ON EFFECTIVE SPEED-UP AND LONG PROOFS OF TRIVIAL THEOREMS IN FORMAL THEORIES*

J. Hartmanis

TR 75-254

July 1975

Department of Computer Science
Cornell University
Ithaca, New York 14853

---

* This research has been supported in part by National Science Foundation Grants GJ-33171X and DCR 75-09433.

ON EFFECTIVE SPEED-UP AND LONG PROOFS OF TRIVIAL THEOREMS IN FORMAL THEORIES*

J. Hartmanis
Department of Computer Science
Cornell University, Ithaca, NY 14853

Abstract

In this note we give a very simple proof which shows that in many interesting formal mathematical theories, axiomatizable as well as decidable ones, for every given formalization we can effectively find infinite subsets of trivially true theorems which require as long proofs in the given formalism as the hardest theorems of the theory. Thus showing that for these theories every formalism is doomed to be blind to the triviality of infinite sets of theorems, which can be found effectively. Furthermore, it follows that for all (sufficiently large) constructable tape and time bounds there exists sets whose recognition can be effectively speeded up on infinite subsets and that such sets appear naturally. Thus showing that for many concrete problems every algorithm can be effectively sped-up on infinite subsets.

* This research has been supported in part by National Science Foundation Grants GJ-33171X and DCR 75-09433.

I. Informal Introduction

The main purpose of this note is to give a simple proof which shows that in many interesting mathematical theories for every formal (mechanical) proof method, there exists infinite sets of "trivially true" theorems which require as long proofs in the given formalism as the hardest theorems. Surprisingly, some of these infinite sets of trivially true theorems with long proofs, can be found effectively for every given proof procedure and thus the proof porcedure can be replaced by another one which shortens the proofs of infinitely many theorems without increasing the length of proofs for any others. For related work see [2,6].

These results hold for many axiomatizable as well as decidable mathematical theories. They also show that there exist recursive functions and recursive sets for which every algorithm computing them can be effectively speeded-up infinitely often. Furthermore, these recursive functions appear naturally, and we observe that specific combinatorial problems fall in the class of problems for which every algorithm can be effectively speeded up. Thus for many computations there do not exist optimal algorithms-- and every given one can be effectively speeded up infinitely often.

These results show very clearly that we pay a price for formalising mathematics. In every formalization, infinite sets of trivial theorems will require very long proofs. Thus giving a very dramatic and quantitative explanation why we should not and in practice do not freeze a formalism when discussing or doing mathematics. It also gives a warning that a necessarily long proof in a formal system does not certify that the result is non-trivial.

These observations are particularly interesting in view of the recent results about decision complexity of decidable mathematical theories, which show that many of these theories, though decidable, are not practically decidable. For example, it has been shown that for the Pressburger Arithmetic, the first order theory of natural numbers under addition, any proof procedure is such that the length of the proofs cannot be bounded by a double exponential in the length of the theorems [3]. Some even more dramatic complexity results are known about other decidable theories [6]. Our observation adds the postscript, that not only are these theories very hard to decide, but that every decision procedure requires as much work (long proofs) as the hardest theorems in the theory on infinite sets of trivially true theorems and that from the description of the decision procedure, we can effectively obtain such infinite sets of trivially true theorems with long proofs. For related observations see also [6].

It is interesting to compare these results with the Blum Speed-up Theorem [1]. We recall that Blum proved that for every recursive function g there exist recursive functions f for which any algorithm can be speeded up by the function g almost everywhere. Unfortunately, even if a function f has an almost everywhere speed-up, the faster algorithms cannot be found effectively from a given algorithm, nor could one effectively compute from which point on the speed-up started. Finally, there is a wide spread belief that functions with an almost everywhere speed-up are artificially constructed and that they do not appear naturally in computing.

The situation with the functions, problems and mathematical theories which have speed-up infinitely often is quite different. This speed-up is effective and we can explicitely show subsets on which computations can be sped-up. Furthermore, we do not have to construct artificial examples to show existence of such speed-up. The most important mathematical theories have for every formalization proof-shortening infinitely often, and many combinatorial problems which we are interested in solving are such that every algorithm can be sped-up infinitely often. As a matter of fact, mathematical theories and sets which have speed-up infinitely often appear to be more natural than those which do not have speed-up.

