Polynomial time bounded truth–table reducibilities to padded sets

Vladimír Glasnák

Abstract. We study bounded truth-table reducibilities to sets of small information content called padded (a set is in the class f-*PAD* of all f-padded sets, if it is a subset of $\{x10^{f(|x|)-|x|-1};\ x \in \{0,1\}^*\}$). This is a continuation of the research of reducibilities to sparse and tally sets that were studied in many previous papers (for a good survey see [HOW1]). We show necessary and sufficient conditions to collapse and separate classes of bounded truth-table reducibilities to padded sets. We prove that depending on two properties of a function f measuring "holes" in its image, one of the following three possibilities happen:

$$R_{\mathrm{m}}(f\text{-}PAD) \subsetneq R_{1\text{-tt}}(f\text{-}PAD) \subsetneq \cdots \subsetneq R_{\mathrm{btt}}(f\text{-}PAD), \quad \text{or}$$
$$R_{\mathrm{m}}(f\text{-}PAD) = R_{1\text{-tt}}(f\text{-}PAD) \subsetneq \cdots \subsetneq R_{\mathrm{btt}}(f\text{-}PAD), \quad \text{or}$$
$$R_{\mathrm{m}}(f\text{-}PAD) = R_{\mathrm{btt}}(f\text{-}PAD).$$

Keywords: computational complexity, sparse set, padded set, reducibility
Classification: 68Q15

1. Introduction

The first polynomial time reducibilities were introduced by Cook [C1] (Turing reducibility) and Karp (many-one reducibility) to delimit computationally hard problems (*NP*-hard problems). The consequent research of reducibilities plays an important role in structural complexity. For example, if one proves that two kinds of reducibilities differ on *NP*, then $NP \neq P$ because all polynomial time reducibilities coincide on *P* (there is one trivial exception for many-one reducibility). This is a motivation for comparison of various kinds of reducibilities that was started by Ladner, Lynch and Selman [LLS1] who proved that many reducibilities differ on *DEXT*. A bit different approach was initiated by Watanabe [W1] who compared the completeness notions for various reducibilities on the class *DEXT*. This work was followed by Buhrman, Homer, Spaan and Torenvliet ([BHT1], [BST1], summarized in [B]). They compared various completeness notions on nondeterministic exponential time complexity classes *NEXT*, *EXPTIME* and *NEXPTIME*.

The author gratefully acknowledges the support of the Grant Agency of the Czech Republic under Grant 201/98/1451.

This paper was written while the author was studying at Department of Theoretical Computer Science, Faculty of Mathematics and Physics, Charles University, Prague.

The problem whether any two reducibilities coincide on *NP* is open. The first approach to this problem mentioned above investigates reducibilities to "more powerful" classes (exponential time). Another approach is to study reducibilities to sets of small information content (or "less powerful sets"). A typical class of such sets is *SPARSE*, the class of all sparse sets (see e.g. [BDG1]). An intensive research has been devoted to the class $R_{\mathrm{T}}(\mathit{SPARSE})$ of all sets that are polynomial time Turing reducible to sparse sets (formal definitions of this and other classes from Introduction are given later). Karp and Lipton [KL1] proved that if $\mathit{NP} \subseteq R_{\mathrm{T}}(\mathit{SPARSE})$, then $\mathit{PH} = \Sigma_2^P$, Mahaney [M1] proved that $\mathit{NP} = P$ follows from the existence of a sparse *NP*-hard problem for many-one reducibility. Ogiwara and Watanabe [OW1] proved the same result for bounded truth-table reducibility. Arvind et al. [AHH+1] and Ranjan and Rohatgi [RR1] resolved the case for conjunctive truth-table reducibility.

The previous results show that in certain cases weaker reducibilities afford stronger results. Book and Ko begins investigation of subclasses of $R_{\mathrm{T}}(\mathit{SPARSE})$ — classes of sets reducible to sparse sets under weaker polynomial time reducibilities. This research led to many results about large hierarchy of classes (see, e.g. [B1], [BHT1], [BK1], [K2], [S1], [S2], some of the results are mentioned below).

From the beginning, reducibilities to sparse sets were compared with reducibilities to tally sets because tally sets are much simpler. In spite of this, there are results where sparse and tally sets have the same properties. Important two examples of this concern classes of sets conjunctively and disjunctively truth-table reducible to sparse and tally sets. Although Ko [K2] proved that $R_{\mathrm{dtt}}(\mathit{SPARSE}) \neq R_{\mathrm{dtt}}(\mathit{TALLY})$, Buhrman, Longpré and Spaan [BLS] proved that $R_{\mathrm{ctt}}(\mathit{SPARSE}) = R_{\mathrm{ctt}}(\mathit{TALLY})$.

Instead of a tally set one can consider a set padded with a long "tail" of zeros, we call such sets padded:

Definition 1. Let f be a nondecreasing time constructible function with $f(n) > n$. A set A is f-padded, if $A \subseteq \{x10^{f(|x|)-|x|-1}; x \in \Sigma^*\}$. The class of all f-padded sets is denoted by f-*PAD*.

It is easy to prove that every 2^n-padded set is m-reducible to a tally set and every tally set is m-reducible to a 2^n-padded set (see [G1]). Hence from the view of polynomial time reducibilities, 2^n-padded sets are equivalent to tally sets.

Special cases of padded sets and their relation to f-sparse sets (sets with $O(f(n))$ words of length n) were studied in [H1], [HIS1], [A1] and [G1]. Hartmanis [H1] proved that $\mathit{NP}-P$ contains a sparse set iff it contains a tally (or, equivalently, 2^n-padded) set. Glasnák [G1] proved that $\mathit{NP} - P$ contains an $n^{O(\log n)}$-sparse set iff it contains a $2^{O(\sqrt{n})}$-padded sets. Surprisingly, Allender [A1] constructed a relativisation such that an analogous result for log-sparse and 2^{2^n}-padded sets does not hold i.e. there is an oracle A such that $\mathit{NP}^A - P^A$ contains a log-sparse set but no 2^{2^n}-padded set. The result is based on the fact that sets with census function $f(n) \in \Omega(n)$ have a variant of Kolmogorov complexity KU depending on f (the greater f the greater KU) while for sets with census function $f(n) \in o(n)$

is KU approximately the same independently on f. Hence the result shows a difference between log-sparse and sparse sets.

The aim of this paper is to classify functions f by the behaviour of bounded reducibilities to f-padded sets. We show that for some functions f, bounded truth-table reducibilities to f-padded sets collapse while for other functions these reducibilities create a hierarchy similar to the boolean hierarchy. We give a full characterization of the behaviour of bounded reducibilities to f-padded sets depending on f. As a consequence we obtain that 2^n-padded sets have different properties than 2^{2^n}-padded sets.

For any unbounded nondecreasing function f, define $f^I(n) = \min\{m;\ f(m) \geq n\}$ (because f^I is a generalization of Inverse function). Define $f_{\mathrm{UP}}(n) = f(f^I(n))$ and $f_{2\mathrm{UP}}(n) = f(2f^I(n))$. Note that $f_{\mathrm{UP}}(n)$ is the smallest value in the image of f greater than or equal to n. Moreover, for every n, $f_{2\mathrm{UP}}(n) \geq f_{\mathrm{UP}}(n)$.

We prove the following results:

Theorem 2. *Let f be a nondecreasing time constructible function with $f(n) > n$ such that $f_{\mathrm{UP}}(n) > p(n)$ infinitely often for every polynomial p. Then*

$$R_{\mathrm{m}}(f\text{-}PAD) \subsetneq R_{1\text{-tt}}(f\text{-}PAD) \quad \textit{and}$$
$$\textit{for every } k > 0,\ R_{k\text{-tt}}(f\text{-}PAD) \subsetneq R_{k+1\text{-tt}}(f\text{-}PAD) \subsetneq R_{\mathrm{btt}}(f\text{-}PAD).$$

Moreover, there are sets $A, B \in R_{\mathrm{m}}(f\text{-}PAD)$ such that $A \cup B \notin R_{\mathrm{btt}}(f\text{-}PAD)$ hence $R_{\mathrm{btt}}(f\text{-}PAD)$ is not closed under union.

