Journal of Universal Computer Science, vol. 5, no. 12 (1999), 828-832
submitted: 3/10/99, accepted: 22/12/99, appeared: 28/12/99 © *Springer Pub. Co.*

On the Power of Positive Turing Reductions

Edith Hemaspaandra
(Department of Computer Science
Rochester Institute of Technology
Rochester, NY 14623
eh@cs.rit.edu)

Abstract: In the early 1980s, Selman's seminal work on positive Turing reductions showed that positive Turing reduction to NP yields no greater computational power than NP itself. Thus, positive Turing and Turing reducibility to NP differ sharply unless the polynomial hierarchy collapses.
We show that the situation is quite different for DP, the next level of the boolean hierarchy. In particular, positive Turing reduction to DP already yields all (and only) sets Turing reducibility to NP. Thus, positive Turing and Turing reducibility to DP yield the same class. Additionally, we show that an even weaker class, $\mathrm{P}^{\mathrm{NP}[1]}$, can be substituted for DP in this context.

Category: F.1
Key Words: computational complexity, NP, positive Turing reductions

1 Background and Definitions

A quarter century ago, Selman initiated the study of polynomial-time positive Turing reductions. A truth-table version of this reducibility had been introduced a few years earlier, by Ladner, Lynch, and Selman [LLS75]. Polynomial-time positive Turing reductions are defined as follows.

Let Σ will be any fixed alphabet having at least two letters. For specificity, in this paper we will take $\Sigma = \{0, 1\}$, but that is not essential. For any machine M, $L(M)$ denotes the set of strings accepted by machine M, and for any set A, $L(M^A)$ denotes the set of strings accepted by machine M running with oracle A. $A \leq_T^p B$ exactly if there is a polynomial-time Turing machine M such that $A = L(M^B)$.

Definition 1. 1. **[Sel82b,Sel82a]** We say that a Turing machine M is *positive* if

$$(\forall A, B \subseteq \Sigma^*)[A \subseteq B \Rightarrow L(M^A) \subseteq L(M^B)].$$

2. **[Sel82b]** Let A and B be sets ($A, B \subseteq \Sigma^*$). We say that A *positive Turing reduces to* B ($A \leq_{pos}^p B$) if there is a polynomial-time positive Turing machine M such that $A \leq_T^p B$ via M.

Since Selman's work, alternate definitions have been examined in some detail [HJ91], and positive reductions have been seen to play a role in a number of places in complexity theory. Most notably, Selman introduced them in the context of the P-selective sets, and to this day they continue to help in the investigation of those sets. Positive reductions have also been used to characterize the class of languages that can be "helped" by unambiguous sets [CHV93].

Henceforth, we will use "positive Turing reductions" as a shorthand for "polynomial-time positive Turing reductions." Selman's seminal work exactly pinpointed the power of positive Turing reductions to NP, namely, the class of languages that positive Turing reduce to NP is in fact NP itself. The class of languages that Turing reduce to NP, P^{NP}, is a strictly larger class than this, unless NP = coNP. So, assuming that the polynomial hierarchy does not collapse to NP, Turing reductions to NP are strictly more powerful than positive Turing reductions to NP.

In this paper, we study the power of positive Turing reductions to DP. DP was introduced by Papadimitriou and Yannakakis [PY84].

Definition 2. **[PY84]** A set C is in DP if there exist an NP set A and a coNP set B such that $C = A \cap B$.

DP is the next level beyond NP in the boolean hierarchy [CGH$^+$88], a structure that has been used in contexts ranging from approximation [Cha] to query order [HHW99]. DP, by definition, is simply the class of languages that are the intersection of an NP and a coNP set, though this class is quite robust and has many equivalent definitions. DP has natural complete problems (Graph Minimal Uncolorability [CM87] and many others [Wag87]), and plays a central role in the study of bounded access to NP, due to its central role in the key normal form for the boolean hierarchy, which turns out to be exactly the finite unions of DP sets [CGH$^+$88]. DP also plays a role in the study of which sets are P-compressible ([GHK92], see also [Wat93]).

