# DENSE AND NON-DENSE FAMILIES OF COMPLEXITY CLASSES

Borodin, A.*, Constable, R.L., Hopcroft, J.E.

Cornell University
Ithaca, New York

* University of Toronto

## Abstract

Let $\Phi$ be any abstract measure of computational complexity, and let L denote the specific measure of memory resource (tape) on one tape Turing machines. Denote by $R^{\Phi}_{t(\ )}$ the class of all total functions whose $\Phi$-complexity is bounded by the function t( ) almost everywhere. Call such classes $\Phi$-complexity classes. We are interested in relationships among these classes, under proper set inclusion ($\subset$). In other words, we are interested in the partially ordered structure $\langle\Sigma^{\Phi}, \subseteq\rangle$ where $\Sigma^{\Phi} = \{R^{\Phi}_{t(\ )} \mid t(\ ) \text{ is recursive}\}$ is called the family of $\Phi$-complexity classes. Of special interest is the subfamily $\Omega^{\Phi} = \{R^{\Phi}_{\Phi_i(\ )} \mid \Phi_i(\ ) \text{ is total}\}$, called the family of exact $\Phi$-complexity classes.

We show that $\Sigma^L$ and $\Omega^L$ are dense under $\subset$ for sufficiently large bounds t( ), but $\Omega^L$ is not dense in $\Sigma^L$. We also construct measures $\Phi$ for which $\Sigma^{\Phi}$ and $\Omega^{\Phi}$ are non-dense, for which $\Sigma^{\Phi}$ is dense but $\Omega^{\Phi}$ is not, for which $\Omega^{\Phi}$ is dense but $\Sigma^{\Phi}$ is not and for which $\Omega^{\Phi}$ is dense in $\Sigma^{\Phi}$. Thus density is not a measure invariant property of $\Sigma^{\Phi}$ or $\Omega^{\Phi}$. These are the first examples of important structural properties of these families which are not measure invariant.

## I. Preliminaries

We assume at least cursory familiarity with the axiomatic approach to computational complexity theory as initiated in Blum [1] and developed recently in [2], [6], and [9]. To establish our notation, we list the following definitions.

Given an acceptable indexing $\{\phi_i(\ )\}$ of the partial recursive functions (of one argument), see Rogers [8], an abstract complexity measure over $\{\phi_i(\ )\}$ is a set $\{\Phi_i(\ )\}$ of partial recursive functions for which there exists a 0,1 valued recursive function M( ) satisfying

Axiom 1: $\phi_i(n)$ is defined iff $\Phi_i(n)$ is defined.

Axiom 2: $M(i,n,m) = 1$ iff $\Phi_i(n) = m$.

We say that $\Phi = \{\Phi_i(\ )\}$ satisfying the axioms is a complexity measure, and the individual $\Phi_i(\ )$ are called exact complexity functions. [The $\Phi_i(\ )$ have also been called "step-counting" functions or "run-time" functions or "difficulty" functions.]

Given a complexity measure $\Phi$, define a $\Phi$-complexity class $R^{\Phi}_{t(\ )} = \{\phi_i(\ ) \mid \phi_i(\ )$ total and $\Phi_i(n) \leq t(n)$ for almost all n (a.e.n.)$\}$. (We use lower case English letters, t,f,g,h, in denoting total as opposed to partial functions.) When $\Phi$ is clear from the context we use $R_{t(\ )}$.

Define $\Sigma^{\Phi} = \{R^{\Phi}_{t(\ )} \mid t(\ ) \text{ is recursive}\}$ and $\Omega^{\Phi} = \{R^{\Phi}_{\Phi_i(\ )} \mid \Phi_i \text{ total}\}$. Call $\Sigma^{\Phi}$ the family of (recursive) complexity classes and $\Omega^{\Phi}$ the family of exact complexity classes (or run-time classes). Our concern is with the families $\Sigma^{\Phi}$ and $\Omega^{\Phi}$ under the partial ordering of set inclusion, $\subseteq$. A set S partially ordered by $\leq$ is dense iff for all a,b in S, $a < b$ implies there is a c in S such that $a < c < b$.

We say that any family $C^{\Phi}$ of complexity classes is dense for sufficiently large t( ) iff there is a function a( ) such that the family $\{R^{\Phi}_{t(\ )} \mid t(\ ) > a(\ )$ & $t(\ )$ recursive$\} \cap C^{\Phi}$ is dense (under $\subset$).

## II. Density

Our Turing machine model is that used in Hartmanis & Stearns [3]. It uses separate input, output and work tapes and uses the standard finite control over a finite alphabet $\Sigma$. We consider the class of all such machines over all finite alphabets $\Sigma$. Let $\{M_i\}$ be an acceptable listing of such machines and $\phi_i(\ )$ the one argument function computed by $M_i$. Define the tape measure $L$ as follows: $L = \{L_i(\ )\}$ where

$$L_i(n) = \begin{cases} \text{the maximum number of tape squares used in the computation of } M_i \text{ on input } n \text{ (in binary)} & \text{if the computation halts} \\ \text{undefined} & \text{otherwise} \end{cases}$$

It is easy to see that $L$ is indeed a complexity measure.

Def. 2.1

$$f(\ ) <\cdot\ g(\ ) \text{ iff } f(x) \leq g(x) \text{ a.e. } x \text{ and } f(x) < g(x) \text{ i.o. (infinitely often)}$$

Theorem 2.1 (Borodin, Hopcroft)

The tape complexity functions are dense under $<\cdot$ .

Proof

Given machines $M_i$ and $M_j$ with tape complexities $L_i(\ ) <\cdot L_j(\ )$ we show how to construct a machine $M_k$ such that $L_k(\ )$ satisfies the theorem, i.e., $L_i(\ ) <\cdot L_k(\ ) <\cdot L_j(\ )$.

If $L_i(\ )$ were non-decreasing, then the proof would be trivial, namely to define $M_k$ on input $x$; run $\phi_i(y)$, mark the amount of tape used, then run $\phi_j(y)$ until it stops or attempts to exceed $L_i(y)$ tape. Record which was greater, i or j. Do this for all $y \leq x$ and determine for how many $y$, $L_i(y) < L_j(y)$. (It is essential to note that this can be done for each $y$ without exceeding $L_i(y)$ (the lower bound) amount of tape.) If the number of such $y$ is even, then stop $M_k$ having used $L_i(x)$ tape; if odd, then let $M_k$ run and use $L_j(x)$ tape. Thus infinitely often $L_k(\ )$ will be above $L_i(\ )$ and infinitely often $L_j(\ )$ will be above $L_k(\ )$. Thus $L_i(\ ) <\cdot L_k(\ ) <\cdot L_j(\ )$ .

However, $L_i(\ )$ need not be non-decreasing. But $L_i(\ ) <\cdot L_j(\ )$ does imply that there exists an infinite sequence $\{n_\ell\}$ such that $L_i(\ )$ is non-decreasing on $\{n_\ell\}$, i.e.,

$$L_i(n_\ell) \leq L(n_{\ell+1}) \text{ and}$$

$$L_i(n_\ell) < L_j(n_\ell) \quad \text{for all } \ell \ .$$

We construct a suitable $M_k$ by guaranteeing that we alternate between $L_i(\ )$ and $L_j(\ )$ on an infinite sequence as above. The amount of tape used by $M_k$ on input $x$ is determined as follows:

Lay off $L_i(x)$ tape (run $M_i$, put down markers to indicate boundaries of the tape used and then erase all other marks on the tape).

