Journal of Universal Computer Science, vol. 9, no. 1 (2003), 34-50
submitted: 16/8/02, accepted: 24/1/03, appeared: 28/1/03 © J.UCS

Relativizing Function Classes

Christian Glaßer
Institut für Informatik,
Julius-Maximilians-Universität Würzburg,
97072 Würzburg, Germany
`glasser@informatik.uni-wuerzburg.de`

Gerd Wechsung
Institut für Informatik,
Friedrich-Schiller-Universität Jena,
07743 Jena, Germany
`wechsung@minet.uni-jena.de`

Abstract: The operators min·, max·, and #· translate classes of the polynomial-time hierarchy to function classes. Although the inclusion relationships between these function classes have been studied in depth, some questions concerning separations remained open.

We provide oracle constructions that answer most of these open questions in the relativized case. As a typical instance for the type of results of this paper, we construct a relativized world where $\min\cdot\mathrm{P} \not\subseteq \#\cdot\mathrm{NP}$, thus giving evidence for the hardness of proving $\min\cdot\mathrm{P} \subseteq \#\cdot\mathrm{NP}$ in the unrelativized case.

The strongest results, proved in the paper, are the constructions of oracles D and E, such that $\min\cdot\mathrm{coNP}^D \subseteq \#\cdot\mathrm{P}^D \wedge \mathrm{NP}^D \neq \mathrm{coNP}^D$ and $\mathrm{UP}^E = \mathrm{NP}^E \wedge \min\cdot\mathrm{P}^E \not\subseteq \#\cdot\mathrm{P}^E$.

Key Words: Polynomial-time hierarchy, Function classes, Oracle separations
Category: F.1.3

1 Introduction

Until recently it has been common practice to study the complexity of computational problems by focusing on decision problems alone. The complexity of computing functions has gained much less attention in the literature. However, most of the known computational problems are more naturally thought of as functional computation problems. We do not care to know only whether a solution exists, but we want a solution to be the output.

Therefore, one started to study the complexity of computational problems exactly, without limiting attention to decision problems. Motivated by these ideas the study of function classes has become an increasingly active area of research in the last few years.

With help of so-called operators one can derive natural classes of functions from existing classes of languages. For example, assume that we are given a language L that contains pairs of words. Then, given some word x, it is a functional problem to determine the smallest word y such that $(x, y) \in L$. We describe this translation from languages to functions with the operator min·. Analogously one obtains operators max$\cdot$ (resp., $\#\cdot$) when looking (in a certain range) for the maximal word y (resp., for the number of words y) such that $(x, y) \in L$. For exact definitions of these operators the reader is referred to Section 2.

As early as 1979, Valiant [Val79a] introduced the famous class #P, which turned out to be $\#\cdot$P. Along this line, Köbler, Schöning, and Torán [KST89, Köb89] considered a class spanP, which has been identified to be $\#\cdot$NP, and furthermore classes $\#\cdot$coNP and $\#\cdot\Delta_2^{\mathrm{P}}$. The study of optimization problems by Krentel [Kre88, Kre92] gave rise to classes max$\cdot$NP and min$\cdot$NP. These classes have been also investigated by Köbler, Schöning, and Torán [KST89, Köb89], and by Vollmer and Wagner [Vol94, VW95]. Closure properties of these and other function classes have been studied by Ogiwara and Hemachandra [OH93]. So in view of certain classes of the polynomial-time hierarchy, the operators min$\cdot$, max$\cdot$, and $\#\cdot$ have gained much interest.

When we apply these operators to all classes of the polynomial-time hierarchy, we obtain corresponding hierarchies of function classes. Hempel and Wechsung [HW97] investigated how these hierarchies are interrelated. After revealing the inclusion structure they concentrated on finding evidences for the separation of certain function classes. For many classes it turned out that the problem of their separation is equivalent to the problem of separating certain complexity classes, e.g., P $\neq$ NP. However, in some cases, no such satisfactory result could be found. Figure 1, which is a copy from [HW97], shows all known results (see [HW97] for a discussion of attribution). Mainly three kinds of results exist:

1. Inclusions between function classes like min$\cdot$NP $\subseteq$ $\#\cdot$coNP.
2. Inclusions between function classes that are equivalent to inclusions between complexity classes. For example, $\#\cdot$coNP $\subseteq$ min$\cdot$coNP $\Longleftrightarrow$ PP $\subseteq$ NP.
3. Inclusions between function classes that imply inclusions between complexity classes. For example, min$\cdot$coNP $\subseteq$ $\#\cdot$P $\Longrightarrow$ NP $\subseteq$ UP.

Whereas the results of type 1 and 2 are fully satisfactory, the results of type 3 raise questions: Is it possible to prove the reverse direction? Can one show the implication above for a strengthened right-hand side (e.g., UP = coNP)? What is the weakest right-hand side that implies min$\cdot$coNP $\subseteq$ $\#\cdot$P? In addition, as an exceptional case from [HW97], we mention that neither was it possible to prove min$\cdot$P $\subseteq$ $\#\cdot$NP nor was it possible to find an equivalent condition or a consequence.

Our attention is on the results of type 3 on one hand, and on the inclusion min$\cdot$P $\subseteq$ $\#\cdot$NP (i.e., the exceptional case from [HW97]) on the other hand.

We show that the results of type 3 fail to be equivalences in suitable relativized worlds. Even if we strengthen the right-hand sides (e.g., to UP = coNP) we do not obtain an equivalence in certain relativized worlds. As a consequence,

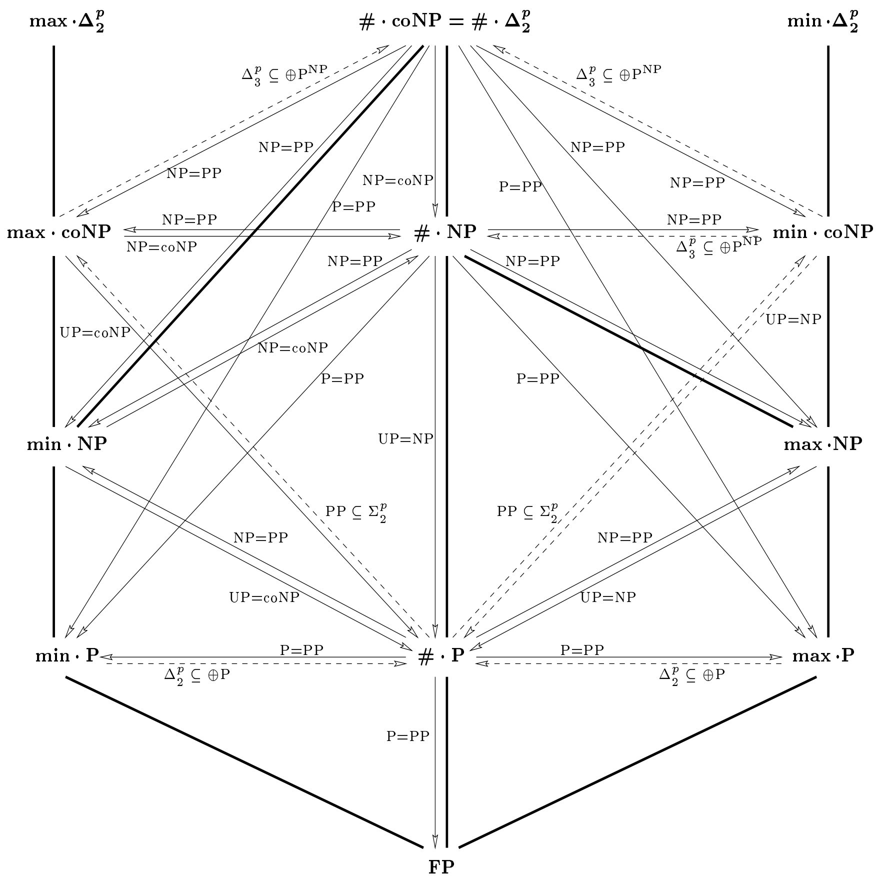

Figure 1: Inclusions between Function Classes [HW97]

