

Syracuse University

SURFACE

Electrical Engineering and Computer Science -
Technical Reports

College of Engineering and Computer Science

12-15-1991

A Note on Many-One and 1-Truth-Table Complete Languages

Steven Homer

Stuart A. Kurtz
University of Chicago

James S. Royer
Syracuse University

Follow this and additional works at: https://surface.syr.edu/eecs_techreports



Part of the [Computer Sciences Commons](#)

Recommended Citation

Homer, Steven; Kurtz, Stuart A.; and Royer, James S., "A Note on Many-One and 1-Truth-Table Complete Languages" (1991). *Electrical Engineering and Computer Science - Technical Reports*. 135.

https://surface.syr.edu/eecs_techreports/135

This Report is brought to you for free and open access by the College of Engineering and Computer Science at SURFACE. It has been accepted for inclusion in Electrical Engineering and Computer Science - Technical Reports by an authorized administrator of SURFACE. For more information, please contact surface@syr.edu.

SU-CIS-91-41

***A Note on Many-One and 1-Truth-Table
Complete Languages***

Steven Homer, Stuart Kurtz, and James Royer

December 16, 1991

*School of Computer and Information Science
Syracuse University
Suite 4-116, Center for Science and Technology
Syracuse, New York 13244-4100*

A Note on Many-One and 1-Truth-Table Complete Languages*

Steven Homer[†] Stuart Kurtz
Boston University University of Chicago

James Royer[‡]
Syracuse University

December 16, 1991

Abstract

The polynomial time 1-tt complete sets for EXP and RE are polynomial time many-one complete.

1 Introduction

Ladner, Lynch, and Selman [LLS75] showed that polynomial time one-one, many-one, truth-table, and Turing reducibilities differ on the exponential time computable sets. For example, there are 1-tt incomparable exponential time computable sets A and B that are 2-tt equivalent. Watanabe [Wat87] improved many of the Ladner-Lynch-Selman theorems by showing that essentially the same behavior occurs within the EXP complete sets¹ of the

*These results were presented by Steven Homer at the Fifth Annual IEEE Structure Complexity Conference (cf. [Hom90]).

[†]Supported in part by National Science Foundation grant CCR-8814339.

[‡]Supported in part by National Science Foundation grant CCR-89011154.

¹We use EXP to denote the class of sets computable in time $2^{p(n)}$ for some polynomial p ; although our results also hold for 2^{cn} .

weaker reducibility, specifically, that there are 2-tt complete sets for EXP that are 1-tt incomplete.

One of the Ladner-Lynch-Selman theorems that Watanabe didn't improve was the existence of 1-tt comparable, many-one incomparable sets. Based on Watanabe's experience with weaker reducibilities, it seemed plausible that there would be a 1-tt complete set that is not m-complete. On the other hand, Berman [Ber77] had shown that every m-complete set for EXP is 1-li complete, and so it was also plausible that there were no 1-tt complete, m-incomplete sets.

In this note, we show that every set that is 1-tt complete for EXP (resp. RE)² is also m-complete for EXP (resp. RE).

Perhaps the most interesting aspect of this result is the light it sheds on the Ko-Long-Du [KLD87] and Kurtz-Mahaney-Royer [KMR88] papers. Ko, Long, and Du show that the 2-tt complete degree for EXP contains a noncollapsing³ 1-li degree if and only if $P = UP$. Kurtz, Mahaney, and Royer show that the 2-tt complete degree for EXP contains a collapsing m-degree. In both cases, it seemed impossible to make the degree constructed complete for stronger reductions than 2-tt. It is now clear why. There are oracles relative to which the 1-li degree of EXP collapses⁴ and oracles relative to which the 1-li degree of EXP does not collapse⁵. As the 1-tt complete degree for EXP is now seen to be a 1-li degree, there can be no relativizing improvement of either the Ko-Long-Du or Kurtz-Mahaney-Royer theorems.

2 Mathematical Preliminaries

We assume familiarity with complexity theory (cf. [BDG88]), and structural complexity theory (cf. [KMR90]).

