Upper bounds for the Complexity of Sparse and Tally Descriptions*

V. Arvind†
Department of Computer Science
Institute of Mathematical Sciences, C.I.T Campus
Madras 600113, India

J. Köbler
Universität Ulm
Abteilung für Theoretische Informatik
D-89069 Ulm, Germany

M. Mundhenk
Universität Trier
Lehrstuhl IV – Informatik
D-54286 Trier, Germany

Abstract

We investigate the complexity of computing small descriptions for sets in various reduction classes to sparse sets. For example, we show that if a set A and its complement conjunctively reduce to some sparse set, then they also are conjunctively reducible to a $\mathrm{P}(A \oplus \mathrm{SAT})$-printable tally set. As a consequence, the class $\mathrm{IC}[\log, \mathrm{poly}]$ of sets with low instance complexity is contained in the EL_1^{Σ}-level of the extended low hierarchy. By refining our techniques, we also show that all word-decreasing self-reducible sets in $\mathrm{IC}[\log, \mathrm{poly}]$ are in NP $\cap$ co-NP and therefore low for NP. We derive similar results for sets in $R_d^p(\mathrm{SPARSE}))$ and $R_{hd}^p(R_c^p(\mathrm{SPARSE}))$, as well as in some nondeterministic reduction classes to sparse sets.

*Parts of this work have been presented at ISAAC'92, MFCS'93, and CIAC'94 [AKM92b, AKM93, Mu94].

†Work done while visiting Universität Ulm. Supported in part by an Alexander von Humboldt research fellowship.

1 Introduction

Sparse sets play a central role in structural complexity theory. The question of the existence of sparse hard sets for various complexity classes under different sorts of reducibilities is well studied (see for example [KL80, Mah82, OW91, AHH$^+$93, AKM92]). Besides, much work has also been done on other issues concerning the complexity of sets reducible to sparse sets (see for example [Kad87, AH92, LS91, AHH$^+$93]). A central motivation for most earlier work (and also for this paper) can be seen as seeking answers to the following two questions.

1. If a set A is reducible to a sparse set, does it follow that A is reducible to some sparse set that is "simple" relative to A?

2. If a set A is reducible to a sparse set, then how easy is it to access all the relevant information contained in the set A when it is used as oracle?

Question 1 originates in the study of the notions of equivalence and reducibility to sparse sets (see for example [TB91, AHOW92, GW93]) and concerns the complexity (relative to A) of small descriptions for sets A which are reducible to sparse sets. For the case of Turing reductions, Gavaldà and Watanabe [GW93] established an NP $\cap$ co-NP lower bound by constructing a set B that is Turing reducible to a sparse set but is not Turing reducible to a sparse set in NP$(B) \cap$ co-NP(B). Hence, the class of sets Turing equivalent to some sparse set is a proper subclass of P$/poly$. The separation of equivalence and reduction classes for restricted truth-table reducibilities is further investigated in [AHOW92]. In this context, it is interesting to know, for various classes of sets that reduce to sparse sets, the complexity of the easiest sparse sets to which such sets reduce. In [AHH$^+$93] this question is investigated for certain truth-table reducibilities and upper bounds are established for the relative complexity of sparse descriptions (see Section 3 for a summary of all previously known upper bound results).

The second question concerns lowness properties of sets reducible to sparse sets. Intuitively speaking, a set is low for a certain class if it gives no power (or limited power) as oracle for that class. Lowness is an important structural tool for systematically classifying sets in different complexity classes according to the power they provide when used as oracles. Building on ideas from recursive function theory, Schöning [Sch83] defined the low hierarchy inside NP. In order to classify sets outside of NP (e.g. sets reducible to sparse sets) Balcázar, Book, and Schöning [BBS86] defined the extended low hierarchy. The extended low hierarchy was subsequently refined by Allender and Hemachandra [AH92] and Long and Sheu [LS91] who showed the optimal location of various reduction and equivalence classes to sparse and tally sets in the (refined) extended low hierarchy.

To formalize the first question, we generalize the definition of sparse set descriptions given in [GW93]. Let $\leq_r$ be a reducibility. A sparse set S is a sparse r-description for a set A if $A \leq_r S$. Similarly, a tally set T is a tally r-description

for A if $A \leq_r T$. In particular, a sparse set S is called a sparse description for A if $A \leq_T^p S$. For every set A in $R_r(\text{SPARSE})$ (respectively $R_r(\text{TALLY})$), we are interested in finding upper bounds for the complexity of sparse (tally) r-descriptions relative to A.[1] We refer to a sparse r-description satisfying the established upper bound as a *simple* sparse r-description (with respect to that upper bound), and we refer to the corresponding upper bound result as a *simplicity* result.

For the simplicity results derived in this paper we give three kinds of applications which provide answers in different forms to the second question above:

- Extended lowness results for the considered reduction classes.
- Lowness results for self-reducible sets and for NP sets in those reduction classes.
- Applying some of our lowness results for self-reducible sets we get new collapse results of the polynomial hierarchy, of the counting hierarchy, and for PSPACE under the assumption that NP (respectively PP, PSPACE) is contained in the considered reduction class.

In general, the approach we use to prove the lowness of a set A that reduces to a sparse set is the following: first we appropriately bound the complexity of a sparse description for A; based on such a simplicity result, we then apply a deterministic enumeration technique similar to that of Mahaney [Mah82] or a census technique similar to that of Hemachandra [Hem87] and Kadin [Kad87] to replace the sparse oracle S (and thus A). Intuitively, we use the simplicity result to extract enough information (a suitable initial segment of S or the census thereof) from the oracle set A in order to avoid further queries to A. When A is self-reducible we obtain stronger (unrelativized) simplicity results. More specifically, we exploit a certain minimization property of the word decreasing and the polynomially related self-reducibilities. Using this property we describe algorithms (for each of the considered reduction classes to sparse sets) which on input 0^n construct an initial segment of the sparse set to which A correctly reduces for all strings upto length n. The corresponding lowness results again follow as a consequence.

A main contribution of this paper is in explicitly bringing out the application of simplicity results to prove new lowness results. Regarding the extended lowness results for reduction classes to sparse sets proved in the appendix, we note that these have been recently subsumed by the containment of P/*poly* in EL_3^{Θ} [Köb93]. However, we have included these proofs since they are direct consequences of the corresponding simplicity results, whereas the proof in [Köb93] is based on a more complicated approximate counting argument.

Our collapse results have the following general pattern: Assume that a complexity class like NP (resp., PP or PSPACE) is contained in a particular reduction class

[1] For a reducibility $\leq_r$ and a class $\mathcal{C}$ of sets, $R_r(\mathcal{C}) = \{A \mid A \leq_r B \text{ for some } B \in \mathcal{C}\}$.

to sparse sets. Since there is a complete set A in NP (resp., PP or PSPACE) that is self-reducible, we can apply the corresponding simplicity result to appropriately bound the complexity of a sparse set S (to which A reduces via the considered reducibility) and locate it in some level of the polynomial hierarchy. Consequently, A is low for an appropriate level of PH.

For example, for every word-decreasing[2] self-reducible set A that is disjunctively reducible to a sparse set, we prove that there is a Δ_2^p-printable sparse set S such that A disjunctively reduces to S. Consequently, A is low for Δ_2^p. An immediate corollary of the above result is:

- If an NP-complete set disjunctively reduces to a sparse set then PH $= \Delta_2^p$.

The above result is interesting since the previous best known collapse of PH under the above assumption was to Σ_2^p [KL80]. Also it is known [Sal93] that in some relativized world the well-known left set technique cannot be used to get a collapse to P under the above assumption.

The paper is organized as follows. In Section 2 we describe the notation and give basic definitions.

In Section 3 we establish simplicity results for various reduction classes to sparse and tally sets. For self-reducible sets in these classes we improve the simplicity results. In particular, we prove the following.

- If a set A and its complement conjunctively reduce to some sparse set, then they are conjunctively reducible to a P($A \oplus$SAT)-printable tally set. Furthermore, all word-decreasing self-reducible sets in IC[log, poly] are in NP$\cap$co-NP.

- For every set $A \in R_{hd}^p(R_c^p(\text{SPARSE}))$ there is a sparse set $\hat{S} \in \text{NP}(A)$ such that $A \in R_{hd}^p(R_c^p(\hat{S}))$. Furthermore, all word-decreasing self-reducible sets A in $R_{hd}^p(R_c^p(\text{SPARSE}))$ are low for Δ_2^p, i.e. $\Delta_2^p(A) = \Delta_2^p$.

The results in Section 4 are obtained by applying proof techniques from Section 3 to classes defined by nondeterministic reductions to sparse and tally sets.

Finally, in Section 5 we apply the simplicity results to derive collapse and lowness results for the considered reduction classes. In particular, we show, using a Kadin like census argument, that all sets in NP $\cap\, R_m^{co\text{-}np}(\text{SPARSE})$ are low for Θ_2^p. This improves all previously known Θ_2^p-lowness results for NP sets reducible to sparse or tally sets (e.g. for NP $\cap\, R_m^p(\text{SPARSE})$ and for NP $\cap$ co-NP/ log, cf. [LS91]). Furthermore, using corresponding simplicity results, we show that IC[log, poly] and $R_m^{co\text{-}np}(\text{TALLY}) \cap R_m^{np}(\text{TALLY})$ are contained in EL_1^Σ. This improves all previously known EL_1^Σ-lowness results regarding sets reducible to sparse or tally sets (e.g. for APT [LS91], for $E_T^p(\text{TALLY})$ [BB86] and for $R_m^p(\text{TALLY})$ [AH92]).

[2]Our self-reducibility results hold for a notion of self-reducibility defined in Section 2 which generalizes the word-decreasing and polynomially related self-reducibilities defined in [Ko83] and [Ba92] respectively.

Name	Notation
many-one	$\leq^p_m$
one truth-table	$\leq^p_{1\text{-}tt}$
bounded truth-table	$\leq^p_{btt}$
conjunctive	$\leq^p_c$
disjunctive	$\leq^p_d$
bounded Hausdorff	$\leq^p_{bhd}$
Hausdorff	$\leq^p_{hd}$
non-monotonic Hausdorff	$\leq^p_{nhd}$
np many-one	$\leq^{np}_m$
co-np many-one	$\leq^{co\text{-}np}_m$

Table 1: Polynomial-time Reductions

2 Preliminaries and notation

For a set A, χ_A denotes the characteristic function of A. $A^{=n}$ ($A^{\leq n}$) denotes the set of all strings in A of length n (up to length n, respectively). The cardinality of A is denoted by $|A|$. A set T is called a tally set if $T \subseteq 0^*$. The census function of a set A is defined as $census_A(1^n) = |A^{\leq n}|$. A set S is called sparse if its census function is bounded above by a polynomial. We use TALLY and SPARSE to represent the classes of tally and sparse sets, respectively.

The join of two sets A and B, denoted $A \oplus B$, is defined as $A \oplus B = \{0x \mid x \in A\} \cup \{1x \mid x \in B\}$. $\langle \cdot, \cdot \rangle$ denotes a standard polynomial time invertible pairing function such that $\langle 0^i, 0^j \rangle \in 0^*$ for all $i, j \geq 0$. Such a pairing function can be extended in a standard fashion to encode arbitrary finite sets F of strings into a string $\langle F \rangle$. For a class $\mathcal{C}$ of sets we denote the union of all sets in $\mathcal{C}$ by $\bigcup \mathcal{C}$. A set S is called $\mathrm{P}^{\mathcal{C}}$-printable (see [HY84]) if the function $0^n \mapsto \langle S^{\leq n} \rangle$ can be computed in $\mathrm{FP}^{\mathcal{C}}$.

The reducibilities discussed in this paper are the standard polynomial time bounded reducibilities defined by Ladner, Lynch, and Selman [LLS75], the Hausdorff reducibility introduced by Wagner [Wag87], and the non-monotonic Hausdorff reducibility (see Table 1 for an overview). At first we give a short overview on the reducibilities we use.

Definition 2.1 *1. A set A is conjunctively reducible to a set B ($A \leq^p_c B$), if there exists a polynomial-time computable function f mapping strings to sets of strings such that for all $x \in \Sigma^*$ it holds that $x \in A \iff f(x) \subseteq B$.*

2. A set A is disjunctively reducible to a set B ($A \leq^p_d B$), if there exists a polynomial-time computable function f mapping strings to sets of strings such that for all $x \in \Sigma^*$ it holds that $x \in A \iff f(x) \cap B \neq \emptyset$.

3. *A set A is co-np many-one reducible to a set B ($A \leq_m^{co\text{-}np} B$), if there exists a polynomial-time nondeterministic Turing transducer M such that for every $x \in \Sigma^*$, $x \in A$ if and only if all outputs of M on input x are in B.*

4. *A set A is np many-one reducible to a set B ($A \leq_m^{np} B$), if there exists a polynomial-time nondeterministic Turing transducer M such that for every $x \in \Sigma^*$, $x \in A$ if and only if at least one output of M on input x is in B.*

Note that $A \leq_c^p B$ if and only if $\overline{A} \leq_d^p \overline{B}$ and that $A \leq_m^{co\text{-}np} B$ if and only if $\overline{A} \leq_m^{np} \overline{B}$. Clearly, for every set B, $R_c^p(B)$ is contained in $R_m^{co\text{-}np}(B)$, and $R_m^{co\text{-}np}(B)$ is closed downward under $\leq_c^p$ and $\leq_m^{co\text{-}np}$ reductions.

