Optimal Proof Systems for Propositional Logic and Complete Sets

Jochen Messner and Jacobo Torán

Universität Ulm
Theoretische Informatik
D-89069 Ulm, Germany
{messner,toran}@informatik.uni-ulm.de

Abstract. A polynomial time computable function $h : \Sigma^* \to \Sigma^*$ whose range is the set of tautologies in Propositional Logic (TAUT), is called a proof system. Cook and Reckhow defined this concept in [5] and in order to compare the relative strength of different proof systems, they considered the notion of p-simulation. Intuitively a proof system h p-simulates a second one h' if there is a polynomial time computable function γ translating proofs in h' into proofs in h. A proof system is called optimal if it p-simulates every other proof system. The question of whether p-optimal proof systems exist is an important one in the field. Krajíček and Pudlák [13, 12] proved a sufficient condition for the existence of such optimal systems, showing that if the deterministic and nondeterministic exponential time classes coincide, then p-optimal proof systems exist. They also gave a condition implying the existence of optimal proof systems (a related concept to the one of p-optimal systems). In this paper we improve this result obtaining a weaker sufficient condition for this fact. We show that if a particular class of sets with low information content in nondeterministic double exponential time is included in the corresponding deterministic class, then p-optimal proof systems exist. We also show some complexity theoretical consequences that follow from the assumption of the existence of p-optimal systems. We prove that if p-optimal systems exist then the class UP (and some other related complexity classes) have many-one complete languages, and that many-one complete sets for NP $\cap$ SPARSE follow from the existence of optimal proof systems.

1 Introduction

A systematic study of the complexity of proof systems for Propositional Logic, was started some time ago by Cook and Reckhow [5]. They were interested in studying the shortest proofs of propositional tautologies in different proof systems, and defined the abstract notion of proof system in the following way:

Definition 1. Let TAUT be the set of all Boolean tautologies (written in a fixed alphabet Σ). A *propositional proof system* (or just *proof system*) is a polynomial time computable function $h : \Sigma^* \to \Sigma^*$ whose range is TAUT.[1]

For example the function h defined as

$$h(w) = \begin{cases} \varphi & \text{if } w = \langle \varphi, v \rangle \text{ and } v \text{ is a resolution proof of } \varphi, \\ x \vee \overline{x} & \text{otherwise.} \end{cases}$$

is a proof system.

If $h(w) = \varphi$ we say that w is a proof of φ in h. Observe that in the given definition a proof system h is not required to be polynomially honest. For a tautology φ, the shortest proof of φ in h, can be much longer than φ.

A *polynomially bounded proof system* h is a proof system in which every tautology has a short proof. More formally, there is a polynomial q such that for every $\varphi \in$ TAUT, there is a string w of length bounded by $q(|\varphi|)$ with $h(w) = \varphi$.

It is not known whether polynomially bounded proof systems exist, on the other hand, many concrete proof systems have been shown not to be polynomially bounded (see for example [5],[16]). There are different motivations for studying the complexity of proof systmes. On the one hand, there are close relations between proof-complexity and Bounded Arithmetic (see e.g. [11]), also concrete proof systems like for example resolution or Frege systems are interesting in their own right, and recently, important connections between these systems and Boolean circuit complexity have been established. Another main motivation for the study of proof systems comes in fact from the following relation between the NP versus *co*-NP question and the existence of polynomially bounded systems.

Theorem 2. [5] NP = *co*-NP *if and only if polynomially bounded proof systems exist.*

This result started the so called *Cook-Reckhow Program*: a way to prove that NP is different from *co*-NP might be to study more and more powerful concrete proof systems, showing that they are not polynomially bounded, until hopefully we have gained enough knowledge to be able to separate NP from *co*-NP (see [3]).

In order to compare the relative powers of two different proof systems, the notion of polynomial simulation (or p-simulation) was introduced in [5].

Definition 3. Let h and h' be two propositional proof system. We say that h simulates h' if there is a function γ that is polynomially bounded in length and translates proofs in h' into proofs in h. In other words, there is a polynomial p such that for every x $|\gamma(x)| \leq p(|x|)$, and for every tautology φ and every proof w of φ in h', $\gamma(w)$ is a proof of φ in h. If in addition γ is computable in polynomial time, we say that h p-simulates h'.

