THE JOURNAL OF SYMBOLIC LOGIC
Volume 54, Number 3, Sept. 1989

PROPOSITIONAL PROOF SYSTEMS, THE CONSISTENCY OF FIRST ORDER THEORIES AND THE COMPLEXITY OF COMPUTATIONS

JAN KRAJÍČEK AND PAVEL PUDLÁK

Abstract. We consider the problem about the length of proofs of the sentences $\mathrm{Con}_S(\underline{n})$ saying that there is no proof of contradiction in S whose length is $\leq n$. We show the relation of this problem to some problems about propositional proof systems.

§0. Introduction. For a finitely axiomatized theory S let $\mathrm{Con}_S(\underline{n})$ denote the statement that there is no proof of contradiction in S whose length is $\leq n$. From the point of view of foundations of mathematics it would be extremely interesting to know how difficult is to prove $\mathrm{Con}_S(\underline{n})$ in another theory T, in particular in a weaker theory. A question that we shall address here is whether there exists a consistent (finitely axiomatized) theory T such that for every consistent (finitely axiomatized) theory S sentences $\mathrm{Con}_S(\underline{n})$ have short proofs in T, and whether such proofs can be easily constructed.

Another problem is connected with propositional proof systems. We accept a very general definition of a propositional proof system which is equivalent to the definition of [CR]. For two systems P, Q we write $P \leq Q$ iff there exists a polynomial p such that for every tautology t, if t has a proof of length n in Q then t has a proof of length $\leq p(n)$ in P. In particular, if P polynomially simulates Q in the sense of [CR], then $P \leq Q$. Clearly any two systems P, Q have a common lower bound. The most interesting open problem about this quasiordering is whether there exists the least element in it. If $\mathcal{NP} = \mathrm{co}\mathcal{NP}$ then the answer is "yes", but it seems that the existence of the least element might be a weaker statement than $\mathcal{NP} = \mathrm{co}\mathcal{NP}$. We shall not solve this problem. We shall only show that the positive answer to it is equivalent to the existence of a theory T in which all $\mathrm{Con}_S(\underline{n})$, for any S consistent, have short proofs. For the stronger statement that there exists a propositional proof system which polynomially simulates every propositional proof system, we prove that it is equivalent to the existence of a theory T such that, for any fixed theory S, proofs of $\mathrm{Con}_S(\underline{n})$ in T can be constructed in polynomial time (in n). Also it is equivalent to the existence of, in a sense, optimal deterministic algorithm for the set of propositional tautologies.

Received October 28, 1987; revised July 6, 1988.

0022-4812/89/5403-0033/$02.70

If one considers the proofs of $\mathrm{Con}_S(\underline{n})$ in T, one may suspect that even in the case when $S = T$ the proofs must be exponentially long (in n). But in [Pu1,2] it has been shown that for T finite and sufficiently strong there are such proofs which are only polynomially long (in n), and in fact can be constructed by a deterministic Turing machine in polynomial (in n) time. We think that it will be considerably more difficult to determine the length of proofs of $\mathrm{Con}_S(\underline{n})$ in T if T is weaker than S. In order to support this belief we shall use relativization, which is so far the only means to prove some kind of independence in the complexity theory. As the relativization of such a problem requires second order language of arithmetic and is not quite well understood, we shall relativize an equivalent problem which can be stated using only the concept of the Turing machine.

In §5 we shall consider what can be proved about propositional proof systems in a weak fragment of arithmetic S_2^1. We observe that results of Cook [Co] and Buss [Bu] imply that the existence of a propositional proof system which polynomially simulates every propositional proof system is in a certain sense consistent with S_2^1. M. Dowd [Do] observed that Cook's theory PV proves the soundness of a Frege system with substitution, which implies that extended Frege systems polynomially simulate Frege systems with substitution. We shall sketch the proof of this result with PV replaced by S_2^1, and then, in §6, we show an explicit simulation.

Related problems were studied especially in [Fr] and [Bu]. Friedman [Fr] proved a lower bound $\Omega(n^{1/4})$ to the length of proof of $\mathrm{Con}_T(\underline{n})$ in T and suggested studying several problems of this kind. Buss [Bu] found a proof-theoretical statement equivalent to $\mathcal{NP} = \mathrm{co}\mathcal{NP}$.

§1. Preliminaries. We shall consider only *finitely axiomatized consistent theories* whose language contain *the language of arithmetic*, i.e. the language $\{0, 1, <, =, +, \cdot\}$. N denotes the set of nonnegative integers. For $n \in N$, $\underline{n}$ will be *the numeral based on the dyadic expansion* of n. Thus the length of n is proportional to $\log n$. All finite objects will be considered as finite sequences in the alphabet $\{0, 1\}$. The one-to-one mapping between the sequences of 0 and 1 and dyadic expansions determines a *Gödel numbering of finite objects*, and this in turn determines the formalization. Thus, e.g. for a formula φ, its formalization $\underline{\varphi}$ is a numeral whose length is proportional to the length of φ. *The length of a sequence* s will be denoted by $|s|$.

We shall denote the relation "*d is a proof of φ in T*" by $d\colon T \vdash \varphi$. We define:

$$T \vdash^n \varphi \Leftrightarrow \exists d(|d| \leq n \;\&\; d\colon T \vdash \varphi).$$

Let T be a theory, R a k-ary relation on N, $\varphi(x_1, \quad, x_k)$ a formula. We say that φ *$\mathcal{P}$-numerates* (*resp. $\mathcal{NP}$-numerates*) R *in* T if

$$R(n_1, \ldots, n_k) \Leftrightarrow T \vdash \varphi(\underline{n_1}, \ldots, \underline{n_k})$$

and there exists a polynomial time Turing machine M such that

$$n_k) \Rightarrow M(n_1, \quad, n_k)\colon T \vdash \varphi(n_1, \quad, n_k),$$

(resp. there exists a polynomial p such that

$$\ldots, n_k) \Rightarrow T \vdash^{p(\log n_1 \cdot \cdots \cdot n_k)} \varphi(n_1, \quad, n_k)).$$

LEMMA 1.1. *There exists a finite fragment T_0 of the true arithmetic such that, for every relation R on N, $R \in \mathscr{P}$ (resp. $R \in \mathscr{NP}$) iff R is $\mathscr{P}$-numerable (resp. $\mathscr{NP}$-numerable) in T_0.*

For the proof see the proof-sketch of Theorem 3.3 of [Pu1]. Essentially the proof consists of verifying that, for a suitable formalization of Turing machines, a given accepting computation can be easily transformed into a proof that such a computation exists. It should be pointed out that natural formalizations of relations in $\mathscr{P}$ (in $\mathscr{NP}$) are actually $\mathscr{P}$-numerations ($\mathscr{NP}$-numerations).

For ordinary Hilbert style proofs the ternary relation d: $T \vdash \varphi$ is in $\mathscr{P}$. Hence let $\mathrm{Prf}(x, y, z)$ be some formula which $\mathscr{P}$-numerates this relation in T_0. (Prf will be used also for propositional proof systems.) We define

$$\mathrm{Con}_T(x) \rightleftarrows \forall y(|y| \leq x \to \neg \mathrm{Prf}(y, \underline{T}, \underline{0 = 1})).$$

(To be quite precise, we should also speak about a $\mathscr{P}$-numeration of the relation $|d| \leq n$, but we shall omit such details here.)

Let TAUT be the set of propositional tautologies. Let $\mathrm{Taut}(x)$ denote a formula such that $\neg \mathrm{Taut}(x)$ $\mathscr{NP}$-numerates the complement of TAUT in T_0.