Definition 2.2 [Wag87] *A set A is Hausdorff-reducible to B ($A \leq_{hd}^p B$), if there exist a polynomial k and a polynomial-time computable function f mapping strings to tuples of strings, such that for all x, f computes a tuple $f(x) = \langle y_1, y_2, \ldots, y_{k(|x|)} \rangle$ such that*

1. *$\chi_B(y_1) \geq \chi_B(y_2) \geq \cdots \geq \chi_B(y_{k(|x|)})$, and*

2. *$x \in A \iff \max\{j \mid 0 \leq j \leq k(|x|)$ and $\chi_B(y_1) = \cdots = \chi_B(y_j) = 1\}$ is odd.*

We call f a bounded Hausdorff reduction ($A \leq_{bhd}^p B$) if k is a constant and we denote the i-th query y_i of $f(x)$ by $f(x,i)$, $1 \leq i \leq k(|x|)$.

Dropping the monotony condition (condition 1 above) we obtain the following generalization of the Hausdorff reducibility.

Definition 2.3 *A is non-monotonic Hausdorff reducible to B ($A \leq_{nhd}^p B$), if there exist a polynomial k and a polynomial-time computable function f such that for all x, f computes a tuple $f(x) = \langle y_1, y_2, \ldots, y_{k(|x|)} \rangle$ such that*

$$x \in A \iff \max\{j \mid 0 \leq j \leq k(|x|) \text{ and } \chi_B(y_1) = \cdots = \chi_B(y_j) = 1\} \text{ is odd.}$$

Non-monotonic Hausdorff reducibility is not a new type of reducibility, it is just the composition of Hausdorff and conjunctive reducibility.

Theorem 2.4 *For every class $\mathcal{K}$ of languages, $R_{nhd}^p(\mathcal{K}) = R_{hd}^p(R_c^p(\mathcal{K}))$.*

Proof If $A \leq_{nhd}^p B$ via f, define f' as $f'(x) = \langle \langle x_1 \rangle, \langle x_1, x_2 \rangle, \ldots, \langle x_1, \ldots, x_n \rangle \rangle$ where $f(x) = \langle x_1, \ldots, x_n \rangle$. Let $B' = \{\langle x_1, \ldots, x_n \rangle \mid \{x_1, \ldots, x_n\} \subseteq B\}$. It is straightforward to see that $A \leq_{hd}^p B'$ via f', and that $B' \in R_c^p(B)$. For the other inclusion direction let $A \leq_{hd}^p B$ via f and $B \leq_c^p C$ via g. Define f' as follows: if $f(x) = \langle y_1, \ldots, y_n \rangle$ and $g(y_i) = \{y_{i,1}, \ldots, y_{i,m_i}\}$, $1 \leq i \leq n$, then

$$f'(x) = \langle \underbrace{y_{1,1}, y_{1,1}, y_{1,2}, y_{1,2}, \ldots, y_{1,m_1-1}, y_{1,m_1-1}, y_{1,m_1}}_{2m_1 - 1 \text{ elements}}$$
$$\underbrace{y_{2,1}, y_{2,1}, \ldots, y_{2,m_2}}_{2m_2 - 1 \text{ elements}}, \ldots, \underbrace{y_{n,1}, y_{n,1}, \ldots, y_{n,m_n}}_{2m_n - 1 \text{ elements}} \rangle$$

Then $A \leq^p_{nhd} C$ via f'. □

Notation [AHOW92] For any reducibility $\leq^\alpha_r$ and any class $\mathcal{C}$ of sets let $R^\alpha_r(\mathcal{C}) = \{A \mid A \leq^\alpha_r B \text{ for some } B \in \mathcal{C}\}$.

A class $\mathcal{K}$ of sets that includes $\emptyset$ and Σ^* and is closed under union and intersection is said to be a set ring. We show how Boolean closures, bounded truth-table closures, and the bounded Hausdorff closures of language classes are related.

Theorem 2.5 [Hau14] *Let $\mathcal{K}$ be a set ring and let $BC(\mathcal{K})$ be the closure of $\mathcal{K}$ under union, intersection, and complement. Then every $A \in BC(\mathcal{K})$ can be represented as $A = \bigcup_{i=1}^{k}(A_{2i-1} - A_{2i})$, where $A_j \in \mathcal{K}$, $1 \leq j \leq 2k$, and $A_1 \supseteq A_2 \supseteq \cdots \supseteq A_{2k}$.*

The above representation for A is called a Hausdorff representation for A over $\mathcal{K}$.

Lemma 2.6 [KSW87] *Let $\mathcal{K}$ be a class that is closed under join and under many-one reductions. Then $BC(\mathcal{K}) = R^p_{btt}(\mathcal{K})$.*

Lemma 2.7 *Let $\mathcal{K}$ be a class closed under many-one reductions and under join. Then A has a Hausdorff representation over $\mathcal{K}$ if and only if A is bounded Hausdorff reducible to some set in $\mathcal{K}$.*

Proof Let $A_1, \ldots, A_{2k}$ be a Hausdorff representation for A. Then A is bounded Hausdorff reducible to $A' = A_1 \oplus \cdots \oplus A_{2k}$ via f with $f(x) = \langle f_1(x), \ldots, f_{2k}(x)\rangle$, where f_i is a many-one reduction function from A_i to A'.

If A is bounded Hausdorff reducible to a set $B \in \mathcal{K}$ via f then $A = \bigcup_{i=1}^{k}(B_{2i-1} - B_{2i})$ where $B_j = \{x \mid f(x) = \langle y_1, \ldots, y_{2k}\rangle \text{ and } y_j \in B\}$. Since $B_j \leq^p_m B$ this is a Hausdorff representation for A over $\mathcal{K}$. □

The next lemma is obtained by combining Theorem 2.5 and the above two lemmas.

Corollary 2.8 *If $\mathcal{K}$ is a set ring which contains* P *and is closed under many-one reductions then $BC(\mathcal{K}) = R^p_{bhd}(\mathcal{K}) = R^p_{btt}(\mathcal{K})$.*

Theorem 2.9

1. $R^p_{bhd}(R^p_c(\text{SPARSE})) = R^p_{btt}(R^p_c(\text{SPARSE}))$
2. $R^p_{bhd}(R^{co\text{-}np}_m(\text{SPARSE})) = R^p_{btt}(R^{co\text{-}np}_m(\text{SPARSE}))$

Proof It suffices to show that $R^p_c(\text{SPARSE})$ and $R^{co\text{-}np}_m(\text{SPARSE})$ are set rings. It is easy to see that $R^p_c(\text{TALLY})$ and $R^{co\text{-}np}_m(\text{TALLY})$ are set rings. But since $\text{SPARSE} \subset R^p_c(\text{TALLY})$ [BLS93] it follows that $R^p_c(\text{SPARSE}) = R^p_c(\text{TALLY})$ and $R^{co\text{-}np}_m(\text{SPARSE}) = R^{co\text{-}np}_m(\text{TALLY})$. □

A reduction is honest if there exists a polynomial p such that for every query y generated on input x it holds that $|x| \leq p(|y|)$. The following lemma shows that we can assume that all reductions to sparse or tally sets are honest.

Lemma 2.10 *If A reduces to a sparse (tally) set S via a polynomial-time bounded reduction of a certain type then A reduces to a sparse (resp. tally) set S' via an honest reduction of the same type.*

Proof Define $S' = \{\langle 0^n, y\rangle \mid n \geq 0, y \in S\}$ and in the new reduction to S' replace each query y on input x by the query $\langle 0^{|x|}, y\rangle$. Clearly, the new reduction is polynomially time-bounded. □

In the case of reductions to sparse sets we can even assume that the query length depends only on the input length (via some strictly increasing polynomial).

Lemma 2.11 *If A reduces to a sparse set S via a polynomially time-bounded reduction of a certain type then A reduces to a sparse set S' via a reduction of the same type such that for all x the queries generated during the computation of the reduction function have length $r(|x|)$ for some strictly increasing polynomial r.*

Proof Let A reduce to a sparse set S, and let t be a polynomial bounding the computation time of the reduction function. Define a "padded" version of S as the sparse set $S' = \{z10^i \mid z \in S, i \geq 0\}$. Then A reduces to S' via a reduction function using the same reduction mechanism, but where each query y is replaced by $y10^{t(|x|)-|y|}$. □

For a class $\mathcal{C}$ of sets and a class $\mathcal{F}$ of functions from Σ^* to Σ^* let $\mathcal{C}/\mathcal{F}$ [KL80] be the class of sets A such that there is a set $B \in \mathcal{C}$ and a function $h \in \mathcal{F}$ such that for all $x \in \Sigma^*$,

$$\chi_A(x) = \chi_B(x, h(0^{|x|})).$$

For a set A, the class $\Theta(A)$ [LS91] contains all languages $L(M, A)$ accepted by a deterministic polymomial time bounded oracle machine M asking on inputs of length n at most $O(\log n)$ queries to A. A deterministic polynomial-time oracle machine M as in the definition above is called a Θ machine. The Θ levels of the (relativized) polynomial time hierarchy are defined as $\Theta_k^p(A) = \Theta(\Sigma_{k-1}^p(A)), k \geq 1$ [Wag90, LS91].

Similarly, the class $\mathrm{F}\Theta(A)$ contains all functions computable by some deterministic polymomial time bounded oracle transducer M asking on inputs of length n at most $O(\log n)$ queries to oracle A, and $\mathrm{F}\Theta_k^p(A) = \mathrm{F}\Theta(\Sigma_{k-1}^p(A)), k \geq 1$.

In the following we give definitions of different kinds of self-reducibilities studied in the literature.

Definition 2.12 *[Ko83] An irreflexive partial order $\prec$ on Σ^* is said to be* polynomially related, *if there is a polynomial p such that*

1. *$x \prec y$ implies $|x| \leq p(|y|)$ (we say that $\prec$ is* polynomially well-founded*),*
2. *$x \prec y$ is decidable in time polynomial in $|x| + |y|$, and*
3. *$x_1 \prec x_2 \prec \cdots \prec x_k$ implies $k \leq p(|x_k|)$.*

Clearly, the length-decreasing order is polynomially related, whereas the word-decreasing order (i.e. the lexicographic order) is not, since it does not fulfil condition 3 above.

Definition 2.13 *A set A is* polynomially well-founded self-reducible, *if there is some polynomial-time oracle machine M and a polynomially well-founded order $\prec$, such that $A = L(M, A)$ and on any input x, M queries the oracle only about words $q \prec x$.*

By restricting $\prec$ to a polynomially related order (resp. to the lexicographic order) we get the polynomially related self-reducibility of Ko (resp. the word-decreasing self-reducibility of Balcázar) [Ko83, Ba92].

Lemma 2.14

1. *Every polynomially well-founded self-reducible set is in $DTIME(2^{n^{O(1)}})$.*
2. **[Ba92]** *Every word-decreasing self-reducible set is in $DTIME(2^{O(n)})$.*
3. **[BBS86]** *Every polynomially related self-reducible set is in PSPACE.*

For other standard definitions used in the paper we refer the reader to standard books on structural complexity theory (for example [BDG, Sch86]).

3 Simple descriptions for deterministic reduction classes

In this section we will investigate simplicity properties of the classes IC[log, poly], $R^p_{hd}(R^p_c(\text{SPARSE}))$, and $R^p_d(\text{SPARSE})$. The inclusion relations between these classes are represented in Figure 1 (see [Ko89, AHOW92, AHH+93, BLS93]). Bold lines represent proper inclusions.

We summarize the previously known simplicity results in the next two theorems.

Theorem 3.1 [LS91] *For every set A in $R^p_m(\text{SPARSE})$ there exists a sparse set $S \in R^{np}_m(A)$ such that $A \leq^p_m S$.*

As shown in [AHH+93], the same upper bound as in the theorem above also holds for $R^p_c(\text{SPARSE})$, i.e., for every set $A \in R^p_c(\text{SPARSE})$ there exists a sparse set $S \in R^{np}_m(A)$ such that $A \leq^p_c S$.