Theorem 3. *Let f be a nondecreasing time constructible function with $f(n) > n$ such that $f_{\mathrm{UP}}(n) \leq p(n)$ for some polynomial p and $f_{2\mathrm{UP}}(n) > q(n)$ infinitely often for every polynomial q. Then $R_{\mathrm{m}}(f\text{-}PAD) = R_{1\text{-tt}}(f\text{-}PAD)$, $R_{\mathrm{btt}}(f\text{-}PAD)$ is the boolean closure of $R_{\mathrm{m}}(f\text{-}PAD)$ and for every $k > 0$,*

$$R_{k\text{-tt}}(f\text{-}PAD) \subsetneq R_{k+1\text{-tt}}(f\text{-}PAD) \subsetneq R_{\mathrm{btt}}(f\text{-}PAD).$$

Theorem 4. *Let f be a nondecreasing time constructible function with $f(n) > n$ such that $f_{2\mathrm{UP}}(n) \leq p(n)$ for some polynomial p. Then for every $k > 0$,*

$$R_{\mathrm{m}}(f\text{-}PAD) = R_{k\text{-tt}}(f\text{-}PAD) = R_{\mathrm{btt}}(f\text{-}PAD)$$

and $R_{\mathrm{m}}(f\text{-}PAD)$ is closed under boolean operations.

Note that the theorems are "complementary" — for every nondecreasing function f, either $f_{\mathrm{UP}}(n)$ is infinitely often greater than every polynomial or it is less than some polynomial. Similarly, either $f_{2\mathrm{UP}}(n)$ is infinitely often greater than every polynomial or it is less than some polynomial.

Another possible view on these results is the following. If we do not consider the "tail" of an f-padded string (because it carries very small amount of information), then we obtain reducibilities with bounded length of query: $R_{\mathrm{red}}(f\text{-}PAD) = \{A;$

there exists a $\leq^P_{\rm red}$-reducibility from A such that every queried string y has the length bounded by $f^I(p(n)) - 1$ where n is the length of the input and p is some polynomial}. Hence our results compare classes of sets which are reducible to some set by bounded truth table reducibilities such that every queried string has the length bounded by $f^I(p(n)) - 1$.

The last section investigates functions. We delimit areas of functions that satisfy the assumptions of the individual theorems and show some examples of such functions. Studying a polynomial bound on $f_{\rm UP}$, we prove that both cases ($f_{\rm UP}$ is or is not polynomially bounded) occur for appropriate functions f between polynomials and double exponential. Similarly, $f_{\rm 2UP}$ may be polynomially bounded or unbounded for f between polynomials and $c_1^{n^{c_2}}$ for some constants c_1, c_2.

2. Preliminaries

Let f be a function on natural numbers. We use the "big O" notation: $O(f) = \{g;$ there exists some constant c such that $g(n) \leq cf(n)$ for almost every $n\}$. Our computation model is a multi-tape Turing machine with the alphabet $\Sigma = \{0, 1\}$. We distinguish transducers that compute mappings on Σ^* and acceptors that compute characteristic functions of subsets of Σ^*. For a transducer F and a string x, let $F(x)$ denote the output of F on x. We say that a function f is computable if there is a transducer F such that $F(1^n) = 1^{f(n)}$. If, in addition, F works in time $O(f(n))$ we say that f is time constructible.

The length of a string x is denoted by $|x|$.

If n is a natural number, then bin(n) is its binary expansion.

The integer part of a real number δ is denoted by $\lfloor\delta\rfloor$.

For a set $A \subseteq \Sigma^*$, define $\overline{A} = \Sigma^* - A$, $A_{\leq n} = \{x \in A; |x| \leq n\}$ and $A_{=n} = \{x \in A; |x| = n\}$.

The characteristic function of a set A is denoted by χ_A.

To code tuples we apply a standard selfdelimiting code from Kolmogorov complexity (see e.g. [LV1]). For a string $x = d_1 \dots d_n$ where d_i is the i-th letter of x, define $l(x) = 1^n 0 d_1 d_2 \dots d_n$. The code of an n-tuple of strings $x_1, \dots, x_n$ is defined by

$$\langle x_1, \dots, x_n\rangle = l(\mathrm{bin}(|x_1|))x_1 l(\mathrm{bin}(|x_2|))x_2 \dots l(\mathrm{bin}(|x_{n-1}|))x_{n-1}x_n.$$

Now $|\langle x_1, \dots, x_n\rangle| = 3(n-1) + |x_n| + \sum_{i=1}^{n-1}(2\lfloor\log|x_i|\rfloor + |x_i|)$. Note that an n-tuple can be unambiguously decoded only if n is known.

Let k be a natural number and let f be a nondecreasing time constructible function with $f(n) > n$. For strings $x_1, \dots x_k$, define

$$[x_1, \dots, x_k]_f = \langle x_1, \dots, x_k\rangle 0 1^{f(|\langle x_1,\dots,x_k\rangle|) - |\langle x_1,\dots,x_k\rangle| - 1}.$$

Note that this definition does not require f to be time constructible or nondecreasing but we use it only for such functions. On the other hand, the condition

$f(n) > n$ is necessary. Therefore we define a useful shortcut. A function is called padding if it is a nondecreasing time constructible function with $f(n) > n$. Note that this definition differs from the common definition of padding function that characterizes certain hard problems ([BH1]).

Let f be a padding function. For a set $A \subseteq \Sigma^*$, define $[A]_f = \{[x]_f;\, x \in A\}$.

Define the inverse function of $[\ldots]_f$ by

$$[y]_{-f} = \begin{cases} x & \text{if } y = [x]_f; \\ \text{undefined} & \text{otherwise.} \end{cases}$$

2.1 Polynomial time reducibilities.

A set A is polynomial time many-one (shortly m) reducible to a set B (we write $A \leq_{\mathrm{m}}^P B$), if there is a transducer F working in polynomial time such that for every x, $F(x)$ is defined and $x \in A$ iff $F(x) \in B$.

Let $k > 0$ be a natural number. A k-truth-table condition is a $k+1$-tuple $\langle \alpha, y_1, \ldots, y_k \rangle$, where $y_1, \ldots, y_k$ are strings and α is a boolean function on k variables given by its table.

A set A is k-truth-table (shortly k-tt) reducible to a set B (we write $A \leq_{k\text{-tt}}^P B$) if there exists a transducer F working in polynomial time which on every input x computes a k-tt condition such that $x \in A$ iff $F(x) = \langle \alpha, y_1, \ldots, y_k \rangle$ and $\alpha(\chi_B(y_1), \ldots, \chi_B(y_k)) = 1$.

Note that if A, B are arbitrary sets, then $A \leq_{1\text{-tt}}^P B$ iff there exists a transducer F working in polynomial time such that $x \in A$ iff ($F(x) = \langle 1, y \rangle$ and $y \in B$ or $F(x) = \langle 0, y \rangle$ and $y \notin B$).

A set A is bounded truth-table (btt) reducible to a set B (we write $A \leq_{\mathrm{btt}}^P B$), if $A \leq_{k\text{-tt}}^P B$ for some k.

A set A is Turing (T) reducible to a set B (we write $A \leq_{\mathrm{T}}^P B$), if there exists some polynomial time oracle Turing machine M such that $x \in A$ iff M^B accepts x.

Let red be any of the above defined reducibilities and C be a class of sets. Define

$$R_{\mathrm{red}}(C) = \{A;\ \text{there exists a set } B \in C \text{ such that } \ A \leq_{\mathrm{red}}^P B\}.$$

Remark 5. Let f be a padding function and red be an arbitrary reducibility of the above defined ones. Note that every set in $R_{\mathrm{red}}(f\text{-}PAD)$ is red-reducible to a set $[B]_f$ for $B \neq \Sigma^*$. Therefore if a set A is in $R_{\mathrm{red}}(f\text{-}PAD)$, then A is red-reducible to a set B via reducibility which queries the set B only for f-padded strings. Precisely,

(1) $A \in R_{\mathrm{m}}(f\text{-}PAD)$ iff A is m-reducible to a set $[B]_f$ via F such that for every string x, the string $F(x)$ is f-padded;

(2) $A \in R_{k\text{-tt}}(f\text{-}PAD)$ iff A is k-tt-reducible to a set $[B]_f$ via F such that for every string x, we have $F(x) = \langle \alpha, y_1, \ldots, y_k \rangle$ for some boolean function α and f-padded strings $y_1, \ldots, y_k$;

(3) $A \in R_{\mathrm{btt}}(f\text{-}PAD)$ iff there is a k such that A is k-tt-reducible to a set $[B]_f$ via F such that for every string x, we have $F(x) = \langle \alpha, y_1, \dots, y_k \rangle$ for some boolean function α and f-padded strings $y_1, \dots, y_k$;

(4) $A \in R_{\mathrm{T}}(f\text{-}PAD)$ iff A is T-reducible to a set $[B]_f$ via M such that for every string x, the set of queries to the oracle contains only f-padded strings.