For a usual propositional proof system P such as Frege systems, Frege systems with substitution, etc., the relation "d is a proof of t", denoted by d: $P \vdash t$, is in $\mathscr{P}$. The general concept of a propositional calculus can be defined just by this requirement. For technical reasons we shall require a little more, namely that this binary relation is computable in deterministic linear time. Thus we define: P is a *propositional proof system* if P is a binary relation computable in deterministic linear time such that

$$\exists d, d\colon P \vdash t \Leftrightarrow t \in \mathrm{TAUT}.$$

Roughly speaking, a propositional proof system is a nondeterministic acceptor for TAUT, and the length of a proof is the length of an accepting computation. Clearly if the provability relation d: $P \vdash t$ is polynomial time computable we can transform it into a linear time computable relation by increasing the lengths of proofs only polynomially. Thus the usual propositional proof systems fall into our definition, except that the length of proofs is polynomially increased. A polynomial increase in the length of proofs is for our purpose irrelevant.

Cook [Co] and Cook and Reckhow [CR] define a propositional proof system to be a polynomial time computable function F such that the range of F is the tautologies. Put otherwise, $F(x)$ is the formula whose proof is x. Thus F corresponds to P, where

$$d\colon P \vdash t \quad \text{iff} \quad P(d) = t,$$

i.e. P is a polynomial time computable relation. As described above, P can be transformed into a relation satisfying our definition of propositional proof systems. On the other hand, if P is a proof system in our sense, define P as follows:

$$\begin{aligned} F(x) &= y && \text{if } \exists d, (d\colon P \vdash y) \wedge x = [d, y], \\ &= p \vee \neg p && \text{otherwise.} \end{aligned}$$

Thus there is no essential difference between these two concepts.

Since for each P we require that the binary relation $d\colon P \vdash t$ is linear time computable, we can choose suitable codes of (the Turing machines computing) the calculi P so that the ternary relation $d\colon P \vdash t$ (where P is identified with its code) is *polynomial time computable.* By Lemma 1.1 it can be $\mathscr{P}$-numerated. We shall use $\mathrm{Prf}(x, y, z)$ to denote its $\mathscr{P}$-numeration, the same formula as for first order theories.

Let P and Q be two propositional proof systems. We say that P *polynomially simulates* Q if there exists a polynomial time computable function F such that $d\colon Q \vdash t \Rightarrow F(d, t)\colon P \vdash t$. Clearly if P polynomially simulates Q then $P \leq Q$.

Recall that the classes $\mathscr{EXP}$ (resp. $\mathscr{NEXP}$) are the classes of languages recognized by deterministic Turing machines in time $2^{O(n)}$ (resp. accepted by nondeterministic Turing machines in time $2^{O(n)}$).

A language X is called *sparse* iff there exists a polynomial p such that for every n, X contains at most $p(n)$ words of length $\leq n$.

For a Turing machine M and an input w, $\mathrm{TIME}(M; w)$ denotes the length of (a longest) computation of M on w. A *machine M with an oracle A* will be denoted by M^A; similarly $\mathscr{P}^A$, $\mathscr{NP}^A$, etc. denote the classes $\mathscr{P}$, $\mathscr{NP}$, etc. relativized to A.

A propositional formula will be sometimes called simply a *proposition.*

§2. The length of proofs of the consistency statements and the length of proofs in propositional proof systems. Let us consider the following statements.

(1) There exists a finitely axiomatized fragment T of the true arithmetic such that for every finitely axiomatized consistent theory S there exists a polynomial p such that $T \overset{p(n)}{\vdash} \mathrm{Con}_S(\underline{n})$, for every $n \in N$.

(2) There exists a propositional proof system which is a least element in the quasiordering $\leq$ (see §0).

(3) There exists a propositional proof system P such that for every $X \subseteq \mathrm{TAUT}$, $X \in \mathscr{NP}$, there exists a polynomial p such that for every $t \in X$, $P \overset{p(|t|)}{\vdash} t$.

(4) There exists a propositional proof system P such that for every $X \subseteq \mathrm{TAUT}$, $X \in \mathscr{P}$, X sparse, there exists a polynomial p such that for every $t \in X$, $P \overset{p(|t|)}{\vdash} t$.

(5) For every $A \in \mathrm{co}\,\mathscr{NP}$ there exists a nondeterministic Turing machine M which accepts A and such that for every $X \subseteq A$, $X \in \mathscr{P}$, X sparse there exists a polynomial p such that, for every $w \in X$, M accepts w in time $\leq p(|w|)$.

(6) There exists a finitely axiomatized fragment T of the true arithmetic such that, for every finitely axiomatized consistent theory S, there exists a deterministic Turing machine M and a polynomial p such that for any given n, M constructs a proof of $\mathrm{Con}_S(\underline{n})$ in T in time $\leq p(n)$.

(7) There exists a propositional proof system which polynomially simulates every propositional proof system.

(8) There exists a deterministic Turing machine M which recognizes TAUT and such that for every deterministic Turing machine M' which recognizes TAUT there exists a polynomial p such that for every $t \in \mathrm{TAUT}$

$$\mathrm{TIME}(M; t) \leq p(|t|, \mathrm{TIME}(M'; t)).$$

(9) For every $A \in \mathrm{co}\mathscr{NP}$ there exists a deterministic Turing machine M which recognizes A and such that for every deterministic Turing machine M' which

recognizes A there exists a polynomial p such that for every $w \in A$

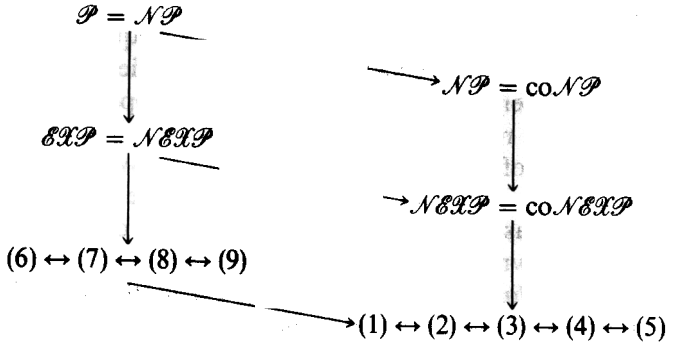
$$\mathrm{TIME}(M; w) \leq p(|w|, \mathrm{TIME}(M'; w)).$$

Suppose (1) were true and T is a reasonable theory, say a fragment of Peano arithmetic, and $p(x)$ is some small polynomial. Then we could realize the Hilbert program in a modified, finitistic sense. We conjecture that this is not possible, i.e. (1) is not true. The only information about this problem is that for T sufficiently strong, every proof of $\mathrm{Con}_T(\underline{n})$ in T has length at least $n^{\varepsilon}, \varepsilon > 0$. On the other hand proofs of polynomial length (in n) can be constructed; in fact there exists a deterministic Turing machine which constructs a proof of $\mathrm{Con}_T(\underline{n})$ in T in time $p(n)$, p a polynomial (cf. [Fr] and [Pu1,2]).

Also, very little is known about propositional proof systems. No nontrivial lower bounds are known for proofs in ordinary propositional proof systems except of the resolution system [Ha]. For instance we cannot rule out that the system in (7) is, say a Frege system.

THEOREM 2.1. (i) *Statements* (1)–(5) *are equivalent, and they are implied by* $\mathcal{NEXP} = \mathrm{co}\mathcal{NEXP}$ *and by statements* (6)–(9).

(ii) *Statements* (6)–(9) *are equivalent, and they are implied by* $\mathcal{EXP} = \mathcal{NEXP}$.



