# A LOGICAL APPROACH TO THE PROBLEM "P=NP?"

V.Yu.Sazonov

Institute of Mathematics, 630090, Novosibirsk, USSR

INTRODUCTION. A possible approach to the problem "P=NP?" or to the problem of exponential lower time complexity bound for NP-complete sets or, more generally and informally, to the so called enumeration problem [1, 8] *) consists in the analysis of those logical means and principles that the problem depends on. This may be 1) a technical analysis of logical and mathematical means and 2) an analysis of mathematical abstractions connected with the problem.

The first attempt of such a technical logical analysis concerned with "P=NP?" was undertaken by J.Hartmanis and J.E.Hopcroft [6]. They proved that in any sufficiently strong axiomatic system Ax some version $(P=NP)_{Ax}$ of P=NP is independent of Ax. Note that the original assertion P=NP may also be written down in the language Ax, and in the formuletion of $(P=NP)_{Ax}$ the provability predicate $Prf_{Ax}$ is used in some way. From an external point of view both assertions are equivalent, and if this equivalence were provable in Ax, then we should have the independence of P=NP itself in the ordinary sense. However, this, obviously, cannot be made for all Ax. There is also no hope to prove such equivalence for some fixed Ax. So, independence of the original problem is not proved.

The Main Result of this paper states, for some natural theory T, unprovability of the assertion on exponential lower time complexity bound of accepting NP-sets (Theorems 1.1, 1.3; the result announced in [13]). But, as opposed to [6] a direct formulation of the problem is used. Since the theory T is rather weak, this result may only be considered as a partial solution of the problem in terms of axiomatic independence. However, the difficulty of proving the Main Result ( §5; compare with 3.8, 3.9 and §6) says that T is strong enough.

*) Unfortunately, the author knows no short, adequate and non-occupied English term for such an important and rather informal problem. It is written in Russian as "проблема перебора". Other English equivalents may be "exhaustive searching problem" or "trial problem". Note that a broader (and earlier) understanding of this problem also arises from [3,16,18]. In this paper the term "enumeration problem" will be appropriately specified.



Questions of technical logical analysis concerned with the complexity theory have also been considered recently by R.J.Lipton, R.A. De Millo, M.O'Donnell [9 - 11]. Besides other facts they obtained only conditional independence results for the enumeration problem.

As to the analysis of abstractions, it seems very important that some specific kind of mathematical abstraction is really and essentially connected with the enumeration problem (and with the computer practice too). § 7 is devoted to this question. There, in particular, the enumeration problem is stated as the problem of "constructive" understanding the sense of bounded quantifiers on finite binary strings. Such an approach to this problem is basic for all this work.

More generally, the main idea of this paper consists in replacing complexity restrictions by logical ones. Note that first steps in this direction were taken evidently by J.H.Bennett (see [5]), N.D.Jones, A.L.Selman and R.Fagin [5,7]. They reformulated polynomial time computability in logical terms. And we do this too (see 7.2). The main novelty of our approach is in attracting appropriate deductive apparatus (theories T, BA, UBA) and finitary mathematical considerations ( § 7).

For the lack of space, many proofs are omitted.

I wish to express gratitude to prof. B.A.Trakhtenbrot, M.K.Valiev, M.I.Dekhtjar' and N.V.Beljakin for discussion and valuable remarks.

## §1. THEORIES T AND BA.

Let $p_1, p_2, \ldots$ be all polynomial time computable functions and predicates (or shortly P-functions and P-predicates) on the set $\{0,1\}^*$ of finite binary strings. For example:

$x = y$ - equality of binary strings;

$\Lambda$ - the empty string constant;

$\sigma_0(x)$, $\sigma_1(x)$ - functions of concatenating zero and unit to x;

$Un(x)$ - x is unary string;

$|x|$, $|x|^2$ - the length and the square of the length of the string x;

Note, that here unary strings play the role of natural numbers. So, $\forall x(Un(|x|) \& Un(|x|^2))$ is true. The ordinary binary coding of natural numbers (by the function $B(i)$ below) is nonadequate for this paper because of exponential complexity of decoding.

$x'$, $x<y$ - lexicographical successor and order ($\Lambda < 0 < 1 < 00 < 01 \ldots$);

$B(i)$ - the $|i|$-th binary string in the lexicographical ordering;

$x \prec y$ - x is an initial part of y;

$\langle x,y \rangle$, $\langle\langle i,j \rangle\rangle$ - binary and unary codes of ordered pairs of binary and unary strings, respectively;

$x[y]$ - the value of the propositional formula x on the string y;

$EXP(x,y)$ - the predicate $2^{|x|} \leq |y|$;

$M(z,x,t)$, $M_2(z,x,y,t)$ - functions denoting the results of computations of Turing machine (T.M.) $z$ with inputs $x$ and $x,y$, respectively, within $\leq |t|$ steps, the result being $\Lambda$ if $z$ does not halt in $|t|$ steps.

Below we also introduce notations for other P-functions and predicates. But for their termal representation those given above are sufficient.

Let also $\mathbb{P}$ denote the model $(\{0,1\}^*, p_1, p_2, \ldots)$;

$\Delta$ - the set of all first-order formulas with bounded quantifiers $\forall x < a$, $\exists x < a$ only, $x$ does not occur in P-term $a$. $\Delta$-formulas, of course, may contain free variables;

$T = \{A \in \Delta \mid \mathbb{P} \models A\}$;

$T_o = \{A \in T \mid A \text{ contains no qantifiers}\}$.