Finally, it should be pointed out that it is not at all clear whether we can reap any practical benefits from the fact that for certain specific problems every algorithm for their solution will work unnecessarily hard on an infinite set of simple cases of this problem. We can always find such an infinite set for these problems effectively from the description of the algorithm, but it may turn out that they do not appear often enough in practice to justify replacing the given algorithm by an improved one which we know can never be optimal.

II. Speed-up for Creative Sets

It is well known that the provable theorems of many important mathematical theories form creative sets [5]. One such mathematical theory is Peano Arithmetic, which is based on Peano's axioms with those instances of the Peano induction axiom that can be expressed in elementary arithmetic. Several other well known examples can be easily found, and it is clear that the creative sets form a very important class of sets in mathematics [5].

We recall that a set A, $A \subseteq \Sigma^*$, is creative iff A is recursively enumerable and there exists a recursive function σ such that for every Turing machine M_i, which enumerate a subset of $\bar{A} = \Sigma^* - A$, that is $T(M_i) \subseteq \bar{A}$, $\sigma(i)$ is in $\bar{A} - T(M_i)$. We can also define equivalently, a recursively enumerable set A to be creative iff every other recursively enumerable set B can be one-one reduced to A, i.e. there exists a one-one recursive function f such that w is in B iff f(w) is in A. From these definitions it easily follows that all creative sets are recursively isomorphic, i.e., if A and C are creative sets, $A \subseteq \Sigma^*$ and $B \subseteq \Gamma^*$, then there exists a recursive surjection $f: \Sigma^* \to \Gamma^*$ such that $w \in A$ iff $f(w) \in B$. Thus, but for a recursive translation, all creative sets are the same, and as it will be seen, our results apply to all of them.

In the first part of this paper we will study the difficulty of recognizing creative sets or equivalently the difficulty of proving theorems in mathematical theories, like the Peano Arithmetic. A natural measure of the difficulty of proving a theorem in a formal mathematical theory is the length of the shortest proof. Unfortunately, to do this, we have to know explicitely how the theory is axiomatized and what proof procedures are used. To avoid such unnecessary difficulties we will measure the difficulty of recognizing sets (of provable theorems) by the amount of tape used by Turing machines recognizing these sets. Intuitively, this measure can be justified by observing that for any (reasonable) proof procedure, we can design a Turing machine which for any given theorem successively checks all possible proofs of increasing length until it finds a proof of the given theorem (or never halts if the input is not a provable theorem). Clearly, the amount of tape used by this Turing machine is a lower bound for the length of the proof and corresponds, intuitively, directly to the amount of erasable blackboard space needed to verify the proof or present it.

Consider the set

$$L_0 = \{(i,j,w) \mid i,j \in N,\ w \in \Sigma^*,\ M_i \text{ on } (i,j,w) \text{ uses a finite amount of tape and does not accept } (i,j,w) \text{ or } M_i \text{ uses no less tape than } M_j \text{ on } (i,j,w)\}.$$

We say that M_i uses less tape than M_j on input x iff. M_i uses a finite amount of tape and M_j uses at least one more tape square than M_i on input x.

We now show that the set L_0 is recursively enumerable. We observe that for any Turing machine M_i, input x and integer k we can recursively decide whether M_i for input x uses more than k tape squares. If M_i uses less than k tape squares for input x we can decide whether x is accepted by M_i or not. Thus to enumerate L_0 we just reject all inputs which are not of the form (i,j,w). For input (i,j,w) we check successively for k = 0,1,2,... whether M_i or M_j uses exactly k tape squares for input (i,j,w). If it is found that one of the Turing machines uses a finite amount of tape then we can decide whether M_i uses the same amount or more tape than M_j; if so (i,j,w) is accepted. If M_i uses less tape than M_j then we can decide whether M_i accepts (i,j,w) and it is accepted iff M_i does not accept it.

Lemma 1: The set L_0 is a creative set.

Proof: We know that L_0 is an r.e. set. To show that it is a creative set we will show that any other r.e. set can be recursively one-one reduced to L_0. Let B be an r.e. set and let the Tm M_k accept B by halting on inputs w in B and by using unbounded amount of tape for inputs not in B. Then the 1-1 mapping

$$\sigma(w) = (k, k, w)$$

yields a 1-1 reduction of B to L_0, since M_k halts an w iff $(k, k, w) \in L_0$. Thus L_0 is a creative set.