- a bold line indicates that the upper class includes the lower class
- $\mathcal{F}_1 \overset{\alpha}{\dashrightarrow} \mathcal{F}_2$ means $\mathcal{F}_1 \subseteq \mathcal{F}_2 \Longrightarrow \alpha$
- $\mathcal{F}_1 \xrightarrow{\alpha} \mathcal{F}_2$ means $\mathcal{F}_1 \subseteq \mathcal{F}_2 \Longleftrightarrow \alpha$

each proof for such an equivalence does not relativize. This gives evidence for the difficulty of solving the results of type 3 in a satisfactory way. Furthermore, our oracle constructions suggest that in the unrelativized case the separation questions of type 3 are *not equivalent* to separations of known complexity classes. So the inclusion structure of function classes reveals problems that differ from separation problems for classes of the polynomial-time hierarchy.

Concerning the exceptional case from [HW97], we show that min$\cdot$P $\not\subseteq$ $\#\cdot$NP in a suitable relativized world. Using a different approach, Kosub, Schmitz, and Vollmer [KSV98] constructed a similar oracle. However, our construction additionally reaches UP = NP. This shows that even under the assumption UP = NP, we cannot prove min·P $\subseteq$ #·NP in a relativizable way. Moreover, we show that from min·P $\subseteq$ #·NP one cannot derive consequences like NP = coNP, UP = NP, or UP = coNP, unless the corresponding proofs do not relativize.

In order to obtain the relativized worlds mentioned above we have to cope with the difficulty of constructing oracles that collapse certain classes while *simultaneously* separating other classes. For this we generalize Rackoff's technique [Rac82] which goes back to Baker, Gill and Solovay [BGS75]. It allows us to construct an oracle D such that $\mathrm{UP}^D = \mathrm{NP}^D \neq \mathrm{coNP}^D$ and additional properties hold. This strengthens Rackoff's oracle E where $\mathrm{UP}^E = \mathrm{NP}^E \neq \mathrm{P}^E$ [Rac82]. The strongest results of our paper are the constructions of oracles D and E such that $\mathrm{min}\cdot\mathrm{coNP}^D \subseteq \#\cdot\mathrm{P}^D \wedge \mathrm{NP}^D \neq \mathrm{coNP}^D$ and $\mathrm{UP}^E = \mathrm{NP}^E \wedge \mathrm{min}\cdot\mathrm{P}^E \not\subseteq \#\cdot\mathrm{P}^E$.

The paper is organized as follows. After the preliminaries, we start in Section 3 with the construction of the oracles D and E. From these oracles and from oracles in the literature we derive all following results. Section 4 deals with oracles that separate certain classes shown in Figure 1. In particular, we provide a relativized world where min·P $\not\subseteq$ #·NP. In Section 5 we consider the implication min·coNP $\subseteq$ #·P $\Longrightarrow$ UP = NP and we show that it cannot be strengthened to an equivalence. Finally, in Section 6 we look again at the open separation question min·P $\not\subseteq$ #·NP. We prove that min·P $\subseteq$ #·NP does not imply consequences like UP = NP or NP = coNP, unless the proofs do not relativize. So a solution of this question is not soon forthcoming.

2 Preliminaries

We summarize some notations. Pol denotes the set of all polynomials with natural coefficients. We use NPM (resp., NPOM) as an abbreviation for nondeterministic polynomial-time bounded (oracle) Turing machine. For any NPOM M and any oracle C, $M^C(x)$ denotes the computation (computation tree) of M on input x where queries are answered according to C. The number of accepting paths of the computation $M^C(x)$ is denoted by $\#_{M^C}(x)$. Similarly, we define $M(x)$ and $\#_M(x)$ for any NPM M. Valiant [Val79a] defined #P to be the class of all functions f such that $f = \#_M$ for a suitable NPM M. Moreover, UP [Val79b] is the class of all languages L such that there exists an NPM M that accepts L and for all x, if $\#_M(x) > 0$ then $\#_M(x) = 1$.

Throughout the paper, we fix the alphabet $\Sigma = \{0, 1\}$. We use the one-one correspondence between natural numbers and Σ^*. This means that any natural number is identified with their corresponding dyadic representation. As a consequence, we consider Σ^* to be ordered such that $\varepsilon < 0 < 1 < 00 < 01 < \ldots$ (i.e., levelwise, and lexicographic within each level, ε being the empty word).

The cardinality of a set X is denoted by $\#X$ or $|X|$. For an NPOM M and an oracle C, the language that is accepted by M^C in the sense of coNP is defined as $L_{\mathrm{coNP}}(M^C) \stackrel{df}{=} \{x : M^C(x) \text{ rejects}\}$. $\Sigma^{\leq i}$ (resp., Σ^i, $\Sigma^{\geq i}$) denotes the set of words of lengths $\leq i$ (resp., $= i$, $\geq i$). We use a standard pairing function $\langle x, y\rangle$ that is polynomial-time computable and polynomial-time invertible. For $v, w \in \Sigma^*$, $v \sqsubseteq w$ means that v is a prefix of w. $\overline{A}$ denotes the complement of $A \subseteq \Sigma^*$.

This paper investigates classes of the form $\max\cdot\mathcal{C}$, $\min\cdot\mathcal{C}$ [HW97], and $\#\cdot\mathcal{C}$ [Tod91] where $\mathcal{C}$ belongs to the polynomial-time hierarchy.

Definition 1 [HW97, Tod91]. Let $\mathcal{C}$ be an arbitrary complexity class.

$$\begin{aligned}
f \in \max\cdot\mathcal{C} &\stackrel{df}{\iff} (\exists A \in \mathcal{C})(\exists p \in \mathrm{Pol})(\forall x \in \Sigma^*) \\
&\qquad [f(x) = \sup\{y \in \Sigma^{\leq p(|x|)} : \langle x, y\rangle \in A\}] \\
f \in \min\cdot\mathcal{C} &\stackrel{df}{\iff} (\exists A \in \mathcal{C})(\exists p \in \mathrm{Pol})(\forall x \in \Sigma^*) \\
&\qquad [f(x) = \inf\{y \in \Sigma^{\leq p(|x|)} : \langle x, y\rangle \in A\}] \\
f \in \#\cdot\mathcal{C} &\stackrel{df}{\iff} (\exists A \in \mathcal{C})(\exists p \in \mathrm{Pol})(\forall x \in \Sigma^*) \\
&\qquad [f(x) = \#\{y \in \Sigma^{\leq p(|x|)} : \langle x, y\rangle \in A\}]
\end{aligned}$$

Note that $\sup\emptyset = 0$ and define $\inf\emptyset = 2^{p(|x|)+1}$.