Let us show by examples the expressive power of the language. So the predicate $SAT(x) = \exists y.x[y]$ represents the set of satisfiable propositional formulas. For any P-function $q(x)$ we denote by $(P = NP)_q$ the sentence $\forall x\,(q(x) \leftrightarrow SAT(x))$. By virtue of NP-completeness of SAT the proposition $P = NP$ is true iff for some $q$ $\mathbb{P} \models (P = NP)_q$. In this very sense the proposition $P = NP$ is expressible in our language. The formula $ACCEPT(z) = \forall x(\exists t M(z,x,t) \neq \Lambda \leftrightarrow SAT(x))$ means obviously, that T.M. $z$ accepts SAT. The following proposition ELB asserts the existence of exponential lower time complexity bound for any algorithm accepting SAT:

$$ELB = \forall z\{ACCEPT(z) \rightarrow \forall x \exists y > x[SAT(y) \,\&\, \forall t(M(z,y,t) \neq \Lambda \rightarrow EXP(y,t))]\}.$$

Now we can formulate the Main Result of the paper. Let EXP be $\forall x \exists y\, EXP(x,y)$ and ACCEPT be $\exists z\, ACCEPT(z)$, both being true in $\mathbb{P}$.

1.1. MAIN RESULT. $T + ACCEPT \not\vdash EXP$. (In particular, $T \not\vdash EXP$.)

Using the following easy

1.2. PROPOSITION. $T_o + ELB + ACCEPT \vdash EXP$ (see Appendix)

we immediately obtain a weakened form of the Main Result:

1.3. THEOREM. $T \not\vdash ELB$, i.e. the lower exponential time complexity bound for accepting SAT is not provable in T.

Note that the Main Result may be trivially strengthened by replacing EXP e.g. by $EXP' = \forall x \exists y\,(2^{\sqrt{|x|}} \leq |y|)$ because $T_o \vdash (EXP \leftrightarrow EXP')$ (see also 6.3).

As the proof method of the Main Result (§ 5) is rather finitary and is applied to suitable axiomatic fragments of T, we introduce here one such important fragment BA (Bounded Arithmetic). Fix its non-logical axioms (below a, b, c are arbitrary terms):

$a = b \,\&\, a = c \rightarrow b = c$, $a = a$, $\sigma_i a = \sigma_i b \rightarrow a = b$ $(i \in \{0,1\})$,



$$|a| \begin{cases} \sigma_0 a \neq \sigma_1 b, \\ |\Lambda| = \Lambda, \\ |\sigma_0 a| = |\sigma_1 a| = \sigma_1 |a|, \end{cases} \qquad a' \begin{cases} \Lambda' = 0, \\ (\sigma_1 a)' = \sigma_0(a'), \\ (\sigma_0 a)' = \sigma_1 a, \end{cases}$$

$\ldots$

$M(a,b,c)$ { $\ldots$

Lex. Ind. $A(\Lambda) \,\&\, \forall x < a[A(x) \rightarrow A(x')] \rightarrow A(a)$ $(A \in \Delta,\ x \notin a)$.

It is clear that for giving functions and predicates enumerated at the begining of the paragraph it suffices to have a finite number of recursive descriptions containing P-functions and P-predicates only. We do not write out them completely as it would be rather cumbersome.

Thus, BA means a sufficiently strong system containing a finite number of obvious quantifier-free axioms and also the lexicographical induction. And a hope is connected with BA that all or most of "obvious" $\Delta$-propositions are provable in BA (see § 7 where a finitary-mathematical content of the theory BA is discussed).

The schema of lexicographical induction may be proved to be equivalent in BA to the following two ones:

$$A(\Lambda) \,\&\, \forall x[A(x) \rightarrow A(x')] \rightarrow A(y) \qquad (A \in \Delta),$$

Lin. Ind. $A(\Lambda) \,\&\, \forall x[A(x) \rightarrow A(\sigma_0 x) \,\&\, A(\sigma_1 x)] \rightarrow A(y)$ $(A \in \Delta)$.

Note, that the last principle of linear induction, if taken on quantifier-free A, is easily provable in the quantifier-free theory $T_o$. Such a variant of induction is strong enough to prove the most of (if not all) "obvious" quantifier-free formulas. But this principle is probably weaker than that of lexicographical induction even on quantifier-free formulas. The reason consists essentially in the fact that we can not arrive at a good constructive (see 7.4) understanding of lexicographical induction as opposite to the linear one on quantifier-free formulas.

To obtain in § 5 the proof of the Main Result 1.1 and discuss in § 7 the finitary-mathematical content of the theory BA we need

1.4. THEOREM. Let Ax be a set of $\Delta$-formulas closed under substitution of terms. Then any proof of a $\Delta$-formula A from axioms Ax (in particular, from BA or T) may be reconstructed into a proof containing $\Delta$-formulas only ($\Delta$-proof $Ax \vdash A$ written also as $Ax \rhd A$).

PROOF is based on a suitable generalization of the Cut-elimination theorem for the Gentzen sequent calculus LK [17](see Appendix).

## §2. OPTIMAL CODING OF BINARY STRINGS

In theories, in which feasibility of exponentiation is not provable (e.g. in T), binary strings can not be identified with unary ones by the usual coding $B(i)$ (see § 1). This follows from

2.1. PROPOSITION. $T_0 \vdash (EXP \leftrightarrow \forall x \exists i\ (x = B(i)))$ □

Nevertheless, such economical coding is possible which is exponentialless in a definite sense (see §§ 3,5).

