Closure of Polynomial Time Partial Information Classes under Polynomial Time Reductions

Arfst Nickelsen and Till Tantau

Technische Universität Berlin
Fakultät für Elektrotechnik und Informatik
10623 Berlin, Germany
nicke@cs.tu-berlin.de, tantau@cs.tu-berlin.de

Abstract. Polynomial time partial information classes are extensions of the class P of languages decidable in polynomial time. A partial information algorithm for a language A computes, for fixed $n \in \mathbb{N}$, on input of words $x_1, \dots, x_n$ a set P of bitstrings, called a pool, such that $\chi_A(x_1, \dots, x_n) \in P$, where P is chosen from a family $\mathcal{D}$ of pools. A language A is in P[$\mathcal{D}$], if there is a polynomial time partial information algorithm which for all inputs $(x_1, \dots, x_n)$ outputs a pool $P \in \mathcal{D}$ with $\chi_A(x_1, \dots, x_n) \in P$. Many extensions of P studied in the literature, including approximable languages, cheatability, p-selectivity and frequency computations, form a class P[$\mathcal{D}$] for an appropriate family $\mathcal{D}$.
We characterise those families $\mathcal{D}$ for which P[$\mathcal{D}$] is closed under certain polynomial time reductions, namely bounded truth-table, truth-table, and Turing reductions. We also treat positive reductions. A class P[$\mathcal{D}$] is presented which strictly contains the class P-sel of p-selective languages and is closed under positive truth-table reductions.

Keywords: structural complexity, partial information, polynomial time reductions, verboseness, p-selectivity, positive reductions.

If a language A is not decidable in polynomial time one may ask whether it nevertheless exhibits some polynomial time behaviour. If there is no polynomial time algorithm that answers the question "$x \in A$?" for all inputs x, there may still exist a *partial information algorithm*. For a tuple of input words $(x_1, \dots, x_n)$ such an algorithm outputs some partial information on membership of these words with respect to A. More precisely it narrows the range of possibilities for values of $\chi_A(x_1, \dots, x_n)$, where χ_A is the characteristic function for A. Many types of partial information have been studied, including verboseness, approximability, (strong) membership comparability, cheatability, frequency computations, easily countable languages, multiselectivity, sortability. For detailed definitions and discussions of these notions see for example [2,1,6,15,8,14,13].

To get a more unified picture, [5] introduced the recursion theoretic notion of $\mathcal{D}$-*verboseness*, where the type of partial information is specified by a family $\mathcal{D}$ of sets of bitstrings. The class of *polynomially* $\mathcal{D}$-verbose languages, whose formal definition given below, is denoted $\mathrm{P}[\mathcal{D}]$. Basic properties of these polynomial time $\mathcal{D}$-verboseness classes are presented in [19].

Reduction closures of partial information classes have been the focus of much interest, mostly due to the fact that the polynomial time Turing reduction closure of P-sel, the class of p-selective sets, is exactly P/poly. Hence, a language is Turing reducible to a p-selective language iff the language has polynomial size circuits. It is also known that P-sel is closed under positive truth-table reductions, but not under 1-tt reductions. Reductions to p-selective sets have been studied in detail in [12]. Opposed to selectivity, *cheatability* behaves quite differently: The class of n-cheatable languages is known [3] to be closed under Turing reductions. For other notions like strongly membership comparable sets [17] only little was previously known concerning their closure properties.

Another important motivation to look at reductions to partial information classes is that one can prove results of the following type: if $\mathrm{P}[\mathcal{D}]$ contains, for certain families $\mathcal{D}$, languages which are NP-hard for certain polynomial time reductions, then P = NP. One of the best results [7,8] in this respect is that if $\mathrm{P}[3\text{-}\mathrm{SIZE}_2]$ contains a language which is NP-hard for $n^{1-\epsilon}$-tt reductions, then P = NP.

We fully characterise the partial information classes $\mathrm{P}[\mathcal{D}]$ which are closed under 2-tt reductions, bounded truth-table reductions and under Turing reductions. It turns out that *exactly* the classes of n-cheatable languages are closed under any of these reductions. We also treat positive truth-table reductions and present a class $\mathrm{P}[\mathcal{D}]$ strictly containing P-sel which is closed under positive truth-table reductions.

This paper is organised as follows. First we give some definitions and basic facts concerning partial information classes. In *Section 2* we briefly discuss closure under many-one and 1-tt reductions and give a simple combinatorial characterisation of the classes $\mathrm{P}[\mathcal{D}]$ closed under these reductions. In *Section 3* we show the main theorem which characterises the classes $\mathrm{P}[\mathcal{D}]$ that are closed under different truth-table and Turing reductions. In *Section 4* we examine positive truth-table reductions. Here again, we reduce the question of whether a class $\mathrm{P}[\mathcal{D}]$ is closed under positive k-tt reductions to finite combinatorics.

1 Preliminaries

Notations. Languages are subsets of $\Sigma^* = \{0,1\}^*$. The *join* of two languages A and B is $A \oplus B := \{\, 0x \mid x \in A \,\} \cup \{\, 1x \mid x \in B \,\}$. Let $\mathbb{B} := \{0,1\}$. For a language A the *characteristic function* $\chi_A\colon \Sigma^* \to \mathbb{B}$ is defined by $\chi_A(x) = 1$ iff $x \in A$. We extend χ_A to tuples of words by $\chi_A(x_1,\dots,x_n) := \chi_A(x_1)\cdots\chi_A(x_n)$. In the following, elements of Σ^* for which membership in languages is of interest are called words, elements of $\mathbb{B}^*$ which are considered as possible values of characteristic functions are called bitstrings. For a bitstring b the number of 1's in b is denoted $\#_1(b)$, $b[i]$ is the i-th bit of b, and $b[i_1,\dots,i_k] := b[i_1]\cdots b[i_k]$. We extend this to sets of bitstrings by setting $P[i_1,\dots,i_k] := \bigl\{\, b[i_1,\dots,i_k] \mid b \in P \,\bigr\}$.

Partial information classes. In order to define partial information classes, we first need the notion of pools and families.

