Institute for Language, Logic and Information

ON SEQUENCES WITH SIMPLE INITIAL SEGMENTS

Domenico Zambella

ITLI Prepublication Series
for Mathematical Logic and Foundations ML-90-05

University of Amsterdam

The ITLI Prepublication Series

1986

86-01 The Institute of Language, Logic and Information
86-02 Peter van Emde Boas — A Semantical Model for Integration and Modularization of Rules
86-03 Johan van Benthem — Categorial Grammar and Lambda Calculus
86-04 Reinhard Muskens — A Relational Formulation of the Theory of Types
86-05 Kenneth A. Bowen, Dick de Jongh — Some Complete Logics for Branched Time, Part I Well-founded Time, Forward looking Operators
86-06 Johan van Benthem — Logical Syntax

1987

87-01 Jeroen Groenendijk, Martin Stokhof — Type shifting Rules and the Semantics of Interrogatives
87-02 Renate Bartsch — Frame Representations and Discourse Representations
87-03 Jan Willem Klop, Roel de Vrijer — Unique Normal Forms for Lambda Calculus with Surjective Pairing
87-04 Johan van Benthem — Polyadic quantifiers
87-05 Víctor Sánchez Valencia — Traditional Logicians and de Morgan's Example
87-06 Eleonore Oversteegen — Temporal Adverbials in the Two Track Theory of Time
87-07 Johan van Benthem — Categorial Grammar and Type Theory
87-08 Renate Bartsch — The Construction of Properties under Perspectives
87-09 Herman Hendriks — Type Change in Semantics: The Scope of Quantification and Coordination

1988

Logic, Semantics and Philosophy of Language:

LP-88-01 Michiel van Lambalgen — Algorithmic Information Theory
LP-88-02 Yde Venema — Expressiveness and Completeness of an Interval Tense Logic
LP-88-03 Year Report 1987
LP-88-04 Reinhard Muskens — Going partial in Montague Grammar
LP-88-05 Johan van Benthem — Logical Constants across Varying Types
LP-88-06 Johan van Benthem — Semantic Parallels in Natural Language and Computation
LP-88-07 Renate Bartsch — Tenses, Aspects, and their Scopes in Discourse
LP-88-08 Jeroen Groenendijk, Martin Stokhof — Context and Information in Dynamic Semantics
LP-88-09 Theo M.V. Janssen — A mathematical model for the CAT framework of Eurotra
LP-88-10 Anneke Kleppe — A Blissymbolics Translation Program

Mathematical Logic and Foundations:

ML-88-01 Jaap van Oosten — Lifschitz' Realizabiility
ML-88-02 M.D.G. Swaen — The Arithmetical Fragment of Martin Löf's Type Theories with weak Σ-elimination
ML-88-03 Dick de Jongh, Frank Veltman — Provability Logics for Relative Interpretability
ML-88-04 A.S. Troelstra — On the Early History of Intuitionistic Logic
ML-88-05 A.S. Troelstra — Remarks on Intuitionism and the Philosophy of Mathematics

Computation and Complexity Theory:

CT-88-01 Ming Li, Paul M.B.Vitanyi — Two Decades of Applied Kolmogorov Complexity
CT-88-02 Michiel H.M. Smid — General Lower Bounds for the Partitioning of Range Trees
CT-88-03 Michiel H.M. Smid, Mark H. Overmars, Leen Torenvliet, Peter van Emde Boas — Maintaining Multiple Representations of Dynamic Data Structures
CT-88-04 Dick de Jongh, Lex Hendriks, Gerard R. Renardel de Lavalette — Computations in Fragments of Intuitionistic Propositional Logic
CT-88-05 Peter van Emde Boas — Machine Models and Simulations (revised version)
CT-88-06 Michiel H.M. Smid — A Data Structure for the Union-find Problem having good Single-Operation Complexity
CT-88-07 Johan van Benthem — Time, Logic and Computation
CT-88-08 Michiel H.M. Smid, Mark H. Overmars, Leen Torenvliet, Peter van Emde Boas — Multiple Representations of Dynamic Data Structures
CT-88-09 Theo M.V. Janssen — Towards a Universal Parsing Algorithm for Functional Grammar
CT-88-10 Edith Spaan, Leen Torenvliet, Peter van Emde Boas — Nondeterminism, Fairness and a Fundamental Analogy
CT-88-11 Sieger van Denneheuvel, Peter van Emde Boas — Towards implementing RL

Other prepublications:

X-88-01 Marc Jumelet — On Solovay's Completeness Theorem

1989

Logic, Semantics and Philosophy of Language:

LP-89-01 Johan van Benthem — The Fine-Structure of Categorial Semantics
LP-89-02 Jeroen Groenendijk, Martin Stokhof — Dynamic Predicate Logic, towards a compositional, non-representational semantics of discourse
LP-89-03 Yde Venema — Two-dimensional Modal Logics for Relation Algebras and Temporal Logic of Intervals
LP-89-04 Johan van Benthem — Language in Action
LP-89-05 Johan van Benthem — Modal Logic as a Theory of Information
LP-89-06 Andreja Prijatelj — Intensional Lambek Calculi: Theory and Application
LP-89-07 Heinrich Wansing — The Adequacy Problem for Sequential Propositional Logic
LP-89-08 Víctor Sánchez Valencia — Peirce's Propositional Logic: From Algebra to Graphs
LP-89-09 Zhisheng Huang — Dependency of Belief in Distributed Systems

Mathematical Logic and Foundations:

ML-89-01 Dick de Jongh, Albert Visser — Explicit Fixed Points for Interpretability Logic
ML-89-02 Roel de Vrijer — Extending the Lambda Calculus with Surjective Pairing is conservative
ML-89-03 Dick de Jongh, Franco Montagna — Rosser Orderings and Free Variables
ML-89-04 Dick de Jongh, Marc Jumelet, Franco Montagna — On the Proof of Solovay's Theorem
ML-89-05 Rineke Verbrugge — Σ-completeness and Bounded Arithmetic
ML-89-06 Michiel van Lambalgen — The Axiomatization of Randomness
ML-89-07 Dirk Roorda — Elementary Inductive Definitions in HA: from Strictly Positive towards Monotone
ML-89-08 Dirk Roorda — Investigations into Classical Linear Logic
ML-89-09 Alessandra Carbone — Provable Fixed points in $I\Delta_0+\Omega_1$

Computation and Complexity Theory:

CT-89-01 Michiel H.M. Smid — Dynamic Deferred Data Structures
CT-89-02 Peter van Emde Boas — Machine Models and Simulations
CT-89-03 Ming Li, Herman Neuféglise, Leen Torenvliet, Peter van Emde Boas — On Space Efficient Simulations
CT-89-04 Harry Buhrman, Leen Torenvliet — A Comparison of Reductions on Nondeterministic Space
CT-89-05 Pieter H. Hartel, Michiel H.M. Smid, Leen Torenvliet, Willem G. Vree — A Parallel Functional Implementation of Range Queries
CT-89-06 H.W. Lenstra, Jr. — Finding Isomorphisms between Finite Fields
CT-89-07 Ming Li, Paul M.B. Vitanyi — A Theory of Learning Simple Concepts under Simple Distributions and Average Case Complexity for the Universal Distribution (Prel. Version)
CT-89-08 Harry Buhrman, Steven Homer, Leen Torenvliet — Honest Reductions, Completeness and Nondeterminstic Complexity Classes
CT-89-09 Harry Buhrman, Edith Spaan, Leen Torenvliet — On Adaptive Resource Bounded Computations
CT-89-10 Sieger van Denneheuvel — The Rule Language RL/1
CT-89-11 Zhisheng Huang, Sieger van Denneheuvel, Peter van Emde Boas — Towards Functional Classification of Recursive Query Processing

Other Prepublications:

X-89-01 Marianne Kalsbeek — An Orey Sentence for Predicative Arithmetic
X-89-02 G. Wagemakers — New Foundations: a Survey of Quine's Set Theory
X-89-03 A.S. Troelstra — Index of the Heyting Nachlass
X-89-04 Jeroen Groenendijk, Martin Stokhof — Dynamic Montague Grammar, a first sketch
X-89-05 Maarten de Rijke — The Modal Theory of Inequality
X-89-06 Peter van Emde Boas — Een Relationele Semantiek voor Conceptueel Modelleren: Het RL-project

1990 SEE INSIDE BACK COVER

Instituut voor Taal, Logica en Informatie
Institute for Language, Logic and Information

Faculteit der Wiskunde en Informatica
(Department of Mathematics and Computer Science)
Plantage Muidergracht 24
1018TV Amsterdam

Faculteit der Wijsbegeerte
(Department of Philosophy)
Nieuwe Doelenstraat 15
1012CP Amsterdam

ON SEQUENCES WITH SIMPLE INITIAL SEGMENTS

Domenico Zambella
Department of Mathematics and Computer Science
University of Amsterdam

ITLI Prepublications
for Mathematical Logic and Foundations
ISSN 0924-2090

Received July 1990

ON SEQUENCES WITH SIMPLE INITIAL SEGMENTS.

Domenico Zambella

0. Introduction.

In this report we will deal with descriptive complexity and with Turing degrees. Sequences can be classified in various classes according to their descriptive complexity; some of this classes have got an important heuristic meaning as the class of random sequences they are definable as the class of sequences with "high" complexity. The connection between descriptive complexity and recursion theory has not been systematically investigated. Some results (by Mayer [1], Chaitin [2] and Solovay [3]) are known for classes of sequences of "low" complexity, namely it is known whether or not certain inequalities characterize the recursive sequences. Other results have been obtained for the class of random sequences by Kurtz [4] and by Kucera [5]; they interpret randomness as a sort of genericity. (In a more general contest, M.van Lambalgen [6] also showed there exist a deep connection between forcing and randomness.) Our contribute is to be intended in the former direction, we mainly deal with low complexity sequences and present two main constructions of nonrecursive sets which look very simple from the point of view of the descriptive complexity of their initial segments. We also would like to add few observations on the interplay between randomness and sequences of low complexity. In [6] the notion of stochastic independence of sequences is introduced using the relativized definition of randomness. We would like to make precise intuitive fact that if a sequence X is independent from Y then Y has few information about X hence Y has in a certain sense very low complexity.

We deal with two quite different kinds of descriptive complexity, namely Chaitin's prefix complexity I [7], and the entropy H as defined in [8]; the exact definitions will be given in the next sections. The first construction we consider is by R. Solovay [3] apart from some minor simplifications due to the author. The second is entirely the author's. The main ingredient of the first construction is priority mixed with estimating the time complexity. The main ingredients of the second proof are the fixed point theorem and the tree construction of hyperimmune free degrees. The two main sections of this paper can be read independently.

Let us first present our notation. We associate to every natural number n the set of the natural numbers less then n, i.e. $n=\{0,\dots,n-1\}$, 2^n is the set of functions from n to 2, and $2^{<\omega}:=\bigcup_{n\in\omega}2^n$. The elements of $2^{<\omega}$ are called (finite) strings if $\sigma\in 2^{<\omega}$ we denote with $|\sigma|$ the length of σ i.e. the domain of σ. The set of the natural numbers is denoted by ω, 2^ω is the set of the functions from ω to 2, called sequences. To each sequence we associate the set $\{n \mid X(n)=1\}$ and we denote it with the same symbol X since no confusion will arise. If $X\in 2^\omega$ we denote by $X\lceil n$ the restriction of X to n. The relation $\subseteq$ between strings or between a string and a sequence is the usual extension relation between partial functions; $\perp$ is the incompatibility relation between partial functions. We assume a given effective bijective enumeration of the set $2^{<\omega}$, we will use the same symbol both for a string and for the number associated to it. There is large freedom in the choice of the enumeration of the strings. We require only that for every string σ, $|\sigma|<\sigma$ and that the operation $|\cdot|$ of length and concatenation * of strings translate into primitive recursive operations. We also write 0^n to mean the string of length n consisting of just 0's. If f is a partial function and $n\notin dom(f)$ (i.e. the domain of f) we write $f(n)\uparrow$ or $f(n)=\infty$. We will consider ∞ as larger than any natural number and we convene that $2^{-\infty}=0$ and $-\log_2 0=\infty$. A function $f:\omega\rightarrow\omega$ is monotone if $\forall n\ f(n)\leqslant f(n+1)$ and divergent if $\lim_n f(n)=\infty$. If B is a (finite) set $\#B$ denote its cardinality. W_e is the domain of the partial recursive function with Gödel number e, as usual $W_{e,s}$ is a finite approximation of W_e. The jump of a set A is indicated as usual with A', thus $\emptyset'$ is a complete r.e. set.

1. Sequences with low prefix complexity.

Let us first briefly recall the definition of prefix complexity. For more details and proofs we refer to [7] and to [9]. A set of string A is prefix free if $\forall\sigma,\tau\in A[\sigma\subseteq\tau\rightarrow\sigma=\tau]$. A prefix algorithm g is a function $g:2^{<\omega}\rightarrow 2^{<\omega}$ such that dom(g) is prefix free. The set of the prefix algorithms is recursively enumerable, so let us fix an effective enumeration and let g_e be the e−th prefix algorithm in this enumeration. The prefix complexity of the string σ with respect to the algorithm g_e is defined by:

$$I_e(\sigma):=|\mu_\tau\, g_e(\tau)\downarrow=\sigma|.$$

If there is no τ such that $g_e(\tau)\downarrow=\sigma$ we define $I_e(\sigma)=\infty$. Consider now the following prefix algorithm U:

$$U(\sigma):=\begin{cases} g_e(\tau) & \text{if for some } e,\ \sigma=0^e*1*\tau \\ \uparrow & \text{otherwise.}\end{cases}$$

U is called the universal prefix algorithm. The reason is that, defining $I(\sigma):=|\mu_\tau U(\tau)\downarrow=\sigma|$, one has that $I(\cdot)$ is asymptotically optimal, namely:

(1.1) $$\forall e\forall\sigma\ I(\sigma)\leqslant I_e(\sigma)+e+1.$$

The proof of the main theorem of this section will need the following basic lemma. We will sketch its proof since in the sequel we refer explicitly to the construction used.