Theorem 3.2

1. **[Sch86, GW93]** *For every set A in P/poly there exists a sparse set $S \in \Delta^p_3(A)$ such that $A \leq^p_T S$.*

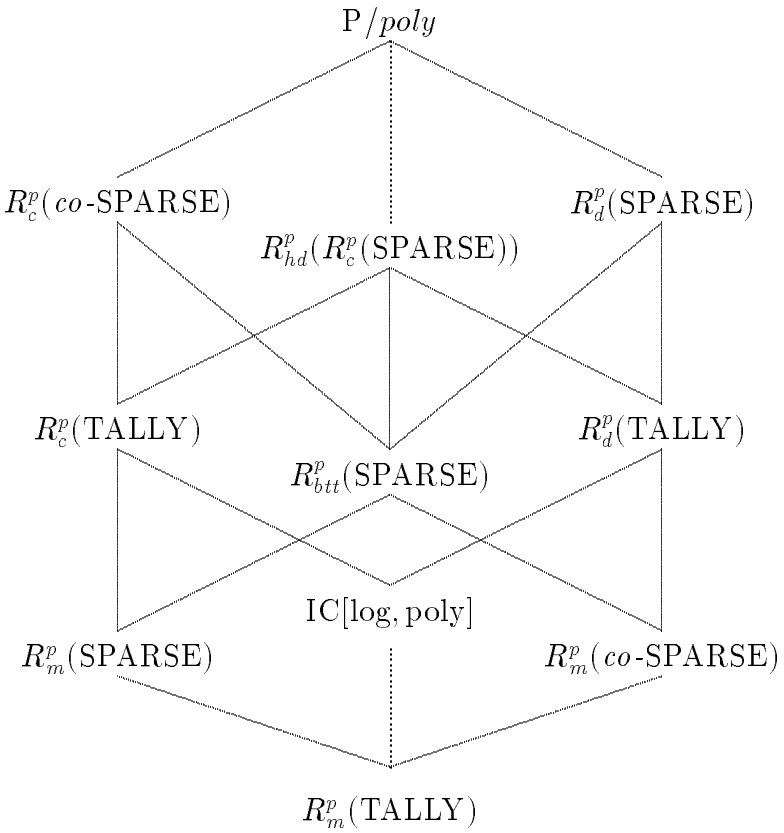

Figure 1: Structure of inclusions

2. **[AHH+93]** *Any set A that disjunctively reduces (respectively, disjunctive bounded truth-table reduces, 2-truth-table reduces) to a sparse set in fact disjunctively reduces (respectively, disjunctive bounded truth-table reduces, 2-truth-table reduces) to a sparse set that is in $\Delta^p_2(A)$ (respectively, $\Theta^p_2(A)$, $\Theta^p_2(A)$).*

3. **[AHH+93]** *For $k = 2$ and $k = 3$: If $A \leq^p_{k\text{-}d} S$ for some sparse set S, then there is a sparse set $\hat{S}$ such that $A \leq_{k\text{-}d} \hat{S}$ and $\hat{S} \in \mathrm{P}(\mathrm{NP} \oplus A)$.*

The first upper bound for the complexity of self-reducible sets which Turing reduce to sparse sets was proved by Karp and Lipton [KL80]. Lozano and Torán [LT91] considered the effect of different self-reducibilities for sets in more restricted reduction classes to sparse sets. We summarize these results in the next theorem.

Theorem 3.3

1. **[KL80]** *Every polynomially related self-reducible set in* P/*poly is low for* Σ^p_2.

2. **[LT91]** *Every word-decreasing and every polynomially related self-reducible set in* $R_m^p(\mathrm{SPARSE})$ *is low for* Δ_2^p.

3. **[OKSW]** *Every polynomially related self-reducible set in* $\mathrm{IC}[\log, \mathrm{poly}]$ *is in* P.

3.1 Sets with low instance complexity

The notion of instance complexity and the class IC[log,poly] of sets of strings with low instance complexity were introduced by Ko, Orponen, Schöning, and Watanabe (see [OKSW]). In [AHH+93], a characterization of IC[log,poly] in terms of a reduction class to tally sets is proved.

Theorem 3.4 [AHH+93] *A set* A *is in* $\mathrm{IC}[\log, \mathrm{poly}]$ *if and only if* A *and its complement are in* $R_c^p(\mathrm{TALLY})$.

In order to prove our simplicity results we need the following lemma.

Lemma 3.5 *If a set* A *and its complement are in* $R_c^p(\mathrm{TALLY})$, *then there is a tally set* T *such that* $A \leq_c^p T$ *and* $\overline{A} \leq_c^p T$ *via honest reduction functions.*

Proof Let T_1, T_2 be tally sets and let f, g be conjunctive reduction functions witnessing $A \leq_c^p T_1$ and $\overline{A} \leq_c^p T_2$, i.e. $x \in A \Leftrightarrow f(x) \subseteq T_1 \Leftrightarrow g(x) \not\subseteq T_2$ for all x. Define the tally set T as follows

$$T = \{0^{2i+1} \mid 0^i \in T_1\} \cup \{0^{2i} \mid 0^i \notin T_2\}.$$

Furthermore, consider the following reduction functions f' and g'

$$f'(x) = \{0^{2i+1} \mid 0^i \in f(x)\},\ g'(x) = \{0^{2i} \mid 0^i \in g(x)\}.$$

Then $A \leq_c^p T$ via f' and $\overline{A} \leq_c^p T$ via g'. Since by Lemma 2.10 the functions f and g can be assumed to be honest, it follows that f' and g' are honest too. □

Theorem 3.6 *If* A *and* $\overline{A}$ *are in* $R_c^p(\mathrm{TALLY})$ *then there is a* $\mathrm{P}(\mathrm{SAT} \oplus A)$*-printable tally set* $\hat{T}$ *such that* A *and* $\overline{A}$ *are in* $R_c^p(\hat{T})$.

Proof Let A and $\overline{A}$ be in $R_c^p(\mathrm{TALLY})$. By Lemma 3.5, there are honest polynomial time reduction functions f and g and a tally set T such that for all strings $x \in \Sigma^*$,

$$x \in A \Longleftrightarrow f(x) \subseteq T \Longleftrightarrow g(x) \not\subseteq T.$$

Let p be a polynomial such that $m \leq p(|x|)$ and $|x| \leq p(m)$ for all queries 0^m in $f(x) \cup g(x)$ (w.l.o.g. we assume that all queries are in 0^*). Consider the following procedure which on input 0^n outputs a tally set T_n. Define $\hat{T} = \bigcup_{n \geq 0} T_n$. We will show that A and its complement conjunctively reduce to $\hat{T}$ via functions f and g respectively.

input 0^n
$I := \emptyset$
for $i := 0$ **to** $p(n)$ **do**
  for $j := 0$ **to** $p(n)$ **do**
    if $\exists x \in \Sigma^{\leq p(n)} : 0^i \in f(x) \wedge 0^j \in g(x)$
      then $I := I \cup \{x\}$ for the lexicographically first such x
    end
  end
end
output $T_n = (\bigcup_{x \in I \cap A} f(x) \cup \bigcup_{x \in I \cap \overline{A}} g(x))$

Let $f(A) = \bigcup_{x \in A} f(x)$ and $g(\overline{A}) = \bigcup_{x \in \overline{A}} g(x)$ respectively. The following obvious claim states that only strings from T are included into the set $\hat{T}$ by the above algorithm.

Claim 1 *$\hat{T} \subseteq f(A) \cup g(\overline{A}) \subseteq T$.*

The next claim ensures that enough strings are collected in the set T_n in order to decide the membership to A of all strings up to length n.

Claim 2 *$f(A^{\leq n}) \subseteq T_n$ and $g(\overline{A}^{\leq n}) \subseteq T_n$.*

Proof To prove the first inclusion, assume to the contrary that for some $x \in A^{\leq n}$ a query $0^i \in f(x)$ is not contained in T_n. Then x is not included in I during the computation of T_n. Hence, for all $0^j \in g(x)$ a string $y_j \neq x$ such that $0^i \in f(y_j)$ and $0^j \in g(y_j)$ is included in I. Since $0^i \notin T_n$, it follows that none of the y_j's are in A. But then $g(x) \subseteq \bigcup_j g(y_j) \subseteq g(\overline{A})$ which implies $x \notin A$ and yields a contradiction.
The second inclusion can be proved similarly. □

Combining Claims 1 and 2 we get that $\hat{T} = f(A) \cup g(\overline{A})$. It immediately follows that $A \leq_c^p \hat{T}$ via f and $\overline{A} \leq_c^p \hat{T}$ via g. The algorithm is clearly a P(SAT $\oplus A$) algorithm. Since f and g are honest reduction functions, it follows that $\hat{T}^{\leq n} = T_n^{\leq n}$. Hence, restricting the output of the above algorithm on input 0^n to strings in $0^{\leq n}$ shows that $\hat{T}$ is P(NP $\oplus$ SAT)-printable. □

It is interesting to observe that the FP(SAT $\oplus A$)-printing algorithm in the above proof first uses the NP oracle to determine a set of queries which are then non-adaptively asked to A.

In [OKSW] it is shown that every polynomially related self-reducible set in IC[log,poly] belongs to P. We next show that all sets in IC[log,poly] which are self-reducible under the weaker notion of polynomially well-founded self-reducibility are in NP $\cap$ co-NP.

Theorem 3.7 *If A is a polynomially well-founded self-reducible set, and A and $\overline{A}$ are in R_c^p(TALLY), then A is in* NP $\cap$ co-NP.

Proof Let A be a polynomially well-founded self-reducible set such that A and $\overline{A}$ are in $R_c^p(\mathrm{TALLY})$. By Lemma 3.5 there exist a tally set T and honest functions $f, g \in \mathrm{FP}$ such that $A \leq_c^p T$ via f and $\overline{A} \leq_c^p T$ via g, i.e. for all x,

$$x \in A \iff f(x) \subseteq T \iff g(x) \not\subseteq T.$$

Furthermore there exists a polynomial time bounded oracle Turing machine M which asks queries to the oracle with respect to a polynomially well-founded order $\prec$, such that $A = L(M, A)$. Let p be a polynomial which bounds $\prec$, as well as the running times of M and of the machines computing f and g. Consider the following Turing machine M' which on input $\langle x, S\rangle$ simulates M as described below.

input $\langle x, S\rangle$
simulate M on input x, but whenever M
queries q to the oracle then do the following:
  if $f(q) \subseteq S$
    then continue with answer "yes"
  else if $g(q) \subseteq S$
    then continue with answer "no"
  else halt without decision **end**
  accept if and only if M accepts

The next claim shows that the algorithm is "robust" in the following sense: for every x and every $S \subseteq T$, if M' on input $\langle x, S\rangle$ decides, then it decides correctly the status of x with respect to A.

Claim 3 *Let $S \subseteq T$. Then for all x, if $M'(\langle x, S\rangle)$ decides its input, then $M'(\langle x, S\rangle)$ accepts iff $x \in A$.*

Proof Let $S \subseteq T$. If $f(q) \subseteq S$, then $f(q) \subseteq T$ and therefore $q \in A$. Similarly, if $g(q) \subseteq S$, then $q \notin A$. Since M' on input $\langle x, S\rangle$ halts with a decision only if for all queries q it holds that $f(q) \subseteq S$ or $g(q) \subseteq S$ the simulation is clearly correct. □

Define the function *Queries* as

$$Queries(x) = \begin{cases} f(x), & x \in A, \\ g(x), & x \notin A, \end{cases}$$

and let $Queries(x_1, \ldots, x_n) = \bigcup_{1 \leq i \leq n} Queries(x_i)$. By definition we have for all x_i that $Queries(x_1, \ldots, x_n) \subseteq T$.

Claim 4 *For every n, there exist an $m \leq p(p(n))$ and a sequence $x_1, \ldots, x_m$ of strings in $\Sigma^{\leq p(n)}$, such that for all $x \in \Sigma^{\leq n}$ it holds that*

$$x \in A \iff f(x) \subseteq Queries(x_1, \ldots, x_m) \iff g(x) \not\subseteq Queries(x_1, \ldots, x_m).$$

Proof For a given n, let $x_1, \ldots, x_m \in \Sigma^{\leq p(n)}$ be a sequence of maximum length fulfilling for all i, $1 \leq i \leq m$, the following two conditions.

(1) $Queries(x_i) \not\subseteq Queries(x_1, \ldots, x_{i-1})$ and

(2) $M'(\langle x_i, Queries(x_1, \ldots, x_{i-1})\rangle)$ decides x_i.

Since T is a tally set, it follows from condition (1) that $m \leq p(p(n))$. It remains to show that for all $x \in \Sigma^{\leq n}$ it holds that $x \in A \iff f(x) \subseteq Queries(x_1, \ldots, x_m) \iff g(x) \not\subseteq Queries(x_1, \ldots, x_m)$. Assume to the contrary that there exists a string $x \in \Sigma^{\leq n}$ for which this is not true. Since $Queries(x_1, \ldots, x_m) \subseteq T$, it follows that neither $f(x) \subseteq Queries(x_1, \ldots, x_m)$ nor $g(x) \subseteq Queries(x_1, \ldots, x_m)$, contradicting the maximality of m. □

The following non-deterministic Turing machine N accepts A.

input y
$S := \emptyset$
guess $m \leq p(p(|y|))$, $x_1, \ldots, x_m \in \Sigma^{\leq p(|y|)}$
for $i := 1$ **to** m **do**
  if $M'(\langle x_i, S\rangle)$ accepts
    then $S := S \cup f(x_i)$
  else if $M'(\langle x_i, S\rangle)$ rejects
    then $S := S \cup g(x_i)$
  else *reject* **end**
end (* of the for-loop *)
if $f(y) \subseteq S$ **then** *accept* **else** *reject* **end**

By Claim 3, it follows inductively that $S \subseteq T$ at every step of the computation of N. Hence, only strings $y \in A$ are accepted. From Claim 4 it follows that for every input $y \in A$ there exists an accepting computation path. It is clear that N runs in polynomial time. This proves $A \in$ NP. Since $\overline{A} \in R_c^p(\text{TALLY})$ and since the class of polynomially well-founded self-reducible sets are closed under complementation, it follows that also $\overline{A} \in$ NP, which completes the proof. □

Corollary 3.8 *Every polynomially well-founded self-reducible set in* IC[log, poly] *is in* NP ∩ co-NP.