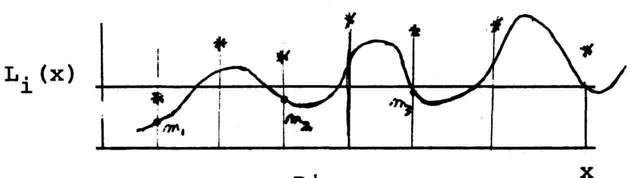
(1) If $L_j(x) \leq L_i(x)$ , then make $L_k(x) = L_i(x)$. (There are possibly finitely many $x$ such that $L_j(x) < L_i(x)$ .)

(2) If $L_i(x) < L_j(x)$ , then construct the unique finite sequence $\{m_1, m_2, m_3, \ldots m_r\}$ where

$$m_1 = \mu m \ \{L_i(m) < L_i(x) \ \& \ L_i(m) < L_j(m)\}$$

$$m_{p+1} = \mu m \ \{m > m_p \ \& \ L_i(m) \leq L_i(x) \ \& \ L_i(m) < L_j(m) \ \& \ L_i(m_p) \leq L_i(m)\}$$

$$m_r = x \ .$$




Diagram

* denotes $L_j(y)$ where $L_j(y) > L_i(y)$

If $r$ is odd, then set $L_k(x) = L_i(x)$.

If $r$ is even, then set $L_k(x) = L_j(x)$.

The $M_k$ and $L_k(\ )$ produced by the construction will satisfy the theorem since $L_k(\ )$ will alternate between $L_i(\ )$ and $L_j(\ )$ on the infinite sequence of points $\{m_1, m_2, \ldots \}$ .

Q.E.D.

Lemma 2.1

$$R^L_{L_i(\,)} - R^L_{t(\,)} \neq \phi \text{ iff}$$

$$\inf_{n\to\infty} \frac{t(n)}{L_i(n)} = 0 .$$

Proof

(1) "only if case" If $\inf_{n\to\infty} \frac{t(\,)}{L_i(n)} \neq 0$, then $\exists c$ such that $t(n) \geq c \cdot L_i(n)$ a.e.n. Since L has a linear global speed-up, $R_{L_i(\,)} \subseteq R_{t(\,)}$ .

(2) "if case" We need to "reprove" the well-known results of [4] since our tape measure has been defined as a function of the actual input rather than of input length. We leave this proof for the interested reader.

Q.E.D.

Theorem 2.2 (Borodin, Hopcroft)

$\Omega^L$ is dense for sufficiently large t( ), i.e., for all sufficiently large $L_i(\,)$ such that $R_{L_i(\,)} \subset R_{L_j(\,)}$, there is an $L_k(\,)$ such that $R_{L_i(\,)} \subset R_{L_k(\,)} \subset R_{L_j(\,)}$ .

Proof

In order to simplify the proof somewhat we will assume that $L_i(x) > 2^x$ for all x (a.e. x). We note, however, that it would be sufficient to require only $\lim\inf_{n\to\infty} L_i(n) \to \infty$.

Without loss of generality we assume that $L_i(x) < L_j(x)$ for all x . ($R_{L_i(\,)} \subset R_{L_j(\,)}$ implies $\exists c$ $L_i(x) \leq c \cdot L_j(x)$ for all x . Replace $L_j(\,)$ by $(c + 1) \cdot L_j(\,)$ .)

Moreover, by Lemma 1.1, $R_{L_i(\,)} \subset R_{L_j(\,)}$ implies $\inf_{n\to\infty} \frac{L_i(n)}{L_j(n)} = 0$.

(1) We will construct a machine $M_k$ so that $L_k(\,)$ oscillates between $L_i(\,)$ and $L_j(\,)$ and $L_k(\,)$ satisfies

$$\inf_{n\to\infty} \frac{L_i(n)}{L_k(n)} = 0 \quad \text{and}$$

$$\inf_{n\to\infty} \frac{L_k(n)}{L_j(n)} = 0 .$$

To this end, we construct $M_k$ so that $\forall c\ \exists x_{c_1}\ \exists x_{c_2}$ such that $c \cdot L_i(x_{c_1}) < L_j(x_{c_2})$ and $L_k(x_{c_1}) = L_i(x_{c_1})$, $c \cdot L_i(x_{c_2}) < L_j(x_{c_2})$ and $L_k(x_{c_2}) = L_j(x_{c_2})$.

(2) Since $L_i(x) \geq 2^x$ and $\inf_{n\to\infty} \frac{L_i(x)}{L_j(x)} = 0$ , there is at least one infinite sequence $S = \{m_1, m_2, m_3, \ldots\}$ such that

(a) for all $\ell$, $L_i(m_\ell) < L_i(m_{\ell+1})$

(b) for all c there is an $m \varepsilon S$ and $y \varepsilon S$ such that $c \cdot L_i(m) < L_j(m) < L_i(y)$ .

(3) We describe the operation of $M_k$ on input x. The definition of $L_k(x)$ requires the reconstruction of $L_k(\,)$ on a finite sequence $S_x = \{n_1, n_2, \ldots, n_r\}$ where

$$n_1 = \mu n[L_i(n) \leq L_i(x)]$$

$$n_{P+1} = \mu n > n_P[L_i(n) \leq L_i(x)]$$

and $n_r = x$ .

First $M_k$ lays off $L_i(x)$ tape. Using a multi-tract tape, $M_k$ finds out which points $z \leq y$ are not in $S_x$, and marks them on one track (put a - on the appropriate track in the cell corresponding to z). This merely requires trying to compute $\phi_i(0), \phi_i(1), \ldots, \phi_i(x)$ . Now for each $y \varepsilon S_x$, reconstruct the computation $M_k$ on y to determine the value of $L_k(y)$. Also attempt to satisfy 2(b) for the largest possible c .

The value of this constant is determined by knowing the history of $L_i(\,)$ and $L_k(\,)$ on $S_x$ , that is, by knowing the largest constant for which 2(b) has already been satisfied. Step (2) guarantees that for each c an appropriate

$x_{c_1}$ and $x_{c_2}$ will come up. Now use an alternating values scheme to determine $L_k(\ )$ just as done in Theorem 1.

During the computation, the work tape of $M_k$ could appear as follows:

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | $L_j$ work track |
| | | | | | | $L_i$ work track |
| - | i | - | ✓ i | j | - | $L_k$ history track |
| | | | | | | constant counter track |

$L_i(x)$ squares long

where

- i in square y of $L_k(\ )$ history implies $y \varepsilon S_x$ and $L_k(y) = L_i(y)$
- j in square y of $L_k(\ )$ history implies $y \varepsilon S_x$ and $L_k(y) = L_j(y)$
- \- in square y of $L_k(\ )$ history implies $y \not\varepsilon S_x$
- ✓ in square z of $L_k(\ )$ history while reconstructing $L_k(y)$ implies that $M_k$ will attempt to determine if $c \cdot L_i(z) < L_j(z)$ using $\leq L_i(y)$ tape.

(4) We give a brief informal description of $M_k$. Our proof depends on our ability to make $L_k(\ )$ oscillate on the sequence $S = \{m_1, m_2, \ldots\}$ so that $\inf_{n\to\infty} \frac{L_i(n)}{L_k(n)} = 0$ and $\inf_{n\to\infty} \frac{L_k(n)}{L_j(n)} = 0$ . Suppose that using our alternating value scheme we have set $L_k(m_p) = L_i(m_p)$ and the maximum constant is $c$ . The idea is to then define $L_k(m_{p+1}) = L_i(m_{p+1})$, $L_k(m_{p+2}) = L_i(m_{p+2})$ , ... until we can detech (at some $y = m_s$) that for some $z$ in S, $c \cdot L_i(z) < L_j(z)$ and $L_k(z)$ was set to $L_i(z)$.