Clearly, $\mathrm{NP} \subseteq \mathrm{DP} \subseteq \mathrm{P}^{\mathrm{NP}}$. Recall that Selman proved that positive Turing reductions to NP are surprisingly weak; they yield just the NP sets. In this paper, we prove that positive Turing reductions to DP are surprisingly strong; they yield all the P^{NP} sets. That is, they yield all the sets that can be computed via Turing reductions to NP (equivalently, via Turing reductions to DP).

We will note that our proof even establishes the same level of power for $\mathrm{P}^{\mathrm{NP}[1]}$, the class of languages computed by P machines making at most one query to an NP oracle.

2 On the Power of Positive Reductions to DP Sets

We now prove our main result. As is standard, for any class $\mathcal{C}$ and any reducibility r,

$$\mathrm{R}^p_r(\mathcal{C}) = \{L \mid (\exists L' \in \mathcal{C})[L \leq^p_r L']\},$$

that is, $\mathrm{R}^p_r(\mathcal{C})$ is the class of sets that r-reduce to sets in $\mathcal{C}$.

Theorem 3. $\mathrm{R}^p_{pos}(\mathrm{DP}) = \mathrm{P}^{\mathrm{NP}}$.

Proof: Clearly $\mathrm{R}^p_{pos}(\mathrm{DP}) \subseteq \mathrm{P}^{\mathrm{DP}} = \mathrm{P}^{\mathrm{NP}}$. So we have only to prove that $\mathrm{R}^p_{pos}(\mathrm{DP}) \supseteq \mathrm{P}^{\mathrm{NP}}$.

We will show that the standard P^{NP}-complete problem `OddMaxSat` [Kre88] is in $\mathrm{R}^p_{pos}(\mathrm{DP})$. `OddMaxSat` is the set of Boolean formulas ϕ whose variables are $x_1, x_2, \ldots, x_{numvars(\phi)}$ and whose lexicographically maximum satisfying assignment is odd. (Here, we are associating 0 with false, 1 with true, and the assignment $x_1 = b_1, x_2 = b_2, \ldots, x_{numvars(\phi)} = b_{numvars(\phi)}$, $b_i \in \{0,1\}$, with the integer $b_1 b_2 \cdots b_{numvars(\phi)}$.)

In order to prove this, we will construct a polynomial-time positive Turing machine M such that $L(M^{\mathtt{Sat}\oplus\overline{\mathtt{Sat}}}) = \mathtt{OddMaxSat}$. For any sets A and B, $A \oplus B$ denotes $\{x0 \mid x \in A\} \cup \{x1 \mid x \in B\}$. The construction is reminiscent of the proof that positive-truth-table reductions to tally sets are as strong as truth-table reductions to tally sets [BHL95].

Define M as follows. M will reject all strings that are not Boolean formulas ϕ whose variables are $x_1, x_2, \ldots, x_{numvars(\phi)}$. Suppose ϕ is a Boolean formula whose variables are $x_1, \ldots, x_n$.

Let $\phi_1 = \phi$. For $i := 1$ to n:

1. Query $\phi_i[x_i := 1]0$ and $\phi_i[x_i := 1]1$.
2. If the answer to $\phi_i[x_i := 1]0$ is "yes" and the answer to $\phi_i[x_i := 1]1$ is "no," then $\phi_{i+1} := \phi_i[x_i := 1]$. If $i = n$, then accept.
3. If the answer to $\phi_i[x_i := 1]0$ is "no" and the answer to $\phi_i[x_i := 1]1$ is "yes," then $\phi_{i+1} := \phi_i[x_i := 0]$. If $i = n$, then reject.
4. If the answer to both queries is "yes," then accept.
5. If the answer to both queries is "no," then reject.

Clearly, M runs in polynomial time. If we run M on input ϕ with oracle $\mathtt{Sat} \oplus \overline{\mathtt{Sat}}$, the answer to $\phi_i[x_i := 1]0$ is "yes" if and only if $\phi_i[x_i := 1] \in \mathtt{Sat}$ and the answer to $\phi[x_i := 1]1$ is "yes" if and only if $\phi_i[x_i := 1] \notin \mathtt{Sat}$. So, for each iteration, M will be in case 2 or 3, so that M will accept ϕ if and only if ϕ's lexicographically maximum satisfying assignment is odd. It follows that $L(M^{\mathtt{Sat}\oplus\overline{\mathtt{Sat}}}) = \mathtt{OddMaxSat}$. Since $\mathtt{Sat} \oplus \overline{\mathtt{Sat}} \in \mathrm{DP}$, it remains to show that M is positive.