Proposition 6. *The class $R_{\mathrm{m}}(f\text{-}PAD)$ is closed under complement for every padding function f.*

Proof: Let A, B be sets such that $A \leq_{\mathrm{m}}^{P} [B]_f$ via F. By Remark 5, assume that the image of F is a subset of f-PAD. Now F reduces $\overline{A}$ to $[\overline{B}]_f$. □

The closure under complement is the main difference between $R_{\mathrm{m}}(TALLY)$ (Book and Ko [BK1] proved that this class is closed under complement) and $R_{\mathrm{m}}(SPARSE)$ (Book and Ko [BK1] proved that this class is not closed under complement).

Köbler [K3] proved the following characterization of the class of sets btt-reducible to a set A.

Theorem 7 ([K3])**.** *Let A be a set. The class $\{B;\, B \leq_{\mathrm{btt}}^{P} A\}$ is exactly the boolean closure of $\{B;\, B \leq_{\mathrm{m}}^{P} A\}$.*

Let us compare this result with Theorem 2, where it is stated that $R_{\mathrm{btt}}(f\text{-}PAD)$ is not closed under boolean operations. In Theorem 2 we have $R_{\mathrm{btt}}(f\text{-}PAD)$ where f-PAD contains many sets while in Theorem 7 we have $R_{\mathrm{btt}}(\{A\})$ for one set A. This causes the difference between the conclusions.

In order to diagonalize we need sequences of transducers and acceptors satisfying certain properties. They are described in the following proposition.

Proposition 8. *There is a sequence $\{F_i\}_{i\in\mathbb{N}}$ of transducers and a sequence $\{M_i\}_{i\in\mathbb{N}}$ of acceptors with oracle such that*

(a) *for every i, F_i and M_i works in time $n^i + i$;*

(b) *for every transducer F working in polynomial time there are infinitely many i such that F_i computes the same mapping;*

(c) *for every oracle acceptor M working in polynomial time there are infinitely many i such that for every oracle A and every string x, M^A accepts x iff M_i^A accepts x.*

The sequences can be easily constructed (see a basic literature on structural complexity, e.g. [BDG1]).

3. Reducibilities to padded sets

This section contains the proofs of main results. First, we prove technical lemmas necessary later. Next, we prove theorems that together imply the main results.

3.1 Technical lemmas.

The first lemma states the properties of f^I. Observe that if f is an increasing function, then f^I coincides with f^{-1} whenever it is defined. Hence f^I is a kind of generalization of inverse function.

Lemma 9. *Let $f : \mathbb{N} \to \mathbb{N}$ be a nondecreasing unbounded function. Then f^I is a nondecreasing unbounded function and*

$$f^I(f(n)) \leq n < f^I(f(n)+1), \quad n \leq f_{\mathrm{UP}}(n) \quad \text{for all } n$$
$$\text{and } \ f(f^I(n)-1) < n \ \text{ for all } n \text{ with } \ f^I(n) > 0.$$

Proof: All of the inequalities follow just from the definition of f^I. □

Lemma 10. *Let f be a padding function and n be a natural number. Then the number of f-padded strings of the length at most n is at most*

$$2^{f^I(n+1)} - 1.$$

Proof: Let z be a string with $|[z]_f| \leq n$. Then $f(|z|) \leq n$. By Lemma 9, $|z| < f^I(f(|z|)+1) \leq f^I(n+1)$. Thus $|z| \leq f^I(n+1) - 1$. The number of the strings with this property is $2^{f^I(n+1)} - 1$. □

The following simple lemma shows that f^I of every padding function f is computable in polynomial time.

Lemma 11. *Let f be a padding function. Then there exists a Turing machine working in time $O(n^2)$ computing $f^I(n)$.*

Proof: On an input 1^n, we search for the smallest i such that $f(i) \geq n$. Since $f(n) > n$ this algorithm works in $O(n^2)$. □

3.2 A polynomial bound on f_{UP}.

Here, we investigate functions f according to whether they satisfy or do not satisfy the property $f_{\mathrm{UP}}(n) \leq p(n)$ for a polynomial p. Note that f_{UP} is the smallest number in the image of f that is greater than or equal to n. Hence $f_{\mathrm{UP}}(n) \leq p(n)$ means that "holes" in the image of f are not too big — they are polynomially bounded.

Proposition 12. *Let f be a padding function such that $f_{\mathrm{UP}}(n) \leq p(n)$ for some nondecreasing polynomial p and all n. Then for any fixed m, there is some nondecreasing polynomial q such that*

$$f(f^I(n)+m) \leq q(n).$$

Proof: The statement holds for $m = 0$. Assume that it holds for $m - 1$. Then

$$\begin{array}{rcccll}
 & & f(f^I(n)+m-1) & \leq & q(n) & \Longrightarrow \\
 & & f(f^I(n)+m-1)+1 & \leq & q(n)+1 & \Longrightarrow \text{ by Lemma 9} \\
f^I(n)+m-1 & < & f^I(f(f^I(n)+m-1)+1) & \leq & f^I(q(n)+1) & \Longrightarrow \\
f^I(n)+m & \leq & f^I(q(n)+1) & & & \Longrightarrow \\
f(f^I(n)+m) & \leq & f(f^I(q(n)+1)) & \leq & p(q(n)+1). &
\end{array}$$

□

Proposition 12 says that a nonzero additive constant m does not break a polynomial bound on $f(f^I(n)+m)$. It follows that if f is a padding function with $f_{\mathrm{UP}}(n) \leq p(n)$ for some polynomial p then for every constant a there exists a polynomial q such that for every f-padded string $[x]_f$ and every string z with $|z| \leq a$, we have $|[zx]_f| \leq q(|[x]_f|)$. Therefore, we can add arbitrary information of a constant length (z) to x and a new f-padded string ($[zx]_f$) is of a polynomially bounded length. This fact is applied in the following theorem.

Theorem 13. *Let f be a padding function with $f_{\mathrm{UP}}(n) \leq p(n)$ for some polynomial p and all n. Then the following statements hold:*

(1) $R_{\mathrm{m}}(f\text{-}PAD) = R_{1\text{-tt}}(f\text{-}PAD)$.

(2) *For every $k > 0$, if $A, B \in R_{k\text{-tt}}(f\text{-}PAD)$, then $A \cup B \in R_{2k\text{-tt}}(f\text{-}PAD)$.*

Proof: Without loss of generality, let p be a nondecreasing polynomial with $f_{\mathrm{UP}}(n) \leq p(n)$.

Statement (1). Here, the additional information has the length 1 and it carries a boolean function of one variable from 1-tt reducibility (it may be either identity or negation).

Let $A \leq^P_{1\text{-tt}} [B]_f$ via some transducer F working in polynomial time (say $p_2(n)$ for some nondecreasing polynomial p_2) such that for every x, $F(x) = \langle y, [z]_f\rangle$, where $y \in \Sigma$, $z \in \Sigma^*$ and $x \in A$ iff either ($y = 1$ and $[z]_f \in [B]_f$) or ($y = 0$ and $[z]_f \notin [B]_f$) (the transducer exists by Remark 5).

Define a mapping $u : \Sigma^* \to \Sigma^*$ by $u(x) = [yz]_f$ where $F(x) = \langle y, [z]_f\rangle$.

Note that $|u(x)| \leq f_{\mathrm{UP}}(p_2(|x|)+1)$ thus $|u(x)| \leq q(|x|)$ for some polynomial q depending on p and p_2. Since f is time constructible, u is computable in polynomial time. In addition, $x \in A$ iff $u(x) \in \{[1x]_f;\, x \in B\} \cup \{[0x]_f;\, x \notin B\}$. Hence $A \in R_{\mathrm{m}}(f\text{-}PAD)$.

Statement (2). Here, additional information is used to differ the elements of C from the elements of D as follows.