We then begin to define $L_k(m_s) = L_j(m_s)$, $L_k(m_{s+1}) = L_j(m_{s+1})$, ... until we detect (at some later $y' = m_t$) that for some $z'$, $c \cdot L_i(z') < L_j(z')$ and $L_k(z')$ was set to $L_j(z')$ . At this point the value of the constant increases to $c+1$ .

The reader is asked to verify the following:

(A) For all $x$ the $L_k(\ )$ history track represents exactly the value of $L_k(y)$ for all $y \varepsilon S_x$.

(B) The computation of $M_k$ on input $x$ uses $L_i(x)$ tape up to the time when we decide whether $L_k(x) = L_i(x)$ or $L_k(x) = L_j(x)$.

(C) $L_k(\ )$ oscillates on at least one infinite sequence $S$ so that $\inf_S \frac{L_i(\ )}{L_k(\ )} = 0$ and $\inf_S \frac{L_k(\ )}{L_j(\ )} = 0$.

Moreover, for all $x$, $L_i(x) \leq L_k(x) \leq L_j(x)$ . Thus,

$$R^L_{L_i(\ )} \subset R^L_{L_k(\ )} \subset R^L_{L_j(\ )} \ .$$

Q.E.D.

<u>Theorem 2.3</u> (Borodin, Hopcroft)

$\Sigma^L$ is dense for sufficiently large $t(\ )$.

<u>Proof</u>

A somewhat more intricate version of the previous (run-time density) proof is needed. The difficulty is that we are no longer assuming that $t_1(\ )$ and $t_2(\ )$ are tape complexity functions, only that they are recursive functions. That is, we assume for sufficiently large $t_1(\ )$, $t_2(\ )$ that $R_{t_1(\ )} \subset R_{t_2(\ )}$ , and we seek a $t_3(\ )$ such that

$$* \ R_{t_1(\ )} \subset R_{t_3(\ )} \subset R_{t_2(\ )} \ .$$

(1) We first notice that we can replace $t_1(\ )$ by another recursive function $\underline{t}(\ )$ such that $R_{\underline{t}(\ )} = R_{t_1(\ )}$ and $\underline{t}(\ ) \leq t_2(\ )$ a.e. To see this, suppose $t_1(\ ) > t_2(\ )$ i.o. Then put $S = \{x | t_1(x) > t_2(x)\}$ . We claim that

$$\underline{t}(x) = \begin{cases} t_1(x) & \text{if } x \not\varepsilon S \\ t_2(x) & \text{if } x \varepsilon S \end{cases}$$ is the

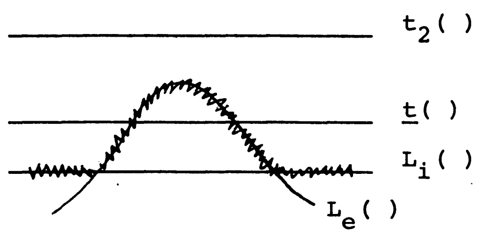
desired function. Clearly $R_{\underline{t}(\ )} \subseteq R_{t_1(\ )}$.

Suppose only $R_{t(\,)} \subset R_{t_1(\,)}$. Then there is an $L_i(\,)$ such that $\inf\limits_{S} \frac{t_2(\,)}{L_i(\,)} = 0$ by Lemma 2.1 and $L_i(\,) \le t_1(\,)$ a.e. Thus $R_{L_i(\,)} \subseteq R_{t_1(\,)}$ and $R_{L_i(\,)} \subset R_{t_2(\,)}$.

But this contradicts $R_{t_1(\,)} \subset R_{t_2(\,)}$.

(2) We now proceed to find a recursive $t_3(\,)$ satisfying * . Let $i$ be such that $L_{i(\,)} \le \underline{t}(\,)$ a.e. Since $R_{\underline{t}(\,)} \subset R_{t_2(\,)}$, there is an $e$ such that $L_e(\,) \le t_2(\,)$ a.e. and

$$\inf_{n\to\infty} \frac{\underline{t}(n)}{L_e(n)} = 0 .$$




Diagram

$L_j$ ww

Let $L_j(x) = \max\{L_i(x) + 1, L_e(x)\}$ . We shall attempt to construct $L_k(\,)$ such that

$$** \quad R_{t(\,)} \subset R_{\max\{\underline{t}(\,), L_k(\,)\}} \subset R_{\max\{\underline{t}(\,), L_j(\,)\}} \subset R_{t_2(\,)} .$$

We accomplish this by making $L_k(\,)$ oscillate between $L_i(\,)$ and $L_j(\,)$, but we must ensure that this oscillation takes place on an infinite set of points $S$ such that $\lim\limits_{S} \frac{\underline{t}(\,)}{L_j(\,)} = 0$ . The desired effect will be that $\inf\limits_{n\to\infty} \frac{\underline{t}(n)}{L_k(n)} = 0$ and $\inf\limits_{n\to\infty} \max\{\underline{t}(n), L_k(n)\}/L_j(n) = 0$ .

(3) We construct $L_k(\,)$ at a point $x$ by reconstructing (as before) as much of the past history of $L_k(\,)$ as we can in $L_i(x)$ tape. The only modification is that we do this reconstruction relative to $\underline{t}(\,)$ (more precisely, relative to a fixed algorithm $\phi_t$ for $t(\,)$ with tape complexity $L_t(\,)$ ).

That is, while computing thie history of $L_i(\,)$, $L_k(\,)$, and $L_j(\,)$ we also compute the history of $\underline{t}(\,)$ on those values for which the computation is possible within $L_i(x)$ tape. We then determine whether $L_k(\,)$ has been set equal to $L_j(\,)$ for an even or odd number of times on a set of points for which $L_j(y) > c\cdot\underline{t}(y)$ for larger and larger $c$ . The more difficult $\underline{t}(\,)$ is to compute, the more ancient is the history which determines how we set the value of $L_k(x)$ . By picking a sequence of points $P$ on which $L_i(\,)$ grows monotonically we can use the history to guarantee oscillation on a set of points $S$ for which $\lim\limits_{S} \frac{\underline{t}(\,)}{L_j(\,)} = 0$ .

We leave to the reader verification that a machine $M_k$ can be defined such that

(A) for all $x$ $L_k(x) = L_i(x)$ or $L_k(x) = L_j(x)$ .

(B) for all $c$ there exists $x_{c_1}$, $x_{c_2}$ such that $c\cdot\underline{t}(x_{c_1}) < L_j(x_{c_1})$ and $L_k(x_{c_1}) = L_i(x_{c_1})$, $c\cdot\underline{t}(x_{c_2}) < L_j(x_{c_2})$ and $L_k(x_{c_2}) = L_j(x_{c_2})$ .

By making some assumptions on the growth of $L_i$ we can guarantee that for all $z$ there is an $x$ such that $L_i(x) > \max\{L_{\underline{t}}(z), c\cdot\underline{t}(z)\}$ where $\phi_{\underline{t}} \simeq \underline{t}$. Thus the construction of $M_k$ is very similar to the construction for run-time density. The reader can easily verify that condition ** of (2) is satisfied.

Q.E.D.

<u>Corollary 2.1</u> (Borodin)

Between any two sufficiently large complexity classes there are an infinite number of incomparable complexity classes.