Definition 1 (n-Pools, n-Families). *Let $n \geq 1$. A subset $P \subseteq \mathbb{B}^n$ is called an* n-pool. *A set $\mathcal{D} = \{P_1, \dots, P_r\}$ of n-pools is called an* n-family *if*

1. *$\mathcal{D}$ covers $\mathbb{B}^n$, that is $\bigcup_{i=1}^r P_i = \mathbb{B}^n$, and*
2. *$\mathcal{D}$ is closed under subsets, that is $P \in \mathcal{D}$ and $Q \subseteq P$ implies $Q \in \mathcal{D}$.*

Definition 2 (Polynomially $\mathcal{D}$-Verbose). *For a given n-family $\mathcal{D}$ a language A is in* the partial information class $\mathrm{P}[\mathcal{D}]$ *(respectively* $\mathrm{P}_{\mathrm{dist}}[\mathcal{D}]$*) iff there is a polynomially time-bounded deterministic Turing machine that on input of n words (respectively distinct words) $x_1, \dots, x_n$ outputs a pool $Q \in \mathcal{D}$ such that $\chi_A(x_1, \dots, x_n) \in Q$. The languages in* $\mathrm{P}[\mathcal{D}]$ *are called* polynomially $\mathcal{D}$-verbose.

We present some definitions that make it easier to deal with polynomial $\mathcal{D}$-verboseness and then state some known facts which will be applied in the following. For more details on polynomial $\mathcal{D}$-verboseness see [19].

Definition 3 (Operations on Bitstrings).

1. *Let S_n be the group of permutations of $\{1, \dots, n\}$. For $\sigma \in S_n$ and $b \in \mathbb{B}^n$ we define $\sigma(b) := b\big[\sigma(1)\big] \cdots b\big[\sigma(n)\big]$.*
2. *For $i \in \{1, \dots, n\}$ and $c \in \mathbb{B}$ define* projections *$\pi_i^c \colon \mathbb{B}^n \to \mathbb{B}^n$ by $\pi_i^c(b) := b[1] \cdots b[i-1]\, c\, b[i+1] \cdots b[n]$.*
3. *For $i, j \in \{1, \dots, n\}$ define a* replacement operation *$\rho_{i,j} \colon \mathbb{B}^n \to \mathbb{B}^n$ by $\rho_{i,j}(b) := b'$ where $b'[k] := b[k]$ for $k \neq j$ and $b'[j] := b[i]$.*

We extend these operations from bitstrings to pools by $\omega(P) := \{\omega(b) \mid b \in P\}$. An n-family $\mathcal{D}$ is said to be closed under permutations, projections and replacements *if for all permutations, projections and replacements ω and all $P \in \mathcal{D}$ we have $\omega(P) \in \mathcal{D}$.*

Definition 4 (Normal Form). *An n-family is* in normal form *if it is closed under permutations, projections and replacements.*

Fact 5 (Normal Form). *For every n-family $\mathcal{D}$ there is a unique n-family $\mathcal{D}'$ in normal form with $\mathrm{P}[\mathcal{D}] = \mathrm{P}[\mathcal{D}']$.*

Fact 6 (Class Inclusion Reduces to Family Inclusion). *For all n-families $\mathcal{D}$ and $\mathcal{E}$ in normal form we have $\mathrm{P}[\mathcal{D}] \subseteq \mathrm{P}[\mathcal{E}]$ iff $\mathcal{D} \subseteq \mathcal{E}$.*

Fact 7 (Change of Tuple Length). *Let $\mathcal{D}$ be an m-family and $m < n$. Define the following n-family $\lceil \mathcal{D} \rceil_n := \{\, P \subseteq \mathbb{B}^n \mid \forall i_1 < \dots < i_m \colon P[i_1, \dots, i_m] \in \mathcal{D} \}$. Then $\mathrm{P}[\mathcal{D}] = \mathrm{P}\big[\lceil \mathcal{D} \rceil_n\big]$. Furthermore, if $\mathcal{D}$ is in normal form so is $\lceil \mathcal{D} \rceil_n$.*

Fact 8 (Intersection). *For all n-families $\mathcal{D}$ and $\mathcal{E}$ in normal form we have $\mathrm{P}[\mathcal{D}] \cap \mathrm{P}[\mathcal{E}] = \mathrm{P}[\mathcal{D} \cap \mathcal{E}]$.*

Definition 9 (Generated n-Family). *For n-pools $D_1, \ldots, D_r$ the minimal n-family $\mathcal{D}$ in normal form for which $\{D_1, \ldots, D_r\} \subseteq \mathcal{D}$ is denoted by $\langle D_1, \ldots, D_r \rangle$. It is the closure of $\{D_1, \ldots, D_r\}$ under subsets, permutations, projections and replacements. We say that $D_1, \ldots, D_r$* generate *$\langle D_1, \ldots, D_r \rangle$.*

Some n-families are of special interest as their partial information classes have been studied extensively in the literature. We write these special families in capital letters with the tuple length attached as index.

Definition 10 (Special Families).

1. *Let* $\text{SEL}_n := \big\langle \big\{ 0^{n-i}1^i \mid 0 \leq i \leq n \big\} \big\rangle$.
2. *For* $1 \leq k \leq 2^n$ *let* $k\text{-SIZE}_n := \{ P \subseteq \mathbb{B}^n \mid |P| \leq k \}$.
3. *For* $1 \leq k \leq n+1$ *let* $k\text{-CARD}_n := \big\{ P \subseteq \mathbb{B}^n \mid |\{ \#_1(b) \mid b \in P \}| \leq k \big\}$.
4. *Let* $\text{BOTTOM}_n := \big\langle \big\{ b \mid \#_1(b) \leq 1 \big\} \big\rangle$ *and* $\text{TOP}_n := \big\langle \big\{ b \mid \#_1(b) \geq n-1 \big\} \big\rangle$.

The class $\text{P}[\text{SEL}_2]$ is exactly the class P-sel of *p-selective languages*, that is languages A which have a polynomial time selector function. Such a selector gets two words u and v as input and selects one of them. Provided $u \in A$ or $v \in A$, the selected word must also lie in A. The class of p-selective languages has been extensively studied, starting with [20].

Fact 11 (SEL). P-sel $= \text{P}[\text{SEL}_2] = \text{P}[\text{SEL}_n]$ *for* $n \geq 2$.

Languages in $\text{P}\big[(2^n-1)\text{-SIZE}_n\big]$ are called *n-approximable*, *n-membership comparable*, *non-n-p-superterse* or *n-p-verbose*. The languages in $\text{P}[n\text{-SIZE}_n]$ are sometimes called *n-cheatable* in the literature, but especially in the older literature this term is also used for the languages in $\text{P}[2^n\text{-SIZE}_{2^n}]$. The following important fact is implicitly proven in [4]:

Fact 12 (SIZE). $\text{P}[k\text{-SIZE}_k] = \text{P}[k\text{-SIZE}_n]$ *for* $n \geq k$.