Proposition 2 [Val79a, Tod91]. $\#\mathrm{P} = \#\cdot\mathrm{P}$

3 Oracle Constructions

Rackoff [Rac82] constructed an oracle E such that $\mathrm{UP}^E = \mathrm{NP}^E \neq \mathrm{P}^E$. We generalize this construction and obtain the following stronger result.

Theorem 3. *There exists an oracle D such that*

$$\min\cdot\mathrm{coNP}^D \subseteq \#\cdot\mathrm{P}^D \textit{ and } \mathrm{NP}^D \neq \mathrm{coNP}^D.$$

The idea of the proof is the following. We define a function m^D which is complete for $\min\cdot\mathrm{coNP}^D$, and a set $L^D \stackrel{df}{=} \{x : \exists y(|x| = |y| \wedge y \in D)\}$ which is in NP^D. We construct D such that $m^D \in \#\cdot\mathrm{P}^D$ and $L^D \notin \mathrm{coNP}^D$. The first property requires coding, while the second property requires separation. Coding is done in all oracle stages that contain words of length $4n$. Separation is done in some of the remaining stages.

We focus on the difficult case for separations. D_i denotes the oracle constructed so far such that D_i satisfies all coding requirements up to words of lengths $\leq i$ and $0^i \notin L^{D_i}$. For some NPOM M, we want to make sure that $0^i \in L^D$ if and only if $M^D(0^i)$ accepts (separation requirement). Suppose that

$M^{D_i}(0^i)$ rejects. Now we extend D_i to some D_j, $j > i$, such that D_j contains all coding requirements up to words of lengths $\leq j$ and $0^i \notin L^{D_j}$. Unfortunately, this extension may result in the acceptance of 0^i by M^{D_j} (since the machine asks for words that came into the oracle during the coding part).

We show that if this happens, then we can add some word of length i to the oracle (thus making 0^i an element of L^D) such that $M(0^i)$ still accepts. Let α be an accepting path of $M^{D_j}(0^i)$. It is not enough to fix all queries on α and to add some other word of length i to the oracle, since this could effect coding requirements. Therefore, if we want to fix a query on α, then we have to make sure that at the same time we fix all coding requirements that depend on this query. This causes new words to fix which in turn causes new coding requirements to fix, and so on. We show that this recursion does not fix too many words. So we find a word of length i that can be added to the oracle.

Proof. Let $M_1, M_2, \ldots$ be an effective enumeration of all NPOM and let p_i be the running time of M_i (independent of the oracle). We may assume $p_i(j) < j^i$ for all i and j. For an arbitrary $D \subseteq \Sigma^*$, let

$$L^D \stackrel{df}{=} \{x : \exists y(|x| = |y| \wedge y \in D)\},$$

$$m^D(w) \stackrel{df}{=} \inf \left\{ \begin{array}{c} y : |y| \leq l \text{ and } M_j^D(\langle x, y\rangle) \text{ has no} \\ \text{path accepting within } k \text{ steps} \end{array} \right\} \text{ if } w = 0^j 1 0^k 1 0^l 1 x, \text{ and}$$

$$m^D(w) \stackrel{df}{=} 0 \text{ otherwise,}$$

where $\inf \emptyset \stackrel{df}{=} 2^{l+1}$ in the second line. Depending on the context we consider m^D either as word function $m^D : \Sigma^* \longrightarrow \Sigma^*$ or as function of natural numbers $m^D : \mathbb{N} \longrightarrow \mathbb{N}$.

For functions f and g we write $f \leq_m^p g$ if there exists a polynomial-time computable function h such that $f(w) = g(h(w))$ for all w. If $\mathcal{F}$ is a function class, then $g \in \mathcal{F}$ is called $\leq_m^p$-complete in $\mathcal{F}$, if $f \leq_m^p g$ for all $f \in \mathcal{F}$. The following is obvious.

Fact 1 $L^D \in \mathrm{NP}^D$ *and* m^D *is* $\leq_m^p$*-complete for* $\min{\cdot}\mathrm{coNP}^D$.

We construct D in such a way that it satisfies two conditions.

1. $L^D \notin \mathrm{coNP}^D$.
2. For all $w \in \Sigma^*$, there exist exactly $m^D(w)$ words z such that $|z| = 3 \cdot |w|$ and $wz \in D$.

Condition 1 implies $\mathrm{NP}^D \neq \mathrm{coNP}^D$. Condition 2 implies $m^D \in \#{\cdot}\mathrm{P}^D$ and therefore, $\min{\cdot}\mathrm{coNP}^D \subseteq \#{\cdot}\mathrm{P}^D$: To see this, consider an arbitrary $f \in \min{\cdot}\mathrm{coNP}^D$. By definition, there exist an NPOM M_j and a $q \in \mathrm{Pol}$ such that $f(x) =$

$\inf\left\{y : |y| \leq q(|x|) \wedge \langle x,y\rangle \in L_{\mathrm{coNP}}(M_j^D)\right\}$. Let p_j be the running time of M_j. The function f belongs to $\#\cdot\mathrm{P}^D$, since $f = \#_M$ for the machine M that is defined as follows. On input x, M first computes the word $w = 0^j 10^{p_j(|x|)}10^{q(|x|)}1x$. Then in a nondeterministic way it generates all words z such that $|z| = 3\cdot|w|$. For each generated z, the machine queries wz and accepts if the answer is positive. By the construction of D, the number of accepting paths is $m^D(w) = f(x)$.

We identify oracles with the sequences of values of their characteristic functions c_D (w.r.t. the fixed ordering of Σ^*). An initial segment D' of D extends up to level n, if $D' = (c_D(\varepsilon), c_D(0), \ldots, c_D(1^n))$.

We define D by constructing a sequence $(D_i)_{i\in\mathbb{N}}$ of its initial segments such that $D_0 = \emptyset$, D_i extends up to level i, and $D_i \sqsubseteq D_{i+1}$. If D_i is used as oracle for a machine that queries words longer than i, then the machine gets negative answers on these queries.

During step i it may happen that words of lengths greater than i must be put into $\overline{D}$. Since in this step, D is constructed only up to level i, we shall consider those words to be frozen, which means that they are reserved for the complement of D. Let F_i be the set of words from $\Sigma^{\geq i}$ that are frozen at the beginning of step i. $F_0 \stackrel{df}{=} \emptyset$. We maintain the following condition.

$$|F_i| \leq 2^{\frac{i}{4}} \tag{$*$}$$

We keep a list L of natural numbers (indices of machines) which is initially empty. If $r \in L$, then $L^D \neq L_{\mathrm{coNP}}(M_r^D)$ is guaranteed. We describe step i of the construction of D (i.e., the construction of D_i).