Lemma 1.1
If f is a recursive partial function such that $\Sigma\{2^{-f(n)}|\ n\in\omega\}\leqslant 1$, then:

$$\exists c\forall\sigma\ \ I(\sigma)\leqslant f(\sigma)+c.$$

Proof. We will construct by stages a prefix algorithm g such that $\forall\sigma[f(\sigma)\downarrow=p\rightarrow\exists\tau\in 2^p\ g(\tau)\downarrow=\sigma]$. Let W_e be the domain of f. Without loss of generality we can assume that for all s, $\#(W_{e,s+1}\setminus W_{e,s})\leqslant 1$. We define g using the following construction. g_s is the finite approximation of the graph of g.

Stage 0. Let $g_0=\emptyset$.
Stage s+1. If there exists an $n\in W_{e,s+1}\setminus W_{e,s}$ then let $p:=f(n)$ and $\sigma:=\mu_\tau(|\tau|=p\ \&\ \forall\delta\in\mathrm{dom}(g_s)\ \delta\perp\tau)$. Define $g_{s+1}=g_s\cup\{\langle\sigma,n\rangle\}$.

We leave to the reader the task of verifying that the construction gives the desired algorithm g.
□

Let us now come to the principal subject of this section. The reader can easily verify that if $Z\in 2^\omega$ is recursive then:

(1.2) $$\exists c\forall n\ I(Z\lceil n)\leqslant I(n)+c.$$

It is quite nontrivial to prove that the converse does not hold. This is the content of the main theorem of this section:

Theorem 1.2
There exists a nonrecursive r.e. set Z such that:

$$\exists c \forall n\ I(Z\lceil n) \leqslant I(n)+c.$$

□

This result is rather striking, in fact one of the first (non trivial) characterization of recursive sequences via descriptive complexity was Chaitin's proof concerning Kolmogorov complexity [2]. He proved that:

$$\exists c \forall n\ K(Z\lceil n) \leqslant K(n)+c \iff Z \text{ is recursive},$$

where the Kolmogorov complexity of the string σ, $K(\sigma)$ is defined similarly to the prefix complexity $I(\sigma)$ but without the condition that the domains of the algorithms are prefixfree. This turns out to be one of the main differences between these two kinds of descriptive complexities and it makes it worthwhile to go in to the details of the proof.

Before commencing the proof of theorem 1.2 we would like to show that there are rather few of such simple sets.

Proposition 1.3
Let $B_c = \{Z \in 2^\omega \mid \forall n\ I(Z|n)-I(n) \leqslant c\}$ then for every c, $\# B_c < 2^c$.
Proof (Sketch). Assume for a contradiction that for a $k > 2^c$ there exist $Z_1, \ldots Z_k \in B$ such that $Z_i \neq Z_j$ whenever $i \neq j$. Let n be such that $Z_i\lceil n \neq Z_j\lceil n$ whenever $i \neq j$. A simple counting argument shows that for any number m, one has $|\{\sigma \mid I(\sigma) \leqslant m\}| < 2^m$. Then k must be less than $2^{I(n)+c}$. For each $i \leqslant k$ define $\sigma_i = Z_i\lceil n$ and consider the σ_i's as k different numbers greater than n, thus $Z_i\lceil\sigma_j \neq Z_u\lceil\sigma_v$ if $\langle i,j\rangle \neq \langle u,v\rangle$. Hence one has:

$$I(Z_i\lceil\sigma_j) \leqslant I(\sigma_j) + c \leqslant I(n) + 2\cdot c$$

There are K^2 such strings, so K^2 must be less than $2^{I(n)+2c}$. Iterating this procedure sufficiently many times one will get a contradiction.
□

We remark that the proposition 1.3 holds also for Kolmogorov complexity; it is not enough to have recursiveness but as a corollary we have that with the help of a $\emptyset'$ oracle one can reconstruct any such low complexity set Z. This result was first obtained by Chaitin by a different argument.

Corollary 1.4

If Z satisfies (1.2) then $Z \leqslant_T \emptyset'$.

Proof. Suppose Z satisfies $\forall n\ I(Z\lceil n)-I(n) \leqslant c$ for a certain constant c. Then by proposition 1.3 there are only finitely many sets X such that $\forall n\ I(X\lceil n)-I(n) \leqslant c$. Choose a string σ such that Z is the unique sequence which satisfies: $X \supseteq \sigma \,\&\, \forall n\ I(X\lceil n)-I(n) \leqslant c$. Define the following set:

$$W := \{\tau \supseteq \sigma \mid \forall \delta < |\tau| .\ I(\delta)-I(|\delta|) \leqslant c\}.$$

Then Z is the unique set such that $\forall n\ Z\lceil n \in W$. By Königs lemma, if $\alpha \in W$ and $\alpha \subseteq Z$, then for $i \in 2$ either $\exists n \forall \tau \in 2^n\ \alpha * i * \tau \notin W$ or $\exists n \forall \tau \in 2^n\ \alpha * (i-1) * \tau \notin W$. Since $W \leqslant_T \emptyset'$, with a $\emptyset'$ oracle we can know which case actually holds and if $\exists n \forall \tau \in 2^n\ \alpha * i * \tau \notin W$ holds then we know that $\alpha * (i-1) \subseteq Z$.

□

Theorem 1.2 is an immediate corollary of the next proposition. We recall that U is the universal prefix algorithm. Define $n^* = \mu_x\ U(x)\downarrow = n$, so that $I(n) := |n^*|$.

Proposition 1.5

There exists a nonrecursive r.e. set Z and a prefix algorithm F such that:

(1.3) $$\forall n\ F(n^*) = Z\lceil n.$$

□

The construction of this prefix algorithm F is due to Solovay. It is a priority construction, which is rather complex because of the presence of a global restraint function. Usually restraint functions arise when one has to combine positive requirements (which require some elements to be enumerated in the set) with negative requirements (which require some elements to be kept out of the set). Each negative requirement gives rise to a restraint function. In some constructions of recursion theory it is possible in special cases to drop the restraints, having elements enter the set to meet the positive requirements while still eventually satisfying all the negative requirements. In our case the interesting feature is that it is impossible to find a strategy to drop the restraints. Nevertheless we prove that they drop out automatically without any "external trigger" so that the positive requirements will be eventually satisfied.