Languages in $\text{P}[n\text{-CARD}_n]$ are called *easily countable* [15]. The languages in $\text{P}[\text{TOP}_n]$ and $\text{P}[\text{BOTTOM}_n]$ have no special names in the literature, but they will come up in different places in the following proofs.

Reductions. In this paper all reductions under consideration will be polynomial time truth-table or Turing reductions defined in the standard way, see [18] for detailed definitions. We write $\leq^{\text{p}}_{\text{m}}$ for many-one reductions, $\leq^{\text{p}}_{k\text{-tt}}$ for truth-table reductions with k queries, $\leq^{\text{p}}_{\text{btt}}$ for truth-table reductions with a constant number of queries, $\leq^{\text{p}}_{\text{tt}}$ for truth-table reductions with a polynomial number of queries, and $\leq^{\text{p}}_{\text{T}}$ for Turing reductions. We write $\leq^{\text{p}}_{k\text{-ptt}}$ for positive truth-table reductions with k queries. In a positive reduction for each input word the boolean function that evaluates the answers to the queries is a monotone function. We write $\leq^{\text{p}}_{\text{ptt}}$ for positive truth-table reductions with a polynomial number of queries.

2 Many-One and 1-tt Reductions

We review what is known [19] about closure under polynomial time many-one and 1-tt reductions for classes $\mathrm{P}[\mathcal{D}]$ and $\mathrm{P}_{\mathrm{dist}}[\mathcal{D}]$.

Theorem 13 (Many-One Reductions). *Let $\mathcal{D}$ be an n-family. Then $\mathrm{P}[\mathcal{D}]$ is closed under $\leq_{\mathrm{m}}^{\mathrm{p}}$-reductions. But $\mathrm{P}_{\mathrm{dist}}[\mathcal{D}]$ is not closed under $\leq_{\mathrm{m}}^{\mathrm{p}}$-reductions, unless $\mathrm{P}_{\mathrm{dist}}[\mathcal{D}] = \mathrm{P}[\mathcal{D}']$ for some n-family $\mathcal{D}'$.*

Not all $\mathrm{P}[\mathcal{D}]$ are closed under 1-tt reductions. To characterize the families $\mathcal{D}$ for which this closure property holds, a new type of operation on bitstrings and pools is needed:

Definition 14 (Bitflip). *For $n \geq 1$ and $i \in \{1, \dots, n\}$ define the* bitflip operation $\mathrm{flip}_i \colon \mathbb{B}^n \to \mathbb{B}^n$ *by* $\mathrm{flip}_i(b) := b[1] \cdots b[i-1]\big(1 - b[i]\big)b[i+1] \cdots b[n]$. *This operation is extended to pools and families of pools in the obvious way. An n-family $\mathcal{D}$ is called* closed under bitflip *if* $\mathrm{flip}_i(\mathcal{D}) = \mathcal{D}$ *for all* $1 \leq i \leq n$.

Theorem 15 (1-tt Reductions). *For all n-families $\mathcal{D}$ in normal form,* $\mathrm{P}[\mathcal{D}]$ *is closed under $\leq_{1\text{-tt}}^{\mathrm{p}}$-reductions iff $\mathcal{D}$ is closed under bitflip.*

If an n-family $\mathcal{D}$ is closed under bitflip then $\mathrm{P}[\mathcal{D}]$ is also closed under complement. The converse does not hold as the family $\mathcal{D} = \mathrm{SEL}_2$ shows.

3 From 2-tt to Turing Reductions

As we know from [20,16] the closure of P-sel under polynomial time Turing reductions equals P/poly. But all $\mathrm{P}[\mathcal{D}]$ are proper subclasses of P/poly. Therefore we have:

Fact 16. *For $\mathrm{SEL}_n \subseteq \mathcal{D} \neq 2\text{-}^n\mathrm{SIZE}_n$ the class $\mathrm{P}[\mathcal{D}]$ is not closed under polynomial time Turing reductions.*

In [8, Theorem 2.9] Beigel, Kummer and Stephan construct languages A and B such that $A \leq_{\mathrm{tt}}^{\mathrm{p}} B \in \mathrm{P}_{\mathrm{dist}}[\mathrm{TOP}_3]$ and $A \notin \mathrm{P}\big[(2^n - 1)\text{-}\mathrm{SIZE}_n\big]$ for all n. In terms of closure under reductions this yields:

Fact 17. *Let $\mathcal{D} \neq 2^n\text{-}\mathrm{SIZE}_n$ be in normal form. If $\mathrm{TOP}_n \subseteq \mathcal{D}$ or $\mathrm{BOTTOM}_n \subseteq \mathcal{D}$, then $\mathrm{P}[\mathcal{D}]$ is not closed under $\leq_{\mathrm{tt}}^{\mathrm{p}}$-reductions.*

On the other hand Amir, Beigel and Gasarch [3,1] and Goldsmith, Joseph and Young [11,10] showed that cheatability classes are closed under polynomial time Turing reductions.

Fact 18. *For all $n \geq 1$ the classes $\mathrm{P}[n\text{-}\mathrm{SIZE}_n]$ are closed under $\leq_{\mathrm{T}}^{\mathrm{p}}$-reductions.*

In the following we show that this cannot be extended to other classes and even that the cheatability classes are the only nontrivial classes in our context which are closed under 2-tt reductions.

To deal with k-tt reductions, we introduce for each language A a language $A_{k\text{-tt}}$ which is many-one complete for the k-tt reduction closure of A.

Definition 19. *For a language A and $k \geq 1$ define*

$$A_{k\text{-tt}} := \left\{ \langle x_1, \dots, x_k, \phi \rangle \mid \phi \colon \mathbb{B}^k \to \mathbb{B},\ \phi\bigl(\chi_A(x_1, \dots, x_k)\bigr) = 1 \right\}.$$

This definition is inspired by the study of btt-cylinders [22,9,8]. A language is a k-tt cylinder iff $A_{k\text{-tt}} \leq^{\mathrm{p}}_{\mathrm{m}} A$.