$\boldsymbol{i \equiv 0 (\mathrm{mod}\ 4)}$ — In this case, we contribute to satisfy condition 2. For each $w \in \Sigma^{\frac{i}{4}}$ we determine $m^D(w)$. This can be done as follows: If $w = 0^j10^k10^l1x$, then we determine the least y such that $|y| \leq l$ and $M_j^{D_{i-1}}(\langle x,y\rangle)$ has no path accepting within k steps. Since only words of lengths $\leq k < i$ can be queried, we can use D_{i-1} instead of D.

Note that $m^D(w) \leq 2^{\frac{i}{4}}$. Because of $(*)$, there are at least $2^{\frac{3i}{4}} - 2^{\frac{i}{4}}$ words of lengths i with prefix w that are not frozen. Hence, for every w we find $m^D(w)$ words z such that wz is not frozen and $|wz| = i$. We obtain D_i by adding for every w exactly $m^D(w)$ such words to D_{i-1}. In this step, no new words are frozen. Hence, condition $(*)$ is maintained.

$\boldsymbol{i \not\equiv 0 (\mathrm{mod}\ 4)}$ — Let r be the smallest number not in L.

Case 1. — $F_i \neq \emptyset$ or $2 \cdot p_r(i)^2 \geq 2^{\frac{i}{4}}$.

Let $D_i = D_{i-1}$. No new words are frozen. Hence, condition $(*)$ is maintained. If Case 1 happens, then after a finite number of steps we reach an i such that the condition of Case 2 is satisfied.

Case 2. — $F_i = \emptyset$ and $2 \cdot p_r(i)^2 < 2^{\frac{i}{4}}$.

We contribute to satisfy condition 1. We want to achieve $L^D \neq L_{\mathrm{coNP}}(M_r^D)$.

Case 2.1. — $M_r^{D_{i-1}}(0^i)$ accepts.

Fix an accepting path α. We extend D_{i-1} such that α is preserved. $M_r^{D_{i-1}}$

on input 0^i runs at most $p_r(i) < 2^{i/4}$ steps. Thus, α contains no more than $2^{i/4}$ queries. Since $F_i = \emptyset$, we can choose a word w of length i such that w is not queried on α. Let $D_i = D_{i-1} \cup \{w\}$ and freeze all queries on α that are of lengths $\geq i$ (these are negative queries). This guarantees $L^D \neq L_{\text{coNP}}(M_r^D)$, since $0^i \in L^D$, but $0^i \notin L_{\text{coNP}}(M_r^D)$. Add r to L. Note that condition $(*)$ is satisfied.

Case 2.2. — $M_r^{D_{i-1}}(0^i)$ rejects.

We continue the construction of oracle segments $D'_i, D'_{i+1}, \ldots, D'_{p_r(i)}$ where all steps $i' \not\equiv 0(\text{mod } 4)$ are skipped. The reason for this *tentative* construction phase is that $D'_{p_r(i)}$ is large enough to answer all queries of M_r on input 0^i.

Case 2.2.1. — $M_r^{D'_{p_r(i)}}(0^i)$ rejects.

Let $D_{p_r(i)} = D'_{p_r(i)}$. We claim that $M_r^D(0^i)$ rejects. This holds, since M_r on input 0^i asks queries of lengths $\leq p_r(i)$. Thus, any extension of $D_{p_r(i)}$ will have no influence on the computation of M_r on input 0^i. During the tentative construction phase we added no words of length i. Hence $0^i \notin L^D$ and therefore, $L^D \neq L_{\text{coNP}}(M_r^D)$. Add r to L. No words are frozen, $(*)$ is satisfied. The next step $p_r(i) + 1$ can be carried out.

Case 2.2.2. — $M_r^{D'_{p_r(i)}}(0^i)$ accepts.

Fix an accepting path α. We will construct a new oracle $D_{p_r(i)}$ such that $w \in D_{p_r(i)}$ for some $w \in \Sigma^i$, α remains an accepting path, and condition 2 remains satisfied. The remainder of the proof makes sure that this is possible. This shows $L^D \neq L_{\text{coNP}}(M_r^D)$.

F_1^- (resp., F_1^+) denotes the set of words of lengths $\geq i$ that are queried on α and that are answered negatively (resp., positively). In order to preserve α under oracle $D_{p_r(i)}$ we have to make sure that all queries are answered the same way as under $D'_{p_r(i)}$. Queries in F_1^- must be frozen, while for queries in F_1^+ an additional argument is needed. This will make sure that $F_1^+ \subseteq D_{p_r(i)}$, $F_1^- \subseteq \overline{D_{p_r(i)}}$, and $D_{p_r(i)}$ satisfies condition 2.

If $F_1^+ = \emptyset$, then let $F^- = F_1^-$ and $F^+ = \emptyset$. In this case we skip the following procedure and go directly to equation $(**)$ below.

Otherwise, $F_1^+ \neq \emptyset$. During the tentative construction we skipped the steps $i' \not\equiv 0(\text{mod } 4)$. Therefore, queries in F_1^+ are of the form $q = 0^j 1 0^k 1 0^l 1xy$ such that $3 \cdot |0^j 1 0^k 1 0^l 1x| = |y|$. Call $0^j 1 0^k 1 0^l 1x$ the prefix of q. We partition F_1^+ into classes $[v]$ of words that have the same length and the same prefix. Let $F_1^+ = [v_1] \cup \ldots \cup [v_s]$. For a finite set of words E, let $\ell(E) \overset{df}{=} \sum_{v \in E} |v|$. Observe that $\ell(F_1^-) + \ell(F_1^+) \leq p_r(i)$.

Let u_1 be the largest word (w.r.t. our fixed ordering) in F_1^+, let $n_1 = \#[u_1]$, and let $w = 0^j 1 0^k 1 0^l 1x$ be its prefix. Since $D'_{p_r(i)}$ satisfies condition 2, there exist exactly $m^{D'_{p_r(i)}}(w)$ words z such that $|z| = 3 \cdot |w|$ and $wz \in D'_{p_r(i)}$. From $[u_1] \subseteq F_1^+ \subseteq D'_{p_r(i)}$ it follows that $n_1 = \#[u_1] \leq m^{D'_{p_r(i)}}(w)$. Therefore, the computations

$$M_j^{D'_{p_r(i)}}(\langle x,0\rangle),\ldots,M_j^{D'_{p_r(i)}}(\langle x,n_1-1\rangle)$$

have accepting paths $\alpha_0,\ldots,\alpha_{n_1-1}$, respectively. The lengths of these paths are $\leq k$.

In order to achieve $[u_1]\subseteq D_{p_r(i)}$ it suffices to preserve the accepting paths $\alpha_0,\ldots,\alpha_{n_1-1}$. For this we have to preserve several things. First of all, we have to freeze all words of lengths $\geq i$ that are queried on $\alpha_0,\ldots,\alpha_{n_1-1}$ and that are answered negatively. F_2^- denotes the set of these words. Similarly, let P^+ be the set of words of lengths $\geq i$ that are queried on $\alpha_0,\ldots,\alpha_{n_1-1}$ and that are answered positively. Let $F_2^+=(F_1^+-[u_1])\cup P^+$. Since the lengths of the paths $\alpha_0,\ldots,\alpha_{n_1-1}$ are $\leq k<|u_1|/4$ it holds that $\ell(P^+)<n_1\cdot|u_1|/4=\ell([u_1])/4$. Therefore, $\ell(F_2^+)<\ell(F_1^+)$.