Let $A \leq^P_{k\text{-tt}} [C]_f$ via F and $B \leq^P_{k\text{-tt}} [D]_f$ via G such that F and G query the oracle only for f-padded strings (by Remark 5). Let

$$E = 1C \cup 0D = \{1x;\, x \in C\} \cup \{0x;\, x \in D\}.$$

Define a $2k$-tt reducibility from $A \cup B$ to $[E]_f$.

For an input x if $F(x)=\langle\alpha_1, [y_1]_f, \dots, [y_k]_f\rangle$ and $G(x)=\langle\alpha_2, [y_{k+1}]_f, \dots, [y_{2k}]_f\rangle$ then output

$$\langle\alpha, [1y_1]_f, \dots, [1y_k]_f, [0y_{k+1}]_f, \dots, [0y_{2k}]_f\rangle,$$

where $\alpha(x_1, \dots, x_{2k}) = \alpha_1(x_1, \dots, x_k) \vee \alpha_2(x_{k+1}, \dots, x_{2k})$. For $1 \le i \le k$, the length of $[1y_i]_f$ is polynomially bounded similarly to the proof of the first statement. Therefore the output can be computed in polynomial time. Hence $A \cup B$ is $2k$-tt-reducible to E. □

Lemma 14. *Let f be a padding function such that $f_{\mathrm{UP}}(n) > p(n)$ infinitely often for every polynomial p. Then there exists an infinite sequence $n_0, n_1, n_2, \dots$ such that for every $i > 0$,*

$$n_i > f^I(n_i) = f^I(n_i^i + i + 1) \text{ and } f(f^I(n_i) - 1) > f(f^I(n_{i-1}) - 1).$$

Proof: We give a construction of such a sequence. Define $n_0 = 1$. Let $i > 0$. Given a sequence $n_0, \dots, n_{i-1}$ satisfying the conditions, we construct the next member of the sequence. Since for every a, $f_{\mathrm{UP}}(n) > n^a + a + 1$ for infinitely many n, there exists a number n_i such that $f_{\mathrm{UP}}(n_i) > n_i^i+i+1$ and $f(f^I(n_i)-1) > f(f^I(n_{i-1}) - 1)$. By Lemma 9,

$$f^I(n_i) \ge f^I(f(f^I(n_i))) \ge f^I(n_i^i + i + 1).$$

Since f^I is nondecreasing, $f^I(n_i) = f^I(n_i^i + i + 1)$. Besides, since $f(n) > n$, $n_i > f^I(n_i)$. □

Theorem 15. *Let f be a padding function with $f_{\mathrm{UP}}(n) > p(n)$ infinitely often for every polynomial p. Then the following statements hold:*

(1) *$R_{\mathrm{m}}(f\text{-}PAD) \subsetneq R_{1\text{-tt}}(f\text{-}PAD)$.*
(2) *There are sets $A, B \in R_{\mathrm{m}}(f\text{-}PAD)$ such that $A \cup B \notin R_{\mathrm{T}}(f\text{-}PAD)$.*

Proof: For f-padded sets C, D, define

$$L_1(C) = \{1xy;\ f^I(|1xy|) \ge 1 \text{ and } [x]_f \in C \text{ and } |x| = f^I(|1xy|) - 1\}$$
$$L_0(D) = \{0xy;\ f^I(|0xy|) \ge 1 \text{ and } [x]_f \in D \text{ and } |x| = f^I(|0xy|) - 1\}.$$

Note that $L_1(C) \le_{\mathrm{m}}^P C$, $L_0(D) \le_{\mathrm{m}}^P D$ and $L(C) = L_1(C) \cup L_0(\overline{C}) \le_{1\text{-tt}}^P C$.

Statement (1). We construct an f-padded set B such that $L(B) \notin R_{\mathrm{m}}(f\text{-}PAD)$. Let $\{n_i\}$ be the sequence from Lemma 14. The set B is constructed in stages.

Stage 0. Let $B = \emptyset$.

Stage $i > 0$. By Lemma 14, $n_i > f^I(n_i) = f^I(n_i^i + i + 1)$ and $f(f^I(n_i) - 1) > f(f^I(n_{i-1}) - 1)$.

Let $\{F_i\}$ be the sequence from Proposition 8. We diagonalize against F_i as a candidate for a reduction of $L(B)$ to an f-padded set on the strings of the length

n_i in $L(B)$. In this stage, we insert into B only strings of the length $f(f^I(n_i)-1)$ and n_i is chosen such that this insertion does not change $L(B)$ strings of length n_j for $0 < j < i$.

If there exists a string x of the length n_i such that $F_i(x)$ is not f-padded, then go to the next stage (recall Remark 5).

Now, assume that for every string x of the length n_i, $F_i(x)$ is f-padded. By our assumptions on F_i, the length of $F_i(x)$ is at most $n_i^i + i$. By Lemma 10,

$$|\{F_i(x);\ |x| = n_i\}| \leq 2^{f^I(n_i^i+i+1)} - 1 = 2^{f^I(n_i)} - 1.$$

Therefore there is some z such that the set $C = F_i^{-1}(z) \cap \Sigma^{n_i}$ has cardinality at least

$$|C| \geq \frac{2^{n_i}}{2^{f^I(n_i)} - 1} > 2^{n_i - f^I(n_i)}.$$

Since there are $2^{n_i - f^I(n_i)}$ strings of the length n_i with the same prefix of the length $f^I(n_i)$, the set C contains at least two strings with different prefixes of this length, say $z_1x_1y_1$ and $z_2x_2y_2$ where $z_1x_1 \neq z_2x_2$, $|z_1x_1| = |z_2x_2| = f^I(n_i)$ and $|z_1| = |z_2| = 1$. If $z_1 = z_2$ then insert $[x_1]_f$ into B. If $z_1 \neq z_2$ then keep B unaltered. In both cases, $z_1x_1y_1 \in L(B)$ iff $z_2x_2y_2 \in \overline{L(B)}$. Since $F_i(z_1x_1y_1) = F_i(z_2x_2y_2)$, F_i cannot reduce $L(B)$ to any f-padded set.

Statement (2). We claim that for any number n with $f^I(n) \geq 1$ and for all subsets C, C', D, D' of $\Sigma^{f^I(n)-1}$,

$$(L_1([C]_f) \cup L_0([D]_f)) \cap \Sigma^n = (L_1([C']_f) \cup L_0([D']_f)) \cap \Sigma^n \text{ iff } C = C' \text{ and } D = D'.$$

The implication from right to left is clear. To prove the opposite implication, it is sufficient to prove that the left equality implies $C \subseteq C'$ because the rest of the proof follows from symmetry. If $x \in C$, then for every $y \in \Sigma^{n-f^I(n)}$, $1xy \in L_1([C]_f)$. Since no string beginning with 1 is in $L_0([D']_f)$ we have $1xy \in L_1([C']_f)$ for every $y \in \Sigma^{n-f^I(n)}$. Hence $x \in C'$ and thus $C \subseteq C'$. The claim is proved.

Let $\{M_i\}$ be a sequence from Proposition 8.

We diagonalize against all T-reducibilities and construct sets C, D such that $L_1([C]_f) \cup L_0([D]_f)$ is not T-reducible to any f-padded set.

Let $\{n_i\}$ be the sequence from Lemma 14.

The construction of C and D is made in stages.

Stage 0. Let $C = D = \emptyset$.

Stage $i > 0$. We diagonalize against T-reducibility via M_i. By Lemma 14, $n_i > f^I(n_i) = f^I(n_i^i + i + 1)$ and $f(f^I(n_i) - 1) > f(f^I(n_{i-1}) - 1)$.

In this stage we insert into sets C, D some strings of the length $f^I(n_i) - 1$ such that $L_1([C]_f) \cup L_0([D]_f)$ will not be Turing reducible to any f-padded set via M_i.

Note that n_i is chosen such that it does not change any computation considered in the previous stages.

Consider all strings of the length n_i. If there is a string of the length n_i such that the computation of M_i on x (with an oracle) queries the oracle for a not f-padded string, then recall Remark 5 and go to the next stage.

Otherwise, since M_i works in time $n_i^i + i$, the computation of M_i with an oracle $[A]_f$ on an input of the length n_i is unambiguously determined by the set $([A]_f)_{\leq n_i^i + i}$. By Lemma 10, there exist at most $2^{2^{f^I(n_i^i+i+1)}-1} = 2^{2^{f^I(n_i)-1}}$ f-padded sets containing strings of the length at most $n_i^i + i$. Since M_i and $([A]_f)_{\leq n_i^i+i}$ unambiguously determine the set of accepted strings of the length n_i, there exists at most $2^{2^{f^I(n_i)-1}}$ subsets L of Σ^{n_i} with $L = \Sigma^{n_i} \cap \{x;\ M_i^{[A]_f}$ accepts $x\}$ for some set A.