Proof of proposition 1.5. We will construct the r.e. set Z as usual by stages and in order to make it nonrecursive we will make it simple (in the sense of Post). Namely, we will satisfy the usual requirements for Post's simple set:

$$P_e\text{: } W_e \text{ infinite} \Rightarrow Z \cap W_e \neq \emptyset$$

and

$$N_e : \#\{x \in Z \mid x < e\} \leqslant e/2.$$

In order to satisfy (1.3) Z has to meet the requirements:

$$R_n : Z\lceil n = F(n^*).$$

These will play the role of negative requirements. At stage s, Z_s will be the finite approximation of Z. To meet the R_n's we compute at each stage s a recursive approximation of n^*, namely, we fix a (for the moment arbitrary) recursive monotone increasing function $t:\omega \rightarrow \omega$ and compute $n_s^* = \mu_x\, U_{t(s)}(x)\downarrow = n$. Then, if at stage s our guess is that $n_s^* = n^*$, we define $F(n_s^*) = Z_s\lceil n$. Once $n_s^* = n$ is guessed, we maintain this guess until there is an evidence that the guess was wrong, i.e. until a stage t is reached at which $n_t^* < n_s^*$. Whilst we are guessing $n_s^* = n^*$, we have to prevent elements less than n from entering Z_s, otherwise, in case our guess were correct, we would end up with $Z\lceil n \neq F(n^*)$. It remains to decide on the time to guess $n_s^* = n^*$. That is when: $n < s$ and $n_s^*\downarrow = n_{s+1}^*\downarrow$. The resulting restraint function is:

$$r(s) := \max\{n < s \mid n_s^*\downarrow = n_{s+1}^*\downarrow\}$$

so that we are allowed to enumerate in Z at stage s only those elements that are greater than r(s). In order to satisfy the N_e's we use the usual (harmless) trick due to Post, namely we will enumerate in Z_s an element x of W_i only if $x > 2\cdot i$. In the following construction F_s is the finite approximation of the graph of F.

Stage 0. $Z_0 = \emptyset$, $F_0 = \emptyset$

Stage s+1. For each $i \leqslant s$, if $W_{i,s} \cap Z_s = \emptyset$ and there is an $x \in W_{i,s}$ such that $x > r(s)+2 \cdot i$, enumerate the minimal such x in Z_{s+1} and $\langle n_{s+1}{}^*, Z_{s+1} \lceil n \rangle$ in F_{s+1}.

The reader will easily verify that $F = \bigcup F_s$ is actually the graph of a prefix algorithm and, since $\lim_s n_s{}^* = n^*$, that $\forall n\ F(n^*)\downarrow = Z\lceil n$. The requirements N_e are also trivially satisfied.

A much more difficult task is to prove that Z meets all the P_e's. Assume for a contradiction that for a certain index e, W_e is infinite and $W_e \cap Z = \emptyset$. Let $G(s) = \max W_{e,s} \dot{-} 2 \cdot e$. G is a monotone divergent primitive recursive function. For all s we have $r(s) \geqslant G(s)$, otherwise some elements of W_e would enter Z at stage s+1 satisfying R_e. The following lemma proves that it is possible to choose the function t in such a way that this does not happen.

Lemma 1.6.
There is a recursive function $t:\omega \rightarrow \omega$ such that for every monotone divergent primitive recursive function G, $\exists s\ r(s) < G(s)$.
Proof. Let A be a strictly monotone recursive function which dominates every primitive recursive function. Let $t(0) := A(0)$ and suppose $t(s-1)$ has already been defined. Then let $t(s) := A(z)$ where z is the least integer such that:

(a) $t(s-1) < z$

(b) $\forall n \leqslant s\ \ \mu_x(U_z(x)\downarrow = n) = \mu_x(U_{A(z)}(x)\downarrow) = n$

Now assume for a contradiction that there exists a monotone divergent primitive recursive function G such that $\forall s\ G(s) \leqslant r(s)$. For some s we will construct new programs for all $n \in [G(s), s]$ which are short (i.e. shorter then $n_s{}^*$) and quick (i.e. converge in less than $t(s+1)$ steps) so that $r(s+1) < G(s+1)$, a contradiction.

Since we need a rough estimate of the time (i.e. the number of steps) a program takes to converge, it is necessary to clarify the model of computation we are going to use. It is out of the question to go into the details During the proof we will merely briefly comment on some facts that the reader can check for himself. One needs a specific representation of algorithms (e.g. the usual deterministic Turing machines would be suitable) to check facts as: "every primitive recursive function can be competed by an algorithm converging in a time which is a primitive recursive function of the input" or: "if P(s,x) is a primitive recursive predicate then the function $f(x)=\mu_s P(s,x)$ can be computed by an algorithm converging in a time which is a primitive recursive function of the input and the output".

With this in mind we start giving some definitions.

Definitions.
Let $I(n,s):=|\mu_x\, U_s(x)\downarrow = n)|$, $\alpha_s(m)=\min\{I(j,t(s)) \mid m \leqslant j\}$
and $\alpha(m)=\lim_s \alpha_s(m)$. Let $s_k= \mu_s\, \alpha_s(G(s))>2\cdot k$.
□

Lemma 1.7.
The function s_k is total and converges in a time which is a primitive recursive function of $t(s_k)$.
Proof. If the function G(s) is eventually increasing, then there is an n such that $\alpha(G(s))>2\cdot k$ and a fortiori $\alpha_s(G(s))>k$. Notice that the function s_k is eventually increasing with k.
□

Lemma 1.8.
For k sufficiently large: $\sum\{2^{-I(n,t(s_k))} \mid n\in[G(s_k),s_k]\} \leqslant 2^{-k}$.
Proof. Assume the opposite, i.e. there are arbitrary large k such that:

$$\sum\{2^{-I(n,t(s_k))} \mid n\in[G(s_k),s_k]\} > 2^{-k}$$

and let:

$$\theta(t)=\sum_{n<t} 2^{-I(n,t)}$$

For every t, $\theta(t)$ is a rational number strictly less then 1. We consider $\theta(t)$ as a finite binary string encoding in the usual way the fractional part of the corresponding rational numbers. In this proof the relation "$\leqslant$" between strings holds if the relation "less than or equal to" holds between the corresponding rational numbers. Consider now the algorithm which on input $|w|*w$ outputs the minimal i such that $\theta(i)\geqslant w$. Let z_k be such that $A(z_k)=t(s_k)$ and let w_k be the substring of $\theta(z_k)$ of length k. We claim that the number i_k produced by this algorithm on input $|w_k|*w_k$ has to be greater then $G(s_k)$. In fact since

$$\sum\{2^{-I(n,A(z_k))} \mid n\in[G(s_k),s_k]\} > 2^{-k}$$

by the condition (b) on the function t we have:

$$\sum\{2^{-I(n,z_k)} \mid n\in[G(s_k),s_k]\} > 2^{-k}.$$

The algorithm just defined will converge on input w_k in a time which is a primitive recursive function of z_k. Therefore if z_k is large enough, it converges in less then $A(z_k)=t(s_k)$ steps so that for some constant c, $I(i_k,t(s_k)) \leqslant | \, |w_k|*w_k \, |+c \leqslant k+\log_2(k)+c$. It being the case that $i_k \geqslant G(s_k)$, by the definition of s_k, we have that $I(i_k,t(s_k)) \geqslant \alpha(s_k) \geqslant 2\cdot k$. Then for k sufficiently large the two inequalities lead to a contradiction.

□

Using lemma 1.1 we can find a prefix function V^k such that for all $n\in[G(s_k),s_k]$ there is an x such that $V^k(x)\downarrow=n$ and $|x|=I(n,t(s_k))-k$. Examining the proof of this basic lemma one realizes that the time that $V^k(x)$ takes to converge is a primitive recursive function of $t(s_k)$.