We now are ready to read off the result showing that every recognition method of L_0 wastes tremendous amounts of tape recognizing trivially recognizable subsets.

Theorem 2: Let M_{i_0} recognize the set L_0, $T(M_{i_0}) = L_0$, and let M_{j_0} be a Tm which halts for all inputs. Then the regular set

$$\{(i_0, j_0, w) \mid w \in \Sigma^*\}$$

is a subset of L_0 and on every input (i_0, j_0, w) M_{i_0} uses no less tape than M_{j_0} on the same input.

Proof: Clearly, for fixed i_0 and j_0,

$$\{(i_0, j_0, w) \mid w \in \Sigma^*\}$$

is a regular set. Since M_{j_0} halts for all inputs we see that on every input (i_0, j_0, w) the Turing machine M_{i_0} uses no less tape than

M_{j_0} on this input. Since otherwise, if M_{i_0} accepts (i_0, j_0, w) using less tape than M_{j_0}, we see that $(i_0, j_0, w) \in T(M_{i_0})$ but not in L_0, contradicting the assumption that $T(M_{i_0}) = L_0$; similarly if M_{i_0} rejects (i_0, j_0, w) using less tape than M_{j_0} on this input we have that $(i_0, j_0, w) \in L_0$ but not in $T(M_{i_0})$; again a contradiction. Thus $\{(i_0, j_0, w) \mid w \in \Sigma^*\}$ is a subset of L_0 and on very element of this subset M_{i_0} uses no less tape than M_{j_0}. This completes the proof.

From this result we immediately get the following.

Corollary 3: For every Turing machine which recognizes L_0 and every recursive function f we can effectively find an infinite regular subset of L_0 on which for each input of length n M_{i_0} uses at least f(n) tape squares for its recognition.

Proof: Immediate from previous theorem. From this result it follows that for every recognizer M_{i_0} of L_0 we can effectively obtain infinite subsets of L_0 on which M_{i_0} uses large amounts of tape but which can be recognized by finite automata, since they are regular subsets.

Considering the members of L_0 as provable theorems of a mathematical theory, the result asserts that every formalization of this mathematical theory will have infinite sets of "trivially" true theorems but which in the formalism require horrendously long proofs.

To illustrate this, assume that we have a first order mathematical theory with the standard logical connectives and in which we can express such concepts as

"M_i uses less tape than M_j on input x"

and

"M_i does not accept the input x"

Assume that the theory is strong enough to prove all true statements of the form

"M_j uses less tape than M_i or input (i, j, w) or M_i uses less tape then M_j and does not accept the input (i, j, w)".

For example, the Peano Arithmetic is such a mathematical theory.

From our previous result we see that any proof procedure for this mathematical theory will have easily recognizable infinite sets of true theorems such that the length of their shortest proofs in the given formalizm will grow faster than any given recursive function (of the length of the theorems to be proved).

We conclude this section by observing that these results hold for all creative sets.

<u>Corollary 4</u>: Let B be a creative set, M_i be a Tm which recognizes B and f as recursive funtions. Then we can effectively find an infinite recursive subset of B which can be recognized by a Tm using L(n) tape but on which the Tm M_i uses $f \circ L(n)$ tape.

<u>Proof</u>: This follows from the fact that all creative sets are recursively isomorphic and that the recursive isomorphic function can be computed on a recursively bounded amount of tape.

The family of recursively enumerable sets which have effectiv speed-up of the type we proved for creative sets has been completely characterized in [2] and Corollary 4 follows from this general characterization, though these proofs are quite complicated.

III. Subrecursive Analogues

Let $L_i(n)$ be the maximal number of tape squares used by the Tm M_i on inputs of length n. We say that a total function $L(n)$ is tape constructable iff there exists a Tm M_i such that $L(n) = L_i(n)$. The following result has been credited to A. Meyer in [6] and is stated there without proof. The auther obtained it independently and only then became aware of its statement in [6].