Lemma 20. *For all languages A and B we have $B \leq^{\mathrm{p}}_{k\text{-tt}} A$ iff $B \leq^{\mathrm{p}}_{\mathrm{m}} A_{k\text{-tt}}$.*

Proof. Suppose $B \leq^{\mathrm{p}}_{k\text{-tt}} A$ via M. If on input x the machine M computes queries $q_1, \dots, q_k$ and a boolean function ϕ_x such that $\phi_x\bigl(\chi_A(q_1, \dots, q_k)\bigr) = \chi_B(x)$, then $x \in B$ iff $\langle q_1, \dots, q_k, \phi_x \rangle \in A_{k\text{-tt}}$. Thus $B \leq^{\mathrm{p}}_{\mathrm{m}} A_{k\text{-tt}}$.

Suppose $B \leq^{\mathrm{p}}_{\mathrm{m}} A_{k\text{-tt}}$ via M. If on input x the reduction machine M computes $\langle q_1, \dots, q_k, \phi_x \rangle$, then asking the oracle A with queries $q_1, \dots, q_k$ and evaluating the answers with ϕ_x constitutes the k-tt reduction to A. □

As an immediate corollary we obtain that a language class $\mathcal{C}$ closed under polynomial time many-one reductions is also closed under k-tt reductions, iff for every $A \in \mathcal{C}$ we also have $A_{k\text{-tt}} \in \mathcal{C}$.

In order to prove the main results of this section we first show a rather basic fact on bounded truth-table reductions. Although this fact should be well known, we could not find it in the literature. It states that polynomial time k-tt reductions can be replaced by sequences of 2-tt reductions.

Lemma 21. *Let A and B be languages with $A \leq^{\mathrm{p}}_{k\text{-tt}} B$. Then there exist intermediate languages $C_1, \dots, C_r$ such that $A \leq^{\mathrm{p}}_{2\text{-tt}} C_1 \leq^{\mathrm{p}}_{2\text{-tt}} \cdots \leq^{\mathrm{p}}_{2\text{-tt}} C_r \leq^{\mathrm{p}}_{2\text{-tt}} B$.*

Proof. Let $\mathbb{B}^k = \{b_1, \dots, b_{2^k}\}$ where the b_i are in lexicographic order. The chain of 2-tt reductions from A to B will consist of the following two subchains

$$\begin{aligned} A &\leq^{\mathrm{p}}_{2\text{-tt}} D_{b_1} \leq^{\mathrm{p}}_{2\text{-tt}} D_{b_2} \leq^{\mathrm{p}}_{2\text{-tt}} \cdots \leq^{\mathrm{p}}_{2\text{-tt}} D_{b_{2^k}} \\ &\leq^{\mathrm{p}}_{2\text{-tt}} H_k \leq^{\mathrm{p}}_{2\text{-tt}} H_{k-1} \leq^{\mathrm{p}}_{2\text{-tt}} \cdots \leq^{\mathrm{p}}_{2\text{-tt}} H_1 \leq^{\mathrm{p}}_{2\text{-tt}} B. \end{aligned}$$

The languages H_i for $i \in \{1, \dots, k\}$ are defined as follows:

$$H_i := \left\{ \langle x_1, \dots, x_i, b \rangle \mid \chi_B(x_1, \dots, x_i) = b \in \mathbb{B}^i \right\}.$$

Clearly, $H_1 \leq^{\mathrm{p}}_{1\text{-tt}} B$ and $H_{i+1} \leq^{\mathrm{p}}_{2\text{-tt}} H_i$. Thus, the second part of the chain from A to B is correct.

The first part consists of the following languages D_b with $b \in \mathbb{B}^k$:

$$D_b := \Bigl(B_{k\text{-tt}} \cap \left\{ \langle x_1, \dots, x_k, \phi \rangle \mid \chi_B(x_1, \dots, x_k) \geq_{\mathrm{lex}} b \right\} \Bigr) \oplus H_k.$$

Note that $D_{b_1} = B_{k\text{-tt}} \oplus H_k$ and hence by Lemma 20 we have $A \leq^{\mathrm{p}}_{1\text{-tt}} D_{b_1}$. Note furthermore that we also have $D_{b_{2^k}} \leq^{\mathrm{p}}_{1\text{-tt}} H_k$.

To show $D_{b_i} \leq^{\mathrm{p}}_{2\text{-tt}} D_{b_{i+1}}$, let x be an input word. If $x = 1y$ then we must decide whether $y \in H_k$ which can trivially be done by passing on $1y$ as a query to $D_{b_{i+1}}$. If $x = 0\,\langle x_1, \dots, x_k, \phi \rangle$ we ask two queries: we ask $q_1 := 0\,\langle x_1, \dots, x_k, \phi \rangle$ and $q_2 := 1\,\langle x_1, \dots, x_k, b_i \rangle$. If the answer to the second query is "yes" we know $\chi_b(x_1, \dots, x_k) = b_i$ and output $\phi(b_i)$, ignoring the answer to the first query. If the answer to the second query is "no" we output the answer to the first query which is, indeed, correct. □

Corollary 22. *A language class $\mathcal{C}$ is closed under $\leq^{\mathrm{p}}_{\mathrm{btt}}$-reductions iff $\mathcal{C}$ is closed under $\leq^{\mathrm{p}}_{2\text{-tt}}$-reductions.*

For the formulation of our Main Theorem 27, we need the following definition:

Definition 23 (k-Cone). *For $n, k \geq 1$ and an n-family $\mathcal{D}$ in normal form an nk-pool P is a* k-cone *for $\mathcal{D}$ if for all tuples $(\phi_1, \ldots, \phi_n)$ of functions $\phi_i \colon \mathbb{B}^k \to \mathbb{B}$, the set of bitstrings*

$$\bigl\{ \phi_1\bigl(b[1,\ldots,k]\bigr)\, \phi_2\bigl(b[k+1,\ldots,2k]\bigr) \cdots \phi_n\bigl(b[(n-1)k+1,\ldots,nk]\bigr) \mid b \in P \bigr\}$$

is a pool of $\mathcal{D}$. The nk-family of all k-cones for $\mathcal{D}$ is denoted by $k\text{-cones}(\mathcal{D})$.

Theorem 24. *For an n-family $\mathcal{D}$ in normal form and $k \geq 1$, $\mathrm{P}[\mathcal{D}]$ is closed under polynomial time k-tt reductions iff $\lceil \mathcal{D} \rceil_{nk} \subseteq k\text{-cones}(\mathcal{D})$.*

Proof. Suppose $\lceil \mathcal{D} \rceil_{nk} \subseteq k\text{-cones}(\mathcal{D})$. Then every pool $D \in \lceil \mathcal{D} \rceil_{nk}$ is a k-cone for $\mathcal{D}$. To prove that $\mathrm{P}[\mathcal{D}]$ is closed under $\leq^{\mathrm{p}}_{k\text{-tt}}$-reductions, by Lemma 20 and Theorem 13 it suffices to show that for every $A \in \mathrm{P}[\mathcal{D}]$ we have $A_{k\text{-tt}} \in \mathrm{P}[\mathcal{D}]$.