2.2. DEFINITION. P-function $\xi^x_j$ of unary argument j ( $\xi^x_j = \xi^x_{|j|}$) and of binary - x is called polynomially optimal sequence (p.o.s. $\xi^x_\Lambda, \xi^x_1, \xi^x_{11}, \xi^x_{111}, \ldots$) if for any sequence $\alpha^x_j$ computable (by a T.M.) in a time $t(j,x)$ there exists a P-function $p(j,x,t)$ with unary values and arguments j and t such that for any j,x

$$\alpha^x_j = \xi^x_{p(j,x,t(j,x))}.$$

If, in particular, $\alpha^x_j$ does not depend on x or j, then, respectively,

2.2.1. (a) $\alpha_j = \alpha^x_j = \xi_{p_1(j,t_1(j))}$, where $\xi_j = \xi^\circ_j$, or

(b) $\alpha^x = \alpha^x_j = \xi^x_{p_2(x,t_2(x))}$ hold for suitable $p_1$ and $p_2$.

If $\alpha^x_j$ is P-function, then we obtain respectively

2.2.2. $\alpha^x_j = \xi^x_{q(j,x)}$, $\alpha_j = \xi_{q_1(j)}$, $\alpha^x = \xi^x_{q_2(x)}$.

So, p.o.s. $\xi^x_j$ gives the "quickest" up to polynomials, enumeration of binary strings. It catches up with any $\alpha^x_i$, if the time $t_a(i,x)$ is taken into account.

2.3. THEOREM. There exists p.o.s. $\xi^x_j$ which may be chosen even injective (i.e. such that $T_0 \vdash |i| \neq |j| \rightarrow \xi^x_i \neq \xi^x_j$). Moreover, if $\alpha^x_j$ is a P-sequence (or a P-term), then equalities 2.2.2 are in $T_0$.

PROOF. Let us first define a sequence $\tilde\xi^x_n$ by $\tilde\xi^x_{\ll i,j,t\gg} \rightleftharpoons \tilde\xi(i,j,t,x) \rightleftharpoons M_2(B(i),j,x,t)$. Let $B(i_0)$ be a program for $\alpha^x_j$ with sutable unary $i_0$. Then, $\alpha^x_j = M_2(B(i_0),j,x,t(j,x)) = \tilde\xi(i_0,j,t(j,x),x) = \tilde\xi^x_{\ll i_0,j,t(j,x)\gg} = \tilde\xi^x_{\tilde p(j,t(j,x))}$ where $\tilde p(j,t) \rightleftharpoons \ll i_0,j,t\gg$ is P-function. The required injective sequence $\xi^x_n$ may be trivially extracted as subsequence of $\tilde\xi^x_n$ so that $\tilde\xi^x_n = \xi^x_{q(n,x)}$ holds for some P-function q. Hence, $\alpha^x_j = \tilde\xi^x_{\tilde p(j,t(j,x))} = \xi^x_{q(\tilde p(j,t(j,x)),x)}$. If $t(j,x)$ is a polynomial of $|j|$ and $|x|$ then, these equalities are true in $\mathbb{P}$ and lie in $T_0$ □

2.4. PROPOSITION. $T_0 + EXP \vdash \forall x \exists i(x = \xi_i)$, $\mathbb{P} \models \forall x\, \exists i(x = \xi i)$.

PROOF follows from 2.1 and from optimality of $\xi_i$: $T_0 \vdash B(i) = \xi_{p(i)}$

So, feasibility of exponentiation implies a possibility of optimal coding $\xi_i$ of <u>all</u> binary strings by unary ones. And, what is more interesting, from §§ 3,5 it follows that this possibility is also compatible with the assumption $\neg EXP$.

## § 3. DETERMINISTIC AND RANDOM FINITE STRINGS

In any theory where EXP is not provable the following abbreviations may be reasonably introduced (cf. 2.4).



3.1. DEFINITION. "x is deterministic string" $\rightleftharpoons DET(x) \rightleftharpoons \exists i(x = \xi_i)$, $DET^z(x) \rightleftharpoons \exists i(x = \xi^z_i)$ (relativization), $DET \rightleftharpoons \forall x DET(x)$, $DET^z \rightleftharpoons \forall x DET^z(x)$,

"x is random string" $\rightleftharpoons RAND(x) \rightleftharpoons \neg DET(x)$.

This definition may be considered as a formalization of the informal notion of a <u>very complex</u> finite binary string by A.N.Kolmogorov. The following simple facts show that $DET^z(x)$ is very natural notion.

3.2. PROPOSITION. $T_0 \vdash Un(i) \,\&\, Un(j) \,\&\, M(B(i),j,z,t) = x \rightarrow DET^z(x)$ □

3.3. COROLLARY. $T_0 \vdash DET^z(z)$, $T_0 \vdash Un(j) \rightarrow DET(j)$ □

3.4. PROPOSITION. $T_0 \vdash DET^z(x) \,\&\, Un(i) \rightarrow DET^{\langle z,y\rangle}(p(i,x,y))$, where p is any term of i,x,y □

3.5. COROLLARY. $T_0 \vdash x \prec y \,\&\, DET^z(y) \rightarrow DET^z(x)$ □

3.6. REMARK. Corollary 3.5 fails if in its formulation $\prec$ is replaced by the lexicographical order $<$.

In § 4 we will show how the hypothesis DET is connected with the possibility of optimal accepting of SAT. We have alredy shown that DET follows from EXP (2.4). But the nonexponential case is more interesting. We will prove in § 5 "nonexponentialness" of the coding $\xi_i$:

3.7. THEOREM. $T + DET \nvdash EXP$.

But a weaker result may be alredy proved.

