. Thus, we ask at most $k - \lfloor k/m \rfloor$ queries. Let $f'(x)$ be the number of these queries that are in SAT. Obviously, f' is a $\#^{\mathrm{NP}[k-\lfloor k/m \rfloor]}$ function.

Note that, given $f'(x)$, one can in polynomial time either compute $f(x)$, if $f(x) \notin \{j_0, (j_0+1) \bmod m\}$, or determine that $f(x) \in \{j_0, (j_0+1) \bmod m\}$. By our choice of j_0, in both cases we can decide whether x is in L. Thus, $L \in \mathrm{P}//\#^{\mathrm{NP}[k-\lfloor k/m \rfloor]}$. □

5 Summary

We have considered the computational model where a P or an NP evaluator gets in addition to the input a function value from a $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ function, for various k and m. We have seen that of all $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ classes, the class of parity functions, i.e., for $m = 2$, provide most information for both P and NP evaluators. In fact, for even m, $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ is as powerful as $\mathrm{Mod}_2^{\mathrm{NP}[k]}$ (Theorem 1.2 and Corollary 4.5).

For odd m, when $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ functions are given to a P evaluator, the resulting class becomes weaker (Theorem 4.8). When $\mathrm{Mod}_m^{\mathrm{NP}[k]}$ functions are given to an NP evaluator, the resulting class is mostly weaker as well. Agrawal, Beigel, and Thierauf [ABT96] show that for odd $m > 2$ and $k \geq 2m$, we have $\mathrm{NP}//\mathrm{Mod}_m^{\mathrm{NP}[k]} = \mathrm{NP}(t)$, where

$$t = k - \lfloor (k+2)/m \rfloor + 3 + (k + \lfloor (k+2)/m \rfloor) (\mathrm{mod} 2).$$

Note that the lower bound of Corollary 4.7 is fairly close: it is at most off by four from the correct value.

Acknowledgements

For helpful discussions, we are grateful to Lane Hemaspaandra and Joel Seiferas.

References

[ABT96] M. Agrawal, R. Beigel, and T. Thierauf. Modulo Information from Nonadaptive Queries to NP. Technical Report ECCC TR96-001. Available at http://www.eccc.uni-trier.de/eccc/

[ABG90] A. Amir, R. Beigel, and W. Gasarch. Some Connections between Bounded Query Classes and Non-Uniform Complexity. In *Proceedings of the 5th Conference in Structure in Complexity Theory*, 232-243, 1990. To appear in *Information and Computation*.

[BDG88] J. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity I.* EATCS Monographs in Theoretical Computer Science. Springer-Verlag, 1988.

[Bei91] R. Beigel. Bounded queries to SAT and the boolean hierarchy. *Theoretical Computer Science*, 84:199–223, 1991.

[BH88] S. Buss and L. Hay. On truth table reducibilities to SAT and the difference hierrachy over NP. In *Proceedings of the 3rd Conference in Structure in Complexity Theory*, 224–233, 1988.

[CGH+88] J. Cai, T. Gundermann, J. Hartmanis, L. Hemachandra, V. Sewelson, K. Wagner, and G. Wechsung. The boolean hierarchy I: Structural properties. *SIAM Journal on Computing*, 17(6):1232–1252, 1988.

[HU79] J. Hopcroft and J. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley, 1979.

[Ka88a] J. Kadin. *Restricted Turing Reducibilities and the Structure of the Polynomial Time Hierarchy.* PhD thesis, Cornell University, February 1988.

[Ka88b] J. Kadin. The polynomial hierarchy collapses if the Boolean hierarchy collapses. *SIAM Journal on Computing*, 17(6):1263–1282, 1994.

[KL82] R. Karp and R. Lipton. Turing machines that take advice. *L'Enseignement Mathématique*, 28:191–209, 1982.

[KSW87] J. Köbler, U. Schöning, and K. Wagner. The difference and truth-table hierarchies of NP. *R.A.I.R.O. Informatique théorique et Applications*, 21(4):419–435, 1987.

[KT94] J. Köbler and T. Thierauf. Complexity-restricted advice functions. *SIAM Journal on Computing*, 23(2):261–275, 1994.

[LLS75] R. Ladner, N. Lynch, and A. Selman. A comparison of polynomial time reducibilities. *Theoretical Computer Science*, 1(2):103–124, 1975.

[W90] K. Wagner. Bounded query classes. *SIAM J. on Computing* 19(5), pages 833-846, 1990.

[W95] K. Wagner. Personal Communication, 1995.

[Wec85] G. Wechsung. On the boolean closure of NP. In *Proceedings of the 5th Conference on Fundamentals of Computation Theory*, pages 485–493. Springer-Verlag *Lecture Notes in Computer Science #199*, 1985. (An unpublished precursor of this paper was coauthored by K. Wagner).