Now F_2^+ is treated similar to F_1^+, i.e., we determine the largest $u_2\in F_2^+$ and obtain sets F_3^+ and F_3^- such that $\ell(F_3^+)<\ell(F_2^+)$. We continue this procedure until $F_\nu^+=\emptyset$ for some $\nu>1$. Observe the following.

1. For every $\mu\geq 1$ we have $|u_\mu|>\max\left\{|v|:v\in F_{\mu+1}^+-F_\mu^+\right\}$. Therefore, $|u_1|\geq |u_2|\geq|u_3|\geq\cdots$ and these words have pairwise different prefixes. Hence, for all $\mu\geq 1$ and all $\nu>\mu$ it holds that F_ν^+ does not contain any words that have the same prefix as u_μ.
2. For every $\mu\geq 1$, in order to reach $[u_\mu]\subseteq D_{p_r(i)}$ in step μ, it suffices to ensure $F_{\mu+1}^+\subseteq D_{p_r(i)}$ and $F_{\mu+1}^-\subseteq\overline{D_{p_r(i)}}$.
3. Since $p_r(i)\geq\ell(F_1^+)>\ell(F_2^+)>\ldots$ is a strictly decreasing sequence, the procedure stops after at most $p_r(i)$ steps with an empty F_ν^+.

Altogether we obtain sets $F^-=F_1^-\cup F_2^-\cup\ldots$ and $F^+=F_1^+\cup F_2^+\cup\ldots$. In each of the at most $p_r(i)$ steps we add at most $p_r(i)$ words to F^- as well as to F^+. So it holds that

$$|F^-|+|F^+|\leq 2\cdot p_r(i)^2<2^{\frac{i}{4}}. \qquad (**)$$

Now we actually redo the steps of the tentative phase (again, we skip the steps $i'\not\equiv 0(\bmod 4)$). We achieve

– $0^i\in L^{D_{p_r(i)}}$ and

– $0^i\notin L_{\mathrm{coNP}}(M_r^{D_{p_r(i)}})$ by maintaining the path α.

By $(**)$, we find a word $w\in\Sigma^i$ that does not belong to F^-. Let $D_i=D_{i-1}\cup\{w\}$. Since $|w|\not\equiv 0(\bmod 4)$, this does not injure condition 2. This yields $0^i\in L^{D_i}$. For maintaining α we need the following consequence of $(**)$.

$$|F^-|<2^{\frac{i}{4}}$$

In any step i' where $i' \equiv 0(\mathrm{mod}\ 4)$, words of the form $w' = 0^j 10^k 10^l 1x$ with $|w'| = \frac{i'}{4}$ will be treated w.r.t. F^- and F^+. This means, if we have to add n words with prefix w' to the oracle, then first we choose words in F^+ that have prefix w', and then words not in F^-. Since $|F^-| < 2^{i/4}$, such words always exist. Note that w does not influence any of those words and paths that have been fixed (via F^- and F^+) during the tentative phase. It follows that $F_1^+ \subseteq F^+ \subseteq D_{p_r(i)}$ and $F_1^- \subseteq F^- \subseteq \overline{D_{p_r(i)}}$. Hence $M_r^{D_{p_r(i)}}(0^i)$ has the accepting path α. Since F^- contains only words of lengths $\leq p_r(i)$ we have no frozen words of lengths $\geq p_r(i)+1$. Therefore, $F_{p_r(i)+1} = \emptyset$ and condition $(*)$ is maintained. Finally, we add r to L. The next step $p_r(i)+1$ can be carried out. $\square$

Figure 1 shows that min·coNP $\subseteq$ #·P implies UP = NP. Since the proof is relativizable, we obtain the following.

Corollary 4. *Relative to oracle D from Theorem 3 it holds that* UP = NP *and* NP $\neq$ coNP.

Hence UP = NP and min·coNP $\subseteq$ #·P relative to the oracle from Theorem 3. However, the second condition is not necessary to reach UP = NP. Even more, the following theorem shows that min·P $\subseteq$ #·P is not necessary to reach UP = NP.

Theorem 5. *There exists an oracle D such that*

$$\mathrm{UP}^D = \mathrm{NP}^D \text{ and } \min\cdot\mathrm{P}^D \not\subseteq \#\cdot\mathrm{P}^D.$$

The idea of the proof is as follows. We use an NP-complete set K^D and a function $f^D(x) = \inf\{y : \text{the } (y+1)\text{-st word of length } |x| \text{ belongs to } D\}$ which is in min·P^D. We construct D such that $K^D \in \mathrm{UP}^D$ and $f^D \notin \#\mathrm{P}^D$. The first property requires coding, while the second property requires separation. Coding is done in all oracle stages that contain words of even lengths. Separation is done in some of the remaining stages.

D_i denotes the oracle constructed so far such that D_i satisfies all coding requirements up to words of lengths $\leq i$, and D_i does not contain any words of length i. For some NPOM M, we want to make sure that $f^D(0^i) \neq \#_{M^D}(0^i)$ (separation requirement). Assume that this is not possible. Hence, for any extension D_j of D_i, if D_j satisfies all coding requirements up to words of lengths $\leq j$, then $f^{D_j}(0^i) = \#_{M^{D_j}}(0^i)$.

Let D_j be an extension of D_i, and let α be an accepting path of $M^{D_j}(0^i)$. In the proof we show that in order to keep α accepting, it suffices to fix a small set of words in D_j. Hence, every path depends on a small number of words. However, from our assumption $f^{D_j}(0^i) = \#_{M^{D_j}}(0^i)$ it follows that there exist paths that depend on a large number of words. This is a contradiction. So we find an appropriate extension D_j.

Proof. Let $M_1, M_2, \ldots$ be an effective enumeration of all NPOM and let p_i be the running time of M_i (independent of the oracle). We may assume $p_i(j) < j^i$ for all i and j. For an arbitrary $D \subseteq \Sigma^*$, let

$$A^D \stackrel{df}{=} \Big\{(x,y) : y + 2^{|x|} - 1 \in D\Big\},$$
$$f^D(x) \stackrel{df}{=} \inf\left\{y : |y| \le |x| \wedge (x,y) \in A^D\right\} \text{ where } \inf\emptyset \stackrel{df}{=} 2^{|x|+1}, \text{ and}$$
$$K^D \stackrel{df}{=} \left\{0^j 10^k 1x : M_j^D(x) \text{ accepts within } k \text{ steps}\right\}.$$

The following fact is obvious.

Fact 2 *$A^D \in \mathrm{P}^D$, $f^D \in \min\cdot\mathrm{P}^D$ and K^D is $\le_m^p$-complete for NP^D.*

We construct D such that it satisfies the following conditions.

1. $f^D \notin \#\mathrm{P}^D$.
2. For every $x \in K^D$ there exists exactly one z such that $|x| = |z|$ and $xz \in D$, and for every $x \notin K^D$ there is no z such that $|x| = |z|$ and $xz \in D$.