Finally consider the prefix algorithm C that on input $0^{|k|}*x$
(1) computes s_k as in the lemma,
(2) computes $t:=t(s_k)$ and $G(s_k)$,
(3) outputs $n:=V^k(x)$.
The time C needs to converge is a primitive recursive function of $t(s_k)$. Therefore, when k is large enough we have for a constant c and all $n\in[G(s_k),s_k]$, $I(n,t(s_k+1))<|x|+|k|+c$. By the definition of $V^k(x)$, $|x|=I(n,t(s_k))-k$, then $I(n,t(s_k+1))<I(n,t(s_k))-k+|k|+c$. Choosing a sufficiently large k one has that for all $n\in[G(s_k),s_k]$, $n_{sk+1}*<n_{sk}*$. This implies $r(s_k)<G(s_k)$.

This concludes the proof of lemma 1.6.

□

2. Sequences with low entropy.

It is hard to find a convincing formalization of the notion of randomness. Since the time of von Mises many attempts turned out to be inconsistent. The work made by Martin Löf [10] in this directions is one of the most convincing and successful. Martin Löf declared nonrandom those sequences which belong to some constructively definable set of measure zero. His key concept is that of sequential test. Sequential tests correspond roughly to the statistical tests that one can actually perform on a computer. We give the precise definition:

Definition 2.7 If $U \subseteq \omega$ we define $U^n = \{x \mid \langle n,x\rangle \in U\}$ and $[U^n] = \{X \in 2^\omega \mid \exists \sigma \in U^n \; X \supseteq \sigma\}$ (recall that we identify natural numbers with binary sequences), $\lambda[U^n]$ is the Lebesgue measure of $[U^n]$. We call U a recursive sequential test if U is r.e. and for every n, $\lambda[U^n] \leqslant 2^{-n}$. The definition can also be relativized to any arbitrary set $A \subseteq \omega$: if such an U is r.e. in A we say U is a sequential test recursive in A.

(Many equivalents definitions could be given also in terms of descriptive complexity; we shall not do so, however, and we refer to the literature for more details; e.g. [9] and [11].)

Recursive sequential tests give rise to one of the most accepted and studied notion of "effective null measure": a set $G \subseteq 2^\omega$ has effective null measure if there is a recursive sequential test U such that $G \subseteq \bigcap_{n \in \omega} \bigcup_{m>n} [U^n]$. A sequence X is said to be random if it is not contained in any set of effective null measure, i.e. if for every recursive sequential test U there are at most finitely many n such that $X \in [U^n]$. When this definition is relativized to a set A we shall speak of A−random sequences.

In a recent paper of M.van Lambalgen [6] it is argued in a very general context that the key notion of randomness is in fact that of stochastic independence. He presented a few quite intuitive axioms that any relation of stochastic independence has to satisfy; he also proved these axioms are consistent. In the same article it is also shown that, in the context of Martin Löf notion of randomness, stochastic independence can be defined by the relativized definition of randomness. Actually one can define "X is independent from Y" if X is Y−random. It is shown for example that the following symmetry property holds: if Y is random and X is Y−random then Y is X−random. It roughly correspond to the Fubini theorem.

As remarked in the introduction, the possible relations between randomness and recursion theoretic notions such as Turing degree have been not completely explored yet. Pioneering work has been done by Kurtz [4] and by

Kucera [5]. In the context of stochastic independence recursion theoretic properties of random sets become much more interesting and urgent.

Some obvious recursion theoretic properties are easily established e.g.: if $X \leqslant_T Y$ then X is not Y-random. From this fact and by the quoted symmetry property we have also that: if X is random and $X \leqslant_T Y$ then Y is not X-random. In other words if X is random then the upper cone of degree above X is contained in a sequential test recursive in X. This is an interesting fact for it corresponds to a well-known result of recursion theory: the upper cone of degrees above the degree of any nonrecursive set has measure zero. (We will refer to this theorem as "Sacks' theorem"; the Sacks' proof [12], is an adaptation of an argument of de Leeuw et al. [13].) So by means of stochastic independence it is possible to achieve a rather strong effectivization of the Sacks' theorem for a set of measure 1 of nonrecursive degree. The effective Sacks' theorem holds for every random set and a fortiori for every set which is above (in the Turing degree upper semilattice) a random set. Since not every nonrecursive set bounds a random set (e.g. there is no random set below an incomplete r.e. degree or with minimal degree [5]), a natural question arises: is the upper cone of degrees above any nonrecursive set A contained in a sequential test recursive in A? In other words: are there no A-random sets above any nonrecursive set A?

Another much stronger condition on low complexity arises from a question of M.van Lambalgen. Are there nonrecursive sets A such that for every random set X, X is also A-random? A negative answer to this latter question would imply a negative answer to the former. In fact Kucera [5] has proved that every complete degree (i.e. every degree above $\emptyset'$) contains a random set. Since for every set A, $A' \geqslant_T \emptyset'$, there is a random set $X \equiv_T A'$. If A is such that every random set is also A-random, then $X \geqslant_T A$ would be A-random and the effective Sacks' theorem fails for A.

Unfortunately we do not know a general answer to these questions but we can show, as promised in the introduction, how they are related with the existence of nonrecursive low complexity sets.

We shall exploit a different concept of descriptive complexity: entropy. The entropy of a string has an intuitive description in terms of Turing machines. We consider a universal Turing machine with three one-way infinite tapes: a read-only input tape, a work tape and a write-only output tape. There is a scanning head in each of the three tapes. The cells of each tape contains a blank, a 0, or a 1. The work tape and the output tape are initially blank, each cell of the input tape is filled with a 0 or a 1 by separate toss of an unbiased coin. The a priori probability of the string σ is the probability that this machine writes the string σ on the first cells of the output tape. From the a priory probability of σ we obtain the entropy of σ by taking the integral part of the base-two logarithm. The reader should not confuse it with Chaitin's entropy. Chaitin requires the machine write in the output tape exactly the sting σ and then halts, while we do not care if, after writing the string σ, the machine keeps on computing and/or eventually writes other digits in the output tape.

Let us introduce some notation to restate these definitions more precisely. As usual by $\{e\}^X$ we mean the partial function computed by the Turing machine with Gödel number e and the oracle X. We write simply $\{e\}$ for $\{e\}^{\emptyset}$. It will simplify our notation to assume that the Gödel numbering is such that for each e there is an $e' \leqslant e$ such that $\{e'\}=\{e\}^{\emptyset}$, and $\{e'\}$ does not query the oracle at all. On 2 $(=\{0,1\})$ we fix the fair coin toss probability measure and in 2^{ω} the product measure λ. For every string σ we define $M_e(\sigma):=\lambda\{\tau \mid \sigma\subseteq\{u\}^{\tau}\}$ and $H_e(\sigma):=$ Integral part of $(-\log_2 M_e(\sigma))$.
In the following we fix a Gödel number u such that for every set X, $\{u\}^X=\{j\}^Z$ where j and Z are such that $0^j*1*Z=X$. Then we define the a priory probability of the string σ, $M(\sigma)=\lambda\{\tau \mid \sigma\subseteq\{u\}^{\tau}\}$ and $H(\sigma):=$ Integral part of $(-\log_2 M(\sigma))$. One immediately realizes that function $\{u\}$ is universal in the same sense as in (1.1), namely:

(2.1) $$\forall e\ H(\sigma)\leqslant H_e(\sigma)+e+1.$$

We call the function H entropy. As the previously defined I, H is a measure of descriptive complexity.