§3. Proofs. In the first part of this section we shall exhibit four basic constructions:

(a1) of a propositional proof system from a true theory,

(a2) of a theory from a propositional proof system,

(b1) of a bounded arithmetical formula from a proposition, and

(b2) of a proposition from a particular bounded arithmetical formula.

The constructions are not difficult, but since they are essential for the proofs and also of some independent interest we describe them explicitly before the proofs.

In the second part of the section we shall prove Theorem 2.1.

From now on we shall consider only finitely axiomatized consistent theories.

Part 1. *The constructions.* We fix T_0, the theory ensured by Lemma 1.1. Let $\mathrm{Taut}(x)$ and $\mathrm{Prf}(x, y, z)$ be the formulas introduced in §1.

(a1) For $T \supseteq T_0$ a finitely axiomatized part of the true arithmetic we define *propositional proof system* $P(T)$ by

$$d^{\frown}0^{p(|d|)}: P(T) \vdash t \quad \text{iff} \quad d: T \vdash \mathrm{Taut}(t).$$

Here a string of zeros of length $p(|d|)$ is added to d so that the binary relation "d is a proof of t" in $P(T)$ is linear time computable; $p(x)$ is a suitable polynomial.

(a2) For P a propositional proof system we construct a finitely axiomatized true *theory* $S(P)$ by:

$$S(P) \rightleftarrows T_0 \cup \{\mathrm{Prf}(x, \underline{P}, y) \to \mathrm{Taut}(y)\}.$$

Further we shall call the formula $\mathrm{Prf}(x, \underline{P}, y) \to \mathrm{Taut}(y)$ *the essential axiom of* $S(P)$.

(b1) From a proposition t we construct a bounded arithmetical *sentence* $\mathrm{Taut}(\underline{t})$. (It is well known that $\mathrm{Taut}(x)$ can be taken bounded, in particular as a Π_1^b-formula; cf. [Bu].)

(b2) Let $\varphi(x)$ be of the form

$$\forall x_1 < 2^x \cdots \forall x_k < 2^x \psi(x, x_1, \ldots, x_k),$$

where ψ is a bounded arithmetical formula of the form

$$Q_1 y_1 < x \cdots Q_l y_l < x \; \sigma(x, x_1, \ldots, x_k, y_1, \ldots, y_l),$$

where $Q_i \in \{\forall, \exists\}$ and σ is open and its atomic subformulas have the form $u = v$, $u + v = w$ or $u \cdot v = w$, where u, v, w are 0, 1 or a variable. We shall transform each closed instance $\varphi(\underline{n})$, $n \in N$, of $\varphi(x)$ into a *proposition* $t_{\varphi,n}$.

For a given n, we can transform $\psi(\underline{n}, x_1, \ldots, x_k)$ into an equivalent open formula $\psi_n(x_1, \ldots, x_k)$ by succesively replacing the bounded quantifiers by n-term conjunctions or disjunctions. Then we introduce an n-tuple of propositional variables for each x_i, $i = 1, \ldots, k$, and replace each atomic formula $u = w$, $u + v = w$ and $u \cdot v = w$ by the conjunction of n formulas for the digits of w. (We can assume without loss of generality that $u, v, w \in \{x_1, \ldots, x_k, \underline{0}, \underline{1}, \ldots, \underline{n}\}$.) The formulas for addition and multiplication are easily obtained from the circuits of depth $O(\log(n))$, whose construction is well known [Sa].

LEMMA 3.1. *For T a finitely axiomatized fragment of the true arithmetic, P a propositional proof system and t a tautology, there exists a polynomial $p(x)$ such that*

(i) *if* $T \vdash^{n} \mathrm{Taut}(\underline{t})$ *then* $P(T) \vdash^{p(n)} t$, *and*

(ii) *there is a polynomial $p(x, y, z)$ which does not depend on P or t such that, for any d, if d: $P \vdash t$ then*

$$S(P) \vdash^{p(|d|,|P|,|t|)} \mathrm{Taut}(\underline{t}).$$

PROOF. (i) is trivial. For (ii), by Lemma 1.1 there is a polynomial $q(x, y, z)$ such that, for any d, P, t, if d: $P \vdash t$ then

$$T_0 \vdash^{q(|d|,|P|,|t|)} \mathrm{Prf}(\underline{d}, \underline{P}, \underline{t}).$$

Using the essential axiom of $S(P)$ we deduce in $S(P)$ the formula $\mathrm{Taut}(\underline{t})$ from $\mathrm{Prf}(\underline{d}, \underline{P}, \underline{t})$ by a proof of length linear in $|d| + |P| + |t|$. Combining these two proofs we obtain the proof of $\mathrm{Taut}(\underline{t})$. □

LEMMA 3.2. (i) *For t a proposition and $T \supseteq T_0$,*

$$t \in \mathrm{TAUT} \quad \textit{iff} \quad N \models \mathrm{Taut}(\underline{t}) \quad \textit{iff} \quad T \vdash \mathrm{Taut}(\underline{t}).$$

(ii) *There exists a deterministic Turing machine M which constructs a proposition $t_{\varphi,n}$ from any $\varphi(x)$ of the form described in the construction* (b2) *and from any $n \in N$*

such that

a) $t_{\varphi,n} \in \mathrm{TAUT}$ *iff* $N \models \varphi(\underline{n})$, *and*

b) $\mathrm{TIME}(M; \varphi(x), n) \leq p(|\varphi|, n)$ *for some polynomial* $p(x, y)$ *not depending on* φ *or* n.

PROOF. (i) is trivial. (ii) The lemma holds with the $t_{\varphi,n}$ defined above; however the proof would not be so easy as the proof for the following construction of $t_{\varphi,n}$.

Since the relation

$$\{[2^n, m_1, \ldots, m_k] \mid N \models \psi(\underline{n}, \underline{m}_1, \ldots, \underline{m}_k)\}$$

is in $\mathscr{P}$, there exists a deterministic Turing machine which in time polynomial in n constructs a circuit $C_{\psi,n}$ such that for $m_1, \ldots, m_k < 2^n$

$$C_{\psi,n}(m_1, \ldots, m_k) = 1 \quad \text{iff} \quad N \models \psi(\underline{n}, \underline{m}_1, \ldots, \underline{m}_k),$$

where $m_1, \ldots, m_k$ are considered to be 0-1 inputs of length n. By introducing new propositional variables $\vec{q}$ for the vertices of $C_{\psi,n}$ we can construct a propositional formula $t_{\varphi,n}(\vec{p}_1, \ldots, \vec{p}_k, \vec{q})$ such that

$$C_{\psi,n}(m_1, \ldots, m_k) = 1 \quad \text{iff} \quad \forall \vec{q}, t_{\varphi,n}(m_1, \ldots, m_k, \vec{q}).$$

Hence $N \models \varphi(\underline{n})$ iff $N \models \forall x_1 < 2^n \cdots \forall x_k < 2^n \psi(\underline{n}, x_1, \ldots, x_k)$ iff $t_{\varphi,n}(\vec{p}_1, \ldots, \vec{p}_k, \vec{q}) \in \mathrm{TAUT}$. □

REMARK. Observe that the proof of Lemma 3.2 works for any co$\mathscr{NEXP}$-predicate instead of $\varphi(x)$. In fact, any co$\mathscr{NEXP}$-predicate can be defined by a formula of the form described in the construction (b2); in particular, $\mathrm{Con}_S(x)$ can be written in such a form (cf. [Bu]).

We shall write $\cdots \vdash^{n}_{*} \cdots$ to mean that there exists a polynomial p such that, for every $n, \cdots \vdash^{p(n)} \cdots$.