By Proposition 2, the first condition implies $\min\cdot\mathrm{P}^D \not\subseteq \#\cdot\mathrm{P}^D$. Condition 2 implies $K^D \in \mathrm{UP}^D$ and therefore, $\mathrm{UP}^D = \mathrm{NP}^D$.

Similar to the proof of Theorem 3 we consider Σ^* to be ordered and we identify oracles with their characteristic sequences. Moreover, we define D by constructing a sequence $(D_i)_{i\in\mathbb{N}}$ of its initial segments such that $D_0 = \emptyset$, D_i extends up to level i, and $D_i \sqsubseteq D_{i+1}$. If D_i is used as oracle for a machine that queries words longer than i, then the machine gets negative answers on these queries. Let F_i be the set of words from $\Sigma^{\ge i}$ that are frozen at the beginning of step i. $F_0 \stackrel{df}{=} \emptyset$. We maintain the following condition.

$$|F_i| < 2^{\frac{i}{2}} \qquad (*)$$

Let L be a list of natural numbers (indices of machines) which is initially empty. If $r \in L$, then $f^D \neq \#_{M_r^D}$ is guaranteed. We describe step i of the construction of D.

***i* ≡ 0(mod 2)** — In this case, we contribute to satisfy condition 2. For each $x \in \Sigma^{i/2}$ we find out whether $x \in K^D$. This can be done as follows: If $x = 0^j 10^k 1y$, then we simulate $M_j^{D_{i-1}}$ on input y for k steps. The oracle D_{i-1} can be used instead of D, since only words of lengths $\le k < i$ are queried. By $(*)$, for every $x \in K^D \cap \Sigma^{i/2}$ there exists at least one $z \in \Sigma^{i/2}$ such that xz is not frozen. Fix such a z_x for each $x \in K^D \cap \Sigma^{i/2}$ and let $D_i = D_{i-1} \cup \{xz_x : x \in K^D \cap \Sigma^{i/2}\}$. In this step, no new words are frozen. Hence, $(*)$ is maintained.

***i* ≡ 1(mod 2)** — Let r be the smallest number not in L.

Case 1. — $F_i \neq \emptyset$ or $8 \cdot p_r(i)^2 \ge 2^{\frac{i}{2}}$.

Let $D_i = D_{i-1}$. No new words are frozen. Hence, $(*)$ is maintained. If Case 1 happens, then after a finite number of steps we reach an i such that the condition of Case 2 is satisfied.

Case 2. — $F_i = \emptyset$ and $8 \cdot p_r(i)^2 < 2^{\frac{i}{2}}$.

We contribute to satisfy condition 1, i.e., we want to reach $f^D(0^i) \neq \#_{M_r^D}(0^i)$. Let $m \overset{df}{=} 8 \cdot p_r(i)^2$ and let w_a denote the $(a+1)$-st word of length i, i.e., $w_a = a + 2^i - 1$.

We consider all continuations of the construction (i.e., segments $D'_i, D'_{i+1}, \ldots, D'_{p_r(i)}$) such that in step i we add at most $m/2$ words to the oracle and skip all odd steps $> i$. In even steps, there are several possibilities to choose the words z_x. We show that among these oracles there exists an $E \overset{df}{=} D'_{p_r(i)}$ such that E extends up to stage $p_r(i)$ and $f^E(0^i) \neq \#_{M_r^E}(0^i)$.

Assumption: $f^E(0^i) = \#_{M_r^E}(0^i)$ for all continuations $D'_i, D'_{i+1}, \ldots, D'_{p_r(i)} = E$.

We will derive a contradiction. Choose a continuation such that $D'_i = D_{i-1} \cup \{w_m\}$. By assumption, $f^E(0^i) = \#_{M_r^E}(0^i) = m$. Denote the accepting paths of the computation $M_r^E(0^i)$ by $q_1, \ldots, q_m$.

In the following we want to preserve these paths. For this we have to fix the positive and negative answers on these paths. Before we do this, let us compare the methods of Theorem 3 and of this theorem.

In the proof of Theorem 3 we are not able to fix single words, but we are able to fix groups of words that have the same prefix. To fix such a group (say of size n' and with prefix p') we have to guarantee that the min·coNP-function induced by p' has a value $\geq n'$. We do this by preserving n' paths of n' different rejecting coNP-computations. For a *single* word w' with prefix p' we cannot determine the set of words that w' depends on; for this we need all words with prefix p'. Consider the following example.

- If u' is the only word with prefix p' that we want to fix, then it suffices to guarantee that the min·coNP-function induced by p' has a value ≥ 1. Hence u' depends on path α'_0.
- If v', v'' are the only words with prefix p' that we want to fix, then it suffices to guarantee that the min·coNP-function induced by p' has a value ≥ 2. Thus either v' or v'' depends on path α'_1.

Hence, for one word w' (either v' or v'') we cannot determine the set of words that w' depends on.

In contrast to Theorem 3, in this proof we are able to determine this set for a single word w'. We show that other words that we have to fix do not change this set. Therefore, the set of words that w' depends on is well defined.

The difference between these methods is important. Otherwise it would be possible to apply the method of Theorem 5 to Theorem 3. This would result in an oracle D' such that min·coNP$^{D'} \subseteq \#$·P$^{D'}$ and min·P$^{D'} \not\subseteq \#$·P$^{D'}$. This is not possible.

We use the following method to fix words. Consider a $w \in E$ of length $> i$. Thus w is of the form $w = 0^j 10^k 1xy$ where $|y| = \frac{|w|}{2}$. Hence, $M_j^E(x)$ accepts within k steps, say on path α. Therefore, if we want to keep w in the oracle, we have to preserve α. Note that the length of α is $\leq k < \frac{|w|}{2}$. N_w denotes the set of negatively answered queries. We freeze all queries that are in this set. P_w denotes the set of positively answered queries. The sum of lengths of words in $P_w \cup N_w$ is smaller than $\frac{|w|}{2}$. We use this argument inductively to fix the words of P_w. Therefore, we can guarantee $w \in E$ by fixing at most $|w|$ words of lengths $\leq |w|$.

Consider a path q of $M_r^E(0^i)$. Note that the length of q is $\leq p_r(i)$. Thus along q, at most $p_r(i)$ words of lengths $\leq p_r(i)$ can be queried. F^+ (resp., F^-) denotes the set of positive (resp., negative) words that have to be fixed in order to preserve q.

$$|F^-| + |F^+| \leq 2 \cdot p_r(i) < 2^{\frac{i}{2}} \qquad (**)$$

For $1 \leq k \leq m$, $F_{q_k}^+$ (resp., $F_{q_k}^-$) denotes the set of positive (resp., negative) fixed words that preserve path q_k. We make a list that contains the following statements.

1. For all paths $q \in \{q_1, \ldots, q_m\}$ we have $w_0 \in F_q^- \cup F_q^+$. Otherwise let $l_1 \in \{q_1, \ldots, q_m\}$ be a path such that $w_0 \notin F_{l_1}^- \cup F_{l_1}^+$. So we can add w_0 to the oracle while preserving path l_1 (via $F_{l_1}^+$ and $F_{l_1}^-$). Hence, $f(0^i) = 0$, but $\#_{M_r}(0^i) > 0$. This contradicts our assumption.