Theorem 5: For every tape constructable $L(n) \geq n$ there exists a recursive set A_L (a recursive function $f_L: \Sigma^* \to \{0,1\}$) such that

a) A_L can be recognized on $L(n)$ tape but not on $L_1(n)$ tape if

$$\lim_{n\to\infty} \frac{L_1(n)}{L(n)} = 0$$

b) For every M_{i_0} such that $T(M_{i_0}) = A_L$ we can effectively find an infinite regular subset R_{i_0} of A_L such that M_{i_0} uses at least $\frac{L(n)}{|M_{i_0}|}$ tape on every member of R_{i_0}.

Proof: Let $M_1, M_2, \ldots$ be a standard enumeration of all one-tape Tm's such that the set $\{M_i\}$ is recognizable in polynomial time using linear tape. Let $L(n) \geq n$ be tape constructable and let

$$A_L = \{(M_i, w) \mid M_i \text{ does not accept } (M_i, w) \text{ on } \frac{L(|(M_i,w)|)}{|M_i|} \text{ tape}\}$$

where $|M_i|$ denotes the length of the description of M_i. Note that (M_i, w) is in A_L if either M_i uses more than $L(|(M_i,w)|)/|M_i|$ tape or it does not accept the input on this amount of tape. We claim that A_L is $L(n)$ tape acceptable by the following method: for input x check if x is of the form (M_i,w), if not reject. For input (M_i,w) lay off $L(|(M_i,w)|)$ tape and on a separate track of the tape simulate M_i on input (M_i,w): accept it if the simulation tries to use more than $L(|(M_i,w)|)/|M_i|$ tape, if the simulation halts on the available tape reject it if accepted and vice versa. Clearly, this simulation can be carried out on $L(|(M_i,w)|)$ tape and we see that A_L is $L(n)$ tape acceptable.

Let M_{i_0} be a recognizer of A_L and consider the regular set

$$R_{i_0} = \{(M_{i_0}, w) \mid w \in \Sigma^*\}.$$

The Tm M_{i_0} on input (M_{i_0},w) either uses more than $L(|(M_{i_0},w)|)/|M_{i_0}|$ tape or it does not. If it does then (M_{i_0},w) is in $T(M_{i_0})$. If it does not then either (M_{i_0},w) is accepted or not. In either case, we get a contradiction: if (M_{i_0},w) is accepted then (M_{i_0},w) is in $T(M_{i_0})$ but not in A_L contradicting $T(M_{i_0}) = A_L$; if (M_{i_0},w) is not accepted then (M_{i_0},w) is in A_L but not in $T(M_{i_0})$, again contradicting the assumption $T(M_{i_0}) = A_L$. Thus we must conclude that $R_{i_0} \subsetneq A_L$ and that on all inputs (M_{i_0},w) the machine M_{i_0} uses more than $L(|(M_{i_0},w)|)/|M_{i_0}|$ tape. From which it also follows that A_L is not recognizable on $L_1(n)$ tape, provided $\lim_{n\to\infty} \frac{L_1(n)}{L(n)} = 0$. This completes the proof.

Corollary 6: For every tape constructable $L(n) \geq n$ there exists a recursive function $f: \Sigma^* \to \{0,1\}$ such that f is computable on L(n) tape and for every M_{i_0} computing f we can effectively find an equivalent algorithm $M_{\sigma(i_0)}$ such that

$$L_{i_0}(n) \leq L_{\sigma(i_0)}(n) \text{ for all } n$$

and for infinitely many w in Σ^*

$$L_{i_0}(|w|) \geq \frac{L(|w|)}{|M_{i_0}|} \text{ and } L_{\sigma(i_0)}(|w|) = 0.$$

Proof: Follows from the observation that A_L and $\bar{A}_L$ are L(n) tape recognizable and therefore we can choose f to be the characteristic function of A_L.

Next we show that many decidable mathematical theories must have trivial theorems requiring long proofs and that we can effectively find infinite set of such trivial theorems with long proofs.

