(12)

Priority Arguments in Complexity Theory

MAY 15 1972

by Nancy Lynch

Albert Meyer

Michael Fischer

Abstract (machine-independent) complexity theory is basically a theory which establishes limits on what may be said about the complexity of specific recursive functions. (For example, the speed-up theorem [1] shows that it is not always possible to discover close upper and lower bounds for the complexity of a function.) As such, its basic proof method is diagonalization.

Sacks, Spector, etc. [2] [3] have developed extensive diagonalization machinery for theorems about degrees of unsolvability. Thus far, we have not applied any except the simplest of the methods devised by the recursion theorists to complexity theory. In this paper, we present two results which arise very naturally out of complexity theory and whose proofs seem to require use of priority constructions.

The problems we deal with involve finding pairs of recursive sets which are complex, but for "different reasons" - that is, they don't "help" each other's computation.

To handle questions of this type, we use a generalization of Blum's axioms for complexity of partial recursive functions, to the case of relative algorithms (as represented by Turing machines with oracles [2]). Specifically, the axioms we use are:

(1) $(\forall i,x,A)\ \ \varphi_i^{(A)}(x)\downarrow \ \Leftrightarrow\ \Phi_i^{(A)}(x)\downarrow$,

(2) $(\exists\psi$, a relative algorithm$)(\forall i,x,y,A)$

Recursive Function Theory Newsletter, 2,
Univ. of Cal., Berkeley (July 1972), 7-10.

$$\psi^{(A)}(i,x,y) = \begin{cases} 1 & \text{if } \Phi_i^{(A)}(x) = y, \\ 0 & \text{otherwise.} \end{cases}$$

However, it is more intuitive to keep in mind the "space measure" (number of worktape squares) on oracle Turing machines when considering our results.

The first theorem is a subrecursive analog to the Friedberg-Muchnik theorem [2]. Instead of producing two sets which do not permit each other's computation, we produce two recursive sets which do not make each other's computation any easier. Another way of interpreting the result is to say it produces pairs of recursive sets which are complex, but for different reasons.

Notation: $\mathrm{Comp}^{(A)}B \le f$ a.e. means $(\exists i)[\varphi_i^{(A)} = C_B$ and $\Phi_i^{(A)} \le f$ a.e.]
$\mathrm{Comp}^{(A)}B > f$ a.e. (i.o.) means $(\forall i)[\varphi_i^{(A)} = C_B \Rightarrow \Phi_i^{(A)} > f$ a.e. (i.o.)]

We write Comp B in place of $\mathrm{Comp}^{(\emptyset)}B$.

Theorem 1: There exists h, a recursive function of two variables, with the following property:

For all sufficiently large running times t_B and t_C, there exist recursive sets B and C such that:

$$\mathrm{Comp}\ B \le h \circ t_B \text{ a.e.,}$$

$$\mathrm{Comp}\ C \le h \circ t_C \text{ a.e.,}$$

$$\mathrm{Comp}^{(C)}B > t_B \text{ a.e.,}$$

and

$$\mathrm{Comp}^{(B)}C > t_C \text{ a.e.}$$

The method of proof we use is to simultaneously construct the two sets B and C, using diagonalization and a finite-injury priority argument.

There is a small recursive bound on the number of injuries to any condition. We omit the proof in favor of a proof of theorem 2.

In theorem 1, both sets are constructed by diagonalization; to make the result more interesting, we would like to fix one of the sets arbitrarily. We may easily obtain the following:

<u>Proposition</u>: There exists h, a recursive function of two variables, with the following property:

For any recursive set A, and any recursive function t_A with the property that Comp A $> h \circ t_A$ i.o., there exist arbitrarily complex recursive sets B such that:

$$\mathrm{Comp}^{(B)} A > t_A \text{ i.o.}$$

The proof idea is partly due to Machtey, and is similar to the initial segment constructions in [2]; there is essentially no priority involved.

Our second theorem is similar to the proposition, but involves a stronger kind of lower bound on the complexity of A. Specifically,

<u>Theorem 2</u>: There exists h, a recursive function of two variables, with the following property:

For any recursive set A, and any total running time t_A, with $t_A \geq \lambda x[x]$, if Comp A $> h \circ t_A$ a.e., then there exist arbitrarily complex recursive sets B such that:

$$\mathrm{Comp}^{(B)} A > t_A \text{ a.e.}$$

The method of proof is a finite-injury priority argument with no apparent recursive bound on the number of injuries for each condition.

References

[1] Blum, Manuel. A Machine-Independent Theory of the Complexity of Recursive Functions, JACM, Vol. 14, No. 2, April 1967, pp. 322-336.

[2] Rogers, Hartley Jr. Theory of Recursive Functions and Effective Computability. McGraw-Hill, 1967.

[3] Sacks, Gerald E. Degrees of Unsolvability. Annals of Mathematical Studies, No. 55, 1963, Princeton, N. J.