The proof requires a more elaborate version of the construction of $M_k$, but the general ideas are the same.

## III. Measures for which $\Sigma^{\Phi}$ and $\Omega^{\Phi}$ are non-dense.

In order to construct our example of a measure $\Phi$ for which $\Sigma^{\Phi}$ and $\Omega^{\Phi}$ are non-dense we need three basic facts. These are given below.

(Compression)
Theorem 3.1 (Blum)

For all $\Phi$ there exists an $h(\ )$, called a jump function for $\Phi$ (sometimes denoted $h_{\Phi}(\ )$ ) such that for all sufficiently large $\Phi_i(\ )$

$$R_{\Phi_i(\ )} \subset R_{h(\Phi_i(\ ))} .$$

This is proved as Theorem 8 of Blum [1] .

(Gap)
Theorem 3.2 (Borodin)

For all $\Phi$ there is a partial recursive $\rho(\ )$ such that for all recursive $\phi_e(\ )$ and for all $\phi_i(\ )$

(a) $\phi_{\rho(i,e)}(n)$ is defined iff $\phi_i(n)$ is defined

(b) if $\phi_i(n)$ is defined, then $\phi_i(n) < \phi_{\rho(i,e)}(n)$

(c) for all $i$ there is no $\Phi_j(\ )$ such that for infinitely many $n$ $\phi_{\rho(i,e)}(n)$ is defined and $\phi_{\rho(i,e)}(n) \leq \Phi_j(n) \leq \phi_e(\phi_{\rho(i,e)}(n))$.

That is, $\phi_{\rho(i,e)}(\ )$ is a "$\phi_e(\ )$ gap above $\phi_i(\ )$."

Proof

(1) Let $P(y,n,e)$ be the predicate "for all $j \leq n$ either $\Phi_j(n) < y$ or $\phi_e(y) < \Phi_j(n)$". This predicate is recursive for recursive $\phi_e(\ )$ because the predicate $M(i,n,m)$ iff "$\Phi_i(n) = m$" is recursive, and consequently so are the predicates "$\Phi_i(n) < m$", "$\Phi_i(n) > m$" .

(2) Define $\phi_{\rho(i,e)}(n) = \mu y[y>\phi_i(n)$ and $P(y,n,e)]$. There are arbitrarily large $y$ such that $P(y,n,e)$. Thus from Kleene's $\mu$-recursion formalism [5] it follows that $\phi_{\rho(i,e)}(\ )$ is partial recursive. Moreover, $\phi_{\rho(i,e)}(n)$ is defined iff $\phi_i(n)$ is defined.

Q.E.D.

We need one more basic fact before constructing our non-dense measure. The fact is simply that for every measure there is a uniform bound on the value of the function given its complexity. For example, if $L_i(x) = y$, then for a binary machine $\phi_i(\ ) \leq 2^y$ .

(Bounding)
Lemma 3.1 (McCreight & Meyer)

For all $\Phi$ there is an increasing $g(\ )$ such that for all $i$ and for all sufficiently large $\Phi_i(\ )$, $\phi_i(n) < g(\Phi_i(n))$ a.e. n.

Proof

The proof is elementary. Blum [1] has proved that any two measures $\Phi$ and $\hat{\Phi}$ are related by an increasing $d(\ )$ such that $\Phi_i(n) < d(\hat{\Phi}_i(n),n)$ and $\hat{\Phi}_i(n) < d(\Phi_i(n),n)$ a.e.n.

To use this fact here we observe that for the tape measure $L = \{L_i(\ )\}$ the lemma surely holds, say for $g_L(\ )$ . Thus picking a $d(\ )$ relating $\Phi$ and $L^m$ , it follows that $\phi_i(n) < g_L(L_i^m(n)\ ) <$ $g_L(d(\Phi_i(n),n)$ a.e.n. Now take $g(n) =$ $g_L(d(n),n)$ , and "sufficiently large" means for all $\Phi_i(\ )$ such that $\Phi_i(n) \geq n$ a.e.n.

Q.E.D.

Theorem 3.3 (Constable)

There exists a measure $\hat{\Phi}$ such that there are arbitrarily large $\hat{\Phi}_{i_1}(\ )$, $\hat{\Phi}_{i_2}(\ )$ satisfying

(i) $R_{\hat{\Phi}_{i_1}(\ )} \subset R_{\hat{\Phi}_{i_2}(\ )}$ and

(ii) there is no recursive $t(\ )$ such that $R_{\hat{\Phi}_{i_1}(\ )} \subset R_{t(\ )} \subset R_{\hat{\Phi}_{i_2}(\ )}$ .

That is, neither $\Sigma^{\hat{\Phi}}$ nor $\Omega^{\hat{\Phi}}$ is dense.

Proof

Our method of proof is quite transparent. We select an appropriate gap size $h(\ ) = \phi_e(\ )$ and use the technique of Borodin's Gap Theorem to find for each $\Phi_i(\ )$ a gap function $\phi_{\rho(i,e)}(\ )$ above

$\phi_i(\ )$. (For reasons which will be clear later, we modify $\rho$ to produce the function above $\Phi_i(\ )$ instead of above $\phi_i(\ )$.) This part of the construction guarantees arbitrarily large functions of the type needed. For brevity we denote $\phi_{\rho(i,e)}(\ )$ for fixed $\phi_e(\ )$ by $\tau_i(\ )$ and when $\tau_i(\ )$ is total we use $t_i(\ )$, and when $i$ is unimportant we use only $t(\ )$. Thus $R^{\Phi}_{t(\ )} = R^{\Phi}_{h(t(\ ))}$. Now define a new measure $\hat{\Phi}$ which puts essentially only one function into the gap, i.e., we "plant a function in the gap". The function $h(t(\ ))$ itself goes into the gap so that $h(t(\ )) \in R^{\hat{\Phi}}_{h(t(\ ))} - R^{\hat{\Phi}}_{t(\ )}$.

For technical reasons (in step 6) we actually define a "double gap", that is a $t(\ )$ such that $R_{t(\ )} = R_{h(t(\ ))} = R_{h(h(t(\ )))}$. For brevity we denote $h(h(t(\ )))$ by $h^{(2)}(t(\ ))$. We now actually plant $h(t(\ ))$ in the gap interval $[t(\ ), h^{(2)}(t(\ ))]$.

The first step in our proof is to select $h(\ )$, then we construct our new measure $\hat{\Phi}$, and finally we prove that it possesses the desired properties.

(1) Given any measure $\Phi = \{\Phi_i(\ )\}$, select an increasing $h(\ )$ so that $h(n) > g(n)$ for all n, for the $g(\ )$ of the Bounding Lemma. Then if $\phi_j(\ ) = h(t_i(\ ))$ it follows that $\Phi_j(n) > t_i(n)$ a.e.n., and hence because of the gap $\Phi_j(n) > h^{(2)}(t_i(n))$ a.e.n. $\Phi_j(n) > t_i(n)$ follows because if $\Phi_j(n) < t_i(n)$ i.o., then $h(\Phi_j(n)) < h(t_i(n)) = \phi_j(n)$ i.o. which contradicts the relation $\phi_j(n) < g(\Phi_j(n)) < h(\Phi_j(n))$ a.e.n. from the Bounding Lemma.