There are $2^{2^{f^I(n_i)-1}}$ subsets of $\Sigma^{f^I(n_i)-1}$, therefore there are

$$\left(2^{2^{f^I(n_i)-1}}\right)^2 = 2^{2^{f^I(n_i)}}$$

different pairs of sets (C', D') such that $C', D' \subseteq \Sigma^{f^I(n_i)-1}$. Thus there are sets C', D' such that for all sets A,

$$(L_1([C']_f) \cup L_0([D']_f)) \cap \Sigma^n \neq \{x;\ M_i^{[A]_f} \text{ accepts } x\}.$$

Now let $C = C \cup C'$, $D = D \cup D'$. Note that n_i is chosen such that

$$(L_1([C]_f) \cup L_0([D]_f)) \cap \Sigma^{n_i} = (L_1([C']_f) \cup L_0([D']_f)) \cap \Sigma^{n_i}.$$

Therefore the diagonalization on M_i succeeds. □

If $f_{\mathrm{UP}}(n) > p(n)$ infinitely often for every polynomial p, then it is possible to diagonalize. Therefore the possibility to add an information of a constant length to every f-padded string without more than polynomial increase is necessary and sufficient condition for $R_{\mathrm{m}}(f\text{-}PAD) = R_{1\text{-tt}}(f\text{-}PAD)$ and for closure of $R_{\mathrm{btt}}(f\text{-}PAD)$ under union.

3.3 A polynomial bound on $f_{2\mathrm{UP}}$.

Now we turn attention to the second condition — $f_{2\mathrm{UP}}(n) \leq p(n)$ for some polynomial p. Similarly to the first condition, this is a measure of "holes" in the image of f but more powerful than the first one and harder to imagine.

Proposition 16. *Let f be a padding function such that, for some real number $\epsilon > 0$ and some nondecreasing polynomial p,*

$$f(\lfloor (1+\epsilon) f^I(n) \rfloor) \leq p(n).$$

Then for every real number $\delta \geq 1$, there is some nondecreasing polynomial q such that

$$f(\lfloor \delta f^I(n) \rfloor) \leq q(n).$$

Proof: The statement holds for every number less or equal to $1+\epsilon$. Assume that it holds for $\sqrt{\delta}$. Then

$$\begin{array}{rcccl} & & f(\lfloor \delta^{\frac{1}{2}} f^I(n) \rfloor) & \leq & q(n) & \Longrightarrow \\ \lfloor \delta^{\frac{1}{2}} f^I(n) \rfloor & < & f^I(f(\lfloor \delta^{\frac{1}{2}} f^I(n) \rfloor)+1) & \leq & f^I(q(n)+1) & \Longrightarrow \\ \lfloor \delta f^I(n) \rfloor & \leq & \lfloor \delta^{\frac{1}{2}}(\lfloor \delta^{\frac{1}{2}} f^I(n) \rfloor + 1) \rfloor & \leq & \lfloor \delta^{\frac{1}{2}} f^I(q(n)+1) \rfloor & \Longrightarrow \\ f(\lfloor \delta f^I(n) \rfloor) & \leq & f(\lfloor \delta^{\frac{1}{2}} f^I(q(n)+1) \rfloor) & \leq & q(q(n)+1). & \end{array}$$

Hence it also holds for all δ. □

Similarly to Proposition 12, now the constant factor δ does not break polynomial bound on $f(\delta f^I(n))$. In particular, if f is a padding function with $f_{2\mathrm{UP}}(n) \leq p(n)$ for some polynomial p then for every constant a there exists a polynomial q such that for every f-padded string $[x]_f$ and every string z with $|z| \leq a|x|$ we have $|[z]_f| \leq q(|[x]_f|)$. Therefore, instead of x we can use a constantly longer information (z) and a new f-padded string ($[z]_f$) is of a polynomially bounded length. This is much greater information than in the first case, hence the results are more powerful than that ones for the first condition.

Theorem 17. *Let f be a padding function such that $f_{2\mathrm{UP}}(n) \leq p(n)$ for some polynomial p. Then the class $R_{\mathrm{m}}(f\text{-}PAD)$ is closed under union.*

Proof: Let $A \leq_{\mathrm{m}}^{P} [C]_f$ via F and $B \leq_{\mathrm{m}}^{P} [D]_f$ via G such that for every string x, both values $F(x)$ and $G(x)$ are f-padded (by Remark 5). Define

$$E = \{[x,y]_f;\ x \in C \ \text{ or } \ y \in D\}.$$

Now, define a mapping $u : \Sigma^* \to \Sigma^*$ by

$$u(x) = [[F(x)]_{-f}, [G(x)]_{-f}]_f.$$

Now, for every x, if $x \in A \cup B$, then either $[F(x)]_{-f} \in C$ or $[G(x)]_{-f} \in D$ thus $u(x) \in E$. Similarly, if $x \notin A \cup B$, then $u(x) \notin E$. Hence u reduces $A \cup B$ to E and it suffices to prove that $u(x)$ can be computed from x in polynomial time. There exists some nondecreasing polynomial p_1 such that $|F(x)| < p_1(|x|) - 2$ and $|G(x)| < p_1(|x|) - 2$ for every x. By Lemma 10 and Proposition 16,

$$\begin{aligned} |u(x)| \leq |[[F(x)]_{-f}, [G(x)]_{-f}]_f| \leq \\ f(|[F(x)]_{-f}| + |[G(x)]_{-f}| + 2\lfloor \log(|[F(x)]_{-f}|) \rfloor + 3) \leq \\ f(3 + 2f^I(p_1(|x|) - 2) + 2\lfloor \log f^I(p_1(|x|)) \rfloor) \leq f(3f^I(p_1(|x|))) \leq q(|x|) \end{aligned}$$

for some polynomial q. Hence the length of $u(x)$ is polynomially bounded. Since f is time constructible, $u(x)$ is computable in polynomial time. □

Lemma 18. *Let f be a padding function such that for every polynomial p, $f_{2\mathrm{UP}}(n) > p(n)$. Then for every natural number $a > 0$, there is an infinite sequence $n_0, n_1, n_2, \ldots$ such that*

$$\lfloor \tfrac{1}{2a} f^I(n_0) \rfloor \geq \tfrac{2^a}{a} + \tfrac{a+1}{a} \lfloor \log a \rfloor + 2\tfrac{a+1}{a}$$

and for every $i > 0$,

$$\begin{gathered} n_i \geq (a+1) f^I(n_i), \\ \lfloor (1 + \tfrac{1}{2a}) f^I(n_i) \rfloor \geq f^I(n_i^i + i + 1) \ \text{ and} \\ f(f^I(n_i) - 1) > f(f^I(n_{i-1}) - 1). \end{gathered}$$

Proof: If there is an $\epsilon' > 0$ and a polynomial p_1 such that $f(\lfloor (1+\epsilon') f^I(n) \rfloor) \leq p_1(n)$, then, by Proposition 16, $f_{2\mathrm{UP}}(n) \leq q(n)$ for some polynomial q. Therefore, for every $\epsilon' > 0$ and every polynomial p_1, $f(\lfloor (1+\epsilon') f^I(n) \rfloor) > p_1(n)$ infinitely often.

Let $a > 0$ be a fixed natural number. By Lemma 9, f^I is unbounded hence there exists n_0 satisfying the first condition. Given a sequence $n_0, \ldots, n_{i-1}$ satisfying all the conditions, we show how to choose n_i.