3.2 Hausdorff reductions

In this section, we prove simplicity results for Hausdorff reductions. In particular we show that for each set A in $R_{hd}^p(R_c^p(\text{SPARSE}))$ there is a sparse $hd(c)$-description in NP(A). As a consequence of the proof, an analogous result holds for bounded Hausdorff reductions, i.e., for each set A in $R_{bhd}^p(R_c^p(\text{SPARSE}))$ there is a sparse $bhd(c)$-description in NP(A). Although $R_{btt}^p(\text{SPARSE}) \subsetneq R_{bhd}^p(R_c^p(S))$,

somewhat anomalously, the only known upper bound for the complexity of sparse *btt*-descriptions for a set A in $R^p_{btt}(\mathrm{SPARSE})$ is $\Delta^p_3(A)$ (this can be seen by a minor modification of the proof of Theorem 3.2).

Theorem 3.9 *For every set* $A \in R^p_{hd}(R^p_c(\mathrm{SPARSE}))$ *there is a sparse set* $\hat{S} \in \mathrm{NP}(A)$ *such that* $A \in R^p_{hd}(R^p_c(\hat{S}))$.

Proof Let A be a set in $R^p_{hd}(R^p_c(\mathrm{SPARSE}))$. Then $A \leq^p_{hd} B$ via a polynomial k and a reduction function f for a set B such that $B \leq^p_c S$ via g for a sparse set S. By Lemma 2.11 we can assume that there exists a strictly increasing polynomial r such that $\bigcup_{i=1}^{k(|x|)} g(f(x,i)) \subseteq \Sigma^{r(|x|)}$ for all x. In the sequel we will use $h(i,x)$ to denote the set $g(f(x,i))$ for $1 \leq i \leq k(|x|)$, and we define $h(0,x) = \emptyset$.

We first observe that A reduces via the given reduction functions to any set T that fulfils the following two conditions for all x.

1. For all $1 \leq i \leq k(|x|)$, $h(i,x) \subseteq T$ implies $h(i-1,x) \subseteq T$, and
2. $x \in A$ if and only if $\max\{i \mid 0 \leq i \leq k(|x|), h(i,x) \subseteq T\}$ is odd.

Let $\hat{S}$ be the intersection of all sets T that fulfil the above conditions. It is straightforward to show that $\hat{S}$ also fulfils the above conditions. Also, since $\hat{S} \subseteq S$, $\hat{S}$ is sparse. Thus, in order to prove the theorem it suffices to give an $\mathrm{NP}(A)$ algorithm that accepts $\hat{S}$.

Intuitively, on input w of length $r(n)$ starting with $T = \emptyset$ the algorithm expands T inside $S^{=r(n)}$ by guessing strings in Σ^n witnessing that T does not yet fulfil one of the two conditions above. Finally w is accepted if $w \in T$.

Let p be a polynomial bounding the census of S, i.e. $census_S(1^{r(n)}) \leq p(r(n))$. Consider the following NP oracle machine M that uses oracle A.

input w
if $|w| \notin \{r(n) \mid n \geq 0\}$ **then** *reject* **end**
compute n such that $|w| = r(n)$
$T := \emptyset$
guess $c \in \{1, \ldots, p(r(n))\}$
guess $x_1, x_2, \ldots, x_c \in \Sigma^n$
for $j := 1$ **to** c **do**
  $m := \max\{i \mid 0 \leq i \leq k(|x_j|), h(i,x_j) \subseteq T\}$
  $T := T \cup \bigcup_{1 \leq i \leq m} h(i,x_j)$
  if $x_j \in A \iff m$ is even
    then $T := T \cup h(m+1, x_j)$
  end
end (* of the for-loop *)
if $w \in T$ **then** *accept* **else** *reject* **end**

We prove that $L(M,A) = \hat{S}$ by establishing the following two claims.

Claim 5 *T is a subset of $\hat{S}$ during each step of the computation, and hence $L(M,A) \subseteq \hat{S}$.*

Proof of Claim 5: We have to show that T is contained in every set C that fulfils the above conditions. This certainly holds when the computation starts, since then $T = \emptyset$. To show that $T \subseteq C$ is an invariant of the for-loop assume that $T \subseteq C$ before the j-th iteration of the for-loop. By Condition 1 above it follows that $\bigcup_{1 \leq i \leq m} h(i, x_j) \subseteq C$. Further, if the condition "$x_j \in A$ iff m is even" holds, then it follows from Condition 2 above that $h(m+1, x_j) \subseteq C$. Therefore it follows that $T \subseteq C$ after the execution of the j-th iteration of the for-loop. □

Claim 6 *For every input of length $r(n)$ there is a computation path along which the set $T = \hat{S}^{=r(n)}$ is computed, and hence $L(M,A) = \hat{S}$.*

Proof of Claim 6: Let $T(n)$ be a set of maximum cardinality that is computed along some path τ on some input of length $r(n)$. By Claim 5 we have that $T(n) \subseteq \hat{S}^{=r(n)}$. Therefore, to prove that $T(n) = \hat{S}^{=r(n)}$, we only need to show that $T(n)$ fulfils Conditions 1 and 2 for all strings of length n.

If $|T(n)| = p(r(n))$, then it follows that $T(n) = S^{=r(n)} = \hat{S}^{=r(n)}$. Otherwise, there is a computation path on which $x_1, x_2, \ldots, x_c$, $c < p(r(n))$, are guessed and along which $T(n)$ is computed. In order to derive a contradiction, assume that there is a string x_{c+1} witnessing the failure of Conditions 1 or 2 for $T(n)$. Consequently there is a computation path along which $x_1, x_2, \ldots, x_c, x_{c+1}$ are guessed and along which a set larger than $T(n)$ is computed. This contradicts the maximality of $T(n)$. □

Since the given Hausdorff and conjunctive reduction functions are not modified in the proof of Theorem 3.9 we immediately obtain the following corollary.

Corollary 3.10 *For every set $A \in R^p_{bhd}(R^p_c(\mathrm{SPARSE}))$ there is a sparse set $S' \in \mathrm{NP}(A)$ such that $A \in R^p_{bhd}(R^p_c(S'))$.*

Since by Theorem 2.9 $R^p_{bhd}(R^p_c(\mathrm{SPARSE})) = R^p_{btt}(R^p_c(\mathrm{SPARSE}))$, and since every bounded Hausdorff reduction is also a bounded truth-table reduction we additionally have the following simplicity result for $R^p_{btt}(R^p_c(\mathrm{SPARSE}))$.

Corollary 3.11 *For every set $A \in R^p_{btt}(R^p_c(\mathrm{SPARSE}))$ there is a sparse set $S' \in \mathrm{NP}(A)$ such that $A \in R^p_{btt}(R^p_c(S'))$.*

Next we consider self-reducible sets in $R^p_{hd}(R^p_c(\mathrm{SPARSE}))$. In order to bound the complexity of such sets we exploit a certain minimality property which we now define.

Definition 3.12 *We say that a polynomially well-founded order $\prec$ is $\mathrm{FP}^{\mathrm{NP}}$-minimizable, if there exists an oracle $\mathrm{FP}^{\mathrm{NP}}$-transducer T such that for all oracles B and every non-negative integer i, T^B on input 0^i outputs a string x (if it exists) for which the following holds:*

1. $x \in B$ *and for all* $z \prec x$, $z \notin B$, *and*

2. *there exists a* $y \in \Sigma^i : \ x \preceq y$.

If such an x *does not exist then* $T^B(0^i)$ *outputs the special symbol* $\perp$.

Note that an $\mathrm{FP}^{\mathrm{NP}}$-computation with oracle B is done by an FP-transducer using a set A as oracle which is accepted by a fixed NP machine that uses oracle B. It is straightforward to prove that the lexicographic order is $\mathrm{FP}^{\mathrm{NP}}$-minimizable using a prefix search method. Using the census technique of [Kad87] as used in [LT91] it also follows that every polynomially related order is $\mathrm{FP}^{\mathrm{NP}}$-minimizable.

Proposition 3.13 *The lexicographic order and every polynomially related order are* $\mathrm{FP}^{\mathrm{NP}}$*-minimizable.*

Definition 3.14 *A set* A *is* $\mathrm{FP}^{\mathrm{NP}}$*-minimizable self-reducible, if there are a polynomial time bounded oracle machine* M *and an* $\mathrm{FP}^{\mathrm{NP}}$*-minimizable order* $\prec$ *such that* $A = L(M, A)$, *and* M *on input* x *only asks queries* $q \prec x$ *to the oracle.*

Since by Proposition 3.13, the lexicographic order and all polynomially related orders are $\mathrm{FP}^{\mathrm{NP}}$-minimizable, we have that all word-decreasing and polynomially related self-reducible sets are $\mathrm{FP}^{\mathrm{NP}}$-minimizable self-reducible.

Since $R^p_{hd}(R^p_c(\mathrm{SPARSE})) = R^p_{nhd}(\mathrm{SPARSE})$ (see Theorem 2.4), instead of the composition of the Hausdorff and the conjunctive reducibilities we use in the following theorem the non-monotonic Hausdorff reducibility which is easier to handle.

Theorem 3.15 *For every* $\mathrm{FP}^{\mathrm{NP}}$*-minimizable self-reducible set* A *in* $R^p_{nhd}(\mathrm{SPARSE})$ *there exists an* P^{NP}*-printable sparse set* S' *such that* $A \leq^p_{nhd} S'$.

Proof Let A be an $\mathrm{FP}^{\mathrm{NP}}$-minimizable self-reducible set, witnessed by the self-reduction machine M and the $\mathrm{FP}^{\mathrm{NP}}$-minimizable order $\prec$, and let T be the $\mathrm{FP}^{\mathrm{NP}}$ oracle transducer computing the minimal elements. Let $A \leq^p_{nhd} S$ via k and f for a sparse set S. Let r be a strictly increasing polynomial such that for all x it holds that $f(x)$ is a list of $k(|x|)$ oracle queries, each query of length $r(|x|)$. For any set B, let

$$j(x, B) = \max\{j \mid 0 \leq j \leq k(|x|) \text{ and for all } i = 1, \ldots, j : f(x, i) \in B\}.$$

Note that for all x, $x \in A$ iff $j(x, S)$ is odd. Similar to the proof of Theorem 3.9 we describe a P^{NP} printing algorithm for a sparse subset $\hat{S}$ of S to which A reduces. We first define below an oracle Turing machine M' that simulates M, answering each self-reducing query q made by M as follows: M' uses its oracle, say O, to compute $j(q, O)$ and continues the simulation of M with answer "yes" if $j(q, O)$ is odd, and with answer "no" otherwise.

input x
simulate M on input x, but whenever M
queries q to the oracle then do the following:
if $j(q, O)$ is odd
then continue with answer "yes"
else continue with answer "no"
end
accept if and only if M accepts

It is clear that M' runs in polynomial time, and that $A = L(M', S)$. For every finite set B of strings, let err_B denote the set

$$err_B = \{x \mid x \in L(M', B) \iff j(x, B) \text{ is odd}\}.$$

Since $\prec$ is $\mathrm{FP}^{\mathrm{NP}}$-minimizable, it is easy to see that there exists an $\mathrm{FP}^{\mathrm{NP}}$ transducer T which on input $(0^i, B)$ outputs a string x (if it exists) with the following properties:

1. $x \in err_B$ and for all $z \prec x$, $z \notin err_B$, and
2. $x \in \Sigma^i$ or there exists a $y \in \Sigma^i :\ x \prec y$.

Otherwise $T(0^i, B) = \perp$.

Now we are ready to give the P^{NP} printing algorithm for $\hat{S}$. The function CONSTRUCT defined below on input 0^i outputs a finite subset Q of S to which all length i instances of A are correctly reduced by f.

function CONSTRUCT
input 0^i
$Q := \emptyset$
while $T(0^i, Q) \neq \perp$ **do**
$x := T(0^i, Q)$
$Q := Q \cup \{f(x, j(x, Q) + 1)\}$
end
return Q

The following claims establish the correctness of function CONSTRUCT.