3.8. THEOREM. $T_0 + DET \nvdash EXP$.

PROOF. Let $\mathbb{M}$ be an arbitrary nonstandard (i.e. $\neq \mathbb{P}$) model of $T_0$, and let $n \in \mathbb{M}$ be a nonstandard unary string (i.e. $\mathbb{M} \models Un(n)$, $n > \Lambda$, $n > 1$, $n > 11, \ldots$). Consider the submodel $\mathbb{D}$ of $\mathbb{M}$ with the carrier $\{x \in \mathbb{M} \mid \mathbb{M} \models \exists i(Un(i) \,\&\, x = \xi_i \,\&\, i < |n|^k)$ for some $k = 0,1,2,\ldots\}$. As follows from 3.4, $\mathbb{D}$, indeed, is closed under operatios of the model $\mathbb{M}$. Obviously $T_0$, DET and $\forall y \neg EXP(n,y)$ hold in $\mathbb{D}$ □

3.9. REMARK. The model-theoretic proof given above also does for stronger variants of Theorem 3.8 such as $T_0 + DET \nvdash \forall x \exists y(|x|^{\log|x|} < |y|)$ and so on, but not for the case with 3.7 (we can not prove $\mathbb{D} \models T$; cf. 3.6). The latter case forces us to make use of §5 proof-theoretic methods.

The following theorem is also proved model-theoretically.

3.10. THEREM. The theories $T + \neg\exists z DET^z$ $(+\neg EXP)$ and $T_0 + \exists z(RAND(z) \,\&\, DET^z)$ $(+\neg EXP)$ are both consistent □

3.11. OPEN QUESTION. Whether $T + \exists z(RAND(z) \,\&\, DET^z)$ or, equivalently, $T + \neg DET + \exists z DET^z$ is consistent? (See also the end of 7.3)

## § 4. OPTIMAL ALGORITHM ACCEPTING SAT

We take p.o.s. $\xi_i$ of § 2 as the main part of the optimal algorithm accepting SAT. Below we will make use of the symbol Pol for polynomials with natural coefficients.

4.1. Note first, that if for $x \in SAT$ we can deterministically compute in a time $t(x)$ a "correct guess", i.e. such a word $g(x)$ for which $\mathbb{P} \models x[g(x)]$, then by 2.2.1 (b) and 2.4 we find some "correct guess" (which may be $\neq g(x)$) also by means of enumeration $\xi_\Lambda^x$, $\xi_1^x$, $\xi_{11}^x$,... in the time $Pol(|x|, t(x))$ (or $Pol(t(x))$, if we consider $t(x) > |x|$). An analogous result was given earlier by L.A.Levin in [8], but without explicit indicating optimal algorithm and also without any definition of optimal coding.

Let us pass from search algorithms of "correct guess" to the ones accepting SAT. Fix any computable total function $\alpha_i^x$. Let $\mathcal{A}_\alpha$ denote the binary code of the algorithm, which, for any given propositional formula $x$, computes in turn its values on the strings $\alpha_\Lambda^x$, $\alpha_1^x$, $\alpha_{11}^x$,... and stops iff the true value $x[\alpha_i^x]$ is obtained.

Let $\mathcal{A}$ and $\mathcal{A}_0$ be $\mathcal{A}_\alpha$ for $\alpha = \xi_i^x$ and $\xi_i$ respectively. By 2.4 it obviously follows that $\mathbb{P} \models ACCEPT(\mathcal{A}) \,\&\, ACCEPT(\mathcal{A}_0)$. Consider running times for just described algorithms: $\tau_\alpha(x)$, $\tau(x)$, $\tau_0(x)$. The following simple fact was drawn to the author's attention by R.Fiby (Bratislava).

4.2. PROPOSITION. If $\alpha_i^x$ does not depend on $x$ and enumerates all binary strings (which is equivalent to acceptance SAT by $\mathcal{A}_\alpha$), then a lower exponential bound for $\tau_\alpha$ takes place:

$$\exists k \; \exists^\infty x \quad 2^{k\sqrt{|x|}} \leqslant \tau_\alpha(x).$$

in particular, it holds for $\tau_0(x)$ □

Optimality of $\xi$ easily implies $\mathcal{A}$ to be the "best" algorithm accepting SAT, if compared with algorithms $\mathcal{A}_\alpha$:

4.3. PROPOSITION. $\tau(x) \leqslant Pol_\alpha(\tau_\alpha(x))$ □

And this result may be strengthened to absolute optimality of $\mathcal{A}$:

4.4. THEOREM. If SAT is deterministically accepted (by a T.M.) in a time $t(|x|)$ (for $x \in SAT$), then SAT is accepted by $\mathcal{A}$ in the time $\tau(x) \leqslant Pol(t(|x|))$ □

In spite of 4.2 we can say that the running time $\tau_0(x)$ of $\mathcal{A}_0$ is in some sense less than the exponentiation:

4.5. THEOREM. $T + ACCEPT(\mathcal{A}_0) \nvdash EXP$.

This follows from the Theorem 3.7 and from the obvious

4.6. PROPOSITION. $T_0 + DET \vdash ACCEPT(\mathcal{A}_0)$ □

So, we conclude, that unprovability in $T$ of the lower exponential time complexity bound for accepting SAT (or EL8; see 1.1-1.3) may be obtained from 3.7.



## § 5. PROOF OUTLINE OF THEOREM 3.7: T + DET ⊬ EXP.

In this paragraph we consider terms and formulas solely from the first-order language $L_\nu$ which only contains finite number of nonlogical symbols for sufficiently many P-functions and P-predicates $p_1$, ..., $p_k$, $\Lambda$, $\sigma_0$, $\sigma_1$, $=$, $\prec$ (e.g. as for BA) and for new (non-P-) function $\nu$ inverse to the P-function $\xi_i$. We relativize the notions $\Delta$ and $\rhd$ (cf. 1.4) to $L_\nu$ with the following restriction (without which 5.1 fails): in $\Delta$-formulas variables in the scope of any $\nu$ must be free. Let $\mathbb{P}_\nu$ be $(\mathbb{P}, \nu)$ and $T_\nu$ be $\{A \in L_\nu \mid A \in \Delta \text{ and } \mathbb{P}_\nu \models A\}$. Theorem 3.7 in question then follows from the next one becouse $T \subseteq T_\nu \vdash DET$.