LEMMA 3.3. *There exists a finite fragment* $T_1 \supseteq T_0$ *of the true arithmetic such that, for every theory* S,

$$T_1 \vdash^{n}_{*} \mathrm{Con}_S(\underline{n}) \leftrightarrow \mathrm{Taut}(t_{\mathrm{Con}_S,n}),$$

and these proofs can be constructed in time polynomial in n.

PROOF. Let $\varphi(\underline{S}, \underline{n}, t_{\mathrm{Con}_S,n})$ be a $\mathscr{P}$-numeration of the construction (b2) or of the construction of the proof of Lemma 3.2 restricted to formulas $\mathrm{Con}_S(\underline{n})$.

Let T_1 be T_0 plus the axiom

$$\varphi(x, y, z) \rightarrow [(\forall u(|u| \leq y \rightarrow \neg \mathrm{Prf}(u, x, \underline{0 = 1}))) \leftrightarrow \mathrm{Taut}(z)].$$

Now for given S we can construct $t_{\mathrm{Con}_S,n}$ in time polynomial in n. By $\mathscr{P}$-numerability we get a proof of

$$\varphi(\underline{S}, \underline{n}, t_{\mathrm{Con}_S,n})$$

again in polynomial time. From this we obtain the required equivalence in T_1 immediately. □

Part 2. Proof of the Theorem. The proof that $\mathscr{NEXP} = \mathrm{co}\mathscr{NEXP}$ implies (1) is easy and has been shown in [Pu1, Proposition 6.2]. The proof that $\mathscr{EXP} = \mathscr{NEXP}$ implies (6) is almost the same.

The following implications are trivial: (2) ⇒ (3), (5) ⇒ (4), (6) ⇒ (1), (9) ⇒ (8). It remains to prove (3) ⇒ (5), (4) ⇒ (1), (1) ⇒ (2) and (6) ⇒ (9), (8) ⇒ (6), (7) ⇒ (6) and (6) ⇒ (7).

Proof of (3) ⇒ (5). If $A \in \text{co}\,\mathcal{NP}$ then A can be reduced to TAUT by a deterministic polynomial time reduction F such that for some $\varepsilon > 0$ we have $|F(w)| \geq |w|^{\varepsilon}$, for every input w. Thus if $X \subseteq A$, $X \in \mathcal{P}$, then $F(X) \subseteq \text{TAUT}$, $F(X) \in \mathcal{NP}$. Now clearly the Turing machine M for A can be constructed from the proof system and F. □

Proof of (4) ⇒ (1). Let P be a propositional proof system with the properties of (4). Let S be an arbitrary consistent theory. Then the set $\{t_{\text{Cons}_S,n} \mid n \in N\}$ is contained in TAUT, is in $\mathcal{P}$ and is sparse. Thus there are proofs d_n, $n \in N$, of size polynomial in n, such that $d_n\colon P \vdash t_{\text{Cons}_S,n}$.

By $\mathcal{P}$-numerability,

$$T_0 \vdash^{n}_{*} \text{Prf}(\underline{d_n}, \underline{P}, \underline{t_{\text{Cons}_S,n}}),$$

whence, by the essential axiom of $S(P)$,

$$S(P) \vdash^{n}_{*} \text{Taut}(\underline{t_{\text{Cons}_S,n}}).$$

Now, by Lemma 3.3, $S(P) \vdash^{n}_{*} \text{Con}_S(\underline{n})$; hence $S(P)$ is the desired theory T satisfying statement (1). □

Proof of (1) ⇒ (2). Suppose T has the property stated in (1). We can assume that T is sufficiently strong, in particular that $T_0 \subseteq T$ and

$$(*) \qquad T \vdash \neg\text{Taut}(x) \to \exists y(|y| \leq r(|x|) \;\&\; \text{Prf}(y, \underline{T_0}, \underline{\neg\text{Taut}(\dot{x})})),$$

where $\underline{\neg\text{Taut}(\dot{x})}$ is used as an abbreviation for the formalization of the function

$$n \mapsto \text{“the Gödel number of } \neg\text{Taut}(\underline{n})\text{”}$$

and $r(x)$ is a suitable polynomial. Note that this is a special case of an antireflection principle which holds in weak fragments of arithmetic; cf. [Bu, Theorem 7.4] and [PW2, Theorem 6.4].

We shall show that $P(T)$ is a least element in the quasiordering of the propositional proof systems. Let Q be an arbitrary propositional proof system. Suppose $d\colon Q \vdash s$, $|d| = n$. Then, by $\mathcal{P}$-numerability in particular,

$$S(Q) \vdash^{n}_{*} \text{Prf}(\underline{d}, \underline{Q}, \underline{s}),$$

and hence, by the essential axiom of $S(Q)$,

$$S(Q) \vdash^{q(n)} \text{Taut}(\underline{s}), \quad \text{for some polynomial } q.$$

Using $\mathcal{NP}$-numerability,

$$T_0 \vdash^{n}_{*} \exists x(|x| \leq \underline{q(n)} \;\&\; \text{Prf}(x, \underline{S(Q)}, \underline{\text{Taut}(s)})).$$

Since $|s| \leq |d| = n$, $T_0 \subseteq S(Q)$ and

$$T \vdash^{n}_{*} \text{Con}_{S(Q)}(\underline{10 \cdot (r(n) + q(n))}),$$

the last formula implies:

$$T \vdash^{n}_{*} \neg\exists x(|x| \leq r(|s|) \;\&\; \text{Prf}(x, \underline{T_0}, \underline{\neg\text{Taut}(s)})).$$

(We assume that, given proofs of some σ of length a and $\neg\sigma$ of length b, a proof of $0 = 1$ of length $10(a+b)$ can be constructed.) Hence, by (*), $T \vdash^{n}_{*} \mathrm{Taut}(\underline{s})$; thus $P(T) \vdash^{n}_{*} s$. □

Proof of (6) ⇒ (9). Let T be the theory of (6) and $A \in \mathrm{co}\,\mathcal{NP}$. By $\mathcal{NP}$-numerability of the complement of A there exists a polynomial p and a formula $\psi(x)$ such that

$$(*) \qquad w \notin A \Rightarrow T_0 \vdash^{p(|w|)} \neg\psi(\underline{w}).$$

Let M_i, $i \in N$, be an enumeration of deterministic Turing machines such that M_0 is a trivial exponential time algorithm for A (given by the Turing machine accepting the complement of A) and such that the relation "C is the computation of the ith Turing machine on w" is a relation in $\mathcal{P}$. Let $\varphi(x,y,z)$ be a $\mathcal{P}$-numeration of this relation in T_0.

Let S_i, $i \in N$, be the theory $T_0 \cup \{\forall x \forall y(\varphi(x,\underline{i},y) \to \psi(y))\}$. Since each S_i contains T_0, (*) is true for every S_i (with the same polynomial p). By the $\mathcal{P}$-numerability and the definition of S_i there exists a polynomial q such that:

(**) if C is an accepting computation of M_i on input w, then

$$S_i \vdash^{q(|C|,i,|w|)} \psi(\underline{w}).$$

Now we shall describe a machine M with the property stated in (9). On an input w, $|w| = n$, M will simulate the work of M_i, $i = 0,\ldots,n$, in several rounds. Namely, in the mth round M simulates

(i) one additional computational step of $M_0,\ldots,M_n$ on w and

(ii) one additional computational step of $M_0,\ldots,M_n$ on inputs $1,2,\ldots,m$ (thus, for instance, it will simulate the first step of $M_0,\ldots,M_n$ on m); and

(iii) M will check if there are $i, k \le n$ and $j, l \le m$ such that M_i has accepted w in j steps and M_k has produced on input l a proof of $\mathrm{Con}_{S_i}(\underline{10(p(n)+q(j,i,n))})$ in T; if so M stops and accepts w; otherwise it goes on until it finishes the simulation of M_0 on w and behaves as M_0.