We identify sets with their characteristic functions: if A is a set, then $A(x) = 1$ means $x \in A$ and $A(x) = 0$ means $x \notin A$. We say that $\langle f_i \rangle_{i \in \Sigma^*}$ is a programming system for a class of functions \mathcal{C} if and only if $\mathcal{C} = \{f_i : i \in \Sigma^*\}$ and the function $\lambda i, x. f_i(x)$ is computable. The *recursion theorem* holds for $\langle f_i \rangle_{i \in \Sigma^*}$ if and only if for each “ f -program” i , there is an f -program e such

²RE denotes the set of recursively enumerable languages.

³A degree is *collapsing* if it consists of a single polynomial time isomorphism type.

⁴Any oracle that makes $P = UP$ suffices.

⁵E.g., a random oracle [KMR89].

that

$$f_e = \lambda x . f_i(\langle e, x \rangle). \quad (1)$$

Intuitively, e is a self-referential program that, on input x , generates a copy of its own program “text” e , builds $\langle e, x \rangle$, and runs the f -program i on this pair.

Let $\langle p_i \rangle_{i \in \Sigma^*}$ be a programming system for the polynomial time computable functions such that $\lambda i, x . p_i(x)$ is computable in time exponential in $|i| + |x|$ and such that the recursion holds for $\langle p_i \rangle_{i \in \Sigma^*}$. See Kozen [Koz80] or Royer-Case [RC87] for examples of such $\langle p_i \rangle_{i \in \Sigma^*}$.

A language A is 1-tt reducible to a language B if there is a function f from strings to expressions of the form **true**, **false**, $(y \in B)$, $(y \notin B)$ such that for all strings x , $x \in A$ if and only if $f(x)$ is a true assertion about B .

3 The 1-tt Complete Sets for EXP.

A precursor of the proof technique of the following theorem can be found in Ganesan and Homer [GH88].

Theorem 1 *The 1-tt complete languages for EXP are m-complete, i.e., the 1-tt complete degree for EXP is an m-degree.*

We give two versions of the proof of this theorem, one using a recursion theorem, and a second which avoids its (explicit) use.

First Proof of Theorem 1: Let E be an m-complete set for EXP. For each $i \in \Sigma^*$, define $A_i = \{x : p_i(x) \in E\}$. If we view EXP as a collection of characteristic functions, then $\langle A_i \rangle_{i \in \Sigma^*}$ is a programming system for EXP for which the recursion theorem holds. (See [KMR90, Page 115] for details.)

Let L be 1-tt complete for EXP. Then L is *uniformly* 1-tt complete: there is a programming system of 1-tt reductions $\langle t_i \rangle_{i \in \Sigma^*}$ such that for each $i \in \Sigma^*$, t_i is a 1-tt reduction of A_i to L and the function $\lambda i, x . t_i(x)$ is computable in exponential time.⁶ Moreover, we assume without any loss of generality

⁶*Proof:* Let t be a 1-tt reduction of E to L and for each i define $t_i = t \circ p_i$.

that for any 1-tt reduction t_i of a set A to L there is no x for which we have $t_i(x) = \mathbf{true}$ or $t_i(x) = \mathbf{false}$.⁷

It suffices to show that E is m-reducible to L . Let t be a 1-tt reduction of E to L . For each $x \in \Sigma^*$, one of the following cases holds.

Case 1. $t(x) = (y_x \in L)$, for some y_x , and so $x \in E \iff y_x \in L$.

Case 2. $t(x) = (y_x \notin L)$, for some y_x , and so $x \in E \iff y_x \notin L$.

On those x 's for which Case 1 holds, $x \mapsto y_x$ acts like an m-reduction of E to L ; and on those x 's for which Case 2 holds, $x \mapsto y_x$ acts like an m-reduction of \overline{E} to L . If Case 1 held for every x , we'd be done. Since we can't assume this, we have to deal with the "twists" introduced by t in Case 2. The idea of the proof is to use the recursion theorem to exhibit an e such that A_e is a version of E that undoes the twists. That is, for all x ,

$$A_e(x) = \begin{cases} E(x), & \text{if (i) } t_e(x) = (y_x \in L); \text{ and} \\ \overline{E}(x), & \text{if (ii) } t_e(x) = (y_x \notin L). \end{cases} \quad (2)$$

If (i) holds for x , then $x \in E \iff x \in A_e$ and $x \in A_e \iff y_x \in L$; hence, $x \in E \iff y_x \in L$. If (ii) holds for x , $x \in E \iff x \notin A_e$ and $x \in A_e \iff y_x \notin L$; hence, $x \in E \iff y_x \in L$.