5.1. THEOREM. $T_\nu \nvdash EXP$.

PROOF. The idea is simple. EXP implies deductive consistency of some subsets $\widetilde{Ax}$ of $T_\nu$, $T_\nu$ being the union of these $\widetilde{Ax}$'s. So, $T_\nu \vdash EXP$ would give $\widetilde{Ax} \vdash Consis_{\widetilde{Ax}}$ for some $\widetilde{Ax}$, which is impossible (Gödel theorem; see e.g. [17], §10). Note that such an idea requires to state a relation between proofs and truthness of the proved theorems. But we can do this for $\Delta$-proofs (with the restriction above) only, which in fact turns out to be quite sufficient. This idea is realized in the formal proof outlined below.

5.2. LEMMA. The function $a \mapsto value(a)$ naturally defined on the set of all closed terms $a$ (from $L_\nu$) is elementary.

PROOF is based on polynomial optimality of $\xi_i$ (see Appendix).

5.3. DEFINITIONS. Fix some natural encoding for $\Delta$-proofs $U$ (and for $\Delta$-formulas $A$, being the trivial proofs) by closed terms in the signature $\{\Lambda, \sigma_0, \sigma_1\}$. The code of $U$ ($A$) will be written as $\ulcorner U \urcorner$ ($\ulcorner A \urcorner$). P-decidable sets $S$ of formulas will be identified with the corresponding open formulas $S(w)$ of one free variable. Introduce several notions which are evidently expressible in $L_\nu$ by open formulas or terms (without $\nu$):

1°. $Prf_S(u,v) \rightleftharpoons$ "$u$ is a $\Delta$-proof $S \rhd v$", the formula $S$ being suitably inserted in that for $Prf_S$;

2°. $PC(w,v) \rightleftharpoons$ the formula $w$ is a partial case of the formula $v$";

3°. $True(u,x,t) \rightleftharpoons$ "the value of $\Delta$-formula $u$ on argument(s) $x$ is true and this fact is verified within $\leqslant |t|$ steps by an appropriate T.M. $\mathcal{M}(u,x)$ evaluating $\Delta$-formulas";

4°. $Ref_\nu(u,x,t) \rightleftharpoons Ref(v,u,x,t) \rightleftharpoons$ "the result (if any) of the following computation within $\leqslant |t|$ steps by an appropriate T.M. $\mathcal{N}(v,u,x)$:

BEGIN given $\Delta$-proof $u$, compute truth value of its final formula with argument evaluation $x$;

IF this value is true THEN let RESULT be $\Lambda$ ELSE seek Refutation

to certain preceeding formulas of the proof u following the proof rules in u in the inverse direction up to some axiom w of u is refuted;

IF PC(w,v) THEN let RESULT be such an argument evaluation (obtained from the Refutation to w) for which the formula v is false ELSE let RESULT be $\Lambda$ END";

5°. Halt(u,x,t) and $\mathrm{Halt}_\nu(u,x,t)$ mean that T.M. $\mathcal{M}(u,x)$ and $\mathcal{M}(v,\mathcal{M}(v,u,x))$, respectively, halt in $\leq |t|$ steps.

5.4. LEMMA. There exists P-decidable set $Ax_0 \subseteq T_\nu$ such that for any P-decidable S and any B in $\Delta$ the following five conditions there hold:

(1) $Ax_0 \rhd (B(x) \,\&\, \mathrm{Halt}(\ulcorner B \urcorner, x, t) \leftrightarrow \mathrm{True}(\ulcorner B \urcorner, x, t))$ $\quad (B \neq B(x))$;

(2) $Ax_0 \rhd (\mathrm{Halt}_\nu(u,x,t) \rightarrow \mathrm{Halt}(v, \mathrm{Ref}_\nu(u,x,t), t))$;

(3) if D is proof $S \rhd B$ then $Ax_0 \rhd \mathrm{Prf}_S(\ulcorner D \urcorner, \ulcorner B \urcorner)$;

(4) $Ax_0 + \mathrm{EXP} \vdash \forall v u x \, \exists t(\mathrm{Halt}_\nu(u,x,t) \,\&\, \mathrm{Halt}(u,x,t))$;

(5) given $B \neq B(\bar{x},y)$ such $E \neq E_B(\bar{x})$ in $\Delta$ may be constructed that

$$Ax_0 \rhd (E(\bar{x}) \leftrightarrow B(\bar{x}, \ulcorner E \urcorner)).$$

PROOF. Conditions (1) - (3) are rather trivial. (4) actually follows for sufficiently strong $Ax_0$ from 5.2. Here the above restriction on $\Delta$ is used. (5) is the well-known Gödel diagonal trick (see e.g. [17], §10) and uses several true axioms in $T_0$ □

Let Ax be arbitrary P-decidable and closed under term substitutions extension of $Ax_0$ such that $Ax \subseteq T_\nu$. (5) gives for some F and G

(6) $Ax_0 \rhd (F(x) \leftrightarrow \neg \mathrm{Prf}_{\widetilde{Ax}}(x, \ulcorner F \urcorner))$ and