(2) From $\Phi = \{\Phi_i(\ )\}$ define a new measure as follows. Going through $\{\Phi_i(\ )\}$ in order, construct an $h^{(2)}(\ ) = \phi_e(\ )$ gap function $\tau_i(\ )$ above each $\Phi_i(\ )$. Make $h(\tau_i(\ ))$ the complexity measure of itself, thereby forming a new measure, say $\hat{\Phi}^i$. Construct successive gap functions so that they are gaps in the cumulative measure, i.e., for $\Phi_{i+1}$ construct the gap $\tau_i(\ )$ over the measure $\hat{\Phi}^i$. Then form the new measure $\hat{\Phi}^{i+1}$. The gap construction thus depends on the index k of the measure over which the gap is constructed as well as on the function $\Phi_i(\ )$ of the original measure $\Phi$. The final measure $\hat{\Phi}$ is the limit of this process. We make this process precise in the next step.

(3) To make the technicalities run smoother it is best to have the original measure $\Phi$ embedded in each stage, $\hat{\Phi}^i$, of constructing $\hat{\Phi}$. To accomplish this, take a new acceptable indexing of the partial recursive functions, say $\{\overline{\phi}_i(\ )\}$ where

$$\overline{\phi}_i(n) = \begin{cases} \phi_{i/2}(n) & i \text{ even} \\ \phi_{(i-1)/2}(n) & i \text{ odd} \end{cases}.$$

Then $\overline{\Phi} = \{\overline{\Phi}_i(\ )\}$ is the measure corresponding to $\Phi$. All we have done is keep a double list. The even list will stay fixed and is actually our given $\Phi$. On the odd list we alter the run-time functions.

At each stage in constructing $\hat{\Phi}$ a function $\hat{M}_k(\ )$ is produced satisfying $\hat{M}_k(i,n,m) = 1$ iff $\hat{\Phi}^k_i(n) = m$. From $\hat{M}_k(\ )$ the gap function of gap size $h^{(2)}(\ )$ can be computed. Denote it by $\tau_{k,i}(\ )$. Let $\alpha(k,i)$ be the index of $h(\tau_{k,i}(\ ))$; $\alpha$ can be made increasing and always odd.

(4) Define $\hat{\Phi}^k$ as follows. $\hat{\Phi}^o = \overline{\Phi}$, $\hat{\Phi}^{k+1}$ is determined from $\hat{\Phi}^k$ by

$$\hat{\Phi}^{k+1}(n) = \begin{cases} \overline{\phi}_i(n) & i \text{ odd and } \alpha(k,2k)=i \\ \hat{\Phi}^k_i(n) & \text{otherwise} \end{cases}.$$

Define $\hat{\Phi} = \{\hat{\Phi}^i_i(\ )\}$, that is $\hat{\Phi}$ is the limit of the $\hat{\Phi}^k$ as $k \to \infty$. The complexity functions of $\hat{\Phi}$ satisfy

$$\hat{\Phi}_i(n) = \begin{cases} \phi_i(n) & i \text{ odd } \& \ \exists k \le i, \ \alpha(k,2k)=i \\ \overline{\Phi}_i(n) & \text{otherwise} \end{cases}.$$

We prove that $\hat{\Phi}$ is indeed a measure by indicating how to decide $\hat{\Phi}_i(n) = m$. If $i$ is even, then just use $\overline{M}(i,n,m)$ to decide. If $i$ is odd, check all $k \leq i$ and ask $\alpha(k,2k)=i$. If not, then again use $\overline{M}(i,n,m)$. But if $\alpha(k,2k)=i$, then first use $\hat{M}_k(2k,n,x)$ for all $x \leq m$ to decide whether $\hat{\Phi}^k_{2k}(n) \leq m$. If the inequality fails, then clearly $\hat{M}(i,n,m)$ is false because $h(\tau_{k,2k}(n)) > \hat{\Phi}^k_{2k}(n)$. If $\hat{\Phi}^k_{2k}(n) < m$, then start the gap defining procedure and see if it can be completed with $h(\tau_{k,2k}(n)) = m$. This can be decided because the possible values for $\tau_{k,2k}(n)$ keep increasing, i.e., they can be only $\hat{\Phi}^k_{2k}(n)+1$, $\hat{\Phi}^k_{2k}(n)+2$, ... and $h(\ )$ is increasing.

The above procedure shows how to decide $\hat{M}(i,n,m)$ and proves that $\hat{\Phi}$ is a measure. Moreover, $\hat{\Phi}$ contains all the "gap functions", $\tau_i(\ )$, of any measure $\hat{\Phi}^k$ because $\alpha(\ )$ is increasing.

(5) It is now easy to see that $\Sigma^{\hat{\Phi}}$ is non-dense. In this step we show that $R^{\hat{\Phi}}_{t(\ )} \subset R^{\hat{\Phi}}_{h(t(\ ))}$ for arbitrarily large $t(\ )$. Let an arbitrarily large $a(\ )$ be given, and let $\Phi_a(\ ) > a(n)$ a.e.n. for $\Phi_a(\ )$ recursive. Then put $t(\ ) = \tau_a(\ )$. Clearly $R_{t(\ )} \subseteq R_{h(t(\ ))}$. But also $R^{\hat{\Phi}}_{t(\ )} \subset R^{\hat{\Phi}}_{h(t(\ ))}$ because $h(t(\ )) \in R^{\hat{\Phi}}_{h(t(\ ))} - R^{\hat{\Phi}}_{t(\ )}$. This holds because there is a $\phi_j(\ ) = h(t(\ ))$ and $\hat{\Phi}_j(\ ) = h(t(\ ))$ by definition of $\tau_a(\ )$, and for any $i \neq \alpha(k,2k)$, $\phi_i(\ ) = h(t(\ ))$ implies $\Phi_i(\ ) > h(t(\ ))$ by step (2). So every index for $h(t(\ ))$ either produces a run-time equal to $h(t(\ ))$ or greater than $h(t(\ ))$.

(6) Finally, we show that there is no $t^1(\ )$ such that $R^{\hat{\Phi}}_{t(\ )} \subset R^{\hat{\Phi}}_{t^1(\ )} \subset R^{\hat{\Phi}}_{h(t(\ ))}$. Suppose such a $t^1(\ )$ existed, then there would be a $\hat{\Phi}_i(\ )$ satisfying

$$*** \quad t(\ ) \underset{\text{i.o.}}{<} \hat{\Phi}_i(\ ) \underset{\text{a.e.}}{\leq} h(t(\ )) .$$

Clearly $i \neq \alpha(k,2k)$ for some $k$ can not occur because $t(\ )$, $h(t(\ ))$ is a gap for the measure $\hat{\Phi}$.

Suppose then that $i = \alpha(k,2k)$. If $k < a$ (recall $t(\ ) = \tau_a(\ )$), then at stage $k$, $\tau_a(\ )$ was constructed to be a gap over $\hat{\Phi}^{k^a}$ so *** is impossible.

Suppose $k > a$, then for sufficiently large $n$ if $\tau_a(\ ) < \hat{\Phi}_i(n)$ holds, that is, $\tau_a(n) < h(\tau_k(n))$,

either $\tau_a(n) < \hat{\Phi}_k(n)$, in which case $h(\tau_a(n)) < h(\tau_h(n)) = \hat{\Phi}_i(n)$ which contradicts ***;

or else $\tau_k(n) \leq \tau_a(n)$, in which case the fact that $\tau_k(n)$ is a two-sided gap function over $\hat{\Phi}^a$ is contradicted, i.e., $h(\tau_k(n)) \leq h(\tau_a(n)) = \Phi^a_a(\ ) \leq h^2(\tau_k(n))$.