Immediately from the definition of H one has that a sequence X satisfies the inequality:

(2.2) $$\exists c \forall n\ H(X\restriction n)\leqslant c$$

if an only if the upper cone of degrees above the degree of X has positive measure. It Thus we have a characterization of the recursive sequences in terms of entropy:

(2.3) $$X \text{ is nonrecursive} \Leftrightarrow \lim_n H(X\restriction n)=\infty$$

An effectivization of Sacks' theorem could be obtained effectivizing the limit $\lim_n H(X\lceil n)=\infty$. The next proposition show how this is related to randomness and more generally to the notion of stochastic independence.

Proposition 2.8
If $\exists f \leqslant_T A$ monotone divergent function such that $\exists^\infty n\, f(n) \leqslant H(A\lceil n)$ then there is no A–random set $X \geqslant_T A$.
Proof Let $g(n):=\mu_m\, f(m) \geqslant n+1$. We define:

$$V_n:=\{\sigma \mid A\lceil g(n) \subseteq \{u\}^\sigma\}.$$

The set $V=\{<n,x> \mid x\in V_n\}$ is r.e. in A. We first show that there are infinitely many n such that $\lambda[V_n] \leqslant 2^{-n}$ i.e., $\forall q \exists n \geqslant q\ \lambda[V_n] \leqslant 2^{-n}$. Fix any q and let p be such that $q \leqslant f(p) \leqslant H(A\lceil p)$, let $n:=f(p)$. By the monotonicity of f, $g(n) \geqslant p$, thus by the monotonicity of H, $n \leqslant H(A\lceil p) \leqslant H(A\lceil g(n))$. So we have:

$$\lambda[V_n] = 2^{-H(A\lceil g(n)} \leqslant 2^{-n}.$$

Now, by standard techniques it is possible to construct a set U, r.e. in A such that:

(i) $\forall n\ \lambda[U_n] \leqslant 2^{-n}$

(ii) $\forall n\ U_n \subseteq V_n$

(iii) if $\lambda[V_n] \leqslant 2^{-n}$ then $U_n=V_n$.

If $X_T \geqslant A$ then $\forall n\ X\in[V_n]$. By (iii) $X\in[U_n]$ for infinitely many n, thus $X\in\bigcap_{n\in\omega}\bigcup_{m>n}[U_n]$ and X is non–random in A.
□

Unfortunately there are nonrecursive sets with very low entropy. The rest of this section is devoted to the proof of the following proposition:

Proposition 2.2
There exists a nonrecursive set Z such that: for every monotone divergent function $f \leqslant_T Z$ and for all but finitely many n,

$$H(Z\lceil n) \leqslant f(n).\square$$

Since to prove this proposition we need a stronger result than Sacks' theorem we give a different proof of the letter.

Lemma 2.1

(i) If $X\lceil n \subseteq \{e\}$ then $H(X\lceil n) \leqslant e+1$

(ii) There exists a recursive function $h(n)$ such that if $H(\sigma) \leqslant n$ then $\sigma \subseteq \{i\}$ for some $i \leqslant h(n)$.

Proof (i) If the algorithm with Gödel number e does not query the oracle then (i) is trivial. Otherwise by our assumption on the numbering there is another Gödel number $e' < e$ such that $\{e'\}=\{e\}$ and $\{e'\}$ never queries the oracle. So (i) follows.

(ii) First observe that there are at most 2^n incompatible strings such that $H(\sigma) \leqslant n$. For each $0 \leqslant i < 2^n$ we now define a (possibly partial) function $X_i:\omega \rightarrow \{0,1\}$ such that for all m, $H(X\lceil m) \leqslant n$. The construction will be effective, thus it is possible to find recursively in n for each i an index e_i such that $\{e_i\}=X_i$. Defining $h(n) := \max\{ e_i \mid 0 \leqslant i < 2^n\}$ the theorem is proved. Let us first construct X_0 as follows:

Let $\sigma_0 := \emptyset$. Assume σ_s has already been defined. Now enumerate recursively $\{\sigma \mid H(\sigma) \leqslant n\}$ till a prolongation τ of σ_s is found. Then put $\sigma_{s+1} := \tau$. Let $X_0 = \bigcup_{s\in\omega} \sigma_s$.

For $i > 0$ we construct X_i in the following way:

Let $\sigma_0 := \emptyset$. Assume σ_s has already been defined. Now enumerate recursively $\{\sigma \mid H(\sigma) \leqslant n\}$ and for all $j < i$ produce X_j, till a prolongation τ of σ_s is found which is incompatible with all the X_j. Put $\sigma_{s+1} := \tau$ and $X_i = \bigcup_{s\in\omega} \sigma_s$.

□

We also need the next lemma to prove proposition 2.1.

Lemma 2.3

For any sequences X,Z the following are equivalent:

(a)$\forall f \leqslant_T X$ monotone divergent functions, for all but finitely many n,

$$H(Z\lceil n) \leqslant f(n).$$

(b)$\forall f \leqslant_T X$ monotone divergent functions, for all but finitely many n,

$$\exists e \leqslant f(n)\ \{e\}\lceil n = Z\lceil n.$$

Proof (a)⇒(b). Suppose f is a counter-example to (b), hence for infinitely many n one has $\{e\}\lceil n = Z\lceil n$ for some $e \leqslant f(n)$. By the lemma 2.1 (i) one has immediately that for infinitely many n, $H(Z\lceil n) \leqslant f(n)+1$.

(b)⇒(a) Suppose f is a counter-example to (a), let h the function defined in lemma 2.1 (ii); we can assume h is monotone divergent. The function $g(n)=\max\{m \mid h(m) \leqslant f(n)\}$ is then a counter-example to (b).

□

To illustrate the method that we are going to use to prove proposition 2.2, we prefer to prove first a weaker proposition.

Proposition 2.4. There exists a nonrecursive set Z such that: for every monotone divergent recursive function f one has that for all but finitely many n,

$$H(Z\lceil n) \leqslant f(n).$$

Proof. (We notice without further comments that this proposition could be also easily proved using the function F constructed in proposition 1.5, but this method would not apply to prove the full proposition 2.2.) By lemma 2.3 the proposition is proved if we can construct a nonrecursive set Z such that:

(*) $\forall f$ monotone divergent recursive function,

$$\exists m \forall n \geqslant m\ \exists e \leqslant f(n)\ \{e\}\lceil n = Z\lceil n.$$

Let $f_0,..,f_s,...$ be an enumeration of all the recursive monotone divergent functions. At stage e we construct an initial segment σ_e of Z and we ensure that σ_e is not an initial segment of $\{e-1\}$. Then $Z=\bigcup_{e\in\omega}\sigma_e$ will not be recursive. At each stage we define also j_e and n_e such that $\{j_e\}\lceil n_e = \sigma_e$. These j_e will play the role of the witnesses of (*) in the sense that we are going to show that for all s and for all but finitely many n: $\exists t\ j_t \leqslant f_s(n)\ \&\ \{j_t\}\lceil n = Z\lceil n$.