(7) $Ax_0 \rhd (G(\langle u,v,x,t\rangle) \leftrightarrow (\mathrm{Prf}_{\widetilde{Ax}}(u,v) \,\&\, \mathrm{Halt}_{\ulcorner G \urcorner}(u,x,t) \,\&\, \mathrm{True}(\ulcorner G \urcorner, \mathrm{Ref}_{\ulcorner G \urcorner}(u,x,t), t) \rightarrow \mathrm{True}(v,x,t)))$,

where $\widetilde{Ax}(w) \rightleftharpoons Ax(w) \vee PC(w, \ulcorner G \urcorner)$. Let us prove also that:

(8) $\widetilde{Ax}$ is closed under term substitutions; (9) $\mathbb{P} \vdash G$ and $\widetilde{Ax} \subseteq T_\nu$;

(10) $\widetilde{Ax} \rhd (\mathrm{Prf}_{\widetilde{Ax}}(u,v) \,\&\, \mathrm{Halt}_{\ulcorner G \urcorner}(u,x,t) \rightarrow \mathrm{True}(v,x,t))$;

(11) $\widetilde{Ax} \not\rhd F$; (12) $\widetilde{Ax} + \mathrm{EXP} \vdash F$.

So, (8) is trivial. (9) is directly verified from (7) and truthness of Ax ( $\subseteq T_\nu$). (10) easily follows from (2), from (1), with G in place of B and $\mathrm{Ref}_{\ulcorner G \urcorner}(u,x,t)$ in place of x, axiom G of $\widetilde{Ax}$ and (7). (11): Suppose, otherwise, $\widetilde{Ax} \rhd F$. Then by (3) $Ax_0 \rhd \mathrm{Prf}_{\widetilde{Ax}}(\ulcorner D \urcorner, \ulcorner F \urcorner)$. But by (6) and the assumption we obtain $\widetilde{Ax} \rhd \neg \mathrm{Prf}_{\widetilde{Ax}}(\ulcorner D \urcorner, \ulcorner F \urcorner)$ which give the contradiction with (9). (12): Let, on the contrary, $\neg F(x)$ holds for some x. Then, by (6), $\mathrm{Prf}_{\widetilde{Ax}}(x, \ulcorner F \urcorner)$ holds, too. From EXP, (4) and (10) there follows $\exists t \mathrm{True}(\ulcorner F \urcorner, x, t)$. But (1), with $B \rightleftharpoons F$, gives F(x). The obtained contradiction proves F □

To complete the proof of 5.1, suppose $T_\nu \vdash \mathrm{EXP}$. Then (12) implies $\widetilde{Ax} \vdash F$ for some $\widetilde{Ax}$ described above. But 1.4 (true for our $\Delta$) and (8) give $\widetilde{Ax} \rhd F$ which contradicts (11). So, theorem 5.1 is proved □



## § 6. ON STRENGTHENING THE MAIN RESULT

In spite of our theory T being rather weak, according to generally accepted canons, the proof of the Main Result uses rather involved technique (Cut elimination, Gödel diagonal trick and the notion of optimal sequence). If we didn't achieve a deadlock here then further steps may probably require some considerable efforts.

Let us only indicate three possible directions of strengthening the Main Result $T \not\vdash \mathrm{ELB}$ (or $T + \mathrm{ACCEPT} \not\vdash \mathrm{EXP}$, or $T + \mathrm{DET} \not\vdash \mathrm{EXP}$):

6.1. Add new axioms (e.g. Collection principle $\forall x < a \, \exists y A \rightarrow$ $\rightarrow \exists b \, \forall x < a \, \exists y < b A$ from 7.3 or $\neg \mathrm{DET}$ as in 3.11) to the theory T.

6.2. Replace accepting SAT by its recognizing (at least for $T_0$).

6.3. Lower the exponential bound (say, up to $n^{\log n}$; cf. 3.9 and the remark to 1.3)

In the meantime our attempts to do this give no result. This may mean that the theory T is still strong enough for the axiomatic investigation of the enumeration problem.

However, we would like to have confidence in that we don't simply replace the enumeration problem by other difficult ones, but investigate the essence of the problem. What does this essence mean? The next paragraph outlines an approach to this question by considering the finitary nature of theory BA. (What the theory T presents itself is rather unclear from the finitary point of view.)

## § 7. FINITARY-MATHEMATICAL CONTENT OF THEORY BA

7.1. Mathematics is not thinkable without abstractions. So, the most important abstraction of constructive mathematics is that of potential feasibility. It consists in the possibility to distract ourselves from any finite bounds (more generally, it would be - from any bounds). However, we assert that there is an axiomatic theory, namely BA, rather rich in content, which may be intuitively interpreted without this abstraction. Some other kind of abstraction is used here, call it abstraction of bounded feasibility. Being a kind of potential feasibility negation */, this absraction consists in the possibility to carry reasonings and conceive constructions

---

*/ A negation of potential feasibility was considered earlier by A.S.Yessenin-Volpin [4] (ultraintuitionism). However, the connection of the present considerations (which also may be characterised as "ultra") with those of [4] is not clear. It would be more appropriate to compare our approach with that of S.Cook [2], just in connection with 7.3 below, and with that of R.Parikh [12].

if they are (or may be) relativized to some parameter bounds (in our case - to finite bounds).

This formulation requires some detailed discussion both on informal and formal levels. Here we only outline such a discussion.

It is clear, that the point of view of "relativization to bounds" (or the abstraction of bounded feasibility) corresponds better to the real computer practice than abstractions of potential feasibility or actual infinity. We always deal with some bounds, e.g. in space (memory overfilling), in time, in resources and in many cases these bounds should be rather taken into consideration, and we must not distract ourselves from them. These bounds may be, of course, rather arbitrary. So, we consider them as parametric. (Note, that infinite bounds are often considered as fixed. E.g. such is the set of natural numbers.)