Let C, D, E, F be such that $C \oplus D \subseteq E \oplus F$. Then $C \subseteq E$ and $D \subseteq F$. We will show that for any string x, if $M^{C\oplus D}$ accepts x, then $M^{E\oplus F}$ accepts x. This is immediate for strings x that are not Boolean formulas, because they are rejected no matter what. So, suppose that x is a Boolean formula ϕ with variables $x_1, \ldots, x_n$ and suppose for a contradiction that $M^{C\oplus D}$ accepts ϕ and $M^{E\oplus F}$ rejects ϕ.

Let i be the first iteration of the "for" loop such that $M^{C\oplus D}$ and $M^{E\oplus F}$ behave differently.

If $\phi_i[x_i := 1] \in E$ and $\phi_i[x_i := 1] \in F$ then $M^{E\oplus F}$ accepts, contradicting our assumption that ϕ is rejected by $M^{E\oplus F}$.

If $\phi_i[x_i := 1] \notin E$ and $\phi_i[x_i := 1] \notin F$, then $\phi_i[x_i := 1] \notin C$ and $\phi_i[x_i := 1] \notin D$, and $M^{C\oplus D}$ rejects, contradicting our assumption that ϕ is accepted by $M^{C\oplus D}$.

So, it must be the case that either $\phi_i[x_i := 1] \in E$ or $\phi_i[x_i := 1] \in F$, but not both. Since $M^{C\oplus D}$ and $M^{E\oplus F}$ behave differently at this stage, and since $C \subseteq E$ and $D \subseteq F$, it follows that $\phi_i[x_i := 1] \notin C$ and $\phi_i[x_i := 1] \notin D$. But this implies that $M^{C\oplus D}$ rejects, contradicting our assumption that ϕ is accepted by $M^{C\oplus D}$.

This concludes the proof that M is positive. So, M is a polynomial-time positive Turing machine such that $L(M^{\mathtt{Sat}\oplus\overline{\mathtt{Sat}}}) = \mathtt{OddMaxSat}$. Since $\mathtt{Sat} \oplus \overline{\mathtt{Sat}} \in \mathrm{DP}$, this implies that $\mathtt{OddMaxSat} \in \mathrm{R}^p_{pos}(\mathrm{DP})$, which implies that $\mathrm{R}^p_{pos}(\mathrm{DP}) \supseteq \mathrm{P}^{\mathrm{NP}}$, since $\mathtt{OddMaxSat}$ is complete for P^{NP}. $\square$

In fact, note that $\mathtt{Sat} \oplus \overline{\mathtt{Sat}}$ is not merely a DP set, but is even in $\mathrm{P}^{\mathrm{NP}[1]}$. Thus, as an immediate corollary to the proof, we can claim the following result

Theorem 4. $\mathrm{R}^{p}_{pos}(\mathrm{P}^{\mathrm{NP}[1]}) = \mathrm{P}^{\mathrm{NP}}$.

Earlier, we mentioned that Selman's positive Turing reductions were themselves inspired by the earlier notion of (polynomial-time) positive truth-table reductions (see [LLS75] for a detailed formal definition of any notions used without definition in this paragraph). The reader may wonder what the power of positive truth-table reductions to DP is. In fact, the answer to this is already implicit in the existing literature. Namely, it is known that $\mathrm{R}^{p}_{disjunctive\text{-}truth\text{-}table}(\mathrm{DP}) = \mathrm{R}^{p}_{truth\text{-}table}(\mathrm{NP})$ [HHR97,BH91]. So, we may immediately conclude that $\mathrm{R}^{p}_{truth\text{-}table}(\mathrm{NP}) = \mathrm{R}^{p}_{disjunctive\text{-}truth\text{-}table}(\mathrm{DP}) \subseteq \mathrm{R}^{p}_{positive\text{-}truth\text{-}table}(\mathrm{DP}) \subseteq \mathrm{R}^{p}_{truth\text{-}table}(\mathrm{NP})$. Thus,

$$\mathrm{R}^{p}_{positive\text{-}truth\text{-}table}(\mathrm{DP}) = \mathrm{R}^{p}_{truth\text{-}table}(\mathrm{NP}).$$