Let $m \geq (a+1)^2 + a + 1$ be a number such that

$$f(\lfloor (1 + \tfrac{1}{2a}) f^I(m) \rfloor) > m^{2i} + i + 1 \quad \text{and} \quad f(f^I(m) - 1) > f(f^I(n_{i-1}) - 1).$$

Let $n_i = (a+1)^2 m + a + 1$. Then the third inequality holds because f^I is nondecreasing. Moreover,

$$\begin{aligned} (1) \quad & f(\lfloor (1 + \tfrac{1}{2a}) f^I(n_i) \rfloor) \geq f(\lfloor (1 + \tfrac{1}{2a}) f^I(m) \rfloor) > \\ & \qquad (m \cdot m)^i + i + 1 \geq (((a+1)^2 + a + 1)m)^i + i + 1 \geq n_i^i + i + 1. \end{aligned}$$

Hence, by Lemma 9,

$$\lfloor (1 + \tfrac{1}{2a}) f^I(n_i) \rfloor \geq f^I(f(\lfloor (1 + \tfrac{1}{2a}) f^I(n_i) \rfloor)) \geq f^I(n_i^i + i + 1).$$

Now, assume that $n_i < (a+1) f^I(n_i)$. Then, by (1) and Lemma 9,

$$\begin{aligned} n_i > f(f^I(n_i) - 1) \geq f(\tfrac{n_i}{a+1} - 1) = f((a+1)m) \geq f((a+1) f^I(m)) \geq \\ f(\lfloor (1 + \tfrac{1}{2a}) f^I(m) \rfloor) \geq n_i^i + i + 1. \end{aligned}$$

This is a contradiction. Thus $n_i \geq (a+1) f^I(n_i)$ and all the conditions are satisfied. □

Theorem 19. *Let f be a padding function such that $f_{2\mathrm{UP}}(n) > p(n)$ infinitely often for every polynomial p. Then the following statements hold:*

(1) *There are sets $A, B \in R_{\mathrm{m}}(f\text{-}PAD)$ such that $A \cup B \notin R_{\mathrm{m}}(f\text{-}PAD)$.*

(2) *For every $k > 0$, $R_{k\text{-tt}}(f\text{-}PAD) \subsetneq R_{k+1\text{-tt}}(f\text{-}PAD)$.*

Note that Theorem 15 has stronger assumption and statement (2) than statement (1) of Theorem 19.

Proof: Statement (1). For f-padded sets C, D, define

$$L_1(C) = \{xy;\ f^I(|xy|) \geq 1 \text{ and } [x]_f \in C \text{ and } |x| = f^I(|xy|) - 1\}$$

$$L_2(D) = \{yx;\ f^I(|yx|) \geq 1 \text{ and } [x]_f \in D \text{ and } |x| = f^I(|yx|) - 1\}.$$

A reducibility u from $L_1(C)$ to C works as follows. On an input z compute $n = f^I(|z|) - 1$ (by Lemma 11). For $n = -1$, let $u(z)$ be a string which is not f-padded and for $n \geq 0$, let x be the first n letters of z and let $u(z) = [x]_f$. Since $|[x]_f| = f(f^I(|xy|) - 1) < |xy|$, $u(z)$ is computable in polynomial time. Thus $L_1(C) \leq^P_{\mathrm{m}} C$. Similarly, $L_2(D) \leq^P_{\mathrm{m}} D$.

We construct sets C, D such that $L_1(C) \cup L_2(D)$ is not m-reducible to any f-padded set. Let $\{F_i\}$ be the sequence of transducers from Proposition 8.

Let $\{n_i\}$ be the sequence from Lemma 18 for $a = 1$. The construction is made in stages.

Stage 0. Let $C = D = \emptyset$.

Stage $i > 1$. We diagonalize against F_i on strings of length n_i in $L_1(C) \cup L_2(D)$.

It follows from Lemma 18 that $\lfloor \frac{1}{2} f^I(n_0) \rfloor \geq 6$. Since f^I is nondecreasing, $f^I(n_i) \geq 12 > 4$ so that $2f^I(n_i) - 2 \geq \frac{3}{2} f^I(n_i)$. Using this and Lemma 18 again, we obtain that $n_i \geq 2f^I(n_i) - 2 \geq \lfloor \frac{3}{2} f^I(n_i) \rfloor \geq f^I(n_i^i + i + 1)$ and $f(f^I(n_i) - 1) > f(f^I(n_{i-1}) - 1)$.

Since we insert in this stage into C only strings of the length $f(f^I(n_i) - 1)$, the last condition guarantees that no computation considered in the previous stages will be changed.

Run F_i on all inputs of the length n_i. Let the outputs be $z_1, \ldots, z_{2^{n_i}}$. Note that $|z_j| \leq n_i^i + i$ for every j, $1 \leq j \leq 2^{n_i}$.

If there is some j such that z_j is not f-padded, then go to the next stage (by Remark 5).

If for every j, z_j is f-padded, then the cardinality of the set $\{z_1, \ldots, z_{2^{n_i}}\}$ is at most $2^{f^I(n_i^i+i+1)} - 1$ by Lemma 10. Therefore there exists some z such that

$$|F_i^{-1}(z) \cap \Sigma^{n_i}| \geq \frac{2^{n_i}}{2^{f^I(n_i^i+i+1)} - 1}.$$

Recall that n_i is chosen such that $2f^I(n_i) - 2 \geq f^I(n_i^i + i + 1)$. This implies that

$$|F_i^{-1}(z) \cap \Sigma^{n_i}| \geq \frac{2^{n_i}}{2^{f^I(n_i^i+i+1)} - 1} > 2^{n_i - 2f^I(n_i) + 2}.$$

Hence there are strings x_1, x_2 with $F_i(x_1) = F_i(x_2) = z$ such that their prefixes of the length $f^I(n_i) - 1$ or their suffixes of the same length differ. If $x_1 = y_1y_2y_3$ and $x_2 = y_4y_5y_6$ such that $y_1 \neq y_4$ and $|y_1| = |y_3| = |y_4| = |y_6| = f^I(n_i) - 1$ then insert $[y_1]_f$ into C. Now, $x_1 \in L_1(C)$ and $x_2 \notin L_1(C) \cup L_2(D)$. If $y_1 = y_4$ insert $[y_3]_f$ into D. Then $x_1 \in L_2(D)$ and $x_2 \notin L_1(C) \cup L_2(D)$. Since F_i computes the same value for x_1 and x_2 it is not a reducibility from $L_1(C) \cup L_2(D)$.

Statement (2). Let $k > 0$ be a natural number. We prove that $R_{k\text{-tt}}(f\text{-}PAD) \subsetneq R_{k+1\text{-tt}}(f\text{-}PAD)$. For any f-padded set C, define

$$\begin{aligned}L_3(C) = \{x_0x_1 \ldots x_ky;\ & f^I(|x_0x_1 \ldots x_ky|) \geq |\operatorname{bin}(k)| + 1 \ \text{ and} \\ & |x_i| = f^I(|x_0x_1 \ldots x_ky|) - |\operatorname{bin}(k)| - 1 \ \text{ and} \\ & \text{for all } i \text{ with } 0 \leq i \leq k,\ [0^{|\operatorname{bin}(k)| - |\operatorname{bin}(i)|}\operatorname{bin}(i)x_i]_f \in C\}.\end{aligned}$$

A $k+1$-tt-reducibility from $L_3(C)$ to C works as follows. On an input z, find $n = f^I(|z|) - 1$. This can be done in polynomial time by Lemma 11. If $n - |\operatorname{bin}(k)| < 0$ or $(1 + k)(n - |\operatorname{bin}(k)|) > |z|$, let $\alpha \equiv 0$ and $y_0 = \cdots = y_k = 0$ be strings. Otherwise, let x_0 be the first $n - |\operatorname{bin}(k)|$ letters of z, x_1 be the next $n - |\operatorname{bin}(k)|$ letters etc. Let α be conjunction of $k + 1$ variables and for all i between 0 and k, let $y_i = [0^{|\operatorname{bin}(k)| - |\operatorname{bin}(i)|}\operatorname{bin}(i)x_i]_f$.

Let i be a number between 0 and k. Since

$$\begin{aligned}|[0^{|\operatorname{bin}(k)| - |\operatorname{bin}(i)|}\operatorname{bin}(i)x_i]_f| = f(|\operatorname{bin}(k)| + f^I(|x_0 \ldots x_ky|) - |\operatorname{bin}(k)| - 1) \leq \\ f(f^I(|x_0 \ldots x_ky|) - 1) < |x_0 \ldots x_ky|,\end{aligned}$$

$\langle \alpha, y_0, \ldots, y_k \rangle$ is computable in polynomial time. Now, we have $x_0 \ldots x_ky \in L_3(C)$ iff

$$\alpha(\chi_C(y_0), \ldots, \chi_C(y_k)) = 1.$$

Thus $L_3(C) \leq^P_{k+1\text{-tt}} C$.

We construct a set C such that $L_3(C)$ is not k-tt-reducible to any f-padded set.