Let $A \in \mathrm{P}\bigl[\lceil \mathcal{D} \rceil_{nk}\bigr]$ via M. Then $A_{k\text{-tt}} \in \mathrm{P}[\mathcal{D}]$ is witnessed by the following algorithm. On input $x_1, \ldots, x_n$ test whether the x_i are in syntactically correct form, that is test whether $x_i = \bigl\langle y^i_1, \ldots, y^i_k, \phi_i \bigr\rangle$ for some y^i_j and ϕ_i. If not, replace x_i by an x'_i of correct syntax. If we have found a pool D for this input of correct syntax, we find a pool for the original input by projecting D in the i-th component to 0, using that $\mathcal{D}$ is closed under projections. So suppose the x_i are all of the form $x_i = \langle y^i_1, \ldots, y^i_k, \phi_i \rangle$. Let M compute a pool $D' \in \lceil \mathcal{D} \rceil_{nk}$ for $y^1_1, \ldots, y^1_k, y^2_1, \ldots, y^2_k, \ldots, y^n_1, \ldots, y^n_k$. Because D' is a k-cone for $\mathcal{D}$, the pool

$$D := \bigl\{ \phi_1(b^1_1, \ldots, b^1_k)\phi_2(b^2_1, \ldots, b^2_k) \cdots \phi_n(b^n_1, \ldots, b^n_k) \mid b^1_1 \cdots b^n_k \in D' \bigr\}$$

is a pool in $\mathcal{D}$ and by definition of $A_{k\text{-tt}}$ we have $\chi_{A_{k\text{-tt}}}(x_1, \ldots, x_n) \in D$.

For the opposite direction, suppose that $\mathrm{P}[\mathcal{D}]$ is closed under $\leq^{\mathrm{p}}_{k\text{-tt}}$-reductions. We will show $\lceil \mathcal{D} \rceil_{nk} \subseteq k\text{-cones}(\mathcal{D})$ by showing $\mathrm{P}\bigl[\lceil \mathcal{D} \rceil_{nk}\bigr] \subseteq \mathrm{P}\bigl[k\text{-cones}(\mathcal{D})\bigr]$ and using Fact 6. Consider a language $A \in \mathrm{P}\bigl[\lceil \mathcal{D} \rceil_{nk}\bigr]$. We will exhibit an algorithm M_A which witnesses $A \in \mathrm{P}\bigl[k\text{-cones}(\mathcal{D})\bigr]$. Because $A \in \mathrm{P}\bigl[\lceil \mathcal{D} \rceil_{nk}\bigr]$, we have $A \in \mathrm{P}[\mathcal{D}]$; and because $\mathrm{P}[\mathcal{D}]$ is closed under k-tt reductions by Lemma 20 there is a machine M which witnesses $A_{k\text{-tt}} \in \mathrm{P}[\mathcal{D}]$.

In order to keep the rest of the proof more readable, we introduce some ad-hoc definitions and abbreviations:

$$\begin{aligned}
\varPhi &:= \bigl\{ (\phi_1, \ldots, \phi_n) \mid \phi_i \colon \mathbb{B}^k \to \mathbb{B} \text{ for } 1 \leq i \leq n \bigr\}, \\
\phi &:= (\phi_1, \ldots, \phi_n), \\
x &:= (x^1_1, \ldots, x^1_k, x^2_1, \ldots, x^2_k, \ldots, x^n_1, \ldots, x^n_k), \\
b &:= b^1_1 \cdots b^1_k\, b^2_1 \cdots b^2_k\, \cdots\, b^n_1 \cdots b^n_k.
\end{aligned}$$

Furthermore, for every input tuple $x \in \bigl(\varSigma^*\bigr)^{nk}$ and every $\phi \in \varPhi$ we write

$$x \circ \phi := \bigl(\langle x^1_1, \ldots, x^1_k, \phi_1 \rangle, \langle x^2_1, \ldots, x^2_k, \phi_2 \rangle, \ldots, \langle x^n_1, \ldots, x^n_k, \phi_n \rangle\bigr).$$

For a bitstring $b \in \mathbb{B}^{nk}$, a pool $D \subseteq \mathbb{B}^{nk}$ and some $\phi \in \Phi$ we write

$$\phi(b) := \phi_1(b_1^1 \cdots b_k^1) \cdots \phi_n(b_1^n \cdots b_k^n),$$
$$\phi(D) := \{\, \phi(b) \mid b \in D \,\}.$$

Finally, for a pool $D \subseteq \mathbb{B}^n$ and $\phi \in \Phi$ we write $\phi^{-1}(D) := \{\, b \mid \phi(b) \in D \,\}$.
We describe how algorithm M_A on input x computes a pool D_x for x:

For all $\phi \in \Phi$:
Compute $M(x \circ \phi) =: D_\phi \in \mathcal{D}$. (We then know $\chi_{A_{k\text{-tt}}}(x \circ \phi) \in D_\phi$.)
Then compute $\phi^{-1}(D_\phi)$. (It holds $\chi_A(x) \in \phi^{-1}(D_\phi)$.)
Now compute

$$D_x := \bigcap_{\phi \in \Phi} \phi^{-1}(D_\phi).$$

Then $\chi_A(x) \in D_x$, and $D_x \in k\text{-cones}(\mathcal{D})$ as $\phi(D_x) \subseteq D_\phi \in \mathcal{D}$ for all $\phi \in \Phi$. □

Lemma 25. *Let $\mathcal{D}$ be an n-family in normal form. If $\mathrm{P}[\mathcal{D}]$ is closed under polynomial time n-tt reductions, then $\mathcal{D} = m\text{-}\mathrm{SIZE}_n$ for some $m \in \mathbb{N}$.*