First we shall show that M recognizes A. If M finishes the simulation of M_0, then this is clear. So suppose that M accepts w because the situation in (iii) occurs. Assume that $w \notin A$. Then, by (*),

$$S_i \vdash^{p(n)} \neg\psi(\underline{w}).$$

On the other hand, M_i accepted w by a computation of length j. Thus, by (**),

$$S_i \vdash^{q(j,i,n)} \psi(\underline{w}).$$

Hence there is a proof of a contradiction in S_i of length, say, $10(p(n)+q(j,i,n))$. But T proves $\mathrm{Con}_{S_i}(\underline{10(p(n)+q(j,i,n))})$. Hence T would be inconsistent, which is a contradiction. Thus $w \in A$.

Now assume that M' recognizes A. Let $M' = M_i$. Then S_i is consistent. By (6), some M_k constructs the proof of $\mathrm{Con}_{S_i}(\underline{n})$ on input n in polynomial time $r(n)$. Let w be an input, $|w| = n \ge i,k$, $w \in A$. Let $j = \mathrm{TIME}(M_i; w)$. Because of M_k, M will find a proof of $\mathrm{Con}_{S_i}(\underline{10(p(n)+q(j,i,n))})$ in T in the $r(10(p(n)+q(j,i,n)))$th round or sooner. Thus M will make at most $\max(r(10(p(n)+q(j,i,n))), j)$ rounds when

working on w. This value is bounded by a polynomial in n and $j = \mathrm{TIME}(M_i; w)$. Since the mth round of M takes polynomially many steps in n and m, the condition of (9) is satisfied. □

Proof of (8) ⇒ (6). Let M be a Turing machine with the property stated in (8). Let S be a consistent theory. Then there exists a polynomial p such that for every n

$$\mathrm{TIME}(M; t_{\mathrm{Con}_S,n}) \leq p(n).$$

If not, we could combine M with a machine which recognizes the tautologies $t_{\mathrm{Con}_S,n}$, $n \in N$, in polynomial time, and thus obtain a machine which is more than polynomially faster than M on an infinite set of inputs. The rest of the proof is the same as the proof of (4) ⇒ (1). Therefore we leave it to the reader. □

Proof of (7) ⇒ (6). Let P be a propositional proof system which polynomially simulates every propositional proof system. As in the proof of (4) ⇒ (1) we shall show that $T = S(P)$ is the theory required in (6). Let S be a consistent theory. Since the set of tautologies $\{t_{\mathrm{Con}_S,n} \mid n \in N\}$ is in $\mathscr{P}$, there exists a propositional proof system Q in which the proofs of these tautologies can be constructed in polynomial time. (E.g. for a suitable polynomial $p(x)$, we can take $0^{p(n)}$ as the proof of $t_{\mathrm{Con}_S,n}$.) As P polynomially simulates Q, the same must be true for P. Thus we can construct in polynomial time proofs d_n: $P \vdash t_{\mathrm{Con}_S,n}$. The rest is the same as in the proof of (4) ⇒ (1). □

Proof of (6) ⇒ (7). Let T be the theory whose existence is assured by (6). Then $P(T)$ is a propositional proof system which polynomially simulates every propositional proof system. The proof is almost identical with the proof of (1) ⇒ (2). One has only to check that all proofs whose existence is claimed can in fact be constructed in polynomial time. □

§4. A relativization.

THEOREM 4.1. *There exists a recursive oracle A such that the following holds. There exists $B \in \mathrm{co}\mathscr{NP}^A$ such that for every nondeterministic oracle Turing machine M^A there exists $X \subseteq B$, $X \in \mathscr{P}^A$ and X sparse, such that either*

(a) *M^A does not accept B, or*

(b) *for infinitely many inputs $w \in X$, $\mathrm{TIME}(M^A; w) \geq 2^{|w|}$.*

Thus (5) is false relative to an oracle. On the other hand if A is such that $\mathscr{NP}^A = \mathrm{co}\mathscr{NP}^A$ then (5) is true relative to A (such an oracle has been constructed in [BGS]).

PROOF. B will be constructed so that there exists an oracle Turing machine M_0^A which accepts B in time $n \cdot 2^n$. Thus we need only consider oracle Turing machines which stop after 2^{2n} steps on every input of length n (since we can combine any M^A with M_0^A without increasing the running time). Let M_i^A, $i \in N$, be an enumeration of such machines such that each machine occurs in it infinitely many times and such that, for every M^A,

$$(*) \qquad \{i \mid M_i^A = M^A\} \in \mathscr{P}.$$

Let $f(n)$ be the recursively defined function $f(1) = 1$, $f(n+1) = 2^{2f(n)} + 1$. Thus M_i^A working on $w, |w| = f(n)$, cannot query the oracle about words of length $f(n+1)$.

Define $A \subseteq \{0,1\}^*$, $A = \bigcup_{n=0}^{\infty} A(n)$, by $A(0) = \varnothing$ and

$$A(n+1) = A(n) \cup \{w \mid |w| = f(n+1) \,\&\, \text{the lexicographically first accepting computation of } M_{n+1}^{A(n)} \text{ on } 0^{f(n+1)} \text{ which does not query all words of length } f(n+1) \text{ does not query } w\}.$$

(Here, and in the sequel, 0^m is the word consisting of m zeros.) Thus $A(n+1) = A(n)$ if there is no accepting computation of $M_{n+1}^{A(n)}$ on $0^{f(n+1)}$, or if every such accepting computation queries all words of length $f(n+1)$.

Claim. $M_{n+1}^{A(n)}$ *accepts* $0^{f(n+1)}$ *iff* M_{n+1}^{A} *accepts* $0^{f(n+1)}$.

Proof. $\Rightarrow$. By the construction of A, the lexicographically first accepting computation of $M_{n+1}^{A(n)}$ on $0^{f(n+1)}$ is also an accepting computation of $M_{n+1}^{A(n+1)}$ and M_{n+1}^{A} can use only the part $A(n+1)$ of A on $0^{f(n+1)}$.

$\Leftarrow$. If $M_{n+1}^{A(n)}$ does not accept $0^{f(n+1)}$, then $A(n+1) = A(n)$. Again M_{n+1}^{A} can use only $A(n+1) = A(n)$ on $0^{f(n+1)}$. The claim is proved.

Let $B = \{w \mid \forall v \in A, |v| \neq |w|\}$. Thus $B \in \text{co}\mathcal{NP}^A$. Now let an arbitrary M be given. We consider three cases.

(i) There exists n such that $M_n^A = M^A$ and there exists an accepting computation of $M_n^{A(n-1)}$ on $0^{f(n)}$ which does not query all words of length $f(n)$. Then M_n^A accepts $0^{f(n)}$ by the claim, but $A(n) \setminus A(n-1) \neq \varnothing$, i.e. $0^{f(n)} \notin B$. Hence M^A does not accept B.

(ii) There exists n such that $M_n^A = M^A$ and no computation of $M_n^{A(n-1)}$ accepts $0^{f(n)}$. Then, by the claim, M_n^A does not accept $0^{f(n)}$, but $A(n) = A(n-1)$, i.e. $0^{f(n)} \in B$. Hence M^A does not accept B.