The class $\mathrm{R}^{p}_{truth\text{-}table}(\mathrm{NP})$, usually referred to as the "Θ^{p}_{2}" level of the polynomial hierarchy (see [Wag90]), has been extensively studied, and is widely believed to differ from P^{NP} (which obviously contains it). However, "$\Theta^{p}_{2} = \mathrm{P}^{\mathrm{NP}}$?" remains a major open research question, and it is not even known whether this equality implies the collapse of the polynomial hierarchy. From the main result of this paper, it is clear that "$\Theta^{p}_{2} = \mathrm{P}^{\mathrm{NP}}$?" can equivalently be stated as "$\mathrm{R}^{p}_{positive\text{-}truth\text{-}table}(\mathrm{NP}) = \mathrm{R}^{p}_{pos}(\mathrm{NP})$?"

Acknowledgments

For helpful conversations, suggestions, and comments, I am grateful to Lane Hemaspaandra, Harald Hempel, Jörg Rothe, Gerd Wechsung, and an anonymous referee. Research supported in part by the National Science Foundation under grants NSF-INT-9513368/DAAD-315-PRO-fo-ab and NSF-INT-9815095/DAAD-315-PPP-gü-ab. Work done in part while visiting Friedrich-Schiller-Universität Jena.

References

[BH91] S. Buss and L. Hay. On truth-table reducibility to SAT. *Information and Computation*, 91(1):86–102, 1991.

[BHL95] H. Buhrman, E. Hemaspaandra, and L. Longpré. SPARSE reduces conjunctively to TALLY. *SIAM Journal on Computing*, 24(4):673–681, 1995.

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

[Cha] R. Chang. Bounded queries, approximations, and the boolean hierarchy. *Information and Computation*. To appear.

[CHV93] J. Cai, L. Hemachandra, and J. Vyskoč. Promises and fault-tolerant database access. In K. Ambos-Spies, S. Homer, and U. Schöning, editors, *Complexity Theory*, pages 101–146. Cambridge University Press, 1993.

[CM87] J. Cai and G. Meyer. Graph minimal uncolorability is D^{P}-complete. *SIAM Journal on Computing*, 16(2):259–277, 1987.

[GHK92] J. Goldsmith, L. Hemachandra, and K. Kunen. Polynomial-time compression. *Computational Complexity*, 2(1):18–39, 1992.

[HHR97] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Raising NP lower bounds to parallel NP lower bounds. *SIGACT News*, 28(2):2–13, 1997.

[HHW99] L. Hemaspaandra, H. Hempel, and G. Wechsung. Query order. *SIAM Journal on Computing*, 28(2):637–651, 1999.

[HJ91] L. Hemachandra and S. Jain. On the limitations of locally robust positive reductions. *International Journal of Foundations of Computer Science*, 2(3):237–255, 1991.

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

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

[PY84] C. Papadimitriou and M. Yannakakis. The complexity of facets (and some facets of complexity). *Journal of Computer and System Sciences*, 28(2):244–259, 1984.

[Sel82a] A. Selman. Analogues of semirecursive sets and effective reducibilities to the study of NP complexity. *Information and Control*, 52:36–51, 1982.

[Sel82b] A. Selman. Reductions on NP and P-selective sets. *Theoretical Computer Science*, 19:287–304, 1982.

[Wag87] K. Wagner. More complicated questions about maxima and minima, and some closures of NP. *Theoretical Computer Science*, 51(1–2):53–80, 1987.

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

[Wat93] O. Watanabe. An NP set with no co-DP sparse subset. Technical Report TR-475, Department of Computer Science, University of Rochester, Rochester, NY, October 1993.