Proof. Suppose $\mathrm{P}[\mathcal{D}]$ is closed under n-tt reductions, where n is a fixed constant. Let m be minimal with $\mathcal{D} \subseteq m\text{-}\mathrm{SIZE}_n$. Let $D = \{b_1, \dots, b_m\} \in \mathcal{D}$ be a pool of maximal size. We have to show that for every pool $E \subseteq \mathbb{B}^n$ with $|E| = m$ already $E \in \mathcal{D}$. Suppose $E = \{e_1, \dots, e_m\}$. By Fact 7 for change of tuple-length we know that the pool $D' := \{\, (b_i)^n \mid b_i \in D \,\}$, where each bitstring consists of n copies of an original bitstring from D, is in $\lceil \mathcal{D} \rceil_{n^2}$. Now define boolean functions $\phi_1, \dots, \phi_n$ such that $\phi_i(b_j) = e_j[i]$. Then the image of D' under $(\phi_1, \dots, \phi_n)$ is E. Because D' is an n-cone for $\mathcal{D}$, it follows that $E \in \mathcal{D}$. □

Lemma 26. *Let $\mathcal{D}$ be an n-family in normal form with $\mathrm{SEL}_n \subseteq \mathcal{D}$ such that $\mathrm{P}[\mathcal{D}]$ is closed under polynomial time 2^n-tt reductions. Then $\mathcal{D} = 2^n\text{-}\mathrm{SIZE}_n$.*

Proof. Because $\mathrm{P}[\mathcal{D}]$ is closed under 2^n-tt reductions, by Theorem 24 we know that $\lceil \mathcal{D} \rceil_{n2^n} \subseteq 2^n\text{-cones}(\mathcal{D})$. It suffices to exhibit a pool $D \in \lceil \mathcal{D} \rceil_{n2^n}$ and 2^n-ary boolean functions $\phi_1, \dots, \phi_n$ such that the image of D under $(\phi_1, \dots, \phi_n)$ is $\mathbb{B}^n$. Note that because $\lceil \mathrm{SEL}_n \rceil_{n2^n} = \mathrm{SEL}_{n2^n}$ by Fact 11, we have $\mathrm{SEL}_{n2^n} \subseteq \lceil \mathcal{D} \rceil_{n2^n}$. We choose $D \subseteq \mathbb{B}^{n2^n}$ as a pool of size 2^n:

$$D := \{b_1, \dots, b_{2^n}\} \text{ with } \; b_i := \left(0^{2^n - i} 1^i\right)^n.$$

This means that each bitstring b_i consists of the concatenation of n copies of a bitstring of length 2^n. Because the bitstrings in D form an ascending chain (see Definition 10) $D \in \mathrm{SEL}_{n2^n} \subseteq \lceil \mathcal{D} \rceil_{n2^n}$. Let $\mathbb{B}^n = \{c_1, \dots, c_{2^n}\}$. Now define boolean functions ϕ_j such that $\phi_j\left(0^{2^n - i} 1^i\right) = c_i[j]$. For those b which are not of the form $0^{2^n - i} 1^i$ choose some arbitrary value for $\phi_j(b)$. The choice of ϕ_j ensures

$$(\phi_1, \dots, \phi_n)(b_i) \;=\; \phi_1\left(0^{2^n - i} 1^i\right) \cdots \phi_n\left(0^{2^n - i} 1^i\right) \;=\; c_i[1] \cdots c_i[n] \;=\; c_i.$$

This yields $(\phi_1, \dots, \phi_n)(D) = \mathbb{B}^n$. □

We sum up the preceding results in the following theorem which characterises the classes closed under 2-tt reductions as well as under Turing reductions:

Theorem 27 (Main Theorem). *For n-families $\mathcal{D}$ in normal form with $\mathcal{D} \neq 2^n\text{-SIZE}_n$ the following are equivalent:*

1. *$\mathcal{D} = m\text{-SIZE}_n$ for some $m \leq n$.*
2. *$\mathrm{P}[\mathcal{D}]$ is closed under polynomial time 2-tt reductions.*
3. *$\mathrm{P}[\mathcal{D}]$ is closed under polynomial time Turing reductions.*

Proof. The class $\mathrm{P}[m\text{-SIZE}_n]$ with $m \leq n$ is equal to $\mathrm{P}[m\text{-SIZE}_m]$ by Fact 12. The class $\mathrm{P}[m\text{-SIZE}_m]$ is closed under polynomial time Turing reductions by Fact 18. A class closed under Turing reductions is also closed under 2-tt reductions.

If for an n-family $\mathcal{D}$ in normal form $\mathrm{P}[\mathcal{D}]$ is closed under 2-tt reductions, then by Corollary 22 it is also closed under polynomial time n-tt and 2^n-tt reductions. By Lemma 25, $\mathcal{D} = m\text{-SIZE}_n$ for some m. By assumption $m \neq 2^n$. If $m > n$ we would have $\text{SEL}_n \subseteq \mathcal{D}$, which is impossible by Lemma 26. Therefore $\mathcal{D} = m\text{-SIZE}_n$ for some $m \leq n$. $\square$

4 Positive Reductions

A motivation to investigate positive reductions is the fact that NP is closed under polynomial time positive Turing reductions. In general, when dealing with classes which are not (known to be) closed under 1-tt reductions it suggests itself to look for closure under some kind of positive reduction. Regarding reductions to languages in partial information classes it can happen that a reduction type, although more powerful than some other in general, looses its extra power when the oracle is taken from $\mathrm{P}[\mathcal{D}]$ for certain $\mathcal{D}$. For example it can be shown that querying an arbitrary number of queries to a language $B \in \mathrm{P}[k\text{-SIZE}_k]$ can always be replaced by putting only $k-1$ of these queries to oracle B. Regarding positive reductions, Selman [21] showed that a polynomial time positive truth-table reduction to a p-selective language can always be replaced by a many-one reduction. It follows that $\text{P-sel} = \mathrm{P}[\text{SEL}_2]$ is closed under ptt-reductions. We extend this result to $\text{SEL}_2 \cup 2\text{-SIZE}_2$, a family in normal form strictly above SEL_2. The partial information class $\mathrm{P}[\text{SEL}_2 \cup 2\text{-SIZE}_2]$ is exactly the class of languages L, for which L and its complement $\overline{L}$ are strictly 2-membership comparable, see [17].

Theorem 28. *If $A \leq^{\mathrm{p}}_{\mathrm{ptt}} B$ and $B \in \mathrm{P}[\text{SEL}_2 \cup 2\text{-SIZE}_2]$ then $A \leq^{\mathrm{p}}_{\mathrm{m}} B$.*

Proof. First, note that $\text{SEL}_2 \cup 2\text{-SIZE}_2 = \text{SEL}_2 \cup \{01, 10\}$. In the following we will call this latter pool the xor-pool.