Therefore, $x \mapsto y_x$ is an m-reduction for E to L as required.

□

The first proof used self-reference to construct an e that "knew" that t_e was a 1-tt reduction of A_e to L . At the price of some clarity, the second proof achieves the same effect and circumvents use of the recursion theorem (i.e., it contains just enough of the proof of the recursion theorem to get by).

Second Proof of Theorem 1: Let E , L and $\langle t_i \rangle_{i \in \Sigma^*}$ be as above. We again show that E is m-reducible to L by constructing an intermediate set which does the untwisting for us.

Define A by

$$A(\langle i, x \rangle) = \begin{cases} E(x), & \text{if (i) } t_i(\langle i, x \rangle) = (y_x \in L); \text{ and} \\ \overline{E}(x), & \text{if (ii) } t_i(\langle i, x \rangle) = (y_x \notin L). \end{cases} \quad (3)$$

⁷To eliminate these two cases, fix some $a \in L$. Interpret $t_i(x) = \mathbf{true}$ as $t_i(x) = (a \in L)$ and interpret $t_i(x) = \mathbf{false}$ as $t_i(x) = (a \notin L)$.

This A is easily seen to be computable in exponential-time. By the 1-tt completeness of L , there is a 1-tt reduction f of A to L . Let j be a t -index for f , i.e., $f = t_j$.

If (i) holds for x in (3), then $x \in E \iff \langle j, x \rangle \in A$ and $\langle j, x \rangle \in A \iff y_x \in L$; hence, $x \in E \iff y_x \in L$. If (ii) holds for x , then $x \in E \iff \langle j, x \rangle \notin A$ and $\langle j, x \rangle \in A \iff y_x \notin L$; hence, $x \in E \iff y_x \in L$.

Therefore, $x \mapsto y_x$ is an m-reduction for E to L as required.

□

Combining our Theorem 1 with Berman's [Ber77] theorem that the m-complete languages for EXP are 1-li complete yields:

Corollary 2 *The 1-tt complete for EXP languages are 1-li complete.*

With hindsight the coincidence of 1-tt and m-completeness for exponential time sets is not surprising. Both types of reductions allow one query to the oracle set, and Watanabe's theorems [Wat87] depend critically on the extra queries available to the weaker reducibilities. We also knew the corresponding result is true for r.e. sets with respect to recursive reductions, although the proof in the r.e. case doesn't generalize to subrecursive classes.

The moral of Theorem 1 is that 1-truth-table reductions should be categorized with the "strong" many-one, one-one and one length-increasing reductions, and not with weaker bounded truth-table reductions.

4 The 1-tt Complete Sets for RE.

We next prove the analogous result for r.e. sets, polynomial time 1-tt complete sets for RE are polynomial time m-complete.

The proof idea is similar Theorem 1. However, unlike EXP, RE is not closed under complementation. We cannot define an r.e. set A by an equation of the form $A(x) = \overline{K}(x)$. What we can do, if $t(x) = (y_x \notin L)$, is define $A(x) = L(y_x)$. The "twisted" case of Theorem 1 becomes a paradoxical case in Theorem 3.

Theorem 3 *The polynomial time 1-tt complete sets for r.e. are polynomial time m-complete.*

Proof: We set ourselves up as in Theorem 1.

Let K be m-complete RE. For each $i \in \Sigma^*$, define $A_i = \{x : p_i(x) \in K\}$. The A_i 's are precisely the r.e. sets, and the recursion theorem holds for A_i .

Let L be polynomial time 1-tt complete for RE. Again, L is uniformly polynomial time 1-tt complete, i.e., there is a programming system $\langle t_i \rangle_{i \in \Sigma^*}$ such that $t_i : A_i \leq_{1\text{-tt}}^P L$ and $\lambda i, x. t_i(x)$ is (exponential time) computable. We can again assume that the cases $t_i(x) = \text{true}$ and $t_i(x) = \text{false}$ don't occur.

It suffices to construct a polynomial time m-reduction from K to L .

Define A as:

$$A(\langle i, x \rangle) = \begin{cases} K(x), & \text{if (i) } t_i(\langle i, x \rangle) = (y_x \in L); \text{ and} \\ L(y_x), & \text{if (ii) } t_i(\langle i, x \rangle) = (y_x \notin L). \end{cases} \quad (4)$$

As K and L are r.e., so is A . By the 1-tt completeness of L there is a 1-tt reduction $f : A \leq_{1\text{-tt}}^P L$. Let j be a t -program for f .