[1] The original definition allows in fact the use of different alphabets for the domain and different languages for the range of h, but for the purposes of this paper the given definition suffices.

Observe that p-simulation is a stronger notion than simulation. It is easy to see that simulation and p-simulation are reflexive and transitive relations. It is also clear that if a proof system h is not polynomially bounded, and h simulates another system h', then h' cannot be polynomially bounded. Cook and Reckhow used p-simulation in order to classify proof systems in different classes with polynomially related derivation strength.

The notion of simulation between proof systems is closely related to the notion of reducibility between problems. Continuing with this analogy, the notion of a complete problem would correspond to the notion of an optimal proof system.

Definition 4. A proof system is optimal (p-optimal) if it simulates (p-simulates) every other proof system.

An important open problem is whether optimal proof systems exist [3]. Observe that if this were the case, then in order to separate NP from *co*-NP it would suffice to prove that a concrete proof system is not polynomially bounded.

Krajíček and Pudlák have given sufficient conditions for the existence of p-optimal and optimal proof systems.

Theorem 5. [13, 12]

If NE = *co*-NE *then optimal proof systems exist.*
If E = NE *then p-optimal proof systems exist.*

On the other hand, to our knowledge, only weak complexity-theoretic consequences of the existence of optimal proof systems were known.[2]

In the present paper we improve the mentioned result from [13, 12] by weakening the conditions that are sufficient for the existence of optimal and p-optimal proof systems. We show in Section 3 that if the deterministic and nondeterministic double exponential time complexity classes coincide (EE = NEE) then p-optimal proof systems exist, and that NEE = *co*-NEE is sufficient for the existence of optimal proof systems. In fact we give a probably weaker sufficient condition, using a special kind of sets with small information content. Let us say that a set is *almost tally*, if its words belong to the set 0^*10^*. We show that if the class of almost tally sets in NEE is included in EE, then p-optimal proof systems exist, and that optimal proof systems exist if almost tally sets in NEE belong also to *co*-NEE.

On the other side, we also show some consequences from the existence of optimal and p-optimal proof systems, proving completeness results for the complexity classes UP and NP ∩ SPARSE, that follow from the existence of such proof systems. Since complete problems for these classes have been unsuccessfully searched for in the past, the results give some evidence of the fact that optimal proof systems might not exist. At the same time they strengthen the connection between the notions of optimal proof systems and complete sets.

[2] From [14] follows that if optimal proof-systems exist, then the class of disjoint pairs in NP has a complete pair under a weak reduction.

The presentation is organized as follows: In Section 4 we show that if p-optimal proof systems exist, then the class UP (unambiguous NP) of problems in NP that can be accepted by nondeterministic polynomial time machines with at most one accepting path for every input [17], has complete problems under the logarithmic space many-one reductions. The existence of complete problems for UP has been studied in [7], where the authors show the existence of a relativization under which this is not possible. Considering that p-optimal proof systems exist, we also show the existence of complete sets for related *promise* classes like FewP. We also consider the weaker hypothesis of the existence of optimal proof systems, and show that the completeness results for the mentioned classes still hold for nonuniform many-one reductions. Finally in Section 5 we prove that optimal proof systems imply the existence of complete problems for the class NP ∩ SPARSE for many-one logarithmic space reductions. The question of the existence of such sets is subtle and has been intensively investigated. Although many-one complete set in NP ∩ SPARSE are not known, Hartmanis and Yesha prove in [8] that there is a sparse set in NP that is Turing complete for NP ∩ SPARSE, (in fact the given set is tally) and ask whether the result can be improved to the many-one case. Hartmanis has also shown that the set of satisfiable formulas with small Kolmogorov complexity SAT ∩ $K[\log, n^2]$ is Turing complete for NP ∩ SPARSE [6], but the completeness of this set under many-one reductions would imply unexpected consequences in the exponential time hierarchy. More recently Schöning has proven that there are sets that are complete for this class under many-one randomized reductions [15].

2 Basic Notions

We assume some familiarity with the standard results and notions about deterministic and nondeterministic complexity classes. For undefined complexity theory notions, and the definition of standard complexity classes, we refer the reader to the standard books in the area like [2]. We will use a pairing function $\langle\, ,\, \rangle$ that is polynomial time computable and invertible. For a set A, $\mathfrak{P}(A) = \{L \mid L \subseteq A\}$ represents the power set of A. Let E and EE, denote the time complexity classes DTIME($2^{O(n)}$) and DTIME($2^{O(2^n)}$), respectively, and NE, NEE their nondeterministic counterparts.