2. For at least $m-1$ paths $q \in \{q_1, \ldots, q_m\}$ we have $w_1 \in F_q^- \cup F_q^+$. Otherwise let $l_1, l_2 \in \{q_1, \ldots, q_m\}$ be different paths such that $w_1 \notin F_{l_1}^- \cup F_{l_1}^+ \cup F_{l_2}^- \cup F_{l_2}^+$. So we can add w_1 to the oracle while preserving the paths l_1 and l_2 (via $F_{l_1}^+$, $F_{l_1}^-$, $F_{l_2}^+$, and $F_{l_2}^-$). Hence, $f(0^i) = 1$, but $\#_{M_r}(0^i) > 1$. This contradicts our assumption.

We continue this list of statements and consider the first $\frac{m}{2}$ of them. From statement k where $k \in \left\{1, \ldots, \frac{m}{2}\right\}$ we obtain the following weakened statement: For at least $\frac{m}{2}$ paths $q \in \{q_1, \ldots, q_m\}$ we have $w_k \in F_q^- \cup F_q^+$. A pigeon hole argument shows that there exists a path $\tilde{q} \in \{q_1, \ldots, q_m\}$ such that $w \in F_{\tilde{q}}^- \cup F_{\tilde{q}}^+$ for at least $\frac{m}{4}$ different words $w \in \{w_0, \ldots, w_{\frac{m}{2}-1}\}$. Together with $(**)$ this implies

$$2 \cdot p_r(i)^2 = \frac{m}{4} \leq |F_{\tilde{q}}^- \cup F_{\tilde{q}}^+| \leq |F_{\tilde{q}}^-| + |F_{\tilde{q}}^+| < 2 \cdot p_r(i)^2.$$

This is a contradiction. Therefore, the assumption is false, i.e., there exists a continuation $D_i', D_{i+1}', \ldots, E$ such that $f^E(0^i) \neq \#_{M_r^E}(0^i)$. We have no frozen words of lengths $\geq p_r(i) + 1$ and therefore, $F_{p_r(i)+1} = \emptyset$. Hence, condition $(*)$ is maintained. We add r to L. The next step $p_r(i) + 1$ can be carried out. $\square$

4 Separations between $\#\cdot$-classes and min$\cdot$-max$\cdot$-classes

We apply the oracles constructed in Section 3. In order to treat the open separation questions between $\#\cdot$-classes and min$\cdot$-max$\cdot$-classes (Figure 1), it suffices to restrict ourselves to the strongest questions. If, for instance, $\max\cdot\mathrm{coNP}^A \subseteq \#\cdot\mathrm{P}^A$ for some oracle A, then $\mathcal{C}^A \subseteq \mathcal{D}^A$ for all classes $\mathcal{C}$ and $\mathcal{D}$ from Figure 1 such that $\mathcal{C} \subseteq \max\cdot\mathrm{coNP}$ and $\#\cdot\mathrm{P} \subseteq \mathcal{D}$. So we have to focus on the following separations.

1. $\#\cdot\mathrm{P} \not\subseteq \max\cdot\mathrm{coNP}$ and $\#\cdot\mathrm{P} \not\subseteq \min\cdot\mathrm{coNP}$
2. $\max\cdot\mathrm{P} \not\subseteq \#\cdot\mathrm{P}$
3. $\min\cdot\mathrm{P} \not\subseteq \#\cdot\mathrm{NP}$

Note the slight asymmetry in the behavior of max·P and min·P. It is caused by the fact that $\max\cdot\mathrm{P} \subseteq \#\cdot\mathrm{NP}$, which is not known for min·P. We show that the separations above are possible in suitable relativized worlds.

Theorem 6. *There exists an oracle D such that*

$$\#\cdot\mathrm{P}^D \not\subseteq \max\cdot\mathrm{coNP}^D \textit{ and } \#\cdot\mathrm{P}^D \not\subseteq \min\cdot\mathrm{coNP}^D.$$

Proof. Yao [Yao85] constructed an oracle D such that PH^D does not collapse. Assume $\#\cdot\mathrm{P}^D \subseteq \max\cdot\mathrm{coNP}^D$ or $\#\cdot\mathrm{P}^D \subseteq \min\cdot\mathrm{coNP}^D$. This implies $\mathrm{PP}^D \subseteq {\Sigma_2^{\mathrm{P}}}^D$ (Figure 1, the proofs are relativizable). By Toda's theorem [Tod89], PH^D collapses. □

Torán [Tor91] constructed an oracle E such that $\mathrm{NP}^E \not\subseteq \oplus\cdot\mathrm{P}^E$. By Figure 1, $\mathrm{NP}^D \subseteq \oplus\cdot\mathrm{P}^D$ for every oracle D such that either $\min\cdot\mathrm{P}^D \subseteq \#\cdot\mathrm{P}^D$ or $\max\cdot\mathrm{P}^D \subseteq \#\cdot\mathrm{P}^D$. Therefore, the oracle from [Tor91] shows the following.

Theorem 7. *There exists an oracle E such that*

$$\min\cdot\mathrm{P}^E \not\subseteq \#\cdot\mathrm{P}^E \textit{ and } \max\cdot\mathrm{P}^E \not\subseteq \#\cdot\mathrm{P}^E.$$

Since $\max\cdot\mathrm{P} \subseteq \#\cdot\mathrm{NP}$ is relativizable, $\max\cdot\mathrm{P}^D \not\subseteq \#\cdot\mathrm{P}^D$ implies $\#\cdot\mathrm{P}^D \neq \#\cdot\mathrm{NP}^D$. Hence, for the oracle in Theorem 7 it holds that $\#\cdot\mathrm{P}^E \neq \#\cdot\mathrm{NP}^E$. However, $\min\cdot\mathrm{P}^D \not\subseteq \#\cdot\mathrm{P}^D$ is compatible with $\#\cdot\mathrm{P}^D = \#\cdot\mathrm{NP}^D$.

Theorem 8. *There exists an oracle D such that*

$$\min\cdot\mathrm{P}^D \not\subseteq \#\cdot\mathrm{P}^D = \#\cdot\mathrm{NP}^D \textit{ and } \mathrm{UP}^D = \mathrm{NP}^D \neq \mathrm{coNP}^D.$$

Proof. This follows from Theorem 5. The proofs for $\mathrm{UP} = \mathrm{NP} \Longrightarrow \#\cdot\mathrm{P} = \#\cdot\mathrm{NP}$ and $\mathrm{NP} = \mathrm{coNP} \Longrightarrow \min\cdot\mathrm{P} \subseteq \#\cdot\mathrm{NP}$ are relativizable (Figure 1). □

This shows that statement 3 (and in particular $\min\cdot\mathrm{coNP} \not\subseteq \#\cdot\mathrm{NP}$) is reachable in some relativized world.

5 Collapse Consequences of min·coNP $\subseteq$ #·P

From [HW97] the following implications are known.