(iii) For every n such that $M_n^A = M^A$ there exists an accepting computation of $M_n^{A(n-1)}$ on $0^{f(n)}$, and each such computation queries all words of length $f(n)$. Then, for every such n, $A(n) = A(n-1)$, i.e. $0^{f(n)} \in B$, and $\text{TIME}(M_n^A, 0^{f(n)}) \geq 2^{f(n)}$. Thus for $X = \{0^{f(n)} \mid M_n^A = M^A\}$ we have $X \subseteq B$, X is sparse, $\text{TIME}(M^A; w) \geq 2^{|w|}$ for $w \in X$, and, by (*), $X \in \mathcal{P} \subseteq \mathcal{P}^A$. □

§5. Propositional proof systems in a weak fragment of arithmetic. In the previous sections we have investigated relations between general first order theories (fragments of arithmetic) and propositional proof systems. In this section we shall concentrate on particular theories and particular propositional proof systems.

We shall recall the definition of a Frege system from [CR]. A *Frege rule* is a rule of the form

$$\frac{C_1, \ldots, C_n}{D}$$

where $C_1, \ldots, C_n, D$ are propositional formulas such that $C_1, \ldots, C_n \models D$. If σ is a substitution of formulas for propositional atoms, then we say that $\sigma(D)$ *follows from* $\sigma(C_1), \ldots, \sigma(C_n)$ by the rule above. A *proof* from a set of rules is, as usual, a sequence of formulas such that each formula follows from the previous one by some rule. An implicationally complete finite set of Frege rules is called a *Frege system*; for details cf. [CR]. The usual textbook systems with finitely many axiom schemes and modus ponens as a rule are Frege systems. The *extension rule* is the rule which allows one to

introduce formulas of the form $p \equiv A$, where A is an arbitrary formula and p is an atom which does not occur in A, in any preceding formula of the proof and in the last formula of the proof. The *substitution rule* allows one to deduce $\sigma(A)$ from A for any substitution σ.

Any two Frege systems polynomially simulate each other, and the same is true for any two Frege systems with the extension rule and any two Frege systems with the substitution rule. Let F, EF, and SF denote respectively some Frege system, some Frege system with the extension, and some Frege system with the substitution rule. We shall not define the system ER of extended resolution, since one can easily show that it has the same power as EF.

In [Bu] Buss introduced several fragments of arithmetic. The most interesting seems to be the one denoted by S_2^1. Roughly speaking, it consists of some basic theory, which includes binary operation $2^{\lceil \log(x+1) \rceil \cdot \lceil \log(y+1) \rceil}$ and induction Σ_1^b-LIND which has the form

$$\varphi(0) \wedge (\forall x, \varphi(x) \to \varphi(x+1)) \to \forall x, \varphi(\lceil \log(x) \rceil)$$

for $\varphi \in \Sigma_1^b$. The formulas in Σ_1^b define in the standard model of arithmetic just the $\mathcal{NP}$-predicates. He proved that the statement $\mathcal{NP} \cap \text{co}\mathcal{NP} = \mathcal{P}$ is in a certain schematic way consistent with S_2^1. Namely if, for $\varphi(x) \in \Sigma_1^b$, $\neg\varphi(x)$ is provably in S_2^1 equivalent to some $\psi(x) \in \Sigma_1^b$, then it provably in S_2^1 defines a set in $\mathcal{P}$.

In the same way our statement (7) is consistent with S_2^1. This follows from a theorem of Cook [Co] and the result of Buss [Bu] that S_2^1 is in a sense a conservative extension of Cook's equational theory PV. Using this conservativity of S_2^1 over PV, Cook's theorem can be stated in our notation as follows.

THEOREM 5.1 (COOK, BUSS). (a) $S_2^1 \vdash \text{Prf}(x, ER, y) \to \text{Taut}(y)$.

(b) *For every propositional proof system P, if*

$$S_2^1 \vdash \text{Prf}(x, P, y) \to \text{Taut}(y),$$

then there exists a function f which is provably (in S_2^1) polynomial time computable and such that

$$S_2^1 \vdash \text{Prf}(x, P, y) \to \text{Prf}(f(x, y), ER, y).$$

In [Do] and [KP] a similar relation between S_2^i, $i \geq 1$, and propositional proof systems for quantified Boolean formulas has been shown. The meaning of the theorem is that (a) S_2^1 proves the consistency of ER, and (b) if S_2^1 proves that P is consistent, then it can be polynomially simulated by ER. Thus our statement (7), with $P = ER$, is consistent with S_2^1 in the same schematic way as $\mathcal{NP} \cap \text{co}\mathcal{NP} = \mathcal{P}$ is consistent with S_2^1.

We observe that a weaker version of another schematic statement, proved by A. Wilkie [W], follows from Theorem 5.1. (Wilkie uses SF, but ER and SF are provably in S_2^1 equivalent.)

COROLLARY 5.2. *Suppose $S_2^1 \vdash \mathcal{NP} = \text{co}\mathcal{NP}$. Then there exists a polynomial $p(x)$ such that*

$$(*) \qquad S_2^1 \vdash \text{Taut}(y) \to \exists x, |x| \leq p(|y|) \wedge \text{Prf}(x, ER, y).$$

PROOF. If $S_2^1 \vdash \mathcal{NP} = \text{co}\mathcal{NP}$, we have

$$S_2^1 \vdash \text{Taut } y \leftrightarrow \exists x, |x| \leq q(|y|) \wedge \varphi(x, y),$$

where $q(x)$ is a polynomial and $\varphi(x, y)$ is provably polynomial time computable. Thus if we think of $|x| \leq q(|y|) \wedge \varphi(x, y)$ as a propositional proof system then S_2^1 proves its consistency, and thus by Theorem 5.1 *ER* has provably (in S_2^1) polynomially long proofs too. □

M. Dowd [Do] observed that the relation between the propositional proof systems *ER* and *PV* can be used to settle a problem posed by Cook and Reckhow [CR]. They have shown that *SF* polynomially simulates *EF*, and conjectured that the converse is not true. Since the following result is only stated in [Do] we shall give a sketch of its proof.

THEOREM 5.3 (cf.[Do]).

$$S_2^1 \vdash \text{Prf}(x, SF, y) \rightarrow \text{Taut}(y).$$

COROLLARY 5.4 (cf.[Do]). *ER, and hence also EF, polynomially simulates SF.*

The corollary follows immediately from 5.1 and 5.3.

PROOF OF THEOREM 5.3 (outline). We assume that formulas are arithmetized in the usual way, see [Bu]. A substitution σ is a finite mapping assigning to some propositional variables p some formulas B. We define $\sigma p = p$ if p is not in the domain of σ. Thus we may consider σ to be a Δ_1^b-definable function assigning to any propositional variable a propositional formula.

The truth value of a formula A for given truth assignment τ will be denoted by $\text{Tr}(A, \tau)$; we again define $\tau p = 0$ if p is not in the domain of τ. Let $\text{Eval}(w, A, \tau)$ be a formalization of "w is a computation of the truth value of A for τ" and let $\text{Val}(w)$ be the last truth value in the computation. Then we have, for a suitable constant c,

$$\begin{aligned} \text{Tr}(A, \tau) = \varepsilon &\leftrightarrow \exists w \leq A^c, \text{Eval}(w, A, \tau) \wedge \text{Val}(w) = \varepsilon, \\ &\leftrightarrow \forall w \leq A^c, \text{Eval}(w, A, \tau) \rightarrow \text{Val}(w) = \varepsilon. \end{aligned}$$

Hence $\text{Tr}(A, \tau) = \varepsilon$ is Δ_1^b. Define

$$\text{Taut}(A) \leftrightarrow \forall \tau \leq A, \text{Tr}(A, \tau) = 1$$

Thus $\text{Taut}(A)$ is Π_1^b.

A truth assignment τ and a substitution σ determine another truth assignment τ^σ in a natural way:

$$\tau^\sigma(p) = \text{Tr}(\sigma p, \tau).$$

Clearly $\tau, \sigma \mapsto \tau^\sigma$ is a Δ_1^b definable function. The following identities are easily provable in S_2^1.