In the introduction we defined the notion of proof system. The next lemma shows that for every set of tautologies that is polynomial time computable, there is a proof system in which the tautologies of the set have short proofs, and moreover, the proof can be found easyly.

Lemma 6. *If $T \subseteq$ TAUT and $T \in$ P, then there exists a proof system h and a function $t \in$ FP that produces proofs in h for every tautology in T. That is, for every $\varphi \in T$, $h(t(\varphi)) = \varphi$.*

Proof. Let $h' \in$ FP be a proof system. We can define a new proof system h as follows:

$$h(w) = \begin{cases} h'(v) & \text{if } w = 0v, \\ v & \text{if } w = 1v \text{ and } v \in T, \\ x \vee \overline{x} & \text{otherwise.} \end{cases}$$

Clearly, h is a proof system. The function t producing proofs in h for the elements of T is just $t(v) = 1v$.

In the definitions of proof systems and p-optimality we allowed both functions, the proof system h and the translation function, to be computable in polynomial time. The following lemma shows that the most part of the computational complexity of both functions can be concentrated in one of them, whereas the other function may be computed, for example, in logarithmic space. To formulate the lemma, let us say that a proof system h is *logspace-optimal* if it simulates in logarithmic space every other proof system h', which means that there is a logspace computable function γ such that $h(\gamma(w)) = h'(w)$ for every word w.

Lemma 7. *The following statements are equivalent*

1. *A p-optimal polynomial time computable proof system exists.*
2. *A logspace-optimal polynomial time computable proof system exists.*
3. *A p-optimal logspace computable proof system exists.*

Proof. Clearly, *2* and *3* imply *1*.

To obtain *2* from *1* let h be a p-optimal polynomial time computable proof system, and let g be defined by

$$g(w) = \begin{cases} h(w') & \text{if } w = \langle M, 0^l, v\rangle, \text{ and } M \text{ is a deterministic Turing trans-} \\ & \text{ducer which on input } v \text{ outputs } w' \text{ in at most } l \text{ steps,} \\ x \vee \neg x & \text{else.} \end{cases}$$

Clearly, g is polynomial time computable. We show that g is logspace-optimal. Let h' be a proof system. By assumption, there is a polynomial time computable translation function γ such that $h(\gamma(w)) = h'(w)$ for any w. Let M be a deterministic Turing transducer computing γ with time bounded by a polynomial $p(n)$. It's easy to see that the logspace computable function γ' with $\gamma'(w) = \langle M, 0^{p(|w|)}, w\rangle$ translates proofs in h' into proofs in g.

We now show that the existence of a p-optimal proof system h implies a logspace computable p-optimal proof system f. Let M be a polynomial time machine computing h. Let $f(w) = \varphi$ if w encodes a complete computation of M (given by the sequence of configurations) with output φ, let $f(w) = x \vee \neg x$ otherwise. Choosing a suitable encoding f is logspace computable. Also f p-simulates h. Therefore f p-simulates every proof system.

3 A sufficient condition

We give now a sufficient condition for the existence of optimal proof systems, based on almost tally sets in double exponential time.

Theorem 8.

If $\mathrm{NEE} \cap \mathfrak{P}(0^*10^*) \subseteq \mathrm{EE}$ *then there exists a p-optimal proof system.*
If $\mathrm{NEE} \cap \mathfrak{P}(0^*10^*) \subseteq$ *co*-NEE *then there exists an optimal proof system.*

Proof. Let M_1, M_2, M_3, ... be some standard enumeration of deterministic Turing transducers with binary input alphabet such that there is an universal Turing machine which is able to simulate k steps of M_i in $(ik)^2$ time for any $k \geq 0$ (clearly, such enumerations exist). Now define the language

$$\mathrm{T} = \{0^j10^i \mid \text{for any word } w \text{ of length at most } 2^{2^n}\text{, where } n = i + j + 1\text{: if } M_i \text{ stops on input } w \text{ in at most } 2^{2^n} \text{ steps, } M_i \text{ outputs a tautology}\}.$$

It is not hard to see that $T \in$ *co*-NEE. Assuming $\mathrm{NEE} \cap \mathfrak{P}(0^*10^*) \subseteq \mathrm{EE}$ we also have *co*-$\mathrm{NEE} \cap \mathfrak{P}(0^*10^*) \subseteq \mathrm{EE}$ and therefore $T \in \mathrm{EE}$. Therefore there is a deterministic Turing machine M_T which decides T in time $2^{c \cdot 2^n}$ for some $c > 0$.