- min·coNP $\subseteq$ #·P $\Longrightarrow$ UP = NP
- min·coNP $\subseteq$ #·P $\Longleftarrow$ UP = coNP

We would like to find a statement that is equivalent to min·coNP $\subseteq$ #·P. However, none of these right-hand sides is likely to serve this purpose. The following theorems show that the implications

$$\begin{aligned} \text{min}\cdot\text{coNP} \subseteq \#\cdot\text{P} &\implies \text{UP} = \text{coNP} \\ \text{UP} = \text{NP} &\implies \text{min}\cdot\text{coNP} \subseteq \#\cdot\text{P} \end{aligned}$$

are not relativizable.

Theorem 9. *There exists an oracle D such that*

$$\text{min}\cdot\text{coNP}^D \subseteq \#\cdot\text{P}^D \textit{ and } \text{UP}^D = \text{NP}^D \neq \text{coNP}^D.$$

Proof. This follows from Theorem 3, since min·coNP $\subseteq$ #·P $\Longrightarrow$ UP = NP is relativizable. □

Theorem 10. *There exists an oracle D such that*

$$\text{min}\cdot\text{coNP}^D \not\subseteq \#\cdot\text{P}^D \textit{ and } \text{UP}^D = \text{NP}^D \neq \text{coNP}^D.$$

Proof. This follows from Theorem 5 and the relativizable facts min·P $\subseteq$ min·coNP and UP = coNP $\Longrightarrow$ min·NP $\subseteq$ #·P. □

6 Collapse Consequences of min·P $\subseteq$ #·NP

The relationship between min·P and #·NP is left open in [HW97]. In Section 4 we proved that there exists an oracle relative to which min·P $\not\subseteq$ #·NP (Theorem 8). This explains why it is hard to transfer the known proof for max·P $\subseteq$ #·NP to the inclusion min·P $\subseteq$ #·NP.

It is also hard to find an implication of the form min·P $\subseteq$ #·NP $\Longrightarrow$ α. At least three possible statements α, that come to mind when considering Figure 1, are not likely to be necessary for min·P $\subseteq$ #·NP. More precisely, the following statements are not relativizable.

– $\min\cdot\mathrm{P} \subseteq \#\cdot\mathrm{NP} \Longrightarrow \mathrm{NP} = \mathrm{coNP}$

– $\min\cdot\mathrm{P} \subseteq \#\cdot\mathrm{NP} \Longrightarrow \mathrm{UP} = \mathrm{NP}$

– $\min\cdot\mathrm{P} \subseteq \#\cdot\mathrm{NP} \Longrightarrow \mathrm{UP} = \mathrm{coNP}$

The first statement follows from Theorem 9, while the last two statements follow from the next theorem.

Theorem 11. *There exists an oracle D such that*

$$\min\cdot\mathrm{P}^D \subseteq \#\cdot\mathrm{NP}^D \text{ and } \mathrm{UP}^D \neq \mathrm{NP}^D = \mathrm{coNP}^D.$$

Proof. Since $\mathrm{NP} = \mathrm{coNP} \Longrightarrow \min\cdot\mathrm{P} \subseteq \#\cdot\mathrm{NP}$ is relativizable (Figure 1), it suffices to construct an oracle for $\mathrm{UP}^D \neq \mathrm{NP}^D = \mathrm{coNP}^D$. Ogiwara and Hemachandra [OH93] constructed a stronger oracle E such that $\mathrm{UP}^E \neq \mathrm{NP}^E = \mathrm{PSPACE}^E$. □

The oracle D from Theorem 9 shows that $\min\cdot\mathrm{P} \subseteq \#\cdot\mathrm{NP} \Longrightarrow \mathrm{NP} = \mathrm{coNP}$ is not relativizable. This oracle has the additional property $\mathrm{UP}^D = \mathrm{NP}^D$ or, equivalently, $\#\cdot\mathrm{P}^D = \#\cdot\mathrm{NP}^D$. We leave it open whether the following oracle E can be constructed.

$$\min\cdot\mathrm{P}^E \subseteq \#\cdot\mathrm{NP}^E \wedge \mathrm{UP}^E \neq \mathrm{NP}^E \wedge \mathrm{NP}^E \neq \mathrm{coNP}^E$$

Acknowledgements

We are grateful to Lane A. Hemaspaandra, Harald Hempel, Sven Kosub, and Kolya Vereshchagin for many helpful comments and suggestions. We are also grateful to our referees whose hints have simplified our paper.

Moreover, we thank for support by Nortel (Northern Telecom) and by grant NSF-INT-9513368/DAAD-315-PRO-fo-ab.

References

[BGS75] T. Baker, J. Gill, and R. Solovay. Relativizations of the P=NP problem. *SIAM Journal on Computing*, 4:431–442, 1975.

[HW97] H. Hempel and G. Wechsung. The operators min and max on the polynomial time hierarchy. In *Proceedings 14th Symposium on Theoretical Aspects of Computer Science*, volume 1200 of *Lecture Notes in Computer Science*, pages 93–104. Springer Verlag, 1997.

[Köb89] J. Köbler. *Strukturelle Komplexität von Anzahlproblemen.* PhD thesis, Universität Stuttgart, Fakultät für Informatik, 1989.

[Kre88] M. W. Krentel. The complexity of optimization problems. *Journal of Computer and System Sciences*, 36:490–509, 1988.

[Kre92] M. W. Krentel. Generalizations of OptP to the polynomial hierarchy. *Theoretical Computer Science*, 97:183–198, 1992.

[KST89] J. Köbler, U. Schöning, and J. Torán. On counting and approximation. *Acta Informatica*, 26:363–379, 1989.

[KSV98] S. Kosub, H. Schmitz, and H. Vollmer. Uniformly defining complexity classes of functions. In *Proceedings 15th Symposium on Theoretical Aspects of Computer Science*, volume 1373 of *Lecture Notes in Computer Science*, pages 607–617. Springer Verlag, 1998.

[OH93] M. Ogiwara and L. Hemachandra. A complexity theory of feasible closure properties. *Journal of Computer and System Sciences*, 46:295–325, 1993.

[Rac82] C. Rackoff. Relativized questions involving probabilistic algorithms. *Journal of the ACM*, 29:261–268, 1982.

[Tod89] S. Toda. On the computational power of PP and ParityP. In *Proceedings 30th IEEE Symposium on Foundations of Computer Science*, pages 514–519. IEEE Computer Society Press, 1989.

[Tod91] S. Toda. *Computational Complexity of Counting Complexity Classes*. PhD thesis, Tokyo Institute of Technology, Department of Computer Science, Tokyo, 1991.

[Tor91] J. Torán. Complexity classes defined by counting quantifiers. *Journal of the Association for Computing Machinery*, 38:753–774, 1991.

[Val79a] L. G. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8:189–201, 1979.

[Val79b] L. G. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal of Computing*, 8(3):411–421, 1979.

[Vol94] H. Vollmer. *Komplexitätsklassen von Funktionen*. PhD thesis, Universität Würzburg, Institut für Informatik, Germany, 1994.

[VW95] H. Vollmer and K. W. Wagner. Complexity classes of optimization functions. *Information and Computation*, 120:198–219, 1995.

[Yao85] A. C. C. Yao. Separating the polynomial-time hierarchy by oracles. In *Proceedings 26th Foundations of Computer Science*, pages 1–10. IEEE Computer Society Press, 1985.