1. Introduction

TURING MACHINES THAT TAKE ADVICE *

by Richard M. KARP [1]) and Richard J. LIPTON [2])

1. INTRODUCTION

Turing machines, random access machines and most of the other abstract computing devices studied in computational complexity theory represent *uniform algorithms,* which can receive arbitrarily long strings of symbols as input. The time and space needed by such devices to recognize a set $S \subseteq \{0, 1\}^*$ are examples of *uniform measures* of the complexity of S. In contrast, Boolean circuits, as well as certain types of decision trees and straight-line programs, compute functions with a finite domain. To study the complexity of recognizing the set $S \subseteq \{0, 1\}^*$ using such computational devices, we can view S as determining an infinite sequence of finite functions. For example, we can introduce, for each n, the Boolean function $S_n : \{0, 1\}^n \to \{0, 1\}$ defined as follows: $S_n(x_1, x_2, ..., x_n) = 1$ if and only if $x_1 x_2 ... x_n \in S$. If $L(S_n)$ denotes the minimum size of a Boolean circuit realizing S_n, then the growth rate of $L(S_n)$ as $n \to \infty$ is a measure of the nonuniform complexity of S.

Let us say that S has *small circuits* if $L(S_n)$ is bounded by a polynomial in n. It is well known that every set in P has small circuits [16]. Adleman [1] has recently proved the stronger result that every set accepted in polynomial time by a randomizing Turing machine has small circuits. Both these results are typical of the known relationships between uniform and nonuniform complexity bounds. They obtain a nonuniform upper bound as a consequence of a uniform upper bound.

The central theme here is an attempt to explore the converse direction. That is, we wish to understand when nonuniform upper bounds can be used to obtain uniform upper bounds. The immediate answer is "not

---

* This article has already been published in *Logic and Algorithmic,* an international Symposium in honour of Ernst Specker, Zürich, February 1980. Monographie de L'Enseignement Mathématique N° 30, Genève 1982.

This research was supported in part by NSF grant [1]) MCS77-09906 and [2]) MCS79-20409. An earlier version of this paper was presented at the Twelfth Annual ACM Symposium on Theory of Computing, 1980 [9].

always": clearly there are sets with small circuits that are not even recursive. The very trivial nature of such "counter-examples" suggests, however, that a more careful investigation may still yield insight. Indeed, as we will show, if one considers not arbitrary sets but rather "well behaved ones" it is possible to achieve our goal. For example, we will show that if SAT has small circuits, then the Meyer-Stockmeyer [19] hierarchy collapses.

Thus, here is an example of a nonuniform upper bound that has uniform consequences. The proof, of course, will depend on the fact that SAT is not a "pathological" set, but is rather well behaved.

Our results also serve to rule out some plausible speculations about the complexity of problems in *NP*. For example, one might imagine that $P \neq NP$, but SAT is tractable in the following sense: for every l there is a very short program that runs in time n^2 and correctly treats all instances of length l. Theorem 5.2 shows that, if "very short" means "of length c log 1", then this speculation is false.

Finally, we mention that the proof techniques presented here were put to use by S. Mahaney in his proof that $P \neq NP$ implies the nonexistence of sparse *NP*-complete problems [11], and by S. A. Cook in his proof that

$$P \subseteq HARDWARE\ (log\ n) \Rightarrow P \subseteq DSPACE\ (log\ n\ log\ log\ n)$$

[5].

2. Nonuniform Complexity Measures

In this section we will define our basic notion of nonuniform complexity and relate it to circuit complexity.

Let S be a subset of $\{0, 1\}^*$. Let $h: N \to \{0, 1\}^*$ where N is the set of natural numbers. Define $S:h = \{x \mid h(|x|) \cdot x \in S\}$. Next, let V be any collection of subsets of $\{0, 1\}^*$ and let F be any collection of functions from N to N. The key definition is

$$V/F = \{S:h \mid S \in V \text{ and } h \in F\}$$

Intuitively, V/F is the collection of subsets of $\{0, 1\}^*$ that can be accepted by V with an amount of advice bounded by F. The idea behind this definition is foreshadowed in papers by Pippenger [14] and Plaisted [15].

We are mainly interested in *poly*, the collection of all polynomially-bounded functions, and *log*, the collection of all functions that are $0\ (log\ n)$. Indeed, many of our results will concern the classes *P/poly* and *P/log*.