We describe a p-optimal proof system h:

> On input $\langle 0^j10^i, 0^s, w \rangle$ examine if $s \geq 2^{2^l}$ and $|w| \leq 2^{2^l}$, where $l = i + j + 1$, and test whether M_T accepts 0^j10^i in at most s steps. If this is the case, output $M_i(w)$ if M_i stops after at most 2^{2^l} steps on input w. (If some other case applies, output some fixed tautology).

Clearly, $0^j10^i \in T$ implies that the Turing machine M_i on input w outputs a tautology if the computation needs at most 2^{2^l} steps. Therefore $h(\Sigma^*) \subseteq \mathrm{TAUT}$. Also, h is computable in polynomial time. We have to show that h p-simulates every other proof system. Let g be a proof system computed by the deterministic Turing transducer M_i with time bound $n^k + k$. A proof w for g is translated into the proof $w' = \langle 0^j10^i, 0^s, w \rangle$ where $s = 2^{c \cdot 2^{i+j+1}}$, and $j = \max(0, \lceil \log\log |w|^k + k \rceil - i - 1)$. By the construction of h, we have $h(w') = g(w)$. We just have to show that the translation $w \mapsto w'$ is computable in polynomial time. Clearly, this is the case if the length of 0^s is polynomially bounded by $|w|$. Now observe that $s = 2^{c \cdot 2^{i+j+1}} \leq 2^{c \cdot 2^{i+1} \cdot 2^{\log\log |w|^k + k}} = (|w|^k + k)^{c \cdot 2^{i+1}}$ which is polynomial in $|w|$.

Similar considerations can be used to show the second part of the theorem.

If in the previous proof we replace each occurrence of the number 2 by some arbitrary constant d (and log by $\log_d$) we obtain that already $\mathrm{NTIME}(2^{O(d^n)}) \cap \mathfrak{P}(0^*10^*) \subseteq \mathrm{DTIME}(2^{O(d^n)})$ for some $d > 0$ implies the existence of a p-optimal proof system (a similar result follows for optimal proof systems).[3] However, the proof seems not to translate directly to a further exponential level as $2^{2^{2^{c + \log\log\log n}}} \notin n^{O(1)}$ for any real $c > 0$.

[3] This improvement emerged from an email discussion with S. Ben-David.

4 Complete problems for UP

In this and the next sections we prove that the existence of optimal proof systems imply the existence of complete sets in certain *promise* complexity classes. The machines computing the sets in these classes can not be guaranteed to keep the condition of the class for all possible inputs, and because of this, complete problems for these classes are not known. We will first prove the existence of complete sets for UP under many-one polynomial time reductions, considering the existence of p-optimal proof systems. Later we will strengthen this result to logarithmic space reductions.

Define the set CAT containing descriptions of machines that are categorical, i.e., have at most one accepting path for all inputs up to a given length.

$$\begin{aligned}\mathrm{CAT} = \{\langle M, 0^l, 0^n\rangle \mid M \ &\text{is a nondeterministic Turing machine and for every}\\ &\text{input } x,\ |x| \le n,\ M \text{ has at most one accepting path of length } \le l\}.\end{aligned}$$

Clearly, CAT $\in$ *co*-NP and since TAUT is *co*-NP complete for polynomial time many-one length-increasing reductions, there is such a function $f \in$ FP reducing CAT to TAUT.

For every fixed nondeterministic Turing machine M and every fixed monotone polynomial q, the set

$$\mathrm{CAT}_{M,q} = \{\langle M, 0^{q(n)}, 0^n\rangle \mid n \ge 1\} \cap \mathrm{CAT},$$

is in P because it is either finite or the set of all such triples. Also the image of $\mathrm{CAT}_{M,q}$ under f, is in P; in order to test whether a given formula ϕ belongs to $f(\mathrm{CAT}_{M,q})$ it suffices to generate the words of $\mathrm{CAT}_{M,q}$ up to a given length, and check whether the image of one of these words after applying f coincides with ϕ.