Claim 7 *For all $x \in \Sigma^*$ and $B \subseteq \Sigma^*$: if for all y with $y \prec x$ it holds that $y \in L(M', B)$ iff $j(y, B)$ is odd, then it follows that $x \in L(M', B) \iff x \in A$.*

Proof Let p be a polynomial such that $x \prec y$ implies $|x| \leq p(|y|)$. Since $\prec$ is transitive and irreflexive, each increasing chain ending with a string x has length bounded by $2^{p(|x|)+1}$. The proof of the claim is by induction on the maximal length l_x of increasing chains ending with x. For the base case, if $l_x = 1$ then $M(x)$ (and therefore $M'(x)$) correctly decides without queries. Now, assume that the claim holds for all z such that $l_z < l_x$. From this it follows that for all $z \prec x$, $j(z, B)$ is

odd $\iff z \in A$. Since M on input x only queries the oracle for strings $z \prec x$, it follows that $x \in L(M', B) \iff x \in A$, which completes the proof. □

Let Q_n be the value of Q in the computation of CONSTRUCT(0^i) after the n-th iteration of the while-loop, and let $x_n = T(0^i, Q_{n-1})$. From Claim 7 it follows immediately that

$$x_n \in A \text{ iff } x_n \in L(M', Q_{n-1}).$$

Using this equivalence, the next claim can be similarly proved as Claim 5 in the proof of Theorem 3.9.

Claim 8 *Q_n is a subset of $S^{\leq r(p(i))}$.*

Claim 9 *The function CONSTRUCT is in* $\mathrm{FP}^{\mathrm{NP}}$.

Proof Since by Claim 8, CONSTRUCT$(0^i) \subseteq S^{\leq r(p(i))}$, and since in every iteration of the while-loop a new element is included in Q, the number of iterations of the loop is bounded by $|S^{\leq r(p(i))}|$. Since T is an $\mathrm{FP}^{\mathrm{NP}}$ transducer, the claim follows. □

Claim 10 *For all instances of length i, $A \leq^p_{nhd} CONSTRUCT(0^i) \cap \Sigma^{r(i)}$ via k and f.*

Proof When the while-loop is exited it holds that $T(0^i, \text{CONSTRUCT}(0^i)) = \perp$. This means that for all $x \in \{x \mid x \in \Sigma^i \text{ or there exists a } y \in \Sigma^i : \ x \prec y\}$,

$$x \in L(M', \text{CONSTRUCT}(0^i)) \iff j(x, \text{CONSTRUCT}(0^i)) \text{ is odd.}$$

By Claim 7 it holds for all $x \in \Sigma^i$ that $x \in L(M', \text{CONSTRUCT}(0^i)) \iff x \in A$, implying that for all $x \in \Sigma^i$, $x \in A \iff j(x, \text{CONSTRUCT}(0^i))$ is odd. □

Define $\hat{S} = \bigcup_{i \geq 0}(\text{CONSTRUCT}(0^i) \cap \Sigma^{r(i)})$. Then $\hat{S}$ is sparse by Claim 8. As a consequence of Claim 9, $\hat{S}$ is P^{NP}-printable. Finally, by Claim 10 we have that $A \leq^p_{nhd} \hat{S}$ via k and f. □

Corollary 3.16 *For every word-decreasing or polynomially related self-reducible set $A \in R^p_{nhd}(S)$, S sparse, there exists a P^{NP}-printable sparse set $S' \subseteq S$ such that $A \leq^p_{nhd} S'$.*

3.3 Disjunctive reductions

For the reduction classes considered up to now we were able to construct a simple description for a set A as the intersection of all sets to which A reduces via a given reduction function. Hence, the simple description was the minimal sparse set to which A would correctly reduce via the considered reduction. For sets which disjunctively reduce to sparse sets it appears that sparse d-descriptions are relatively harder to compute.

We show that every set A in $R^p_d(\mathrm{SPARSE})$ has a sparse d-description in $\mathrm{NP}(A \oplus \mathrm{SAT})/\mathrm{F}\Theta^p_2(A)$. Furthermore we show that there exists a set in $\mathrm{NP}(A)/\mathrm{F}\Theta^p_2(A)$ such that every element $\langle 0^n, W\rangle$ in this set encodes a finite set W to which A is disjunctively reducible with respect to strings of length n, i.e., $A^{=n} \leq^p_d W$ via the given disjunctive reduction function (cf. the notion of $\mathrm{CIR}(A)$ [KoSc85]).

In [AHH+93] it is proved that for every set A in $R^p_d(\mathrm{SPARSE})$ there exists a sparse set $S \in \Delta^p_2(A)$ such that $A \leq^p_d S$ (Theorem 3.2). Our theorem below improves this result, and the proof technique is a refinement of the one used in [AHH+93] to prove Theorem 3.2.

Theorem 3.17 *Let A be a set in $R^p_d(\mathrm{SPARSE})$ witnessed by the sparse set S and the reduction function f. Then*

1. *there exists a set $C \subseteq \{\langle 0^n, W\rangle \mid A \leq^p_d W$ via f for inputs of length $n\}$ and a polynomial p such that $C \in \mathrm{NP}(A)/\mathrm{F}\Theta^p_2(A)$ and for every n there is at least one pair of the form $\langle 0^n, W\rangle$ in $C^{\leq p(n)}$,*
2. *there exists a sparse set $\hat{S} \in \mathrm{NP}(A \oplus \mathrm{SAT})/\mathrm{F}\Theta^p_2(A)$ such that $A \leq^p_d \hat{S}$ via f.*

Proof By Lemma 2.11 we can assume that A disjunctively reduces to a sparse set via a reduction function f such that for all x and $z \in f(x)$, $|z| = r(|x|)$ and $|f(x)| \leq r(|x|)$, where r is a strictly increasing polynomial. Then it holds for all x that $x \in A$ if and only if $f(x) \cap S^{=r(|x|)} \neq \emptyset$. We construct sparse sets $S_1 \subseteq A$ and $S_2 \subseteq \overline{A}$ which define a sparse set $\hat{S} = \bigcup_{y \in S_1} f(y) - \bigcup_{y \in S_2} f(y)$ such that for all x, $x \in A$ if and only if $f(x) \cap \hat{S} \neq \emptyset$. Consider the set L defined by

$$\begin{aligned}
&\langle 0^n, \langle x_1, \ldots, x_j\rangle, \langle y_1, \ldots, y_k\rangle\rangle \in L \;\Leftrightarrow \\
&\quad \text{for all } l, 1 \leq l \leq j,\; x_l \in A^{=n} \wedge f(x_l) \cap \left(\bigcup_{1 \leq i < l} f(x_i) - \bigcup_{i=1}^{k} f(y_i)\right) = \emptyset \\
&\quad \text{and} \\
&\quad \text{for all } l, 1 \leq l \leq k,\; y_l \in \overline{A}^{=n} \wedge f(y_l) \cap \left(\bigcup_{i=1}^{j} f(x_i) - \bigcup_{1 \leq i < l} f(y_i)\right) \neq \emptyset
\end{aligned}$$

Clearly L is in $\mathrm{P}(A)$. For $\langle 0^n, \langle x_1, \ldots, x_j\rangle, \langle y_1, \ldots, y_k\rangle\rangle$ in L the set $\bigcup_{i=1}^{k} f(y_i)$ is a subset of $\overline{S}$ and therefore it holds for every l, $1 \leq l \leq k$, that $f(x_l) \cap \bigcup_{1 \leq i < l} f(x_i) \cap S = \emptyset$, i.e., $f(x_l)$ generates at least one query in S that is not covered by any of the sets $f(x_1), \ldots, f(x_{l-1})$. The advice that we use are the values of the following two functions,

$$j(n) = \max\{j \mid \exists k\, \exists x_1, \ldots, x_j, y_1, \ldots, y_k :\; \langle 0^n, \langle x_1, \ldots, x_j\rangle, \langle y_1, \ldots, y_k\rangle\rangle \in L\}$$

$$k(n) = \max\{k \mid \exists x_1, \ldots, x_{j(n)}, y_1, \ldots, y_k :\; \langle 0^n, \langle x_1, \ldots, x_{j(n)}\rangle, \langle y_1, \ldots, y_k\rangle\rangle \in L\}$$

Claim 11 *For all n it holds that $j(n) \leq census_S(1^{r(n)})$ and $k(n) \leq r(n) \cdot census_S(1^{r(n)})$.*

Proof of Claim 11: Let $\langle 0^n, \langle x_1, \ldots, x_{j(n)} \rangle, \langle y_1, \ldots, y_k \rangle \rangle \in L$. As noted above, it holds for every l, $1 \leq l \leq j(n)$, that $f(x_l) \cap \bigcup_{1 \leq i < l} f(x_i) \cap S = \emptyset$. Since $f(x_i) \cap S \neq \emptyset$, $1 \leq i \leq j(n)$, it follows that

$$census_S(1^{r(n)}) \geq \left| \bigcup_{1 \leq i \leq j(n)} f(x_i) \cap S \right| = \sum_{i=1}^{j(n)} |f(x_i) \cap S| \geq j(n)$$

This proves the first part of the claim. The second part follows immediately by the definition of L since for all x, $|f(x)| \leq r(|x|)$, i.e., $|\bigcup_{1 \leq i \leq j(n)} f(x_i)| \leq j(n) \cdot r(n)$. □

Since $r(n)$ is a polynomial and $census_S(1^n)$ is bounded by some polynomial the functions $j(n)$ and $k(n)$ are also polynomially bounded, and the following claim can be easily proved.

Claim 12 *On input 0^n, $j(n)$ and $k(n)$ can be computed by an $\mathrm{F}\Theta_2^p(A)$ computation.*

Every element $z = \langle 0^n, \langle x_1, \ldots, x_{j(n)} \rangle, \langle y_1, \ldots, y_{k(n)} \rangle \rangle$ in L contains enough information to construct a set S_z containing at most $r(n) \cdot j(n)$ elements such that $A^{=n} \leq_d^p S_z$ via f.

Claim 13 *Let $\langle 0^n, \langle x_1, \ldots, x_{j(n)} \rangle, \langle y_1, \ldots, y_{k(n)} \rangle \rangle \in L$. Then for all $x \in \Sigma^n$ it holds that $x \in A$ if and only if $f(x) \cap (\bigcup_{i=1}^{j(n)} f(x_i) - \bigcup_{i=1}^{k(n)} f(y_i)) \neq \emptyset$.*

Proof of Claim 13: In order to derive a contradiction let $x \in \Sigma^n$ and assume that $x \in A$ but $f(x) \cap (\bigcup_{i=1}^{j(n)} f(x_i) - \bigcup_{i=1}^{k(n)} f(y_i)) = \emptyset$. Then it follows that the string $\langle 0^n, \langle x_1, \ldots, x_{j(n)}, x \rangle, \langle y_1, \ldots, y_{k(n)} \rangle \rangle$ is contained in L which contradicts the maximality of $j(n)$. To show the converse assume that $x \notin A$ and $f(x) \cap (\bigcup_{i=1}^{j(n)} f(x_i) - \bigcup_{i=1}^{k(n)} f(y_i)) \neq \emptyset$. Then $\langle 0^n, \langle x_1, \ldots, x_{j(n)} \rangle, \langle y_1, \ldots, y_{k(n)}, x \rangle \rangle \in L$ contradicting the maximality of $k(n)$. □

Now consider the sets C and C' defined by

$$\langle 0^n, W, j, k \rangle \in C' \Leftrightarrow \exists \langle 0^n, \langle x_1, \ldots, x_j \rangle, \langle y_1, \ldots, y_k \rangle \rangle \in L : \; W = \bigcup_{i=1}^{j} f(x_i) - \bigcup_{i=1}^{k} f(y_i)$$

and

$$\langle 0^n, W \rangle \in C \Leftrightarrow \langle 0^n, W, j(n), k(n) \rangle \in C'$$

Clearly, $C' \in \mathrm{NP}(A)$, and therefore $C \in \mathrm{NP}(A)/\mathrm{F}\Theta_2^p(A)$. By Claim 13 it holds for every $\langle 0^n, W \rangle \in C$ that $A^{=n} \leq_d^p W$ via f. Furthermore, by the definitions of L and C it follows for all $\langle 0^n, W \rangle \in C$ that $W \subseteq \Sigma^{r(n)}$ and $|W| \leq r(n) \cdot j(n)$. This proves the first part of the theorem.