Stage 0. Let $\sigma_0=0$ and let j_0 be any number such that $\{j_0\}\in 2^\omega$, $n_0=0$

Stage e+1. Case I. If $\{e\}\notin 2^\omega$ then do nothing, i.e. let $j_{e+1}=j_e$ and $n_{e+1}=n_e$ and define $\sigma_{e+1}=\{j_{e+1}\}\lceil n_{e+1}=\sigma_e$. Case II. Otherwise let $F_e(n)=\min\{f_0(n),..,f_e(n)\}$. Let h be the recursive function such that:

$$\{h(j)\}(n)=\begin{cases}\{j_e\}(n) & \text{if } n<n_e \text{ or } j>F_e(n)\\ 1\dot{-}\{e\}(n) & \text{otherwise.}\end{cases}$$

Let j_{e+1} be a fixed point of the function h (i.e. $\{h(j_{e+1})\}=\{j_{e+1}\}$) and let $n_{e+1}=1+\mu_n(n>n_e \text{ and } j_{e+1}\leqslant F_e(n))$. Finally let $\sigma_{e+1}=\{j_{e+1}\}\lceil n_{e+1}$.

The next lemma proves that our construction works.

Lemma 2.5. For every e,
(i) $\sigma_e\subseteq\sigma_{e+1}$
(ii) if $\sigma_{e+1}\subseteq\{e\}$ then $\{e\}\notin 2^{\omega}$ (i.e. Z is not recursive).
(iii) $\exists i\ |\sigma_i|>e$ (i.e. $Z\in 2^{\omega}$).
(iv) for all but finitely many n, $\exists t\ j_t\leqslant f_e(n)\ \&\{j_t\}\lceil n=Z\lceil n$.
Proof. **(i)** If case I applies then it is trivial. Otherwise if {e} is total then by the definition of h, $\{j_{e+1}\}=\{h(j_{e+1})\}$ coincide with $\{j_e\}$ on $n<n_e$. Therefore $\sigma_e=\{j_e\}\lceil n_e\subseteq\{j_{e+1}\}\lceil n_{e+1}=\sigma_{e+1}$.
(ii) Suppose $\{e\}\in 2^{\omega}$. The definition of h ensures that $\{j_{e+1}\}(n_{e+1}-1)\neq\{e\}(n_{e+1}-1)$.
(iii) Case II applies infinitely many times and at each stage i at which it applies one has $|\sigma_{i+1}|>|\sigma_i|$.
(iv) First observe that for every n, $F_e(n)\leqslant f_e(n)$. So the lemma is proved if we can show that for every e and for all but finitely many n,

$$\exists t\ j_t\leqslant F_e(n)\ \&\{j_t\}\lceil n=Z\lceil n.$$

Let us fix an e and take an m so large that $\forall t\leqslant e\ \{j_t\}\lceil m\neq Z\lceil m$ (by the previous lemma (ii) such an m exists). Assume for a contradiction that there exists an $n\geqslant m$ such that:

$$\forall t\ [\ j_t\leqslant F_e(n)\rightarrow\{j_t\}\lceil n\neq Z\lceil n].$$

Let v be the minimal index such that $\{j_v\}\lceil n=Z\lceil n$, then $j_v>F_e(n)$. Since $n\geqslant m$, by our choice of m we have that $v>e$ and $F_e(n)$ being monotone increasing in n and monotone decreasing in e one has $\forall p<n\ j_v>F_v(p)$. Hence $\{j_v\}\lceil n=\{j_{v-1}\}\lceil n$ against the minimality of v.
□

If we want to strengthen proposition 2.4 to prove our claim, we could replace (*) with (ii) of lemma 2.1. This would force us to a radical change of method. Instead, we will by-pass the difficulty of looking at all the monotone divergent functions recursive in Z by constructing a Z of hyperimmune free degree. In the usual definition, a set Z has hyperimmune free degree if and only if every function $f \leqslant_T Z$ is bounded by a recursive function. One realizes immediately that this is equivalent to: a set Z has hyperimmune free degree if and only if every monotone divergent function $f \leqslant_T Z$ bounds a monotone divergent recursive function. Then satisfying (*) and simultaneously making Z of hyperimmune free degree we still prove our claim.

We need only to reproduce the tree construction of Martin and Miller [14] of hyperimmune free degrees and substitute for the simple original diagonalization method with the one used in proposition 2.4.

We recall some definitions and some basic lemmas, for the proof of which we refer to the literature (e.g. [15] p. 493-498).

Definitions.

A recursive tree is a recursive function $T:2^{<\omega}\to 2^{<\omega}$ such that for all σ, $T(\sigma*0)$ and $T(\sigma*1)$ are incompatible extensions of $T(\sigma)$.

$B\in 2^{\omega}$ is a branch of a tree T (or, B is on the tree T) iff $B=\bigcup_{n\in\omega}T(X\lceil n)$ for some $X\in 2^{\omega}$.

If Q and T are two trees, then $Q\subseteq T$ (i.e. Q is a subtree of T) iff every branch of Q is a branch of T.□

Totality Lemma.

Given e and a recursive tree, there is a recursive tree $Q\subseteq T$ such that one of the following holds:

(a) for every X on Q, $\{e\}^X$ is not total.

(b) for every X on Q, $\{e\}^X$ is total and $\forall n\forall\sigma\in 2^n\ \{e\}^{Q(\sigma)}\downarrow$.□

Proof of proposition 2.2. Since in the end we want to get a set of hyperimmune free degree, the whole construction has to be performed on a tree. We will define a sequence of trees $\{T_e\}_{e\in\omega}$ such that $T_{e+1}\subseteq T_e$. Finally we will take $Z=\bigcup_{e\in\omega}T_e(\emptyset)$. As in the previous proof let $f_0,\ldots,f_n,\ldots$ be an enumeration of all the monotone divergent recursive functions.

Stage 0. Let T_0 be the identity tree and let $n_0=0$ and j_0 be any number such that $\{j_0\}\in 2^{\omega}$. Notice that $\{j_0\}$ is on T_0.