(7) We now have a measure which is not dense. We can improve it to be a measure which is not run-time dense by making $\tau_k(\ )$ a run-time for $\Phi_k(\ )$ in the same manner that $h(\tau_k(\ ))$ was made a run-time for itself. If this were done from the very beginning we could then use a "single gap", $h(\ )$ instead of $h^{(2)}(\ )$ because in step (6) $\tau_k(\ )$ would itself be a run-time. However, for expository purposes it seemed best to keep the definition of $\hat{\Phi}$ as simple as possible.

Q.E.D.

The construction of a non-dense family, $\Sigma^{\hat{\Phi}}$, raises the question whether $\Sigma^L$ is dense <u>because</u> $\Omega^L$ is dense. In the next section <u>we will</u> show that this can not be the case in general because there is a measure $\Phi$ for which $\Omega^\Phi$ is dense but $\Sigma^\Phi$ is not. We then also show that in general the density of $\Sigma^\Phi$ need not transfer to $\Omega^\Phi$ because there is a measure $\Phi$ for which $\Sigma^\Phi$ is dense but $\Omega^\Phi$ is not.

## <u>IV. Relative Density</u>

<u>Def. 4.1</u>

The run-time classes, $R^\Phi_{\Phi_i(\ )}$, <u>are dense in</u> $\Sigma^\Phi$ for sufficiently large $t(\ )$ iff for all sufficiently large $\overline{t}(\ )$, $\underline{t}(\ )$ such that $R_{\underline{t}(\ )} \subset R_{\overline{t}(\ )}$, there is a $\Phi_i$ such that $R_{\underline{t}(\ )} \subset R_{\Phi_i(\ )} \subset R_{\overline{t}(\ )}$.

The notion of denseness in $\Sigma^{\Phi}$ is a notion of relative density. It is analogous to the notion that the rationals are dense in the reals. Such concepts lend further insight into the structure of $\Sigma^{\Phi}$. We shall show that this relative density notion is not measure invariant.

As a prelude to the first theorem we recall the diagonalization procedure of Blum's Compression Theorem, Theorem 8 of [1]. For any measure $\Phi$ a function $h(\ )$, a jump function for $\Phi$, is determined such that a diagonalization construction places a function $\phi(\ )$ in $R_{h(\Phi_j(\ ))} - R_{\Phi_j(\ )}$ for all $i$.
The construction is roughly this (We do not require $\phi(\ )$ to be 0,1 valued as Blum does, and thus the procedure is simpler.): at input $n$,

(1) Compute $\Phi_j(n)$.

(2) Form the set $L_n$ of those indices $k$ such that $k \leq n$ & $\Phi_k(n) \leq \Phi_j(n)$.

(3) Let $\phi(n) = \sum_{i \in L_n} \phi_i(n)$.

Clearly $\phi(\ ) \notin R_{\Phi_j(\ )}$. Also since the complexity of $\Phi_j(\ )$ can be bounded uniformly by a function of its value, we can produce an $h(\ )$ such that the $\Phi$-complexity of steps (1)-(3) is bounded by $h(\Phi_j(\ ))$. Thus $\phi(\ )$ is a.e. more difficult than $\Phi_j(\ )$ but can be done in $h(\Phi_j(\ ))$.

The above procedure can be modified so that it takes place only on a set S. If membership in S is easy to decide, then $\phi(\ )$ is made more difficult than $\Phi_j(\ )$ only on S. To modify the above procedure, add the instruction

(0) Determine whether $n \in S$,
if not, go to (1)
if yes, then put $\phi(\ ) = 0$.

For this procedure there is a function $h(n,m)$ increasing in both variables such that $\phi(\ ) \in R_{h(\Phi_j(\ ),\Phi_s(\ ))}$ where $\Phi_s(\ )$ is the complexity of deciding $n \in S$ by a procedure with index $s$.

We use these facts below, first in the simple but useful observation

Lemma 4.1 (Constable)

For all $\Phi$ there are arbitrarily large $\underline{t}(\ )$, $\overline{t}(\ )$ such that

(i) $R_{\underline{t}(\ )} \overset{\cdot}{\subset} R_{\overline{t}(\ )}$ and

(ii) there is no $\Phi_j(\ )$ such that $\underline{t}(\ ) \underset{\text{i.o.}}{<} \Phi_j(\ ) \underset{\text{a.e.}}{\leq} \overline{t}(\ )$ and $R_{\underline{t}(\ )} \overset{\cdot}{\subset} R_{\Phi_j(\ )} \overset{\cdot}{\subset} R_{\overline{t}(\ )}$.

Proof

(1) Let an arbitrarily large $a(\ )$ be given. Choose an infinite set S whose complement is infinite and for which there is a characteristic function $\phi_s(\ )$ such that $\Phi_s(\ ) < a(\ )$ a.e..

(2) Determine the function $d(\ )$ such that $g(n) = h(n,n)$ for the $h(\ )$ of the remark preceding the theorem. Let $\phi_{i_o}(\ )$ be a function which is a.e. more complex than $a(\ )$ and pick a gap function $t(\ )$ of gap size $d^{(2)}(\ )$ above $\Phi_{i_o}(\ )$.

(3) Define
$$\overline{t}(n) = d^{(2)}(t(n))$$
$$\underline{t}(n) = \begin{cases} a(n) & \text{if } n \in S \\ d(t(n)) & \text{otherwise.} \end{cases}$$

Notice that $R_{\underline{t}(\ )} \overset{\cdot}{\subseteq} R_{\overline{t}(\ )}$ since $t(\ ) \leq \overline{t}(\ )$ a.e. But also $R_{\underline{t}(\ )} \overset{\cdot}{\subset} R_{\overline{t}(\ )}$ since $\phi_{i_o}(\ ) \in R_{\overline{t}(\ )} - R_{\underline{t}(\ )}$.

(4) Suppose $\Phi_j(\ )$ satisfies $\underline{t}(\ ) \underset{\text{i.o.}}{<} \Phi_j(\ ) \underset{\text{a.e.}}{\leq} \overline{t}(\ )$. Then notice that actually $\Phi_j(\ ) \leq t(\ )$ a.e. since the interval $[t(\ ), d^{(2)}(t(\ ))]$ is a complexity gap. Thus it is not possible that $R_{\underline{t}(\ )} \overset{\cdot}{\subset} R_{\Phi_j(\ )}$ because on $\overline{S}$, the complement of S, it is possible to define a function $\phi(\ )$ which is more complex than $\Phi_j(\ )$ but less complex than $\underline{t}(\ )$. This follows since on $\overline{S}$, $\Phi_j(\ ) \ll \underline{t}(\ )$.

Q.E.D.

We now conclude a more interesting fact. First a definition (due to McCreight and Meyer [6]).

Def. 4.2

A measure $\Phi$ is proper iff for all i,

$$\Phi_i(\ ) \in R^{\Phi}_{\Phi_i(\ )} .$$

Theorem 4.1 (Constable)

If $\Phi$ is proper, then the run-time classes are not dense in $\Sigma^{\Phi}$ .