In order to prove the second part we select for every n the lexicographically smallest pair $\langle 0^n, W_n \rangle$ in C and define the desired sparse set $\hat{S}$ as the union of all

the W_n, $n \geq 0$. Consider the NP set D defined by

$\langle 0^n, W, j, k\rangle \in D \Leftrightarrow$
there exists a string $\langle 0^n, \langle x_1, \ldots, x_j\rangle, \langle y_1, \ldots, y_k\rangle\rangle$ such that
for all $l, 1 \leq l \leq j$, $f(x_l) \cap W \neq \emptyset \ \wedge \ f(x_l) \cap (\bigcup_{1\leq i<l} f(x_i) - \bigcup_{i=1}^{k} f(y_i)) = \emptyset$,
for all $l, 1 \leq l \leq k$, $f(y_l) \cap W = \emptyset \ \wedge \ f(y_l) \cap (\bigcup_{i=1}^{j} f(x_i) - \bigcup_{1\leq i<l} f(y_i)) \neq \emptyset$,
and $\langle 0^n, \bigcup_{i=1}^{j} f(x_i) - \bigcup_{i=1}^{k} f(y_i)\rangle < \langle 0^n, W\rangle$

Claim 14 *A string $\langle 0^n, W, j(n), k(n)\rangle \in C'$ is contained in D if and only if there exists a string $\langle 0^n, W'\rangle \in C$ which is lexicographically smaller than $\langle 0^n, W\rangle$.*

Proof of Claim 14: Let $\langle 0^n, W, j(n), k(n)\rangle$ be a string in C'. Then there exists a string $\langle 0^n, \langle x'_1, \ldots, x'_{j(n)}\rangle, \langle y'_1, \ldots, y'_{k(n)}\rangle\rangle \in L$ such that $W = \bigcup_{i=1}^{j(n)} f(x'_i) - \bigcup_{i=1}^{k(n)} f(y'_i)$. Thus it follows by Claim 13 that for all x of length n, $x \in A$ if and only if $f(x) \cap W \neq \emptyset$, and the claim follows by the definition of D. □

An immediate consequence of Claim 14 is

Claim 15 *For every n there is exactly one element of the form $\langle 0^n, W, j, k\rangle$ in $C' - D$.*

Now we are ready to define the sparse set $\hat{S} = \bigcup_n \{W \mid \langle 0^n, W, j(n), k(n)\rangle \in C' - D\}$. For $r(n) = |w|$ the following algorithm accepts an input $\langle w, j(n), k(n)\rangle$ if and only if w is in $\hat{S}$.

input $\langle w, j, k\rangle$
guess $n \leq |w|$
if $|w| \neq r(n)$ **then** *reject* **end**
guess $x_1, \ldots, x_j, y_1, \ldots, y_k \ \in \Sigma^n$
if $\langle 0^n, \langle x_1, \ldots, x_j\rangle, \langle y_1, \ldots, y_k\rangle\rangle \notin L$ **then** *reject* **end**
$W := \bigcup_{i=1}^{j} f(x_i) - \bigcup_{i=1}^{k} f(y_i)$
if $\langle 0^n, W, j, k\rangle \notin D$ **and** $w \in W$ **then** *accept* **else** *reject* **end**

Since the second if-statement queries oracle L (which is in $\mathrm{P}(A)$), the third if-statement queries oracle D (which is in NP) and since the advice (namely $j(n)$ and $k(n)$) can be computed in $\mathrm{F}\Theta_2^p(A)$ it follows that $\hat{S} \in \mathrm{NP}(A \oplus \mathrm{SAT})/\mathrm{F}\Theta_2^p(A)$. □

The next theorem is a simplicity result for $R^p_{1\text{-}tt}(R^p_d(\mathrm{SPARSE}))$ which follows quite easily from the previous theorem.

Theorem 3.18 *For every set A in $R^p_{1\text{-}tt}(R^p_d(\mathrm{SPARSE}))$ there is a sparse set S' in $\mathrm{NP}(A \oplus \mathrm{SAT})/\mathrm{F}\Theta_2^p(A)$ such that $A \in R^p_{1\text{-}tt}(R^p_d(S'))$.*

Proof For a set A which 1-tt-reduces to a set B, Σ^* can be splitted into two disjoint parts $C_1, C_2 \in \mathrm{P}$ such that $A \cap C_1 \leq_m^p B$ and $\overline{A \cap C_2} \leq_m^p B$. Assume that $B \in R_d^p(\mathrm{SPARSE})$. Then there exist sparse sets $S_1, S_2 \in \mathrm{NP}(A \oplus \mathrm{SAT})/\mathrm{F}\Theta_2^p(A)$ such that $A \cap C_1 \leq_d^p S_1$ and $\overline{A \cap C_2} \leq_d^p S_2$. Since C_1 and C_2 are in P, combining both reductions yields that $A \in R_{1\text{-}tt}^p(R_d^p(S_1 \oplus S_2))$, and since SPARSE is closed under join the theorem follows. □

We now turn to the complexity of simple descriptions for self-reducible sets in $R_d^p(\mathrm{SPARSE})$.

Theorem 3.19 *For every* $\mathrm{FP}^{\mathrm{NP}}$*-minimizable self-reducible set* A *in* $R_d^p(\mathrm{SPARSE})$ *there exists an* P^{NP}*-printable sparse set* S' *such that* $A \leq_d^p S'$.

Proof Let $A \leq_d^p S$ via f, and let r and t be polynomials such that $f(x) \subseteq \Sigma^{r(|x|)}$ and $|f(x)| \leq t(|x|)$ for all $x \in \Sigma^*$ (see Lemma 2.11). Let M be a self-reduction machine for A w.r.t. some $\mathrm{FP}^{\mathrm{NP}}$-minimizable order $\prec$. Consider the following oracle machine M' that uses its oracle O to simulate M in the following way.

input x
simulate M on input x, but whenever M
queries q to the oracle then do the following:
  if $f(q) \cap O \neq \emptyset$
    then continue with answer "yes"
    else continue with answer "no"
  end
accept if and only if M accepts

For every finite set B of strings, let err_B denote the set

$$err_B = \{x \mid x \in L(M', B) \iff f(x) \cap B \neq \emptyset\}.$$

Since $\prec$ is $\mathrm{FP}^{\mathrm{NP}}$-minimizable, there exists an $\mathrm{FP}^{\mathrm{NP}}$ oracle transducer T which on input $\langle 0^i, B\rangle$ computes a string x (if it exists) with the following properties:

1. $x \in err_B$ and for all $z \prec x$, $z \notin err_B$, and
2. $x \in \Sigma^i$ or there exists a $y \in \Sigma^i : \ x \prec y$.

Otherwise $T(0^i, B) = \perp$.

Using the oracle machine M' defined above, we now define a function CONSTRUCT that computes on input 0^i a set Q such that $A \leq_d^p Q$ via f for all instances up to length i.

function CONSTRUCT
  input 0^i
  $Q := \emptyset$

while $T(0^i, Q) \neq \perp$ **do**
  $x := T(0^i, Q)$
  if $x \in L(M', Q)$
    then $Q := Q \cup f(x)$
    else $Q := Q - f(x)$
  end
return Q

We establish through the following claims that A correctly reduces for instances upto length i via the disjunctive reduction f to the set CONSTRUCT(0^i).

The next claim can be proved exactly as Claim 7 in the proof of Theorem 3.15.

Claim 16 *For all $x \in \Sigma^*$ and $Q \subseteq \Sigma^*$: if for all y with $y \prec x$ it holds that $y \in L(M', Q) \iff f(y) \cap Q \neq \emptyset$, then it follows that $x \in L(M', Q) \iff x \in A$.*

Let Q_n be the value of Q in the computation of CONSTRUCT(0^i) after the n-th iteration of the while-loop, and let $x_n = T(0^i, Q_{n-1})$. The next claim follows from Claim 16.

Claim 17 *$x_n \in A$ iff $x_n \in L(M', Q_{n-1})$.*

Claim 18 *If $x_n \in A$, then $Q_{n-1} \cap S \subsetneq Q_n \cap S$.*

Proof Assume $x_n \in A$. Then it holds that $f(x_n) \cap S \neq \emptyset$. Since $x_n \in L(M', Q_{n-1})$ (by Claim 17) it follows that $Q_n = Q_{n-1} \cup f(x_n)$. Since $x_n \in err_{Q_{n-1}}$, we get that $f(x_n) \cap Q_{n-1} = \emptyset$, and therefore $Q_{n-1} \cap S \subsetneq (Q_{n-1} \cup f(x_n)) \cap S$. □

Claim 19 *If $x_n \notin A$, then $Q_{n-1} \cap S = Q_n \cap S$ and $Q_n \subsetneq Q_{n-1}$.*

Proof Assume $x_n \notin A$. Then $f(x_n) \cap S = \emptyset$. By Claim 17 it holds that $x_n \notin L(M', Q_{n-1})$, and therefore $Q_n = Q_{n-1} - f(x_n)$ implying that $Q_{n-1} \cap S = (Q_{n-1} - f(x_n)) \cap S$. Since $x_n \in err_{Q_{n-1}}$, we have $f(x_n) \cap Q_{n-1} \neq \emptyset$, and the claim follows. □

The next claim is a direct consequence of Claims 18 and 19.

Claim 20 *For all n, $Q_{n-1} \cap S \subseteq Q_n \cap S$.*

Claim 21 *For all n, $|\{x_1, \ldots, x_n\} \cap A| \leq |S^{\leq r(i)}|$.*

Proof Since by Claim 18 every x_i in A puts at least one new element of $S^{\leq r(i)}$ into Q which, by Claim 20, is not removed from Q. The claim follows. □

Claim 22 *For all n, $\{x_{n+1}, \ldots, x_{n+|Q_n - S|+1}\} \cap A \neq \emptyset$.*

Proof Assume the contrary. If $\{x_{n+1}, \ldots, x_{n+|Q_n - S|+1}\} \subseteq \overline{A}$, then by Claim 19 it follows that $|Q_{n+|Q_n-S|+1}| \leq |Q_n| - (|Q_n - S|+1) < |Q_n \cap S|$. But then $Q_{n+|Q_n-S|+1} \cap S \subsetneq Q_n \cap S$, yielding a contradiction to Claim 20. □

From Claim 21 it follows for all n that $|Q_n| \leq |S^{\leq r(i)}| \cdot t(i)$, where t is the polynomial bounding the number of queries produced by f on inputs of length at most $p(i)$. From Claims 21 and 22, it follows that the number of iterations of the while-loop is at most $|S^{\leq r(i)}| \cdot t(i)$ which is polynomial in i. Therefore the set $\bigcup_{i \geq 0}(\text{CONSTRUCT}(0^i) \cap \Sigma^{r(i)})$ is sparse and P^{NP}-printable. Similar as in the proof of Theorem 3.15 it follows that for all instances in Σ^i, A correctly reduces via f to the set $\text{CONSTRUCT}(0^i) \cap \Sigma^{r(i)}$.

□

4 Simplicity results for non-deterministic reduction classes

We now consider nondeterministic reducibilities to sparse sets. The following theorem can be proved by easy modifications of the proofs of Theorems 3.6 and 3.7.

Theorem 4.1

1. *For every set* $A \in R_m^{np}(\text{TALLY}) \cap R_m^{co\text{-}np}(\text{TALLY})$ *there is a* $\text{P}(\text{SAT} \oplus A)$*-printable tally set* T' *such that* $A \in R_m^{np}(T') \cap R_m^{co\text{-}np}(T')$.

2. *Every polynomially well-founded self-reducible set in the class* $R_m^{np}(\text{TALLY}) \cap R_m^{co\text{-}np}(\text{TALLY})$ *is in* P^{NP}.

For sets in $R_m^{co\text{-}np}(\text{SPARSE})$ we have a simplicity result similar to sets in $R_c^p(\text{SPARSE})$.

Theorem 4.2 *For every set* $A \in R_m^{co\text{-}np}(\text{SPARSE})$ *there exists a sparse set* $S' \in R_m^{np}(A)$ *such that* $A \in R_m^{co\text{-}np}(S')$.

Proof Let S be a sparse set, and let M be an NP transducer witnessing $A \in R_m^{co\text{-}np}(S)$. By Lemma 2.11 there is a strictly increasing polynomial p such that for all output strings y of M on input x it holds that $|y| = p(|x|)$. Consider the following subset S' of S.

$$S' = \{y \mid \exists x \in A : M \text{ on input } x \text{ outputs } y \}$$

Then $A \in R_m^{co\text{-}np}(S')$ via M. Furthermore, the following NP transducer witnesses $S' \leq_m^{np} A$.

input y
guess a string $x, |y| = p(|x|)$
guess a computation path
if $M(x)$ produces the output y on the guessed path
  then output x
  else output a fixed string not in A
end ■

By modifying the proof of Theorem 3.9 we can easily obtain a simplicity result for sets in the reduction class $R^p_{hd}(R^{co\text{-}np}_m(\mathrm{SPARSE}))$.

Theorem 4.3 *For every set* $A \in R^p_{hd}(R^{co\text{-}np}_m(\mathrm{SPARSE}))$ *there is a sparse set* $S' \in \mathrm{NP}(\mathrm{SAT} \oplus A)$ *such that* $A \in R^p_{hd}(R^{co\text{-}np}_m(S'))$.

Proof The proof is quite similar to that of Theorem 3.9 and therefore we omit most of the details. Let A be a set in $R^p_{hd}(R^{co\text{-}np}_m(S))$ for a sparse set S. We can combine the Hausdorff reduction function and the NP transducer witnessing the *co-np* many-one reduction to obtain a polynomial k and an NP transducer M such that for all x,

$$x \in A \iff \max\{i \mid 0 \le i \le k(|x|), \forall l, 1 \le l \le i : Out_M(l, x) \subseteq S\} \text{ is odd,}$$

where $Out_M(l, x)$ is the set of all output strings produced by M on input (l, x).