Stage 2e+1. (We have to replace the diagonalization technique with the construction of the previous proposition.) Case I. If {e} is not a characteristic function then let $j_{e+1}=j_e$ and $n_{e+1}=n_e$ and define $T_{2\cdot e+1}=T_{2\cdot e}$. Case II. Otherwise let $F_e(n)=\min\{f_0(n),..,f_e(n)\}$. Define $m_j=\mu_n(n>n_e \,\&\, j\leqslant F_e(n))$. Let B be either the leftmost or the rightmost branch of $T_{2\cdot e}$ which extends $\{j_e\}\lceil m_j$ according to which one is not equal to {e}. The branch B is recursive uniformly in j, i.e. there is a recursive function h(j) such that {h(j)}=B. Let j_{e+1} be a fixed point of the function h and let $n_{e+1}=\mu_n(n>n_e$ and $j_{e+1}\leqslant F_e(n))$. Observe that for each j {h(j)} is on $T_{2\cdot e}$. Then also $\{j_{e+1}\}$ is on $T_{2\cdot e}$. Finally let $T_{2\cdot e+1}=$ the full subtree of $T_{2\cdot e}$ above $\{j_{e+1}\}\lceil n_{e+1}$.

Stage 2·e+2. Let $T_{2\cdot e+2}=$ the Q of the totality lemma for $T_{2\cdot e+1}$.

The next and last lemma proves that the construction works.

Lemma 2.6. For every e:

(o) $T_{e+1}(\emptyset)\subseteq T_e(\emptyset)$

(i) If X is on T_{2e+1} then $X\neq\{e\}$

(ii) $\exists i\ |T_i(\emptyset)|>e$.

(iii) If X is on T_{2e+1} then for all but finitely many n, $\exists t\ j_t\leqslant f_e(n)\ \&\{j_t\}\lceil n=X\lceil n$.

(iv) If X is T_{2e+2} and $\{e\}^X$ is total then there is a recursive function bounding $\{e\}^X$ (i.e. Z has hyperimmune free degree).

Proof. **(o)**, **(i)**, **(ii)** and **(iii)** are similar to the corresponding ones of lemma 2.5, **(iv)** can be found in [15].

□

Aknowledgment.

This work has been done during a visit at the University of Amsterdam with the frequent and friendly help Dr. Michiel van Lambalgen. I wish to express my sincere thanks also to Dr. Dick de Jongh for making it possible for me to stay in Amsterdam in an enjoyable way, and for his kind help in editing this report.

References.

[1] D.W. Loveland, A variant of the Kolmogorov concept of complexity, *Inf. and Control* **15**, (1969) 510–526

[2] G.J. Chaitin, Information -heoretic characterization of recursive infinite strings, *Th. Comp. Sci.* 2, (1976) 45-48

[3] R.M. Solovay, Handwritten notes.

[4] S.A. Kurtz, Randomness and genericity in the degrees of unsolvability, Ph.D. thesis University of Illinois at Urbana Champaign (1981)

[5] A. Kucera, Measure, Π^0_1 classes and complete extensions of PA, in *Recursion theory week*, G.E. Sacks, G.H. Müller, H.D. Edbinghaus (Eds), Lectures Notes in Mathematics, **1141**, Springer Verlag, Heidelberg, (1985) 245–259

[6] M. van Lambalgen, The axiomatization of randomness, to appear in the *Journal of Symbolic Logic.*

[7] G.J. Chaitin, Algorithmic information theory, *I.B.M. J. Res. Dev.*, **21** (1977), 350–359

[8] P. Gacs, On the relation between descriptional complexity and algorithic probability, *Th. Comp. Sci.* **22** (1983) 71–93

[9] M. van Lambalgen, Random sequences, Ph.D. thesis, University of Amsterdam.

[10] P. Martin Löf, The definition of Random sequences, *Inf. and Control* **9** (1966), 602–619.

[11] C.P. Schnorr, *Zufälligkeit und Wahrscheinlichkeit*, Lectures Notes in Mathematics, **218**, Springer Verlag, Heidelberg (1971).

[12] G.E. Sacks, *Degrees of unsolvability*, Princeton University Press, Princeton, New Jersey (1966).

[13] K. de Leeuw, E.F. Moore, C.E. Shannon and N. Shapiro, in *Automata Studies*, C.E. Shannon and J. McCarthey (Eds.), Princeton University Press, Princeton, New Jersey (1956), 183-212.

[14] W. Miller and D.A. Martin, The degree of Hyperimmune sets, *Zeit. Math. Log. Grund. Math.* **14** (1968), 159-166.

[15] P.Odifreddi, *Classical Recurstion Theory*, North Holland, Amsterdam (1989).

The ITLI Prepublication Series

1990

Logic, Semantics and Philosophy of Language

LP-90-01 Jaap van der Does	A Generalized Quantifier Logic for Naked Infinitives
LP-90-02 Jeroen Groenendijk, Martin Stokhof	Dynamic Montague Grammar
LP-90-03 Renate Bartsch	Concept Formation and Concept Composition
LP-90-04 Aarne Ranta	Intuitionistic Categorial Grammar
LP-90-05 Patrick Blackburn	Nominal Tense Logic
LP-90-06 Gennaro Chierchia	The Variablity of Impersonal Subjects
LP-90-07 Gennaro Chierchia	Anaphora and Dynamic Logic
LP-90-08 Herman Hendriks	Flexible Montague Grammar
LP-90-09 Paul Dekker	The Scope of Negation in Discourse, towards a flexible dynamic Montague grammar
LP-90-10 Theo M.V. Janssen	Models for Discourse Markers

Mathematical Logic and Foundations

ML-90-01 Harold Schellinx	Isomorphisms and Non-Isomorphisms of Graph Models
ML-90-02 Jaap van Oosten	A Semantical Proof of De Jongh's Theorem
ML-90-03 Yde Venema	Relational Games
ML-90-04 Maarten de Rijke	Unary Interpretability Logic
ML-90-05 Domenico Zambella	Sequences with Simple Initial Segments

Computation and Complexity Theory

CT-90-01 John Tromp, Peter van Emde Boas	Associative Storage Modification Machines
CT-90-02 Sieger van Denneheuvel, Gerard R. Renardel de Lavalette	A Normal Form for PCSJ Expressions
CT-90-03 Ricard Gavaldà, Leen Torenvliet, Osamu Watanabe, José L. Balcázar	Generalized Kolmogorov Complexity in Relativized Separations
CT-90-04 Harry Buhrman, Leen Torenvliet	Bounded Reductions

Other Prepublications

X-90-01 A.S. Troelstra	Remarks on Intuitionism and the Philosophy of Mathematics, Revised Version
X-90-02 Maarten de Rijke	Some Chapters on Interpretability Logic
X-90-03 L.D. Beklemishev	On the Complexity of Arithmetical Interpretations of Modal Formulae
X-90-04	Annual Report 1989
X-90-05 Valentin Shehtman	Derived Sets in Euclidean Spaces and Modal Logic
X-90-06 Valentin Goranko, Solomon Passy	Using the Universal Modality: Gains and Questions
X-90-07 V.Yu. Shavrukov	The Lindenbaum Fixed Point Algebra is Undecidable
X-90-08 L.D. Beklemishev	Provability Logics for Natural Turing Progressions of Arithmetical Theories
X-90-09 V.Yu. Shavrukov	On Rosser's Provability Predicate
X-90-10 Sieger van Denneheuvel, Peter van Emde Boas	An Overview of the Rule Language RL/1
X-90-11 Alessandra Carbone	Provable Fixed points in $I\Delta_0+\Omega_1$, revised version