On the Limitations of Locally Robust Positive Reductions

Lane A. Hemachandra* and Sanjay Jain†
Department of Computer Science
University of Rochester
Rochester, New York 14627, USA

Abstract

Polynomial-time positive reductions, as introduced by Selman, are by definition globally robust — they are positive with respect to all oracles. This paper studies the extent to which the theory of positive reductions remains intact when their global robustness assumption is removed.

We note that two-sided locally robust positive reductions — reductions that are positive with respect to the oracle to which the reduction is made — are sufficient to retain all crucial properties of globally robust positive reductions. In contrast, we prove absolute and relativized results showing that one-sided local robustness fails to preserve fundamental properties of positive reductions, such as the downward closure of $NP$.

1 Introduction

In this paper we study the relative powers of different positive reducibilities. Informally, a reduction is positive if converting some “no” answers to “yes” does not cause a previously accepted string to be rejected.

Selman, in his seminal paper [Sel82b], defines and considers the properties of polynomial-time positive reductions. His positive reductions are by definition globally robust in the positivity.

An oracle machine, or a set of oracle machines, is said to robustly have a property $P$ if it has property $P$ for all oracles. Recent work on the power of robustness [Sch85, BI87, HH87, Ko87, Tar87] suggests that global robustness is a strong restriction. For example, it is known that if two nondeterministic machines $N_1$ and $N_2$ are robustly complementary — that is, complementary for every oracle — then for all oracles $A$, $L(N_1^A) \in P^{A\oplus NP}$ [HH87]. This, and the desire to broaden the domain of application of Selman’s techniques, motivate us to relax the global robustness restriction.

Accordingly, we introduce three notions of locally robust polynomial-time positive reductions. We show that the Turing versions of these reducibilities differ. However, our ability to distinguish among the truth-table versions of these reducibilities depends on the structure of $NP$. In particular, we show that if $P=NP$ then these polynomial-time locally robust truth-table reducibilities are the same. However, if there exist uniformly log*-sparse tally sets in $NP-P$, then the reducibilities differ.

We study the extent to which the theory of positive reductions, as studied by Selman, remains intact for locally robust reductions. We prove results identifying the crucial properties of positive reductions required to obtain the results of [Sel82b]. One reason for introducing new reducibilities is that it is more likely that a set $A$ reduces to $B$ by locally positive reductions than by globally positive reductions. Our results thus enrich the domain in which Selman’s techniques can be applied.

*Supported in part by the National Science Foundation under grants CCR-8809174, CCR-8996198 and Presidential Young Investigator Award CCR-8957604.
†Supported by National Science Foundation grant CER 5-285-25 to the University of Rochester.
2 Notations

Let \( \mathcal{N} \) denote the set of natural numbers. \( \Sigma \) is an alphabet set, usually \( \{0,1\} \). A language is a subset of \( \Sigma^* \). \( \emptyset \) denotes the empty set. \( M_0, M_1, \ldots \) denotes some standard enumeration of polynomial-time deterministic Turing machines. \( N_1, N_2, \ldots \) denotes some standard enumeration of polynomial-time nondeterministic Turing machines. We assume that the running times of machine \( M_i \) (\( N_i \)) is bounded by deterministic (nondeterministic) time \( n^i + i \). \( P \) denotes the class of all languages accepted by some polynomial-time deterministic Turing machine \([HU79]\). \( NP \) denotes the class of languages accepted by some polynomial-time nondeterministic Turing machine and \( coNP \) denotes the class of languages whose complement is in \( NP \) \([HU79]\).

\( L(M) \) denotes the language accepted by the machine \( M \). \( E \) and \( NE \) denote respectively the class of languages accepted by exponential time deterministic and nondeterministic Turing machines; that is, \( E = \bigcup_{c>0} \text{DTIME}[2^{cn}] \) and \( NE = \bigcup_{c>0} \text{NTIME}[2^{cn}] \). \( L(M^A) \) denotes the language accepted by the oracle machine \( M \) with the oracle \( A \) \([HU79]\).

\( A \preceq_T B \) means there exists a machine \( M \) such that \( A = L(M^B) \). \( \preceq_T \) denotes polynomial-time Turing reduction. \( \preceq_t \) and \( \preceq_m \) similarly denote truth-table and many-one reductions. \( P(A) \) denotes the set of strings in \( A \) with length at most \( n \).

Positive reducibility was first studied for polynomial-time truth-table reductions in \([LLS75]\). Selman, in \([Sel82b]\), extended the definition to Turing reductions. We first give the definition of globally positive reducibility due to Selman.\(^1\)

**Definition 3.1** \([Sel82b, Sel82a]\) A query machine \( M \) is globally positive if \( (\forall x)(\forall A, B)[x \in L(M^B) \Rightarrow x \in L(M^{AUB})] \).

Intuitively, a machine \( M \) is positive if converting some "no" answers to "yes" answers does not make the machine reject a previously accepted string. Moreover, this property holds for all oracles given to the machine (hence the term globally positive). Positive reducibility can now be defined using these globally positive machines.

**Definition 3.2** \([Sel82b, Sel82a]\) \( A \preceq_{p} C \) if \( A \preceq_T C \) by some polynomial-time, globally positive Turing machine \( M \).

The conditions placed here on positive Turing reductions are analogous to those in the definition of positive truth-table reductions in \([LLS75]\), which defined globally positive truth-table reductions.

**Definition 3.3** \([LLS75]\) \( A \preceq_{p} C \) if \( A \preceq_T C \) by some polynomial-time, globally positive machine \( M \), and there is a polynomial-time computable function \( f : \{0,1\}^* \rightarrow \{c,0,1,\}^* \) such that \( M \) on input \( x \) makes queries only from the list \( f(x) \) (here \( c \) acts as a separator of elements of the list). \( M \) above can be equivalently represented by a polynomial time evaluator \( e \) such that for all oracles \( C \), \( e(x, XC(y_1), XC(y_2), \ldots) = M^C(x) \), where \( y_1, y_2, \ldots \) are the elements in the list \( f(x) \).\(^2\)

**Definition 3.4** \([LLS75]\) \( A \preceq_{p} C \) if \( A \preceq_T C \) by some polynomial-time, globally positive machine

---

\(^1\)This reducibility is simply referred to as "positive" in \([Sel82b]\). However, we'll refer to it throughout this paper as "globally positive" in order to distinguish it from the locally positive reducibilities we define.

\(^2\)In \([LLS75]\) the first argument of \( e \) is \( o(x) \), however without loss of generality we can take this to be \( x \) and let \( e \) do the (polynomial time) computations required to obtain \( o \).