Proof

Let the $d(\ )$ of Lemma 4.1 be chosen larger than $\max\{h(n,n),g(n)\}$ for the $g(\ )$ of the Bounding Lemma. Construct $t$, $\underline{t}(\ )$ and $\overline{t}(\ )$ as in the preceding lemma. Now suppose

$$R^{\Phi}_{\underline{t}(\ )} \subset R^{\Phi}_{\Phi_m(\ )} \subset R^{\Phi}_{\overline{t}(\ )}$$

and $\Phi$ is proper. Then $\Phi_m(\ ) \in R_{\Phi_m(\ )}$ so there is a $\phi_i(\ ) = \Phi_m(\ )$ with $\Phi_i(\ ) \leq t(\ )$. Then by the definition of $d(\ )$ and $t(\ )$,

$\Phi_m(\ ) = \phi_i(\ ) < g(\Phi_i(\ )) \leq g(t()) < \overline{t}(\ )$

a.e. $R^{\Phi}_{\underline{t}(\ )} \subset R^{\Phi}_{\Phi_n(\ )}$ implies $\Phi_n > \underline{t}$ ,

i.o. Hence $\Phi_m(\ )$ contradicts Lemma 4.1 .

Q.E.D.

The tape measure L is proper, thus the run-time classes are not dense in L although L is both dense and run-time dense. Is there then a measure $\Phi$ in which the run-time classes are dense in $\Sigma^{\Phi}$? To answer this question, we refer to the concept of similar measures introduced by McCreight & Meyer in [6].

Def. 4.3

Two measures $\Phi$ and $\hat{\Phi}$ are similar iff $\Phi_i(\ ) = \hat{\Phi}_i(\ )$ for all $i$ such that range $\phi_i(\ ) \neq \{0\}$ .

We also need

Def. 4.4

A set S of functions is class determining (C.D.) for $\Phi$ iff for all sufficiently large $t(\ )$ there is an $s(\ ) \in S$ such that $R_{t(\ )} = R_{s(\ )}$ .

A measure $\Phi$ is class determing iff $S = \{\Phi_i(\ )\}$ is c.d. Remark: McCreight & Meyer observe that no proper measure can be c.d. But they prove the deep result that there is a c.d. measure similar to every measure.

We now conclude

Theorem 4.2 (Borodin)

There is a measure $\Phi$ for which the run-time classes are dense in $\Phi$ .

Proof

Take $\hat{L}$ to be a c.d. measure similar to L . Then $\hat{L}$ is both dense and run-time dense; and since $\hat{L}$ is c.d., the run-time classes are dense in $\hat{L}$ .

Q.E.D.

Finally, we answer the questions left open from section III.

Theorem 4.3 (Constable)

There is a measure $\Phi$ for which $\Sigma^{\Phi}$ is dense but $\Omega^{\Phi}$ is not.

Proof

Start with the dense measure L . Since the run-time classes are not dense in L , there are arbitrarily large $\underline{t}(\ )$ $\overline{t}(\ )$ such that $R^{L}_{\underline{t}(\ )} \subset R^{L}_{\overline{t}(\ )}$ but no $L_i(\ )$ satisfies

$$R^{L}_{\underline{t}(\ )} \subset R^{L}_{L_i(\ )} \subset R^{L}_{\overline{t}(\ )} .$$

Now we form, by the methods of Theorem 3.3 , a measure $\hat{\Phi}$ which makes $\underline{t}(\ )$ and $\overline{t}(\ )$ into run-times of some $\overline{L}_i(\ )$ below them. For instance, make the $\tau_i(\ )$ of Theorem 3.3 a run-time of $L_i(\ )$ rather than of itself.

We leave the details of this construction to the reader.

Q.E.D.

Theorem 4.4 (Constable)

There is a measure $\Phi$ for which $\Omega^{\Phi}$ is dense but $\Sigma^{\Phi}$ is not.

Proof

The plan of the proof is to proceed essentially as in Theorem 3.3 to construct a measure $\Phi$ for which $\Sigma^{\Phi}$ is non-dense. By taking as the base measure for this construction (the $\Phi$ of Theorem 3.3) the measure L and by introducing several special conditions on the construction of the new complexity functions (gap functions in L), it is possible to keep $\Omega^\Phi$ dense.

Intuitively $\Omega^\Phi$ remains dense because we scatter the new complexity functions into L sparsely enough that there are tape complexity functions of difficult functions between them. The special conditions mentioned above are designed to insure the sparseness.

To simplify presentation of this proof we use the following abbreviations:

A: $R^\Phi_{\Phi_i(\,)} \subset R^\Phi_{\Phi_j(\,)}$

B. $R^L_{\Phi_i(\,)} \subset R^L_{\Phi_j(\,)}$

C. $\exists k$ such that $R^\Phi_{\Phi_i(\,)} \subset R^\Phi_{\Phi_k(\,)} \subset R^\Phi_{\Phi_j(\,)}$ .

The difficult part of the proof is demonstrating that $\Omega^\Phi$ is dense. That $\Sigma^\Phi$ is non-dense is proved exactly as in Theorem 3.3 . We must show that A implies C . We distinguish four possible ways in which A can hold.

(1) $\Phi_i(\,)$ is a tape complexity function ($L_i(\,)$) and $\Phi_j(\,)$ is a tape complexity function ($L_j(\,)$).

(2) $\Phi_i(\,)$ is a newly defined complexity function in $\Phi$ (gap function in L), but $\Phi_j(\,)$ is a tape complexity function ($L_j(\,)$).

(3) $\Phi_i(\,)$ is a tape complexity function ($L_i(\,)$) and $\Phi_j(\,)$ is a new complexity function.

(4) Both $\Phi_i(\,)$ and $\Phi_j(\,)$ are new complexity functions.

We will abbreviate these cases by using the terms "new" and "tape" and by using $L_i(\,)$ for $\Phi_i(\,)$ when $\Phi_i(\,)$ is a tape function, likewise for $\Phi_j(\,)$. Thus in summary form:

(1) $R^\Phi_{L_i(\,)} \subset R^\Phi_{L_j(\,)}$ , "tape in tape" case.

(2) $R^\Phi_{\Phi_i(\,)} \subset R^\Phi_{L_j(\,)}$ , "new in tape" case.

(3) $R^\Phi_{L_i(\,)} \subset R^\Phi_{\Phi_j(\,)}$ , "tape in new" case

(4) $R^\Phi_{\Phi_i(\,)} \subset R^\Phi_{\Phi_j(\,)}$ , "new in new" case .

The strategy of the proof is to show that (I) A implies B, and (II) (A implies B) implies (A implies C).

We show each of these parts by examining the four possible cases. Further special notation is used in the proofs, specifically:

let $L_{ii}(\,)$ denote the tape complexity function used as the base for constructing $\Phi_i(\,)$, likewise for $L_{jj}(\,)$ and $\Phi_j(\,)$;

let $L_{\Phi_i(\,)}$ and $L_{\Phi_j(\,)}$ denote respectively the tape complexities of the gap functions $\Phi_i(\,)$ and $\Phi_j(\,)$ as defined by the uniform procedure of Theorem 3.3 (as modified slightly below) .

Similar conventions will apply to other $\Phi$-complexity functions which arise, i.e., if $\Phi_p(\,)$ is introduced, then $L_{pp}(\,)$ and $L_{\Phi_p(\,)}$ are understood to be defined.

We shall consider in detail only case (2) which is the difficult case. The general principles will be clear from a careful examination of this case.

We show that A implies B , i.e.,

$R^\Phi_{\Phi_i(\,)} \subset R^\Phi_{L_j(\,)}$ implies $R^L_{\Phi_i(\,)} \subset R^L_{L_j(\,)}$.