7.2. Consider first <u>what constructions and computations are relativized to (parametric) bounds</u>. The simplest example is a notion of "primitive recursive" functions f over a parametric finite segment $\omega_n = \{0,\dots,n\}$ ($\bar{x} \leq n \Rightarrow f(n;\bar{x}) \leq n$, $\bar{x} \neq x_1,\dots,x_k$, k is fixed for f). The definition of such recursive functions differs from the ordinary one in: 1) the basic successor function $x + 1$ is replaced by the successor function modulo $n + 1$ (so $n + 1 = 0$; one may also take $n + 1 = n$) and 2) the bound n is considered as an additional argument. In passing from primitive recursive descriptions to arbitrary recursive ones relativized to some total functions $g_j: \omega^{k_j} \to \omega$ (in the ordinary case) or $g_j: \omega_n^{k_j} \to \omega_n$ (in our case) we just obtain ordinary partial recursive functions $f(\bar{x},\bar{g})$ relative to $\bar{g}$ or just all polynomial time computablr functions of the kind $\bar{g} \mapsto \lambda \bar{x} \leq n.f(n;\bar{x},\bar{g})$, respectively. But this very naturally leads to the following <u>analogue of Church's thesis</u>:

> the polynomial time computability proves to be just the computability uniformly relativized (or relativizable) to finite parametric bounds.

It seems to the author that an appropriate thesis is necessary for good understanding the real nature of the polynomial time computability (and of the enumeration problem, too). Note that the above representation of P-functions (obtained by the author in 1976 [14]) resembles the known B.A.Trakhtenbrot's representation of computations in finite models [15] and R.Fagin's spectral representation of NP-sets [5].

7.3. Let us pass to discussion of <u>reasonings relativized to bounds</u>. It is probably most natural to formalize them in the ordinary language of second-order arithmetic, but interpreted in the finite parametric



domain $\{0, \dots n\}$. So, quantifiers are taken both on elements and on many place predicates over the domain. Let us dwell on the axiom system UBA (Uniformly Bounded Arithmetic), consisting of several quantifier-free properties of o,n, =, +1, $<$ ($0 \leq x \leq n$, $n+1=0$ and so on), extensionality axioms, induction and comprehension schemes (possibly with a restriction on using predicate quantifiers). There may be some doubts on what logic should be used in carrying out proofs in UBA. But even if in spite of these doubts the classical logic is chosen, it may be asserted that in a reasonable sense the proofs in UBA are uniformly relativized to finite bounds (o and n). Moreover, we advance the hypothesis that

> the system UBA, and the system BA together with it, adequately formalizes the intuitive notion of reasoning uniformly relativized to finite parametric bounds.

We may be convinced in the fact on the ground of the experience in working with the systems UBA and BA. Let us explain why these systems are intuitively equivalent. (It would be better to say that BA is "conservative extension" of UBA). Naturally, only bounded formulas are of our interest. In UBA they are bounded yet and may be easily represented by bounded BA-formulas. By 1.4 unbounded quantifiers in proofs from BA of $\Delta$-formulas may be eliminated. Hence, we may confine ourselves to the provability notion $\rhd$ from 1.4. Taking into account considerations of 7.2 (Fagin's or our representation result) we get intuitive equivalence of the notions "BA$\rhd$" and "UBA$\vdash$" and hence, equivalence of BA and UBA. By the way, remember that naturally arising here provability notion $\rhd$ plays an essential role in the proof of the Main Result. So, our informal considerations are, indeed, helpful. (Note that more general notion of reasonings relativized to bounds is possible. So, there is an essentially new "tautology" - the Collection principle:

$$\forall x \leq a\ \exists y\ A \longrightarrow \exists b\, \forall x \leq a\ \exists y \leq b\ A.$$

If every "truth" is relativized to a bound b, then "b-truthness" of the antecedent should naturally mean just "b-truthness" of $\forall x \leq a\ \exists y \leq b\ A$, so the consequent is also "b-true" together with the antecedent. A formalization of this idea in terms of Kripke truthness will be published elsewhere. The corresponding logical sequent calculus is also obtained for which an analogue of Harrop theorem holds:

$$\vdash \exists x A \implies \vdash \exists x < b\ A \quad \text{for some term } b.$$

This confirms the proofs in such a calculus are, indeed, relativized to bounds. Let $\Delta$-Coll mean Collection principle with A bounded. Then the theorem also holds for T + $\Delta$-Coll with classical logic

(cf. [12] where a weaker result is stated). In particular, $T$ + $+\Delta$-Coll $\nvdash$ EXP. Compare also this and

7.3.1. THEOREM. $T_0 + \Delta\text{-Coll} + \exists z \mathrm{DET}^z \vdash \mathrm{EXP}$, $T_0 + \mathrm{Coll} \vdash \mathrm{EXP}$

with 3.7, 3.10, 3.11 and 4.2 (7.3.1 having been obtained before 4.2)▯.

7.4. Intuitively, the enumeration problem is the problem of constructive (in a new sense) understanding of bounded quantifiers on binary strings (or on predicates over the domain $\{0,\dots, n\}$ . A direct interpretation of bounded quantifiers uses actual enumeration of binary strings, say, in lexicographical order. But we can imagine no uniform (on n) determenistic process of enumeration of all binary strings of the length n, so that this process, as finite discrete object, could be described, say, as a k-dimensional $n \times \dots \times n$-matrix with k independent of n. I.e., we can not relativize such a process to the bound n. That is why bounded quantifiers on binary strings in BA (or in UBA) are used rather formally as opposite to bounded quantifiers on unary strings (see also the discussion on lexicographical and linear inductions in § 1). This leads to the question: what is an arbitrary binary string? (For example, it is rather unclear what is an arbitrary binary string of a thousand of signs). It is worth comparing it with analogous problem of an arbitrary set of natural numbers, i.e. essentially with the continuum problem.