Let $\varphi_{M,l,n}$ denote the formula $f(\langle M, 0^l, 0^n\rangle)$. Clearly, if the machine M is categorical and its running time is bounded by q, then for every n, $\varphi_{M,q(n),n}$ is a tautology.

The following lemma intuitively says that under the hypothesis of the existence of p-optimal proof systems, for every categorical machine M there is a polynomial time computable function producing proofs of the categoricity of M.

Lemma 9. *Let $h \in$ FP be a p-optimal proof system. For every categorical machine M with running time bounded by a polynomial q, there is a function $g_{M,q} \in$ FP such that for every $l \in \mathbb{N}$, $g_{M,q}(0^l)$ produces an output w and $h(w) = \varphi_{M,q(l),l}$.*

Proof. Let M be a categorical machine polynomially time bounded by a polynomial q. Every formula in the set $f(\mathrm{CAT}_{M,q})$ is a tautology. As we have seen this set is in P and by Lemma 6 there is a proof system h' and a function $t \in$ FP that produces short proofs in h' for the tautologies in $f(\mathrm{CAT}_{M,q})$. Formally, for every tautology $\varphi_{M,q(l),l} \in f(\mathrm{CAT}_{M,q})$, $h'(t(\varphi_{M,q(l),l})) = \varphi_{M,q(l),l}$.

Since h is p-optimal, it p-simulates h'. This means that there is a function $\gamma \in$ FP, translating proofs in h' into proofs in h, and for every tautology $\varphi_{M,q(l),l} \in f(\text{CAT}_{M,q})$ we have $h(\gamma(t(\varphi_{M,q(l),l}))) = \varphi_{M,q(l),l}$.

The claimed function $g_{M,q}$ on input 0^l computes the formula $\varphi_{M,q(l),l} = f(\langle M, 0^{q(l)}, 0^l \rangle)$, and applies functions t and γ to it. Clearly $g_{M,q} \in$ FP.

We can now prove that p-optimal proof systems imply the existence of complete sets for UP.

Theorem 10. *If p-optimal proof systems exist then there are sets that are complete for* UP *under polynomial time many-one reductions.*

Proof. Let h be a p-optimal proof system and consider the set

$$A = \{\langle M, 0^l, w, x\rangle \mid M \text{ is the description of a NDTM and } h(w) = \varphi_{M,l,|x|} \text{ and } M \text{ accepts } x \text{ in } l \text{ steps or less}\}.$$

The set A is clearly in UP since $h(w) = \varphi_{M,l,|x|}$ means that this formula is a tautology, and therefore for every input of length smaller or equal than $|x|$ (and in particular for x), M has at most one accepting path of length l.

For the hardness part, let B be a set in UP, accepted by a machine M in time bounded by a polynomial q. W.l.o.g. we can suppose that for every n, $q(n) \leq q(n+1)$, and that on any input x, every computation path of M halts after exactly $q(|x|)$ steps. Consider the function $g_{M,q} \in$ FP whose existence was proved in the above lemma. The function $\lambda \in$ FP defined for every $x \in \Sigma^*$ as

$$\lambda(x) = \langle M, 0^{q(|x|)}, g_{M,q}(0^{|x|}), x\rangle$$

many-one reduces B to A since $h(g_{M,q}(0^{|x|})) = \varphi_{M,q(|x|),|x|}$.

To see that p-optimal proof systems also imply the existence of logspace many-one complete problem for UP, we show that the construction in the proof of Lemma 9 can be modified so that $g_{M,q}$ is logspace computable. In fact, by the same observations it can be seen that even weaker reductions are possible. Remember $g_{M,q}(0^l) = \gamma(t(f(\langle M, 0^{q(l)}, 0^l\rangle)))$ where f is a length-increasing reduction from CAT to TAUT, $t(x) = 1x$ is the function from the proof of Lemma 6, and γ is a function translating proofs in a proof system h' into proofs in h, whose existence is guaranteed by the p-optimality of system h. Clearly, $\langle M, 0^{q(l)}, 0^l\rangle$ can be computed in logarithmic space from 0^l, and the function t is logspace computable. Also f can be chosen to be a logspace computable (and length-increasing) many-one reduction from CAT to TAUT. By Lemma 7 we can assume h to be logspace-optimal, and therefore we can also assume γ to be logspace computable. As the composition of logspace computable functions is again logspace computable we obtain:

Theorem 11. *If p-optimal proof systems exist then there are sets that are complete for* UP *under logarithmic space many-one reductions.*

The completeness result for UP can be extended to the related complexity classes FewP and Few as stated in the next theorem. The classes FewP and Few were defined in [1] and [4] as a generalizations of the class UP. For space reasons we omit the definition of these classes and the proof of the next theorem.