We diagonalize against polynomial time transducers represented by a sequence $\{F_i\}$ from Proposition 8.

Let $n_0, n_1, \ldots$ be a sequence from Lemma 18 for $a = k$.

Stage 0. Let $C = \emptyset$.

Stage $i > 0$. Since we insert in this stage into C only strings of the length $f(f^I(n_i) - 1)$, it guarantees that no computation considered in the previous stages will be changed.

Run F_i on all inputs of the length n_i. If there is some x of the length n_i such that $F_i(x) = \langle \alpha, z_1, \ldots, z_k \rangle$, where for some j, z_j is not f-padded, then go to the next stage (by Remark 5).

In the opposite case, by Lemma 10, there are at most $2^{f^I(n_i^i+i+1)}-1$ f-padded strings possibly computed by F_i. There are at most 2^{2^k} boolean functions of k variables. Therefore there exists some x of the length n_i such that

$$|F_i^{-1}(F_i(x))\cap\Sigma^{n_i}|\geq\frac{2^{n_i}}{2^{2^k}(2^{f^I(n_i^i+i+1)}-1)^k}>2^{n_i-2^k-kf^I(n_i^i+i+1)}.$$

Recall that n_i is chosen such that

$$f^I(n_i^i+i+1)+\frac{2^k}{k}+\frac{k+1}{k}\lfloor\log k\rfloor+2\frac{k+1}{k}\leq$$
$$\lfloor(1+\tfrac{1}{2k})f^I(n_i)\rfloor+\lfloor\tfrac{1}{2k}f^I(n_i)\rfloor\leq\tfrac{k+1}{k}f^I(n_i).$$

Therefore

$$kf^I(n_i^i+i+1)+2^k+(k+1)|\operatorname{bin}(k)|+k+1\leq(k+1)f^I(n_i).$$

This implies that

$$|F_i^{-1}(F_i(x))\cap\Sigma^{n_i}|>2^{n_i-(k+1)f^I(n_i)+(k+1)|\operatorname{bin}(k)|+(k+1)}.$$

Hence there are strings $y,y'\in\Sigma^{n_i}$ with $F_i(x)=F_i(y)=F_i(y')$ such that their prefixes of the length

$$(k+1)f^I(n_i)-(k+1)|\operatorname{bin}(k)|-(k+1)$$

differ. Let $y=x_0x_1\dots x_kz$ and $y'=x'_0x'_1\dots x'_kz'$, where $|x_i|=|x'_i|=f^I(n_i)-1-|\operatorname{bin}(k)|$ for $0\leq i\leq k$. For every $0\leq i\leq k$, insert $0^{|\operatorname{bin}(k)|-|\operatorname{bin}(i)|}\operatorname{bin}(i)x_i$ into C. Now $y\in L_3(C)$ and $y'\notin L_3(C)$ hence F_i is not a k-tt reducibility from $L_3(C)$ to any f-padded set. □

3.4 Proofs of the main results.

Proof of Theorem 2: By Theorem 15, $R_{\mathrm{m}}(f\text{-}PAD)\subsetneq R_{1\text{-tt}}(f\text{-}PAD)$ and there exist sets A,B such that $A\cup B\notin R_{\mathrm{T}}(f\text{-}PAD)\supseteq R_{\mathrm{btt}}(f\text{-}PAD)$. Since $f_{2\mathrm{UP}}(n)\geq f_{\mathrm{UP}}(n)$ we can apply Theorem 19 and obtain $R_{k\text{-tt}}(f\text{-}PAD)\subsetneq R_{k+1\text{-tt}}(f\text{-}PAD)$. □

Proof of Theorem 3: By Theorem 19 we have $R_{k\text{-tt}}(f\text{-}PAD)\subsetneq R_{k+1\text{-tt}}(f\text{-}PAD)$. By Theorem 13, $R_{\mathrm{m}}(f\text{-}PAD)=R_{1\text{-tt}}(f\text{-}PAD)$ and $R_{\mathrm{btt}}(f\text{-}PAD)$ is closed under union. Together with Theorem 7 we obtain that $R_{\mathrm{btt}}(f\text{-}PAD)$ is the boolean closure of $R_{\mathrm{m}}(f\text{-}PAD)$. □

Proof of Theorem 4: By Proposition 6 and Theorem 17, $R_{\mathrm{m}}(f\text{-}PAD)$ is closed under boolean operations hence $R_{\mathrm{m}}(f\text{-}PAD)=R_{\mathrm{btt}}(f\text{-}PAD)$. □

4. Examples

The aim of this section is to investigate properties of functions f_{UP} and $f_{2\mathrm{UP}}$. We prove that all cases from Theorem 2, 3 and 4 occur. Further we give necessary conditions for satisfying $f_{\mathrm{UP}}(n) \leq p(n)$ or $f_{2\mathrm{UP}}(n) \leq p(n)$ for some polynomial p.

Proposition 20. *Let f be a padding function with $f(n) \leq p(n)$ for some polynomial p and every n. Then $f_{2\mathrm{UP}}(n) \leq q(n)$ for some polynomial q.*

Proof: If $m < n \leq f(m) \leq p(m)$, then $f_{2\mathrm{UP}}(n) \leq f(2m) \leq p(2m) \leq p(2n)$. □

It is not surprising that functions f which are polynomially bounded have also $f_{2\mathrm{UP}}(n)$ polynomially bounded. But a padding function bounded by a polynomial is out of interest because $R_{\mathrm{m}}(f\text{-}PAD)$ consist of all subsets of Σ^*.

Proposition 21. *Let f be a padding function with $f_{\mathrm{UP}}(n) \leq p(n)$ for some polynomial p and every n. Then $f(n) \leq c_1^{c_2^n}$ for some constants c_1, c_2.*

Proof: Let $p'(n) = p(n+1)$. Note that $f(n+1) \leq f_{\mathrm{UP}}(f(n)+1) \leq p'(f(n))$ by Lemma 9. Applying this procedure n times, we obtain

$$f(n) \leq \underbrace{p'(p' \ldots p'}_{n}(f(0)) \ldots) \leq c_1^{c_2^n}$$

for some constants c_1, c_2. □

Hence every function f with f_{UP} polynomially bounded lies between polynomials and double exponential. The following proposition shows that the bounds given by Propositions 20 and 21 are almost optimal.

Proposition 22. *There exists a padding function f such that $f(n) \leq n^{\log n}$ almost everywhere and $f_{\mathrm{UP}}(n) > p(n)$ infinitely often for every polynomial p.*

On the other hand, the function $g(n) = 2^{2^n}$ satisfies $g_{\mathrm{UP}}(n) \leq n^2 + 2$.

Proof: Define

$$f(n) = \begin{cases} 4 & \text{if } n = 0; \\ n^{\log n} & \text{if } f(n-1) = n; \\ f(n-1) & \text{otherwise.} \end{cases}$$

We claim that for all n and for all m

$$2^{2^{2^n}} \leq m < 2^{2^{2^{n+1}}} \text{ implies } f(m) = 2^{2^{2^{n+1}}}.$$

To prove the claim note that $f(0) = \cdots = f(3) = 2^2$. Moreover, if $f(2^{2^{2^n}} - 1) = 2^{2^{2^n}}$ then $f(2^{2^{2^n}}) = 2^{2^{2^{n+1}}}$. If $f(2^{2^{2^n}}) = 2^{2^{2^{n+1}}}$, then for every m between $2^{2^{2^n}}$ and $2^{2^{2^{n+1}}}$, $f(m) = 2^{2^{2^{n+1}}}$. The claim is proved. It implies that f is time constructible.

Next, for every m,

$$f_{\mathrm{UP}}\left(2^{2^{2^m}}+1\right)=2^{2^{2^{m+1}}}.$$

Therefore, for infinitely many n,

$$f_{\mathrm{UP}}(n)=2^{2^{2^{1+\log\log\log(n-1)}}}=(n-1)^{\log(n-1)}.$$

This is greater than any polynomial in n.