Let T be a consistant, decidable mathematical theory, let L and F be tape constructable functions such that for all n $F(n+1) > F(n) \geq n$ and $L(n) \geq n$. Let there exist a Tm M which for each input (M_i, w) writes a formula $\gamma_{i,w}$ in T such that:

a) $\gamma_{i,w}$ is provable in T iff $(M_i,w) \in A_L$, that is iff the turing machine M_i does not accept input (M_i,w) on $L(|(M_i,w)|)/|M_i|$ tape.

b) $|\gamma_{i,w}| \leq F(|(M_i,w)|)$ and M use no more than $|\gamma_{i,w}|$ tape to write $\gamma_{i,w}$ for input (M_i,w).

Corollary 7: Any decision procedure for the theory T requires at least $L \circ F^{-1}(n)$ tape and for every Tm M_{i_0} which recognizes the theorems of T we can effectively find an infinite subset T' of theorems of T such that T' is recognizable on linear tape but M_{i_0} uses at least $c \cdot L \circ F^{-1}(n)$ tape on every element of T', for a constant $c > 0$.

Proof: Let M_{i_0} recognize the theorems of T. We now convert M_{i_0} to a recognizer of A_L, $M_{i_{00}}$. For input x $M_{i_{00}}$ checks if $x = (M_i, w)$; if not it is rejected. For input (M_i, w) the machine $M_{i_{00}}$ computes $\gamma_{i,w}$ and applies M_{i_0} to $\gamma_{i,w}$. Clearly, $M_{i_{00}}$ accepts A_L. Assume now that M_{i_0} operates on L_{i_0} tape such that

$$\lim_{n\to\infty} \frac{L_{i_0}(n)}{L \circ F^{-1}(n)} = 0.$$

Then $M_{i_{00}}$ would operate on $L_{i_0}[F(n)]$ tape; but then

$$\lim_{n\to\infty} \frac{L_{i_0}[F(n)]}{L \circ F^{-1} \circ F(n)} = \lim_{n\to\infty} \frac{L_{i_0}[F(n)]}{L(n)} = 0,$$

contradicting the fact that no such recognizer exists for A_L.

We now show how to obtain the set T' using $M_{i_{00}}$. We know that the regular set

$$R = \{(M_{i_{00}}, w) \mid w \in \Sigma^*\}$$

is a subset of A_L and that $M_{i_{00}}$ uses at least $\frac{L(n)}{|M_{i_{00}}|}$ tape on inputs from R. We let

$$T' = \{\gamma_{i_{00},w} \mid w \in \Sigma^*\}$$

clearly, T' is an infinite set of true theorems of T. To recognize the set T' we proceed as follows: for input γ (on track below γ) check for $(M_{i_{\circ\circ}},w)$, $|(M_{i_{\circ\circ}},w)|\leq|\gamma|$, whether $\gamma = \gamma_{i_{\circ\circ},w}$.

If no such $(M_{i_{\circ\circ}},w)$ is found reject the input, otherwise accept it. Clearly, T' is recognizable on linear tape, since $|\gamma_{i,w}|\geq|(M_i,w)|$, and the computation of $\gamma_{i,w}$ from (M_i,w) require no more than $|\gamma_{i,w}|$ tape.

On the other hand, since $M_{i_{\circ\circ}}$ uses $\dfrac{L(n)}{|M_{i_{\circ\circ}}|}$ tape on R, $M_{i_\circ}$ must use at least $\dfrac{L\circ F^{-1}(n)}{|M_{i_{\circ\circ}}|}$ tape, since $|\gamma_{i_{\circ\circ},w}|\leq F(|(M_{i_{\circ\circ}},w)|)$.

This completes the proof.

Recent work on the complexity of decision procedure for decidable mathematical theories has shown how statements about Turing machine computations can be efficiently encoded in formulas in these theories so that the statement about the Tm computation is true iff the formula is provable in the theory [3,6]. These results can be used not only to establish the complexity of the decision procedures, but also the existance of speed-up for most of these decision procedures [6].

For example, it has been shown [6] that the set

$\mathscr{R}$ = {R | R is a regular expression over 0, 1, (.), · +, ¬, *, denoting the empty set}

cannot be recognized on tape bounded by an elementary function. By the same techniques it follows that every Tm recognizing this set can be effectively speeded-up by any desired "stack of exponent on infinite subsets of R_0 of $\mathscr{R}$.

We conclude with a look at time complexity of decision procedures and speed-up of proofs for Pressburger Arithmetic [3].