Given A we know that there is a $\Phi_p(\,)$ such that $\Phi_i(\,) \underset{\substack{i.o.\\S}}{<} \Phi_p(\,) \underset{a.e.}{\leq} L_j(\,)$ (i.e., $\Phi_i(n) < \Phi_p(n)$ for $n \in S$ and S an infinite subset of the integers), and if $\phi_s(\,) = \phi_p(\,)$, then $\Phi_i(\,) \underset{i.o.}{<} \Phi_s(\,)$. If $\Phi_p(\,)$ is a tape complexity, then B follows using $\phi_p(\,)$ for properness. However, if $\Phi_p(\,)$ is a new complexity function, then $\phi_p(\,)$ may not be in $R^L_{L_j(\,)}$ . We show how to construct an $L_q(\,)$ such that

* $\Phi_i(\,) \underset{i.o.}{<} L_q(\,) \underset{a.e.}{\leq} L_j(\,)$ and

$\phi_s(\ ) = \phi_q(\ )$ implies $\Phi_i(\ ) < \Phi_s(\ )$ i.o. and $\Phi_i(\ ) \leq L_q(\ )$ a.e. .

We distinguish two cases, $p > i$ and $p < i$ , i.e., $\Phi_p(\ )$ constructed after $\Phi_i(\ )$ or $\Phi_p(\ )$ constructed before $\Phi_i(\ )$ . We also specify our first restriction on the construction of the new complexity functions.

R1: If $\Phi_r(n)$ is constructed after $\Phi_s(n)$ and $\Phi_s(n) < \Phi_r(n)$, then we require that $L_{\Phi_s}(n) < \Phi_r(n)$.

It is easily seen that R1 does not alter the results of Theorem 3.3

Now for $p > i$ we know $\Phi_i(n) < L_{\Phi_i}(n) < \Phi_p(n)$ for $n \varepsilon s$ . Moreover we recall from Theorem 3.3 that if $\phi_s(n) = \phi_i(\ )$ , then $L_s(\ ) > \Phi_i(\ )$ a.e. . We now tkae $\phi_q(\ )$ to be the function obtained by running in the minimum of $L_{\Phi_i}(\ )$ and $L_j(\ )$ . It is easy to see that $L_q(\ )$ satisfies * .

If $p < i$ then we use a different trick. For this we need another restriction.

R2: Given the tape complexity $L_k(\ )$, define the associated gap function, $\tau_k(\ )$, so that it lies lies above the tape complexity, $L_{k'}(\ )$, of the function $\phi_{k'}(\ )$ obtained by applying the compression (or jump) procedure (of our Theorem 3.1) to $L_k(\ )$.

Now we observe that if $p < i$ and $\Phi_i(n) < \Phi_p(n)$, then $L_{pp}(n)$ must satisfy $\Phi_i(n) < L_{pp}(n) < \Phi_p(n)$ , for otherwise $\Phi_i(n)$ would be forced above $\Phi_p(n)$ by the gap defining procedure. If $\Phi_i(\ ) < \Phi_p(\ )$ a.e., then $L_{pp}(\ )$ suffices for $L_q(\ )$ and R2 is used to prove that $L_{pp}(\ )$ satisfies * . If on the other hand $\Phi_p(\ ) < \Phi_i(\ )$ i.o., then we must use a more careful observation to discover the right $\phi_q(\ )$. We first need another restriction.

R3: The gap size $h(\ )$ must be taken to be a tape complexity function.

Again the restriction does not negate any of the conclusions of Theorem 3.3 since there is an increasing tape function above every recursive function.

Using this restriction we notice that it is possible to decide whether $\Phi_i(n) < \Phi_p(n)$ in less than $h^{(2)}(L_{pp}(n))$ tape. This fact is tedious to verify in detail, but it follows informally because if $\Phi_i(n) < \Phi_p(n)$, then $\Phi_i(n) < L_{pp}(n)$, and the interval between $L_{ii}(n)$ and $L_{pp}(n)$ can be examined using only $h^{(2)}(L_{pp}(n))$ tape to determine whether $\Phi_i(n)$ lies in it.

We then observe that $h^{(2)}(L_{pp}(n)) < L_j(n)$ for a.e.n. using R1 and the fact that each new complexity function is a two sided gap of size $h(\ )$ . We define $\phi_q(\ )$ as the function obtained by applying compression to $L_{pp}(\ )$ if $\Phi_i(n) < \Phi_p(n)$ and as $\phi_j(\ )$ otherwise. It is easy to show that $\phi_q(\ )$ satisfies * . This concludes the proof that A implies B .

We now turn to showing (II) for case (2), that (A implies B) implies (A implies C) . From A implies B and A we conclude that $R^L_{\Phi_i(\ )} \subset R^L_{L_j(\ )}$ and moreover from * that

$$R^L_{\Phi_i(\ )} \subset R^L_{L_q(\ )} \subseteq R^L_{L_j(\ )} \ .$$

But because each new complexity function, $\Phi_p(\ )$, is a two sided gap, i.e., if $L_k(\ ) \leq \Phi_p(\ )$ a.e., then $h(L_k(\ )) \leq \Phi_p(\ )$ a.e., we can construct a function $L_r(\ )$ such that

$$L_q(\ ) \underset{\text{i.o.}}{<} L_r(\ ) \leq L_j(\ ) \quad \text{and}$$

$$L_q(\ ) \leq L_r(\ ) \quad \text{a.e.} \quad \text{and}$$

$$\phi_s(\ ) = \phi_r(\ ) \text{ implies } L_q(\ ) < L_s(\ ).$$

We do this by applying compression to the part of $L_q(\ )$ which is below $L_j(\ )$ . We have shown that this part car be recognized within $L_j(\ )$ so that the resulting function $\phi_r(\ )$ satisfies the above conditions. Thus

$$R^{L}_{\Phi_i}(\ ) \subset R^{L}_{L_q}(\ ) \subset R^{L}_{L_j}(\ ) .$$

Now it is easy to observe that this relationship carries over to $\Phi$. So

$R^{\Phi}_{\Phi_i}(\ ) \subset R^{\Phi}_{L_q}(\ ) \subset R^{\Phi}_{L_j}(\ )$, and A implies C.

Q.E.D.


## Acknowledgement

We would like to thank Paul R. Young, who has independently constructed a measure $\Phi$ for which neither $\Sigma^{\Phi}$ nor $\Omega^{\Phi}$ is dense, for the stimulation he provided in the writing of this paper.


## References


[1] Blum, M. "Machine-Independent Theory of the Complexity of Recursive Functions" JACM, 14, 1967, p.322-36.

[2] Borodin, A. "Complexity Classes of Recursive Functions and the Existence of Complexity Gaps" Proc. ACM Symposium on the Theory of Computing, 1969, p.67-78.

[3] Hartmanis, J. & Stearns, R.E. "On the Computational Complexity of Algorithms" Trans. AMS, 117, 5, 1965, p.285-306.

[4] Hennie, F.C. & Stearns, R.E. "Two Tape Simulation of Multitape Machines" JACM, 13, 4, 1966, p.533-546.

[5] Kleene, S.C. Introduction to Mathematics , Princeton, 1952.

[6] McCreight, E.M. & Meyer, A.R. "Classes of Computable Functions Defined by Bounds on Computation" Proc. ACM Symposium on Theory of Computing, Marina del Rey, 1969, p.79-88.

[7] Rogers, H. Theory of Recursive Functions and Effective Computability, New York, 1967.

[8] Rogers, H. "Gödel Numberings of Partial Recursive Functions" J. Symbolic Logic, 22, 3, 1958, p.331-341.

[9] Young, Paul R. "Speed-Up by Changing the Order in which Sets are Enumerated" ACM Symposium on Theory of Computing, May 1967, p.89-92.