(i) $\text{Tr}(\neg A, \tau) = 1 - \text{Tr}(A, \tau)$; $\text{Tr}(A \wedge B, \tau) = \text{Tr}(A, \tau) \cdot \text{Tr}(B, \tau), \ldots$.

(ii) $\sigma(\neg A) = \neg\sigma(A)$, $\sigma(A \wedge B) = \sigma(A) \wedge \sigma(B), \ldots$.

Claim. $S_2^1 \vdash \text{Tr}(\sigma(A), \tau) = \text{Tr}(A, \tau^\sigma)$.

This is proved using the identities (i) and (ii) and induction over the length of A.

As an immediate corollary we have

$$S_2^1 \vdash \text{Taut}(A) \rightarrow \text{Taut}(\sigma(A)).$$

Thus S_2^1 proves that the substitution rule preserves tautologies. For the other rules of SF the same fact is easily provable. As Taut is Π_1^b and S_2^1 proves also Π_1^b-LIND (see [Bu, Chapter 2]), we can now use induction over the length of the proof to show that each formula in an SF-proof is a tautology. □

§6. Explicit polynomial simulation of the substitution rule by the extension rule. The proof above did not give an explicit polynomial simulation of SF by EF. We shall describe such a simulation here.

In order to simplify the exposition we shall assume that SF contains only modus ponens and substitution as rules. If the system contained other Frege rules the simulation would be essentially the same.

Let $\varphi_1(\bar{p}), \dots, \varphi_n(\bar{p})$ be an SF proof, where $\bar{p} = (p_1, \dots, p_m)$ are all propositional variables occurring in the proof. Let $\bar{q}_1, \dots, \bar{q}_n$ be vectors of propositional variables which contain distinct elements, and let $\bar{q}_n = \bar{p}$.

Let ψ_i denote $\varphi_i(\bar{q}_i)$, $i = 1, \dots, n$. For $j = 1, \dots, n$ define a vector $\bar{\beta}_j$ of m formulas as follows:

(1) If $\varphi_j(\bar{p})$ is an axiom or has been obtained from previous formulas by modus ponens, then $\bar{\beta}_j = \bar{q}_j$.

(2) If $\varphi_j(\bar{p})$ has been obtained from $\varphi_i(\bar{p})$ by a substitution, say $\varphi_j(\bar{p}) = \varphi_i(\bar{\alpha}(\bar{p}))$, then $\bar{\beta}_j = \bar{\alpha}(\bar{q}_j)$.

Denote by

$$\Psi_{i,j} = \psi_i \wedge \psi_{i+1} \wedge \cdots \wedge \psi_j, \qquad j, i = 0, \dots, n-1,$$

$j \geq i - 1$, where $\Psi_{i,i-1}$ is the empty expression or the truth.

Now the simulation of the proof above in EF proceeds as follows. First we introduce variables $\bar{q}_{n-1}, \dots, \bar{q}_1$ by the extension rule by putting

$$q_{i,k} \equiv (\Psi_{i+1,i} \wedge \neg\psi_{i+1} \wedge \beta_{i+1,k}) \vee \cdots \vee (\Psi_{i+1,n-1} \wedge \neg\psi_n \wedge \beta_{n,k}).$$

Clearly, for $i < j$,

$$\Psi_{i+1,j-1} \wedge \neg\psi_j \to q_{i,k} \equiv \beta_{j,k}$$

can be derived by a polynomial proof. Hence we have polynomial proofs also of

$$\Psi_{i+1,j-1} \wedge \neg\psi_j \to \varphi_i(\bar{q}_i) \equiv \varphi_i(\bar{\beta}_j),$$

which can be written also as

$$(*) \qquad \Psi_{i+1,j-1} \wedge \neg\psi_j \to \psi_i \equiv \psi_i(\bar{\beta}_j).$$

Now we shall prove succesively $\psi_1, \psi_2, \dots, \psi_n$. Since $\psi_n = \varphi_n(\bar{p})$ we obtain a polynomial simulation.

Suppose $\psi_1, \dots, \psi_{j-1}$ has already been proved.

(1) First suppose that $\varphi_j(\bar{p})$ is an axiom. Then ψ_j is also an axiom.

(2) Suppose that $\varphi_j(\bar{p})$ follows from $\varphi_u(\bar{p})$ and $\varphi_v(\bar{p})$, $u, v < j$, by modus ponens. First derive $\Psi_{u+1,j-1}$ and $\Psi_{v+1,j-1}$. Thus we get from (*), with $i = u$ and $i = v$,

$$\neg\psi_j \to \varphi_u(\bar{\beta}_j) \wedge \varphi_v(\bar{\beta}_j).$$

Applying modus ponens to $\varphi_u(\bar{\beta}_j)$ and $\varphi_v(\bar{\beta}_j)$, we obtain $\neg\psi_j \to \varphi_j(\bar{\beta}_j)$, which is $\neg\psi_j \to \psi_i$; hence ψ_i follows.

(3) Suppose that $\varphi_j(\bar{p})$ has been obtained by a substitution, say $\varphi_j(\bar{p}) = \varphi_i(\bar{\alpha}(\bar{p}))$, $i < j$. In the same way as above, from (*) we obtain $\neg\psi_j \to \varphi_i(\bar{\beta}_j)$. But by definition

$$\varphi_i(\bar{\beta}_j) = \varphi_i(\bar{\alpha}(\bar{q}_j)) = \varphi_j(\bar{q}_j) = \psi_j.$$

Thus ψ_j follows. □

§7. Conclusions, problems, remarks. Apparently it will be difficult to settle whether the statements (1)–(9) are true or not. Thus we would like to pose more accessible problems.

Assume that (1) is not true, which seems to be likely. Then for every fragment T of the true arithmetic there must be a consistent (finitely axiomatized) theory S such that the $\mathrm{Con}_S(\underline{n})$'s do not have polynomially long (in n) proofs in T. Is it possible to construct S from T using only the information that (1) is false? We state it as our first problem.

PROBLEM 1. Find a construction of a consistent finitely axiomatized theory $S(T)$ from a fragment (finitely axiomatized) T of the true arithmetic such that the following is true: If $\mathrm{Con}_{S(T)}(\underline{n})$ have polynomially long proofs in T, then, for every consistent (finitely axiomatized) theory S, $\mathrm{Con}_S(\underline{n})$ have polynomially long proofs in T.

Natural candidates for $S(T)$ are $T + \mathrm{Con}_T$ (cf. Problem 1(1) in [Pu1]) or the "jump of T" (cf. [Bu]).

One can ask analogous questions for propositional proof systems. The proofs of this paper suggest the following candidate of a set of propositions which do not have polynomially long proofs in a given system. Given P, let $t_n \rightleftarrows t_{\mathrm{Con}_{S(P)},n}$ (cf. §3).

PROBLEM 2. Do the t_n's have polynomially long proofs in P?

For similarly constructed propositions Cook conjectured in [Co] that they do not have short proofs in the extended resolution. A construction of propositional formulas from bounded arithmetical formulas was used also in [PW1].

Let $S(P)$ ($P(T)$) be the theory (the propositional proof system) constructed from P (from T) in §3. It would be interesting to find out that $S(P)$ is some familiar theory for some particular propositional proof system P, and similarly for $P(T)$. However, it follows from the results of §5 that for ordinary propositional proof systems $S(P)$ is a very weak theory. Namely $S(SF)$ (SF is the Frege system with substitution) is contained in S^1_2, and S^1_2 proves that SF polynomially simulates other usual propositional proof systems (Gentzen systems, the extended resolution, extended Frege systems). On the other hand $P(T)$ seems to be a very powerful propositional proof system already for weak fragments of arithmetic. Since $S(SF) \subseteq S^1_2$, $P(S^1_2)$ polynomially simulates all ordinary propositional proof systems. We are not able to prove that it is strictly stronger, but there is evidence for it. Namely, the consistency of $P(T)$ (as it is defined) entails the consistency of T, but already the consistency of a very weak theory, Robinson's arithmetic, is not provable in bounded arithmetic with exponentiation.