Theorem 12. *If p-optimal proof systems exist then there are sets that are complete under logarithmic space many-one reductions for the classes* FewP *and* Few.

Let us mention at this point that in [7] an oracle is constructed under which the class UP does not have many-one complete sets. In [9] this result is improved from many-one to Turing reducibility, and it is also shown that under certain relativizations the class FewP does not have Turing complete sets. Since the proofs in this paper relativize, we can state the following corollary:

Corollary 13. *There exists a relativization under which p-optimal proof systems do not exist.*

If we only consider the existence of optimal proof systems (instead of p-optimal systems), then we can prove a version of the above result for nonuniform reductions. Intuitively a set is polynomial time nonuniformly many-one reducible to a second one, if the reduction function is not necessarily in FP, as in the usual polynomial time reductions, but it is computed by a family of polynomial size circuits [10].

Due to space reasons we omit the formal definition of non-uniform reductions as well as the proof of the next theorem.

Theorem 14. *If optimal proof systems exist then there are sets that are complete for* UP *under nonuniform polynomial time many-one reductions.*

As expected, the many-one completeness results for Few and FewP from Theorem 12 become completeness results for nonuniform many-one reductions if only the existence of optimal proof systems is considered.

5 Complete sets for NP ∩ SPARSE

We prove now that there are many-one complete sets for NP ∩ SPARSE under the hypothesis of the existence of optimal proof systems. The proof follows the same lines as the previous one for complete sets in UP, but in this case we do not need p-optimality, and the existence of optimal proof systems suffices.

Let us define the set SP containing descriptions of nondeterministic machines that do not accept too many strings up to a given length:

SP $= \{\langle M, 0^l, 0^n\rangle \mid M$ is a nondeterministic Turing machine and there are at most l pairs (x_i, y_i), $|x_i| \leq n$, $|y_i| \leq l$, such that $x_i \neq x_j$ for $i \neq j$, and y_i is an accepting path of M on input $x_i\}$.

It is not hard to see that SP $\in$ *co*-NP, and therefore SP is polynomial-time many-one reducible to TAUT. Let $f \in$ FP be a length increasing function that reduces SP to TAUT.

Let$M$be a fixed nondeterministic Turing machine with running time bounded by a polynomial q, that for every length l accepts at most $q(l)$ words of length l. The set

$$\mathrm{SP}_{M,q} = \{\langle M, 0^{q(n)}, 0^n\rangle \mid n \geq 1\} \cap \mathrm{SP},$$

is in P, and the image of $\mathrm{SP}_{M,q}$ under f, is also in P;

Let $\zeta_{M,l,n}$ denote the formula $f(\langle M, 0^l, 0^n\rangle)$. Clearly, if the machine M runs in time bounded by q and accepts a q-sparse set of inputs, then for every n, $\varphi_{M,q(n),n}$ is a tautology.

The following lemma is analogous to Lemma 9, and says that in an optimal proof system, a proof of the fact that a machine accepts a sparse language up to a given length, can be polynomially bounded. The proof of the lemma follows the same lines as the one for Lemma 9 and it is omitted.

Lemma 15. *Let $h \in$ FP be a p-optimal proof system. For every nondeterministic Turing machine M with running time bounded by a polynomial q, and such that for every $n \in \mathbb{N}$, M accepts at most $q(n)$ words of length n, there is a polynomial r such that for every $l \in \mathbb{N}$, there is a string $w \in \Sigma^*$, with $|w| \leq r(l)$ and $h(w) = \zeta_{M,q(l),l}$.*

We can now prove that optimal proof systems imply the existence of complete sets for NP $\cap$ SPARSE.