To verify the property of g note that $g_{\mathrm{UP}}(0)=g_{\mathrm{UP}}(1)=g_{\mathrm{UP}}(2)=2$ and for every n,m such that

$$m<2^{2^{n+1}}-2^{2^n},\ \text{we have } g_{\mathrm{UP}}(2^{2^n}+m+1)=2^{2^{n+1}}\leq(2^{2^n}+m+1)^2.$$

Hence $g_{\mathrm{UP}}(n)\leq n^2+2$. □

Corollary 23. *For all $k>1$,*

$$R_{\mathrm{m}}(2^{2^n}\text{-}PAD)=R_{1\text{-tt}}(2^{2^n}\text{-}PAD)\subsetneq R_{k\text{-tt}}(2^{2^n}\text{-}PAD)\subsetneq$$
$$R_{k+1\text{-tt}}(2^{2^n}\text{-}PAD)\subsetneq R_{\mathrm{btt}}(2^{2^n}\text{-}PAD).$$

Hence there is a significant difference between tally sets and 2^{2^n}-padded sets (or "double tally" sets).

Proposition 24. *Let f be a padding function with $f_{2\mathrm{UP}}(n)\leq p(n)$ for some polynomial p and every n. Then $f(n)\leq c_1^{n^{c_2}}$ for some constants c_1,c_2.*

Proof: Let $p'(n)=p(n+1)$. Note that $f(2n)\leq f_{2\mathrm{UP}}(f(n)+1)\leq p'(f(n))$. Hence

$$f(n)\leq f(2^{\lfloor\log n\rfloor+1})\leq\underbrace{p'(p'\dots p'}_{\lfloor\log n\rfloor+1}(f(1))\dots)\leq c_1^{c_2^{\lfloor\log n\rfloor}}\leq c_1^{n^{\log c_2}}$$

for some constants c_1,c_2. □

Hence every function f with $f_{2\mathrm{UP}}$ polynomially bounded lies between polynomials and $c_1^{n^{c_2}}$. The following proposition shows that the bound given by Propositions 20 and 24 is almost optimal.

Proposition 25. *There exists a padding function f such that for all n,*

$$f(n)\leq n^{1+\log n}$$

and $f_{2\mathrm{UP}}(n)>p(n)$ infinitely often for every polynomial p and $f_{\mathrm{UP}}(n)\leq 2n$.

On the other hand,

$$g(n) = 2^{n^2}$$

satisfies $g_{2\mathrm{UP}}(n) \leq n^{16} + 1$.

Proof: Define a sequence as follows:

$$m_0 = 2; \quad m_{i+1} = 2^{\lfloor \log m_i \rfloor^2}(m_i + 1).$$

Now define f by

$$f(n) = \begin{cases} 1 & \text{if } n = 0 \\ 2 & \text{if } n = 1 \\ 2^k(m_i + 1) & \text{if } n = m_i + k \text{ where } 0 \leq k \leq \lfloor \log m_i \rfloor^2 \\ m_{i+1} & \text{if } n = m_i + k \text{ where } \lfloor \log m_i \rfloor^2 < k < m_{i+1} - m_i. \end{cases}$$

It is easy to see that f is time constructible and nondecreasing.

Moreover $f(n) \leq 2^{\lfloor \log n \rfloor^2} n \leq n^{1+\log n}$ and $f_{\mathrm{UP}}(n) \leq 2n$ because $f(n+1) \leq 2f(n)$. On the other hand,

$$f_{2\mathrm{UP}}(m_i + 1) = f(2m_i) = f(m_i + \lfloor \log m_i \rfloor^2) = 2^{\lfloor \log m_i \rfloor^2}(m_i + 1).$$

This is greater than any polynomial in $m_i + 1$.

To verify the property of g note that $g_{2\mathrm{UP}}(0) = g_{2\mathrm{UP}}(1) = 1$ and for every n, m such that $m < 2^{(n+1)^2} - 2^{n^2}$,

$$g_{2\mathrm{UP}}(2^{n^2}+m+1) = g(2(n+1)) = 2^{4(n+1)^2} = 2^{4n^2+8n+4} \leq 2^{16n^2} \leq (2^{n^2}+m+1)^{16}.$$

Hence $g_{2\mathrm{UP}}(n) \leq n^{16} + 1$. □

References

[A1] Allender E., *Limitations of the upward separation technique*, Math. Systems Theory **24** (1991), 53–67.

[AHH+1] Arvind V., Han Y., Hemachandra L., Köbler J., Lozano A., Mundhenk M., Ogiwara M., Schöning U., Silvestri R., Thierauf T., *Reductions to sets of low information content*, in Proceedings of the 19th International Colloquium on Automata, Languages and Programming, Springer-Verlag Lecture Notes in Computer Science, vol. 623, 1992, pp. 162–173.

[BDG1] Balcázar J. L., Díaz J. and Gabarró J., *Structural complexity I*, volume 11 of EATCS Monographs on Theoretical Computer Science, Springer Verlag, Berlin, 1988.

[B1] Buhrman H., *Resource Bounded Reductions*, PhD Thesis, Universiteit van Amsterdam, Amsterdam, 1993.

[BH1] Berman L., Hartmanis J., *On isomorphism and density of NP and other complete sets*, SIAM J. Comput. **6** (1977), 305–327.

[BHT1] Buhrman H., Homer S., Torenvliet L., *Completeness for nondeterministic complexity classes*, Math. Systems Theory **24** (1991), 179–200.

[BK1] Book R., Ko K., *On sets truth-table reducible to sparse sets*, SIAM J. Comput. **17** (1988), 903–919.

[BST1] Buhrman H., Spaan E., Torenvliet L., *Bounded reductions*, in K. Ambos-Spies, S. Homer and U. Schöning, editors, Complexity Theory, Cambridge University Press, 1993, pp. 83–99.

[C1] Cook S.A., *The complexity of theorem proving procedures*, Proc. 3rd Annual Symposium on Theory of Computing, 1971, pp. 151–158.

[G1] Glasnák V., *Sparse sets and collapse of complexity classes*, submitted for publication.

[H1] Hartmanis J., *On sparse sets in NP−P*, Inform. Process. Lett. **16** (1983), 55–60.

[HIS1] Hartmanis J., Immerman N., Sewelson V., *Sparse sets in NP−P: EXPTIME versus NEXPTIME*, Inform. and Control **65** (1985), 158–181.

[HOW1] Hemachandra L., Ogiwara M., Watanabe O., *How hard are sparse sets?*, in Proc. Structure in Complexity Theory seventh annual conference, pp. 222–238; IEEE Computer Society Press, 1992.

[KL1] Karp R.M., Lipton R.J., *Some connections between uniform and nonuniform complexity classes*, Proc. 12th ACM Symposium on Theory of Computing, 1980, pp. 302–309.

[K1] Karp R. M., *Reducibility among combinatorial problems*, in Miller, Thatcher (ed.), Complexity of Computer Computations, Plenum Press, New York, 1972, pp. 302–309.

[K2] Ko K., *Distinguishing conjunctive and disjunctive reducibilities by sparse sets*, Inform. and Comput. **81** (1989), 62–87.

[K3] Köbler J., *Unterschung verschiedener polynomieller Reduktionsklassen von NP*, Diplom. thesis, Institut für Informatik, Univ. Stuttgart, 1985.

[LLS1] Ladner R., Lynch N., Selman A., *A comparison of polynomial-time reducibilities*, Theoret. Comput. Sci. **1** (1973), 103–123.

[LV1] Li M., Vitányi P., *An Introduction to Kolmogorov Complexity and its Applications*, Springer-Verlag, New York, 1993.

[OW1] Ogiwara M., Watanabe O., *On polynomial-time bounded truth-table reducibility of NP sets to sparse sets*, SIAM J. Comput. **20** (1991), no. 3, 471–483.

[M1] Mahaney S., *Sparse complete sets for NP: solution of a conjecture of Berman and Hartmanis*, J. Comput. System Sci. **25** (1982), 130–143.

[RR1] Ranjan D., Rohatgi P., *On randomized reductions to sparse sets*, in Proceedings of the 7th Structure in Complexity Theory Conference, IEEE Computer Society Press, 1992, pp. 239–242.

[S1] Saluja S., *Relativized limitations of left set technique and closure classes of sparse sets*, Proc. of the 8th IEEE Conf. Structure in Complexity Theory, 1993, pp. 215–222.

[S2] Schöning U., *On random reductions from sparse to tally sets*, Inform. Process. Lett. **46** (1993), 239–241.

[W1] Watanabe O., *A comparison of polynomial time completeness notions*, Theoret. Comput. Sci. **54** (1987), 249–265.

APP Czech, Na Strži 63, Prague 4, Czech Republic
E-mail: vglasnak@appg.com

(*Received* May 11, 1999)