Let $A \leq^{\mathrm{p}}_{\mathrm{ptt}} B$ via a reduction R and let $B \in \mathrm{P}[\text{SEL}_2 \cup 2\text{-SIZE}_2]$ via a machine M. To show $A \leq^{\mathrm{p}}_{\mathrm{m}} B$, upon input x we must compute a query q such that $\chi_A(x) = \chi_B(q)$. Let $q_1, \dots, q_k$ be the polynomially many queries produced by R upon input x and let ϕ be the monotone evaluation function used by R. Note that for once k depends on the length of the input x.

We define a graph $G = (V, E)$ with coloured edges as follows. The vertices $V = \{q_1, \ldots, q_k\}$ are exactly the queries q_i. For each pair (q_i, q_j) with $q_i <_{\text{lex}} q_j$ the machine M outputs (possibly a subset of) one of the three pools $\{00, 01, 11\}$, $\{00, 10, 11\}$ and the xor-pool $\{01, 10\}$. In the first case there is a black directed edge from q_i to q_j, in the second case there is a black directed edge from q_j to q_i. If the xor-pool is output, q_i and q_j are connected by an undirected red edge.

This graph has the following two properties: First, if $q_i \in B$ then for every black edge going from q_i to a vertex q_j we also have $q_j \in B$. Second, if q_i and q_j are connected by a red edge, then $q_i \in B$ iff $q_j \notin B$.

We compute a single query $q = q_i$ such that knowing $\chi_B(q)$ yields $\chi_B(q_j)$ for all other vertices. To do so, we apply the following pruning algorithm to the graph:

Search for a red edge (q_i, q_j) plus another node q_ℓ which is connected to both q_i and q_j by black edges. If both black edges go from q_ℓ to q_i and q_j we *know* $\chi_B(q_\ell) = 0$, for being connected by a red edge exactly one of the two words q_i and q_j must be in B. Thus we can remove q_ℓ from the graph. Likewise, if both edges go from q_i and q_j to q_ℓ we know $\chi_B(q_\ell) = 1$ and we can also remove q_ℓ. Now, if the first black edge goes from q_i to q_ℓ and the second from q_ℓ to q_j, the graph property yields $\chi_B(q_i) = 1$ and $\chi_B(q_j) = 0$. Conversely, if the black edges go the other way round, we know $\chi_B(q_i) = 0$ and $\chi_B(q_j) = 1$. In either case we can remove both q_i and q_j.

For the remaining graph $G' = (E', V')$ there are two possible situations: either the graph no longer contains a red edge or it contains a red spanning tree.

If the graph contains no red edge, it is a tournament; that is for all vertices in the graph the machine R behaves like a selector. Using this selector we can compute a pool from $\text{SEL}_{|E'|}$ for the characteristic string of the vertices in the graph. To compute the special query q, we proceed as follows: for each bitstring $b \in P$, in order of increasing number of 1's, compute $\phi(b)$. If $\phi(b) = 0$ (or $\phi(b) = 1$) for all b, we do not have to query the oracle at all. Otherwise there is exactly one bitstring such that $\phi(b) = 0$, but $\phi(b') = 1$ for the next bitstring. We take the word at the position where a 0 in b changes into a 1 in b' as our query q.

If the graph has a red spanning tree, knowing the characteristic value of any vertex in the graph immediately yields the characteristic values of all other vertices. Hence, we can compute two bitstrings b_0 and b_1 such that $\chi_B(q_1, \ldots, q_k) \in \{b_0, b_1\}$. Compute $\phi(b_0)$ and $\phi(b_1)$. If these are equal, we do not need to query B at all. If they are different, there must exists a position i such that $b_0[i] = \phi(b_0)$ and $b_1[i] = \phi(b_1)$ and we can ask $q := q_i$. □

Corollary 29. *Let $\mathcal{D}$ be a family with $\text{P}[\mathcal{D}] \subseteq \text{P}[\text{SEL}_2 \cup 2\text{-SIZE}_2]$. Then $\text{P}[\mathcal{D}]$ is closed under polynomial time positive truth-table reductions.*

Proof. If $A \leq^{\text{p}}_{\text{ptt}} B \in \text{P}[\mathcal{D}]$, then $A \leq^{\text{p}}_{\text{m}} B$ by Theorem 28. But $\text{P}[\mathcal{D}]$ is closed under many-one reductions by Theorem 13. Therefore $A \in \text{P}[\mathcal{D}]$. □

We now exhibit some families $\mathcal{D}$ for which $\text{P}[\mathcal{D}]$ is not even closed under 2-ptt reductions. To do this, we proceed similarly as in the previous section where we showed that certain $\mathrm{P}[\mathcal{D}]$ are not closed under k-tt reductions. For every A we introduce a typical language $A_{k\text{-ptt}}$ which is k-ptt reducible to A. We then define positive k-cones and with these we characterize the families $\mathrm{P}[\mathcal{D}]$ closed under k-ptt reductions. Finally we present some special families that do not comply with this characterization.

Definition 30. *For a language A and $k \geq 1$ define*

$$A_{k\text{-ptt}} := \bigl\{ \langle x_1, \dots, x_k, \phi \rangle \mid \phi\colon \mathbb{B}^k \to \mathbb{B} \text{ is monotone, } \phi\bigl(\chi_A(x_1, \dots, x_k)\bigr) = 1 \bigr\}.$$

The following lemma is the analogue of Lemma 20 for positive reductions. A proof is almost identical to the proof of that lemma and is therefore omitted.

Lemma 31. *For all languages A and B we have $B \leq^{\mathrm{p}}_{k\text{-ptt}} A$ iff $B \leq^{\mathrm{p}}_{\mathrm{m}} A_{k\text{-ptt}}$.*

Definition 32 (Positive *k*-Cone). *For $k \geq 1$ and an n-family $\mathcal{D}$ in normal form, an nk-pool P is a* positive k-cone for $\mathcal{D}$ *if for all tuples $(\phi_1, \dots, \phi_n)$ of k-ary monotone boolean functions ϕ_i the set of bitstrings*

$$\bigl\{ \phi_1\bigl(b[1, \dots, k]\bigr)\, \phi_2\bigl(b[k+1, \dots, 2k]\bigr) \cdots \phi_n\bigl(b[(n-1)k+1, \dots, nk]\bigr) \mid b \in P \bigr\}$$

is a pool of $\mathcal{D}$. By k-pcones$(\mathcal{D})$ we denote the nk-family of all positive k-cones for $\mathcal{D}$.