In this paper we have stated that several different answers to the above question are consistent with the theory T+¬EXP (see 3.7, 3.10) all being in terms of determinisity (or constructivity) of finite binary strings (see also the Open Question 3.11).

## APPENDIX

PROOF OF THEOREM 1.2. Let x be arbitrary. Suppose $\mathrm{ACCEPT}(z_0)$ holds for some $z_0$. Then ELB implies $\mathrm{SAT}(y)$ and $\forall t(M(z_0,y,t) \neq \Lambda \rightarrow \mathrm{EXP}(y,t))$ for some $y > x$. $\mathrm{SAT}(y)$ and $\mathrm{ACCEPT}(z_0)$ give $M(z_0,y,t) \neq \Lambda$ and, therefore, $\mathrm{EXP}(y,t)$ for some t. Finally, $\mathrm{EXP}(x,t)$ is obtained from the evident fact $T_0 \vdash \mathrm{EXP}(y,t) \,\&\, x<y \rightarrow \mathrm{EXP}(x,t)$ (the only axiom of $T_0$ needed for this proof). As x is arbitrary, this proves $\forall x \exists t \mathrm{EXP}(x,t)$ □

PROOF OF LEMMA 5.2. Let $\widetilde{f}$ denote the superposition $\lambda i.\nu(f(\xi_i))$ and $\widetilde{c}$ denote $\nu(c)$ for any function f and any constant c. Polynomial optimality of $\xi_i$ easily implies that $\widetilde{f}$ is P-function, if f is. Also such is $\widetilde{\nu}$ (which coincides with $\nu$ on unary arguments). Replace our term a by $\xi_{\widetilde{a}}$, where $\widetilde{a}$ is the result of replacing in a any symbol for function or constant f by that for $\widetilde{f}$. Note that value(a) = = value($\xi_{\widetilde{a}}$), becouse of the identity $\xi_{\nu(x)} = x$, and the process of evaluating non P-term a is contained in that of evaluating



P-term $\xi_{\widetilde{a}}$. But for P-terms of sufficiently large length n (and a fixed finite signature) the time of this evaluation is $\leq n^{k^n}$ for some constant number k. So, the evaluating function in question is elementary □

1.4'. THEOREM. Let Ax be a set of sequents closed under term substitutions. Then Cuts by formulas, which do not belong to sequents of Ax, can be eliminated from any $LK_{Ax}$-proof □

## REFERENCES


1. Cook, S.A., The Complexity of Theorem Proving Procedures, Conf.Record of 3rd ACM Symposium on Theory of Computing, 1971, 151-158.
2. Cook, S.A., Feasibly Constructive Proofs and the Propositional Calculus, 17th Ann.Symp. on Theory of Computing, Conf.Record, May 1975, 83-97.
3. Dekhtjar', M.I., On the impossibility to eliminate the complete enumeration under the computation of functions relatively to their graphs, Doklady Akad.Nauk SSSR, v. 189, 1969, 748-751 (in Russian).
4. Yesenin-Volpin, A.S., An analysis of potential feasibility, Logiceskije issledovanija, Akad.Nauk SSSR, Moscow, 1959, 218-262 (in Russian).
5. Fagin, R., Generalized First-Order Spectra and Polynomial Time Recognizable sets, Complexity of Computations, SIAM-AMS Proc., vol.7, 1974, 43-73.
6. Hartmanis, J., Hopcroft, J.E., Independence Result in Computer Science, ACM SIGACT News, 8 (1976), № 4, 3-24.
7. Jones, N.D., Selman, A.L., Turing Machines and the Spectra of First-Order Formulas with Equality, Proc. 4th ACM Symp. on Theory of Computing, 1972, 157-167.
8. Levin, L.A., Universal Enumeration Tasks, Problemy Peredaci Informacii, IX, 3, 1973, 115-116 (in Russian).
9. Lipton, R., Model Theoretic Aspects of Computational Complexity, Proc. of the 1978 FOCS Symposium, 193-200 (1978).
10. De Millo, R.A., Lipton, R.J., Some connections between mathematical Logic and Complexity Theory, Proc. of the 1979 STOC, 153-159 (1979).
11. O'Donnell, M., A Programming Language Theorem which is Independent of Peano Arithmetic, Proc. of the 1979 STOC, 176-188 (1979).
12. Parikh, K., Existence and Feasibility in Arithmetic, the Journal of Symbolic Logic, vol. 36, № 3, 1971, 494-508.
13. Sazonov, V.Yu., Theory in which lower exponential complexity bound for NP complete tasks is unprovable, 5th All-Union Conf.on math. logic, Institute of Mathem., Novosibirsk, 1979, p. 133 (in Russian).
14. Sazonov, V.Yu., Polynomial computability and recursivity in finite domains, to be published.
15. Trakhtenbrot, B.A., The impossibility of an algorithm for the decidability problem on finite classes, Doklady AN SSSR, 70, № 4(1950), 569-572 (in Russian).
16. Trakhtenbrot, B.A., The formalization of some notions in terms of the complexity of computations, P.Suppes et al., eds., Logic, Methodology and Philosophy of Sciences IV, North-Holl., 1973, 205-213 (in Russian).
17. Takeuti, G., Proof Theory, North-Holland, 1975.
18. Yablonsky, S.V., On the algorithmic difficulties of minimal schemes synthesis, Problemy Kybernetiki, 2, 1959, Moscow, 75-121 (in Russian).