Let $T_i(n)$ be the maximal number of steps taken by the Tm M_i on inputs of length n. We say that T(n) is <u>time-constructable</u> iff $T(n) = T_i(n)$ for some M_i.

<u>Corollary 8</u>: Let $T(n) \geq n^2$ be time constructable. Than there exist a set A_T,

$$A_T = \{(M_i, w) \mid M_i \text{ does not accept } (M_i, w) \text{ in time } \frac{T(n)}{|M_i| \log T(n)}$$

such that A_T is recognizable in time $T(n)$ and if $T(M_{i_0}) = A_T$ then the regular set

$$R = \{(M_{i_0}, w) \mid M_{i_0} \text{ does not accept } (M_{i_0}, w) \text{ in time } \frac{T(n)}{|M_{i_0}| \log T(n)}\}$$

is a subset A_T and M_{i_0} uses at least

$$\frac{T(n)}{|M_{i_0}| \log T(n)}$$

steps for every member of R.

Proof: Similar to the proof of Theorem 5. The only difference is in the factor $\log T(n)$ which appears because we have to use a less efficient simulation algorithm than in the tape complexity case [4].

We recall that the Pressburger Arithmetic (PA) is the first order theory of addition of natural numbers [3]. This theory is decidable and it "is one of the simplest, most basic, imaginable mathematical theories" [3]. Furthermore, it has been shown that any non-deterministic Tm recognizing the true theorems of PA must use at least time

$$T(n) = 2^{2^{cn}} \quad \text{for some } c > 0.$$

From this, one concludes that any "reasonable" proof procedure for PA will have theorems of length n whose shortest proof must be longer than $2^{2^{cn}}$ [3].

We add to this result the following observation about effective shortening of proofs.

Theorem 9: Let M_{i_0} be a Tm which recognizes the theorems of PA. Then we can effectively find an infinite set T of theorems of PA such that T is recognizable in polynomial time but M_{i_0} uses at least time

$$T(n) \geq 2^{2^{qn}}, \text{ for some } q > 0,$$

for every element of T.

Proof: By a close inspection fo the proof techniques used in [3] and using reasoning similar to that in the proof of Corollries 7 and 8, we can construct the set T' of sentences $\gamma_{i_0,w}$ in PA asserting that M_{i_0} does not accept the input (M_{i_0},w) in time

$$\frac{2^{2^{dn}}}{|M_{i_0}|\log 2^{2^{dn}}} = \frac{2^{2^{dn}}}{2^{dn}\,|M_{i_0}|}, \text{ for some } d > 0.$$

From this we then conclude that for some q > 0 and sufficiently large n

$$\frac{2^{2^{dn}}}{2^{dn}|M_{i_0}|} \geq 2^{2^{qn}}.$$

Thus, the recognition of the set T by M_i requires at least $2^{2^{qn}}$ steps for almost all elements of T'. By removing the short elements of T' we get the desired set T . On the other hand, it can be seen from [3] that the set T is recognizable in polynomial time. This completes the proof.

Thus again we see that for this very basic and simple theory we can effectively find, for every proof procedure, infinite sets of theorems which are recognizable as true in polynomial time, but for which the shortest proofs in the formalism are horrendously long. Again pointing out the limitations of formal methods and mechanical proof procedures, even for this very simple mathematical theory.

References

1. M. Blum, "A Machine-Independent Theory of the Complexity of Recursive Functions", JACM 14 (1967), 322-336.

2. M. Blum and I. Marques, "On Complexity Properties of Recursively Enumerable Sets", J. Symbolic Logic, 38 (1973) 579-593.

3. M.J. Fischer and M.O. Rabin, "Super-Exponential Complexity of Pressburger Arithmetic". in SIAM-AMS Proceedings Volume 7, 27-41. American Math. Soc. Providence, RI, 1974.

4. J. Hartmanis and J.E. Hopcroft, "An Overview of the Theory of Computational Complexity", JACM 18 (1971) 444-475.

5. H. Rogers, <u>The Theory of Recursive Functions and Effective Computability</u>, McGraw Hill, New York, 1967.

6. L.J. Stockmeyer, "The Complexity of Decision Problems in Automata Theory and Logic", Project MAC, Report MAC TR-133, MIT, Cambridge, Mass., July 1974.