We now give an $\mathrm{NP}(\mathrm{SAT} \oplus A)$ algorithm that accepts a sparse set $\hat{S} \subseteq S$ such that A reduces to $\hat{S}$ via M and k.

input y
if $|y| \notin \{r(n) \mid n \ge 0\}$ **then** *reject* **end**
compute n such that $|y| = r(n)$
$T := \emptyset$
guess $c \in \{1, \ldots, p(r(n))\}$
guess $x_1, x_2, \cdots, x_c \in \Sigma^n$
for $j = 1$ **to** c **do**
  $m := \max\{i \mid 0 \le i \le k(n), \forall l, 1 \le l \le i : Out_M(l, x_j) \subseteq T\}$
  $T := T \cup \bigcup_{1 \le i \le m} Out_M(i, x_j)$
  if $x_j \in A \iff m$ is even
    then $T := T \cup Out_M(m + 1, x_j)$
  end
end for
if $y \in T$ **then** *accept* **else** *reject* **end**

The correctness of the above algorithm can be similarly proved as in the proof of Theorem 3.9. □

The following corollary is an immediate consequence of the proofs of Theorem 4.3 and Theorem 2.9.

Corollary 4.4 *For every set* $A \in R^p_{btt}(R^{co\text{-}np}_m(\mathrm{SPARSE}))$ *there is a sparse set* $S' \in \mathrm{NP}(\mathrm{SAT} \oplus A)$ *such that* $A \in R^p_{btt}(R^{co\text{-}np}_m(S'))$.

For the sake of completeness we mention the following result that can be proved along the lines of Theorem 3.15.

Theorem 4.5 *For every* $\mathrm{FP}^{\mathrm{NP}}$*-minimizable self-reducible language* A *in the class* $R^p_{hd}(R^{co\text{-}np}_m(\mathrm{SPARSE}))$ *there exists a* $\mathrm{P}^{\Sigma^p_2}$*-printable sparse set* S' *such that* $A \in R^p_{hd}(R^{co\text{-}np}_m(S'))$.

5 Lowness

We apply the simplicity results obtained in the earlier sections to prove new lowness and collapse results concerning reduction classes to sparse sets.

Theorem 5.1 *Every* $\mathrm{FP}^{\mathrm{NP}}$*-minimizable self-reducible set* A *in* $R^p_{nhd}(\mathrm{SPARSE})$ *and in* $R^p_d(\mathrm{SPARSE})$ *is low for* Δ^p_2*, i.e.* $\Delta^p_2(A) = \Delta^p_2$.

Proof By Theorems 3.15 and 3.19 we have that for every $\mathrm{FP}^{\mathrm{NP}}$-minimizable self-reducible set A in $R^p_{nhd}(\mathrm{SPARSE})$ (resp., $R^p_d(\mathrm{SPARSE})$) there exists a P^{NP}-printable set S such that A is in $R^p_{nhd}(S)$ (resp., $R^p_d(S)$). Thus, a $\Delta^p_2(A)$ computation can be simulated by a Δ^p_2 machine which first computes a suitable initial segment of S, and then simulates the original computation, answering every oracle query to A by reducing it to S. □

Lozano and Torán [LT91] show that in some relativized world there exists a word-decreasing self-reducible sparse set that is not low for Θ^p_2. Therefore, Theorem 5.1 cannot be improved to lowness for Θ^p_2 by relativizable proof techniques.

Since the NP-complete set SAT and the PSPACE-complete set QBF are polynomially related self-reducible, and since $\mathrm{C_=P}$ has a word-decreasing self-reducible complete set [OL91] we get the following collapse results.

Theorem 5.2

1. *If* SAT *reduces to a sparse set via a* $\leq^p_{nhd}$*- or* $\leq^p_d$*-reduction, then* $\mathrm{PH} = \Delta^p_2$.
2. *If* QBF *reduces to a sparse set via a* $\leq^p_{nhd}$*- or* $\leq^p_d$*-reduction, then* $\mathrm{PSPACE} = \Delta^p_2$.
3. *If every set in* $\mathrm{C_=P}$ *reduces to a sparse set via a* $\leq^p_{nhd}$*- or* $\leq^p_d$*-reduction, then* $\mathrm{CH} = \Delta^p_2$.

Next we apply the simplicity result of Theorem 4.2 to prove in Theorem 5.5 an extension of Kadin's result [Kad87] that every sparse NP set is low for Θ^p_2. For the proof we need the following technical lemmas.

Lemma 5.3 *For every sparse set S, $census_S \in \mathrm{F}\Theta(R_c^{np}(S))$.*

Proof The value of $census_S(1^n)$ can be computed by a binary search using the oracle set $C := \{\langle 0^n, 1^m\rangle \mid \text{there are at least } m \text{ different strings of length at most } n \text{ in } S\}$ which can easily be seen to be in $R_c^{np}(S)$. □

Lemma 5.4 *For every set A and any $k \geq 2$, $\Theta_k^p(\mathrm{SPARSE} \cap \mathrm{NP}(A)) \subseteq \Theta_2^p(\Sigma_{k-2}^p \oplus A)$.*

Proof Let B be a set in $\Theta_k^p(S)$ for a sparse set S in $\mathrm{NP}(A)$, and let M be a Θ machine deciding B relative to some oracle $C \in \Sigma_{k-1}^p(S)$ that is computed by a Σ_{k-1}^p oracle machine M'. Let p be a polynomial bounding the length of the queries to S in the $\Theta_k^p(S)$ computation of B. We outline how B can be computed in $\Theta_2^p(\Sigma_{k-2}^p \oplus A)$. On input x, at first $m = census_S(1^{p(|x|)})$ is computed by an $\mathrm{F}\Theta_2^p(A)$ computation, as described in Lemma 5.3. Then machine M is simulated on input x where each query y to C is replaced by the query $\langle y, 1^m, 1^{p(|x|)}\rangle$ to the $\mathrm{NP}(\Sigma_{k-2}^p \oplus A)$ set C' defined as

$$C' = \{\langle y, 1^i, 1^j\rangle \mid \exists\, y_1 < \cdots < y_i \in S^{\leq j} : y \in L(M', \{y_1, \ldots, y_i\})\}.$$

□

Theorem 5.5 *If $A \in \mathrm{NP}(\mathrm{NP} \cap \mathrm{SPARSE})$ and $A \leq_m^{co\text{-}np} S$ for a sparse set S then A is low for Θ_2^p, i.e. $\Theta_2^p(A) \subseteq \Theta_2^p$.*

Proof Let $A \in \mathrm{NP}(S')$ for a sparse set S' in NP and $A \leq_m^{co\text{-}np} S$ for a sparse set S. By Theorem 4.2 we can assume that $S \in R_m^{np}(A)$. Since $A \in \mathrm{NP}(S')$ and since $\mathrm{NP}(S')$ is closed under $\leq_m^{np}$-reductions it follows that $S \in \mathrm{NP}(S')$. This shows that $A \in \text{co-NP}(\mathrm{NP}(S') \cap \mathrm{SPARSE}) \cap \mathrm{NP}(S')$. Let $S'' \in \mathrm{NP}(S') \cap \mathrm{SPARSE}$ witness the above inclusion for A, i.e. $A \in \text{co-NP}(S'') \cap \mathrm{NP}(S')$. It is not hard to infer that $\mathrm{NP}(A) \subseteq \mathrm{NP}(S' \oplus S'')$. Hence, it follows that $\Theta_2^p(A) \subseteq \Theta_2^p(S' \oplus S'')$. Since $S' \oplus S''$ is in $\mathrm{NP}(S') \cap \mathrm{SPARSE}$ we can invoke Lemma 5.4 (for the case $k = 2$) to get $\Theta_2^p(S' \oplus S'') \subseteq \Theta_2^p(S')$. Since $S' \in \mathrm{NP} \cap \mathrm{SPARSE}$, we invoke Lemma 5.4 (for $k = 2$) once more to get $\Theta_2^p(S') \subseteq \Theta_2^p$. Combining the above inclusions yields that $\Theta_2^p(A) \subseteq \Theta_2^p$. □

As a corollary we can improve the result of Lozano and Torán [LT91] that every disjunctive self-reducible set that is many-one reducible to some sparse set is low for Θ_2^p. Since $\text{co-NP}/\log \subseteq R_m^{co\text{-}np}(\mathrm{TALLY})$ [AKM92] the corollary also improves all previously known Θ_2^p-lowness results for NP sets reducible to sparse or tally sets (e.g. for $\mathrm{NP} \cap R_m^p(\mathrm{SPARSE})$ and for $\mathrm{NP} \cap \text{co-NP}/\log$, cf. [LS91]).

Corollary 5.6

1. *If $A \in \mathrm{NP} \cap R_c^p(\mathrm{SPARSE})$ then A is low for Θ_2^p, i.e. $\Theta_2^p(A) \subseteq \Theta_2^p$.*

2. *If* $A \in \mathrm{NP} \cap R_m^{co\text{-}np}(\mathrm{SPARSE})$, *then* A *is low for* Θ_2^p, *i.e.* $\Theta_2^p(A) \subseteq \Theta_2^p$.

In order to classify sets outside the Polynomial Hierarchy, the notion of lowness was extended by Balcázar, Book, and Schöning [BBS86] who defined the extended low hierarchy. Allender and Hemachandra [AH92] and Long and Sheu [LS91] refined the extended low hierarchy and proved the optimality of the location of various classes therein.

Definition 5.7 [BBS86, AH92, LS91] *The* Σ, Δ, *and* Θ *levels of the extended low hierarchy (denoted* EL_k^Σ, EL_k^Δ, *and* EL_k^Θ, *respectively) are defined as below.*

1. $EL_k^\Sigma = \{A \mid \Sigma_k^p(A) \subseteq \Sigma_{k-1}^p(\mathrm{SAT} \oplus A)\},\ k \geq 1$
2. $EL_k^\Delta = \{A \mid \Delta_k^p(A) \subseteq \Delta_{k-1}^p(\mathrm{SAT} \oplus A)\},\ k \geq 2$
3. $EL_k^\Theta = \{A \mid \Theta_k^p(A) \subseteq \Theta_{k-1}^p(\mathrm{SAT} \oplus A)\},\ k \geq 2$

Many classes of sets reducible to sparse and tally sets have been shown to be included in the extended low hierarchy (see for example [BBS86, AH92, LS91]). It is interesting to see how the simplicity results immediately yield corresponding extended lowness results for the reduction classes. (For extended lowness results concerning the EL_3^Θ-level see the appendix.)

To derive the EL_1^Σ-lowness results we need the following theorem.

Theorem 5.8 *If* A *is in* $\mathrm{NP}(S) \cap \text{co-NP}(S)$ *for a sparse set* S *that is* $\mathrm{P}(\mathrm{SAT} \oplus A)$*-printable then* $A \in EL_1^\Sigma$.

Proof Let S be a $\mathrm{P}(\mathrm{SAT} \oplus A)$-printable set to which A reduces. We have to show that $\mathrm{NP}(A) \subseteq \mathrm{P}(\mathrm{SAT} \oplus A)$. The $\mathrm{NP}(A)$ computation can be simulated by the following $\mathrm{P}(\mathrm{SAT} \oplus A)$ computation: first compute a large enough initial segment of the sparse set S and then simulate the $\mathrm{NP}(A)$ computation where oracle queries to A are answered using the initial segment of S. Since the reduction mechanism from A to S is strong, the theorem follows. □

The class APT (almost polynomial time, see [MP79]) is easily seen to be contained in IC[log, poly], since it is in $R_m^p(\mathrm{SPARSE})$ and is closed under complement. Therefore the following corollary improves all previously known EL_1^Σ-lowness results regarding sets reducible to sparse or tally sets (e.g. for APT [LS91], for $E_T^p(\mathrm{TALLY})$[3] [BB86] and for $R_m^p(\mathrm{TALLY})$ [AH92]).

Corollary 5.9

1. $\mathrm{IC}[\log, \mathrm{poly}] \subseteq EL_1^\Sigma$
2. $R_m^{co\text{-}np}(\mathrm{TALLY}) \cap R_m^{np}(\mathrm{TALLY}) \subseteq EL_1^\Sigma$

[3] $E_T^p(\mathrm{TALLY})$ is the class of sets which are Turing equivalent to tally sets.

Proof Let A be a set in IC[log, poly]. By Theorems 3.6 and 3.4 there is a P(SAT $\oplus$ A)-printable tally set T such that $A \in R_c^p(T) \cap R_d^p(T)$. So, lowness follows by Theorem 5.8. The second part is proved similarly. □

Although we have simplicity results concerning nondeterministic reduction classes to sparse sets (e.g. Theorem 4.3) extended lowness is not a meaningful measure of lowness for such reduction classes since they contain either NP or co-NP. Nevertheless, the next theorem is a kind of lowness result. We show that the full power of a Θ_3^p computation relative to an oracle in $R_{hd}^p(R_m^{co\text{-}np}(\text{SPARSE}))$ is not needed in order to access the information contained in the oracle.

Theorem 5.10 *If* $A \in R_{hd}^p(R_m^{co\text{-}np}(\text{SPARSE}))$ *then* $\Theta_3^p(A) \subseteq \Theta_2^p(\Sigma_2^p \oplus A)$.