Theorem 33. *Let $\mathcal{D}$ be an n-family. Then $\mathrm{P}[\mathcal{D}]$ is closed under $\leq^{\mathrm{p}}_{k\text{-ptt}}$-reduction iff $\lceil \mathcal{D} \rceil_{nk} \subseteq k$-pcones$(\mathcal{D})$.*

The omitted proof is essentially the same as for Theorem 24 for k-tt reductions.

Theorem 34. *Let $\mathcal{D}$ be a 2-family with $\mathrm{BOTTOM}_2 \subseteq \mathcal{D}$. If $\mathrm{P}[\mathcal{D}]$ is closed under 2-ptt reductions, then $\mathcal{D} = 4\text{-}\mathrm{SIZE}_2$.*

Proof. The pool $\{00, 01, 10\}$ is in $\mathcal{D}$ and therefore with the definition from Fact 7 for change of tuple-length, it is easy to check that $D = \{0000, 0010, 0100, 1001\}$ is in $\lceil \mathcal{D} \rceil_4$. Choose $\phi_1 = \phi_2$ as the 2-ary boolean or-function. Then we have $(\phi_1, \phi_2)(D) = \{00, 01, 10, 11\} = \mathbb{B}^2$. Because $\mathcal{D}$ is closed under 2-ptt reductions $(\phi_1, \phi_2)(D)$ has to be in $\mathcal{D}$. It follows that $\mathcal{D} = 4\text{-}\mathrm{SIZE}_2$. ☐

If a class $\mathcal{C}$ is closed under 2-ptt reductions then co-$\mathcal{C}$ also is closed under 2-ptt reductions. Therefore Theorem 34 also holds if $\mathrm{TOP}_2 \subseteq \mathcal{D}$.

We have now characterised the families $\mathcal{D}$ for tuple-length $n = 2$ for which $\mathrm{P}[\mathcal{D}]$ is closed under positive truth-table reductions:

Theorem 35. *For 2-families $\mathcal{D}$ in normal form with $\mathcal{D} \neq 4\text{-}\mathrm{SIZE}_2$ the following are equivalent:*

1. *$\mathrm{P}[\mathcal{D}]$ is closed under polynomial time ptt reductions.*
2. *$\mathrm{P}[\mathcal{D}]$ is closed under polynomial time 2-ptt reductions.*
3. *$\mathcal{D} \subseteq \mathrm{SEL}_2 \cup 2\text{-}\mathrm{SIZE}_2$.*
4. *$\mathrm{BOTTOM}_2 \nsubseteq \mathcal{D}$ and $\mathrm{TOP}_2 \nsubseteq \mathcal{D}$.*

While in the previous section on general k-tt and Turing reductions a complete picture was presented, in the case of positive reductions there remains work to be done. The above result should be extended to general tuple-lengths n and to positive Turing reductions.

References

1. A. Amir, R. Beigel, and W. Gasarch. Some connections between bounded query classes and non-uniform complexity. In *Proc. 5th Structure in Complexity Theory*, 1990.
2. A. Amir and W. Gasarch. Polynomial terse sets. *Inf. and Computation*, 77, 1988.
3. R. Beigel. *Query-Limited Reducibilities*. PhD thesis, Stanford University, 1987.
4. R. Beigel. Bounded queries to SAT and the boolean hierarchy. *Theoretical Comput. Sci.*, 84(2), 1991.
5. R. Beigel, W. Gasarch, and E. Kinber. Frequency computation and bounded queries. In *Proc. 10th Structure in Complexity Theory*, 1995.
6. R. Beigel, M. Kummer, and F. Stephan. Quantifying the amount of verboseness. In *Proc. Logical Found. of Comput. Sci.*, volume 620 of *LNCS*. Springer, 1992.
7. R. Beigel, M. Kummer, and F. Stephan. Approximable sets. In *Proc. 9th Structure in Complexity Theory*, 1994.
8. R. Beigel, M. Kummer, and F. Stephan. Approximable sets. *Inf. and Computation*, 120(2), 1995.
9. L. Berman and J. Hartmanis. On isomorphisms and density of NP and other complete sets. *SIAM J. Comput.*, 6(2):305–322, 1977.
10. J. Goldsmith, D. Joseph, and P. Young. Using self-reducibilities to characterize polynomial time. *Inf. and Computation*, 104(2):288–308, 1993.
11. J. Goldsmith, D. A. Joseph, and P. Young. Using self-reducibilities to characterize polynomial time. Technical Report CS-TR-88-749, University of Wisconsin, Madison, 1988.
12. L. Hemaspaandra, A. Hoene, and M. Ogihara. Reducibility classes of p-selective sets. *Theoretical Comput. Sci.*, 155:447–457, 1996.
13. L. A. Hemaspaandra, Z. Jiang, J. Rothe, and O. Watanabe. Polynomial-time multi-selectivity. *J. of Universal Comput. Sci.*, 3(3), 1997.
14. M. Hinrichs and G. Wechsung. Time bounded frequency computations. In *Proc. 12th Conf. on Computational Complexity*, 1997.
15. A. Hoene and A. Nickelsen. Counting, selecting, and sorting by query-bounded machines. In *Proc. STACS 93*, volume 665 of *LNCS*. Springer, 1993.
16. K.-I. Ko. On self-reducibility and weak p-selectivity. *J. Comput. Syst. Sci.*, 26:209–221, 1983.
17. J. Köbler. On the structure of low sets. In *Proc. 10th Structure in Complexity Theory*, pages 246–261. IEEE Computer Society Press, 1995.
18. R. E. Ladner, N. A. Lynch, and A. L. Selman. A comparison of polynomial time reducibilities. *Theoretical Comput. Sci.*, 1(2):103–123, Dec. 1975.
19. A. Nickelsen. On polynomially $\mathcal{D}$-verbose sets. In *Proc. STACS 97*, volume 1200 of *LNCS*, pages 307–318. Springer, 1997.
20. A. Selman. P-selective sets, tally languages and the behaviour of polynomial time reducibilities on NP. *Math. Systems Theory*, 13:55–65, 1979.
21. A. L. Selman. Reductions on NP and p-selective sets. *Theoretical Comput. Sci.*, 19:287–304, 1982.
22. P. Young. On semi-cylinders, splinters, and bounded-truth-table reducibility. *Trans. of the AMS*, 115:329–339, 1965.