We may define the quasiordering of propositional proof systems less effectively. Put $P \leq' Q$ iff for any polynomial $q(x)$ there is a polynomial $p(x)$ such that for any tautology t if Q accepts t in time $\leq q(|t|)$ then P accepts t in time $\leq p(|t|)$. It is easy to construct P, Q such that $P \leq' Q$ but *not* $P \leq Q$. But there is a $\leq'$-least system iff there is a $\leq$-least one (the latter implies the former and the former implies (3)).

Statements similar to (8) and (9) but different have been studied elsewhere. For instance let us define, for a deterministic Turing machine M and a set A accepted by M,

$$f_M(n) = \max_{|w| \leq n, w \in A} \mathrm{TIME}(M; w).$$

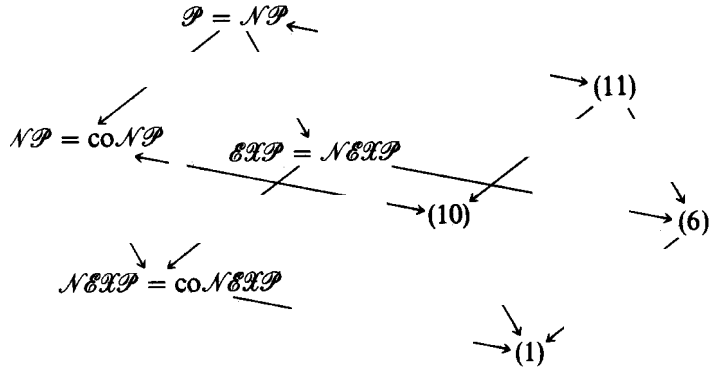
Then the well-known construction of Levin [Le, Theorem 2], gives an M for an $\mathcal{NP}$-complete set A such that for any other deterministic Turing machine M' which recognizes A, $f_M(n)$ can be bounded by a polynomial in $f_{M'}(n)$. There is no simple relation between this result and our statements.

The statement (6) is a strengthening of (1). There are at least two other straightforward strengthenings of (1):

(10) There is a finitely axiomatized fragment T of the true arithmetic and a polynomial $p(x, y)$ such that for any consistent finitely axiomatized theory S and any $n \in N$ there is a proof of $\mathrm{Con}_S(\underline{n})$ in T of length $\leq_p(|S|, n)$.

(11) There is a finitely axiomatized fragment T of the true arithmetic, a deterministic Turing machine M and a polynomial $p(x, y)$ such that for any consistent finitely axiomatized theory S and any $n \in N$, M constructs from (S, n) a proof of $\mathrm{Con}_S(\underline{n})$ in T in time $\leq p(|S|, n)$.

It is easy to prove that $\mathcal{NP} = \mathrm{co}\mathcal{NP} \Leftrightarrow$ (10) and $\mathcal{P} = \mathcal{NP} \Leftrightarrow$ (11). Thus the following diagram holds:



So it is natural to pose the following questions.

PROBLEM 3. Does (1) $\Rightarrow \mathcal{NEXP} = \mathrm{co}\mathcal{NEXP}$? Does (6) $\Rightarrow \mathcal{EXP} = \mathcal{NEXP}$? Find, at least, oracles relative to which the first or the second of these implications is not true.

Note that in [BGS] an oracle A has been constructed such that $\mathcal{P}^A \neq \mathcal{NP}^A$ and $\mathcal{NP}^A = \mathrm{co}\mathcal{NP}^A$, and that in [Wi] oracles B and C were found such that $\mathcal{P}^B \neq \mathcal{NP}^B$ and $\mathcal{EXP}^B = \mathcal{NEXP}^B$, and $\mathcal{EXP}^C \neq \mathcal{NEXP}^C$ and $\mathcal{NEXP}^C = \mathrm{co}\mathcal{NEXP}^C$.

Observe that in (1), (6), (10) and (11) we need only Δ_0-soundness of T (cf. construction (a1)). Since Robinson's arithmetic Q proves all true Σ_1^0-sentences, the assumption "T is a finite fragment of the true arithmetic" can be equivalently replaced by "T is a finite consistent extension of Q". Moreover we can assume that $T = Q \cup \{\varphi\}$, for some Π_1^0-sentence φ.

REFERENCES

[BGS] T. BAKER, J. GILL, and R. SOLOVAY, *Relativizations of the $\mathcal{P} =? \mathcal{NP}$ question*, ***SIAM Journal on Computing***, vol. 4 (1975), pp. 431–442.

[Bu] S. R. BUSS, ***Bounded arithmetic***, Bibliopolis, Naples, 1986.

[Co] S. A. COOK, *Feasibly constructive proofs and the propositional calculus*, ***Proceedings of the seventh annual ACM symposium on the theory of computing***, 1975, pp. 83–97.

[CR] S. A. COOK and R. A. RECKHOW, *The relative efficiency of propositional proof systems*, this JOURNAL, vol. 44 (1979), pp. 36–50.

[Do] M. DOWD, *Model-theoretic aspects of $\mathcal{P} \neq \mathcal{NP}$*, preprint, 1985.

[Fr] H. FRIEDMAN, *On the consistency, completeness and correctness problems*, preprint, Ohio State University, Columbus, Ohio, 1979.

[Ha] A. HAKEN, *The intractability of resolution*, ***Theoretical Computer Science***, vol. 39 (1985), pp. 297–308.

[KP] J. KRAJÍČEK and P. PUDLÁK, *Quantified propositional calculi and fragments of bounded arithmetic*, ***Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*** (to appear).

[Le] L. A. LEVIN, *Universal sorting problems*, ***Problemy Peredači Informacii***, vol. 9 (1973), no. 3, pp. 115–116; English translation, ***Problems of Information Transmission***, vol. 9 (1973), pp. 265–266.

[PW1] J. B. PARIS and A. J. WILKIE, *Counting problems in bounded arithmetic*, ***Methods in mathematical logic (proceedings, Caracas, 1983)***, Lecture Notes in Mathematics, vol. 1130, Springer-Verlag, Berlin, 1985, pp. 317–340.

[PW2] ———, *On the scheme of induction for bounded arithmetic formulas*, ***Annals of Pure and Applied Logic***, vol. 35 (1987), pp. 261–302.

[Pu1] 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, 1986, pp. 165–196.

[Pu2] ———, *Improved bounds to the length of proofs of finitistic consistency statements*, ***Logic and combinatorics*** (S. G. Simpson, editor), Contemporary Mathematics, vol. 65, American Mathematical Society, Providence, Rhode Island, 1987, pp. 309–332.

[Sa] J. SAVAGE, ***The complexity of computing***, Wiley, New York, 1976.

[W] A. J. WILKIE, *Subsystems of arithmetic and complexity theory*, Lecture at the eighth international congress of logic, methodology and philosophy of science, Moscow, 1987.

[Wi] G. B. WILSON, ***Relativization, reducibilities and the exponential hierarchy***, Technical Report No. 140, University of Toronto, Toronto, 1980.

MATHEMATICAL INSTITUTE
CZECHOSLOVAK ACADEMY OF SCIENCES
115 67 PRAGUE 1, CZECHOSLOVAKIA