Proof By Theorem 4.3 we can assume that A is in $R_{hd}^p(R_m^{co\text{-}np}(S))$ for a sparse set $S \in \text{NP}(\text{SAT} \oplus A)$. Therefore, $\Theta_3^p(A) \subseteq \Theta_4^p(S) \subseteq \Theta_4^p(\text{NP}(\text{SAT} \oplus A) \cap \text{SPARSE})$, and by Lemma 5.4 (for the case $k = 4$) we get $\Theta_4^p(\text{NP}(\text{SAT} \oplus A) \cap \text{SPARSE}) \subseteq \Theta_2^p(\Sigma_2^p \oplus (\text{SAT} \oplus A))$. Now the theorem follows since $\Theta_2^p(\Sigma_2^p \oplus (\text{SAT} \oplus A)) \subseteq \Theta_2^p(\Sigma_2^p \oplus A)$. □

Conclusion

From the surprising result that P/*poly* is located optimally in EL_3^Θ [Köb93] and also that SPARSE is located optimally in EL_3^Θ [AH92], it is clear that extended lowness cannot be used to distinguish between different restricted reduction classes to sparse sets. The results of this paper give some evidence that the notion of simplicity is a better suited tool to study properties of reduction classes to sparse sets. In our opinion, the most significant of the applications of simplicity are the collapse results stated in Theorem 5.2.

In the following tables we summarize some of our results on simplicity and lowness.

For A in the reduction class	simplicity	lowness
$R_c^p(\text{TALLY}) \cap R_d^p(\text{TALLY})$	P(SAT $\oplus$ A)-printable	EL_1^Σ
$R_m^{np}(\text{TALLY}) \cap R_m^{co\text{-}np}(\text{TALLY})$	P(SAT $\oplus$ A)-printable	EL_1^Σ
$R_m^{co\text{-}np}(\text{SPARSE}) \cap \text{NP}^{\text{NP}\cap\text{SPARSE}}$	$R_m^{np}(A)$	$\Theta_2^p(A) \subseteq \Theta_2^p$
$R_{hd}^p(R_c^p(\text{SPARSE}))$	NP(A)	EL_3^Θ
$R_{1\text{-}tt}^p(R_d^p(\text{SPARSE}))$	NP(SAT $\oplus$ A)/F$\Theta_2^p(A)$	EL_3^Θ

For a self-reducible set A in the reduction class	simplicity	lowness
$R^p_c(\mathrm{TALLY}) \cap R^p_d(\mathrm{TALLY})$	NP	$\mathrm{NP}(A) \subseteq \mathrm{NP}$
$R^{np}_m(\mathrm{TALLY}) \cap R^{co\text{-}np}_m(\mathrm{TALLY})$	P(NP)-printable	$\Delta^p_2(A) \subseteq \Delta^p_2$
$R^p_{hd}(R^p_c(\mathrm{SPARSE}))$	P(NP)-printable	$\Delta^p_2(A) \subseteq \Delta^p_2$
$R^p_{1\text{-}tt}(R^p_d(\mathrm{SPARSE}))$	P(NP)-printable	$\Delta^p_2(A) \subseteq \Delta^p_2$

Acknowledgments

The first author is grateful to Uwe Schöning for his hospitality and for the research environment provided at Universität Ulm during the year 1991-92. The authors thank Ricard Gavaldà for helpful comments.

Appendix

The following theorems have been recently subsumed by the containment of P/*poly* in EL^{Θ}_3 [Köb93]. We include them in order to show how they can be obtained as direct consequences of the corresponding simplicity results.

Theorem 5.11 $R^p_{hd}(R^p_c(\mathrm{SPARSE})) \subseteq EL^{\Theta}_3$.

Proof Let A be in $R^p_{hd}(R^p_c(\mathrm{SPARSE}))$. Using Theorem 3.9 we can assume that $A \in R^p_{hd}(R^p_c(S))$ for a sparse set $S \in \mathrm{NP}(A)$. In order to prove the theorem we have to show that every set $B \in \Theta^p_3(A)$ is already contained in $\Theta^p_2(\mathrm{SAT} \oplus A)$. Since $A \in R^p_{hd}(R^p_c(S))$, it follows that $B \in \Theta^p_3(S)$. Since S is in $\mathrm{NP}(A)$ we can apply Lemma 5.4 (for the case $k = 3$) and obtain that B is in $\Theta^p_2(\mathrm{SAT} \oplus A)$. □

The next lowness proof uses the existence of simple sparse d-descriptions for sets in $R^p_d(\mathrm{SPARSE})$ (see Theorem 3.17). For the proof we need the following lemma.

Lemma 5.12 *If L is in $\Theta^p_k(\mathrm{SPARSE} \cap \mathrm{NP}(A \oplus \mathrm{SAT})/\mathrm{F}\Theta^p_2(A \oplus \mathrm{SAT}))$ for some set A and $k \geq 3$, witnessed by a Θ^p_k machine that on input x asks the sparse set only queries of length $p(|x|)$ for some fixed polynomial p, then L is in $\Theta^p_2(\Sigma^p_{k-2} \oplus A)$.*

Proof Let B be a set in $\Theta^p_k(S)$ for a sparse set S in $\mathrm{NP}(A \oplus \mathrm{SAT})/\mathrm{F}\Theta^p_2(A \oplus \mathrm{SAT})$, and let M be a Θ machine deciding B relative to some oracle $C \in \Sigma^p_{k-1}(S)$ that is computed by a Σ^p_{k-1} oracle machine M'. The proof is similar to the one of Lemma 5.4. The only difference is that now on input x the advice string for instances of S of length $p(|x|)$ and the cardinality of $S^{=p(n)}$ has to be precomputed by an $\mathrm{F}\Theta^p_2(A \oplus \mathrm{SAT})$ computation before the simulation of M on input x is started. □

Theorem 5.13 $R_d^p(\text{SPARSE}) \subseteq EL_3^{\Theta}$.

Proof Let A be in $R_d^p(\text{SPARSE})$. We have to show that $\Theta_3^p(A) \subseteq \Theta_2^p(A \oplus \text{SAT})$. Since A is many-one equivalent to the set $\{x10^i \mid i \geq 0 \wedge x \in A\}$ we can w.l.o.g. assume that all queries to A in the $\Theta_3^p(A)$ computation on input x are of length $p(|x|)$ for some fixed polynomial p. By Theorem 3.17 it follows that there is a sparse set $S \in \text{NP}(A \oplus \text{SAT})/\text{F}\Theta_2^p(A)$ such that A is in $R_d^p(S)$ via some function f, where all queries in $f(x)$ are of length $q(|x|)$ for some fixed polynomial q. Then it follows by Lemma 5.12 (for the case $k = 3$) that $\Theta_3^p(A) \subseteq \Theta_3^p(S) \subseteq \Theta_2^p(A \oplus \text{SAT})$. □

References

[AH92] E. Allender and L. Hemachandra. Lower bounds for the low hierarchy. *Journal of the ACM*, 39(1):234-250, 1992.

[AHOW92] E. Allender, L. Hemachandra, M. Ogiwara, and O. Watanabe. Relating equivalence and reducibility to sparse sets. *SIAM Journal on Computing*, 21(3):529-539, 1992.

[AHH+93] V. Arvind, Y. Han, L. Hemachandra, J. Köbler, A. Lozano, M. Mundhenk, M. Ogiwara, U. Schöning, R. Silvestri, and T. Thierauf. Reductions to sets of low information content. In *Complexity Theory, Current Research*, Cambridge University Press, 1–45, 1993.

[AKM92] V. Arvind, J. Köbler, and M. Mundhenk. On bounded truth-table, conjunctive, and randomized reductions to sparse sets. In *Proceedings 12th Conference on the Foundations of Software Technology & Theoretical Computer Science*, Lecture Notes in Computer Science, #652:140–151, Springer Verlag, 1992.

[AKM92b] V. Arvind, J. Köbler und M. Mundhenk. Lowness and the complexity of sparse and tally descriptions. *Proceedings Third International Symposium on Algorithm and Computation*, Lecture Notes in Computer Science, #650:249–258, Springer Verlag, 1992.

[AKM93] V. Arvind, J. Köbler, and M. Mundhenk. Hausdorff reductions to sparse sets and to sets of high information content. *Proceedings 18th Conference on Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science, #711:232–241, Springer Verlag, 1993.

[Ba92] J. Balcázar. Self-reducibility. *Journal of Computer and System Sciences*, 41:367-388, 1990.

[BB86] J. Balcázar and R. Book. Sets with small generalized Kolmogorov complexity. *Acta Informatica*, 23(6):679-688, 1986.

[BBS86] J.L. Balcázar, R. Book, and U. Schöning. Sparse sets, lowness and highness. *SIAM Journal on Computing*, 15:739-747, 1986.

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

[BLS93] H. Buhrman, L. Longpré, and E. Spaan. SPARSE reduces conjunctively to TALLY. In *Proceedings of the 8th Structure in Complexity Theory Conference*, IEEE Computer Society Press, 208–214, May 1993.

[GW93] R. Gavaldà and O. Watanabe. On the computational complexity of small descriptions. *SIAM Journal on Computing*, December 1993, to appear.

[Hau14] F. Hausdorff. *Grundzüge der Mengenlehre.* Leipzig, 1914.

[HY84] J. Hartmanis and Y. Yesha. Computation times of NP sets of different densities. *Theoretical Computer Science,* 34:17-32, 1984.

[Hem87] L. Hemachandra. The strong exponential hierarchy collapses. *Journal of Computer and System Sciences,* 39:299-322, 1989.

[Kad87] J. Kadin. $P^{NP[\log n]}$ and sparse Turing-complete sets for NP. *Journal of Computer and System Sciences,* 39(3):282-298, 1989.

[KL80] R. Karp and R. Lipton. Some connections between nonuniform and uniform complexity classes. *Proceedings of the 12th ACM Symposium on Theory of Computing,* 302-309, April 1980.

[Ko89] K. Ko. Distinguishing conjunctive and disjunctive reducibilities by sparse sets. *Information and Computation*, 81(1):62-87, 1989.

[Ko83] K. Ko. On self-reducibility and weak p-selectivity. *Journal of Computer and System Sciences*, 26:209-221, 1983.

[KoSc85] K. Ko and U. Schöning. On circuit-size complexity and the low hierarchy in NP. *SIAM Journal on Computing,* 14:41-51, 1985.

[Köb93] J. Köbler. Locating P/poly optimally in the extended low hierarchy. *Proceedings of the 10th Annual Symposium on Theretical Aspects of Computer Science,* Lecture Notes in Computer Science, #665:28–37, Springer-Verlag, 1993.

[KSW87] J. Köbler, U. Schöning, and K.W. Wagner. The difference and truth-table hierarchies of NP. *Theoretical Informatics and Applications*, 21(4):419-435, 1987.

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

[LS91] T.J. Long and M.-J. Sheu. A refinement of the low and high hierarchies. Technical Report OSU-CISRC-2/91-TR6, The Ohio State University, 1991.

[LT91] A. Lozano and J. Torán. Self-reducible sets of small density. *Mathematical Systems Theory,* 24:83-100, 1991.

[Mah82] S. Mahaney. Sparse complete sets for NP: Solution of a conjecture of Berman and Hartmanis. *Journal of Computer and System Sciences*, 25(2):130-143, 1982.

[MP79] A. Meyer, M. Paterson. With what frequency are apparently intractable problems difficult? Tech. Report MIT/LCS/TM-126, Lab. for Computer Science, MIT, Cambridge, 1979.

[Mu94] M. Mundhenk. On self-reducible sets of low information content. *Proceedings of 2nd Italian Conference on Algorithms and Complexity,* February 1994, to appear.

[OKSW] P. Orponen, K. Ko, U. Schöning, and O. Watanabe. Instance complexity. *Journal of the ACM*, to appear.

[OL91] M. Ogiwara and A. Lozano. On one query self-reducible sets. *Theoretical Computer Science,* 112:255-276, 1993.

[OW91] M. Ogiwara and O. Watanabe. On polynomial-time bounded truth-table reducibility of NP sets to sparse sets. *SIAM Journal on Computing*, 20(3):471-483, 1991.

[Sal93] S. Saluja. Relativized limitations of the left set technique and closure classes of sparse sets. *Proceedings of the 8th Structure in Complexity Theory Conference*, 215-222. IEEE Computer Society Press, May 1993.

[Sch83] U. Schöning. A low hierarchy within NP. *Journal of Computer and System Sciences*, 27:14-28, 1983.

[Sch86] U. Schöning. *Complexity and Structure,* Lecture Notes in Computer Science, #211, Springer-Verlag, 1985

[TB91] S. Tang and R. Book. Reducibilities on tally and sparse sets. *Theoretical Informatics and Applications*, 25:293-302, 1991.

[Wag87] K.W. Wagner. More complicated questions about maxima and minima, and some closures of NP. *Theoretical Computer Science*, 51:53-80, 1987.

[Wag90] K.W. Wagner. Bounded query classes. *SIAM Journal on Computing*, 19(5):833-846, 1990.