Theorem 16. *If optimal proof systems exist then there are sets that are complete for* NP $\cap$ SPARSE *under logarithmic space many-one reductions.*

Proof. Let h be an optimal proof system, and let S be the set

$$S = \{0^{\langle M,l,j,n\rangle}1x \mid M \text{ is the description of a NDTM which accepts } x \text{ in } l \text{ steps or less, } |x| = n, \text{ and there is a string } w,\ |w| \leq j, \text{ such that } h(w) = \zeta_{M,l,n}\}.$$

S belongs clearly to NP. Also, the number of string x such that $0^{\langle M,l,j,n\rangle}1x \in S$ is bounded by l, since $0^{\langle M,l,j,n\rangle}1x \in S$ implies that $\zeta_{M,l,|x|}$ is a tautology. Therefore for every length n there are at most n words of this length in S. This proves that S is sparse.

In order to see that S is hard for the class, let S' be a set in NP $\cap$ SPARSE, accepted by a nondeterministic Turing machine M with time bounded by a polynomial q, and with density also bounded by q. By Lemma 15 there is a polynomial r such that for every $l \in \mathbb{N}$, there is a string w with $|w| \leq r(l)$ and $h(w) = \zeta_{M,q(l),l}$. The reduction from S' to S is given by the function

$$\lambda(x) = 0^{\langle M,q(|x|),r(|x|),|x|\rangle}1x.$$

Observe that this function is computable in logarithmic space, one-to-one, length increasing and also invertible in logarithmic space.

Let us mention at this point that contrary to the UP case, there is no known relativization under which the class NP ∩ SPARSE does not have many-one complete sets. For this reason, and considering the existing results on sparse sets mentioned in the introduction, we feel that Theorem 16 only provides a weak consequence of the existence of optimal proof systems.

References

1. E. Allender. Invertible functions. Ph.D. dissertation, Georgia Institute of Technology, 1985.
2. J. L. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity I*, volume 11 of *EATCS Monographs on Theoretical Computer Science.* Springer-Verlag, 1988.
3. S. Buss. Lectures on Proof Theory. Tech Report No. SOCS-96.1, McGill University, 1996. (http://www.cs.mcgill.ca/˜denis/TR.96.1.ps.gz)
4. J. Cai and L. Hemachandra. On the power of parity polynomial time. *Mathematical Systems Theory* **23**, pp. 95–106, 1990.
5. S. Cook and R. Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic* **44**, pp. 36–50, 1979.
6. J. Hartmanis. Generalized Kolmogorov complexity and the structure of feasible computations. In *Proceedings of the 24th IEEE Symposium on Foundations of Computer Science* (FOCS'83), pp. 439–445, 1983
7. J. Hartmanis and L. Hemachandra Complexity classes without machines: On complete languages for UP. *Theoretical Computer Science*, **58**, pp. 129–142, 1988.
8. J. Hartmanis and J. Yesha. Computation times of NP sets of different densities. *Theoretical Computer Science*, **34**, pp. 17–32, 1984.
9. L. Hemaspaandra, S. Jain and N. Vereshchagin. Banishing robust Turing completeness. *Int. Journal of Foundations of Computer Science*, **4**, pp. 245–265, 1993.
10. R. Karp and R. Lipton. Some connections between nonuniform and uniform complexity classes. In *Proceedings of the 12th ACM Symposium on Theory of Computing*, pp. 302–309, 1980.
11. J. Krajíček. *Bounded Arithmetic, Propositional Logic and Complexity Theory* Cambridge University Press 1995.
12. J. Krajíček and P. Pudlák. Propositional proof systems, the consistency of first order theories and the complexity of computations. *Journal of Symbolic Logic* **54**, pp. 1063–1079, 1989.
13. P. Pudlák. On the length of proofs of finitistic consistency statements in first order theories. Logic Colloquium'84 (J. B. Paris et al., editors), North-Holland, Amsterdam, pp. 165–196, 1986
14. A. A. Razborov. On provably disjoint NP-pairs. Technical Report RS-94-36, Basic Research in Computer Science Center, Aarhus, 1994.
15. U. Schöning. On random reductions from sparse sets to tally sets. *Information Processing Letters*, **46**, pp. 239–241, 1993.
16. A. Urquhart. The complexity of propositional proofs. *Bulletin of Symbolic Logic* **1**, pp. 425-467, 1995.
17. L. Valiant. The relative complexity of checking and evaluating. *Information Processing Letters*, **5**, pp. 20–23, 1976.