If (i) holds for x , then $x \in K \iff \langle j, x \rangle \in A$ and $\langle j, x \rangle \in A \iff y_x \in L$; hence, $x \in K \iff y_x \in L$. If (ii) holds for x , then $\langle j, x \rangle \in A \iff y_x \in L$, but also $\langle j, x \rangle \in A \iff y_x \notin L$; hence $y_x \in L \iff y_x \notin L$. This is impossible, so (ii) never holds!

Therefore, $x \mapsto y_x$ is a polynomial time m-reduction of K to L , as required.

□

Note that Theorem 3 cannot be restated in the degree theoretic language used in Theorem 1. The problem is that the polynomial time 1-tt degree for RE contains sets that are not themselves r.e. Our proof depends critically on the fact that L is r.e. For example, \overline{K} is a member of the polynomial time 1-tt degree complete for RE, but \overline{K} is not even recursively m-complete for RE.

Corresponding to Corollary 2, and this time combining our result with Dowd's theorem [Dow78] that the polynomial time m-complete sets for RE are polynomial time 1-complete, we have,

Corollary 4 *The polynomial time 1-tt complete r.e. sets are polynomial time 1-complete.*

5 Remarks

Our proofs do not work for nondeterministic subrecursive classes such as NEXP or NP. Harry Buhrman [Buh90] has succeeded in proving the analogous theorem for NEXP—nondeterministic exponential time. That is, every 1-tt complete for NEXP set is m-complete for NEXP. The problem for NP remains open and interesting.

Our methods *do* work for logspace reductions, the analogous theorems can be obtained *mutatis mutandis*.

6 Acknowledgments

We would like to thank Eric Allender for bringing this problem to our attention. Klaus Ambos-Spies, Steffen Lempp and Robert Soare provided a solution to the analogous problem from recursion theory that got us started.

References

- [BDG88] J. L. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity I*, volume 11 of *EATCS Monographs on Theoretical Computer Science*. Springer Verlag, 1988.
- [Ber77] L. Berman. *Polynomial reducibilities and complete sets*. PhD thesis, Cornell University, 1977.
- [Buh90] H. Buhrman. Personal communication, 1990.
- [Dow78] M. Dowd. On isomorphism. Unpublished Manuscript, 1978.
- [GH88] K. Ganesan and S. Homer. Complete problems and strong polynomial reducibilities. In *Proc. Symposium on Theoretical Aspects of Computer Science*, pages 240–250, 1988. Springer Lecture Notes in Computer Science 349.
- [Hom90] S. Homer. Structural properties of nondeterministic complete sets. In *Proceedings of the 5th Annual IEEE Structure in Complexity Theory Conference*, pages 3–10, 1990.

- [KLD87] K. Ko, T. Long, and D. Du. A note on one-way functions and polynomial-time isomorphisms. *Theoretical Computer Science*, 47:263–276, 1987.
- [KMR88] S. Kurtz, S. Mahaney, and J. Royer. Collapsing degrees. *Journal of Computer and System Science*, 37:247–268, 1988.
- [KMR89] S. Kurtz, S. Mahaney, and J. Royer. The isomorphism conjecture fails relative to a random oracle. In *Proceedings of the 21st annual ACM Symposium on Theory of Computing*, pages 157–166, 1989.
- [KMR90] S. Kurtz, S. Mahaney, and J. Royer. On the structure of complete degrees. In A. Selman, editor, *Complexity Theory Retrospective*. Springer-Verlag, 1990.
- [Koz80] D. Kozen. Indexings of subrecursive classes. *Theoretical Computer Science*, 11:277–301, 1980.
- [LLS75] R. Ladner, N. Lynch, and A. Selman. Comparison of polynomial-time reducibilities. *Theoretical Computer Science*, 1:103–123, 1975.
- [RC87] J. Royer and J. Case. Intensional subrecursion and complexity theory. Technical Report 87-007, Department of Computer Science, University of Chicago, 1987.
- [Wat87] O. Watanabe. A comparison of polynomial time completeness notions. *Theoretical Computer Science*, 54:249–265, 1987.