The full non-renameability result; a lost tale

Dedicated to Cor Baayen at the occasion of his retirement from the CWI

Peter van Emde Boas

ILLC, FWI, UvA; Plantage Muidergracht 24, 1018 TV Amsterdam; peter@fwi.uva.nl

supported in part by HCM program COLORET (Complexity, logic and recursion theory, nr. CHRX-CT93-0415 (DG 12 COMA))

The *naming theorem*, one of the classical results in Abstract Complexity Theory states that the entire hierarchy of complexity classes under an arbitrary complexity measure can be renamed using an effective measured transformation by a honest collection of names preserving the extension of the classes. The *non-renameability result* which was proven by the author two decades ago states the opposite to be the case for the hierarchy of honesty classes: every attempted measured transformation must destroy the extension of at least one honesty class. However, the published version of the theorem uses the fact that in the theory partial functions are first class citizens; a version involving total functions only is proven under restrictions on the names of the classes. In this note we present the full version of the theorem; this result was obtained and announced twenty years ago but has remained unpublished since.

1 Introduction

Abstract Complexity Theory is a research subject which connects Recursion theory and Theoretical Computer Science. It finds its origin in the seminal paper by Manuel Blum [2], was intensively studied during the early seventies, but it has become obsolete and forgotten by 1980. The subject can be found in several textbooks from that period, but an almost complete survey will also be included in the second volume of Odifreddis textbook on recursion theory [6].

At the time I was completing my thesis on this subject [8] under supervision of A. van Wijngaarden and Cor Baayen with Juris Hartmanis serving as referee, interest in complexity theory already had shifted to the study of the fundamental complexity classes based on standard computational devices and to the study of the fundamental questions about the power of nondeterminism and the relation between time and space which are unsolved until today. Actually I know only of two ph.d. projects which have been completed on Abstract Complexity Theory since 1974 [1, 3]. So by the end of the seventies the subject had quite well become stable. It was my intent to transform my thesis into a two-volume textbook on Abstract Complexity Theory which has in fact been announced for many years and which has been consuming an open slot in the Mathematical Centre Tract series until this series was replaced by the CWI tracts series. This book however was never completed.

Some central results from my thesis have been published in two papers [9, 10]. Other results from the research only have appeared in the thesis itself or were never published at all. The oldest of these two papers presents the so-called *non-renameability result*. This theorem establishes the most visible distinction between the structure of the hierarchy of complexity classes and the hierarchy of honesty classes: whereas according to the *naming theorem* of Mc Creight and Meyer [5] the entire collection of complexity classes can be renamed by means of a measured transformation preserving the extension of all classes, it is shown that every attempt to rename honesty classes in a similar way must destroy the extension of at least one class.

The result in the present note is an improvement of the results in [9]; it was obtained during my residence at Cornell after my ph.d. defense. When the galley proofs of this paper where send to the printer I inserted a *note added in proof* announcing the full version of theorem 7 in that paper. Since the commercial edition of the thesis (where the promised improvement was stated to be appearing) was never completed the result only exists as a manuscript. Evidently after these many years the result might legally have been claimed by an independent researcher but this has not happened either. I am therefore grateful to be offered the opportunity to use the invitation to contribute to this volume dedicated to Cor Baayen to retrieve it from my archives in order to preserve it for prosperity.

2 Preliminaries

By a function in this paper we mean a *partial recursive function* from the set of integers $\mathbf{N}$ into itself. Functions which are defined for all arguments are called *total*. The symbol $\mathcal{P}(\mathcal{R})$ denotes the set of all partial (total) functions. The set of arguments x for which $f(x)$ is defined is denoted $\mathcal{D}f$. We write $f(x) \leq \infty (f(x) = \infty)$ for $x \in \mathcal{D}f (x \notin \mathcal{D}f)$.

The inequality $f \leq g$ means that $\mathcal{D}f \subseteq \mathcal{D}g$ and $g(x) \geq f(x)$ for $x \in \mathcal{D}g$. Strict inequality $f < g$ means that $\mathcal{D}f \subseteq \mathcal{D}g$ and $g(x) > f(x)$ for $x \in \mathcal{D}g$. If $\mathcal{D}f \subseteq \mathcal{D}g$ and $g(x) = f(x)$ for $x \in \mathcal{D}g$ the we write $g \subseteq f$. The range of f is denoted $\Re f$.

For finite k the inequalities $k \leq \infty$ and $\infty \leq \infty$ are taken to be true whereas $\infty \leq k$ is false. Beside inequalities on all arguments we also have inequalities holding *almost everywhere*. If $P(x)$ is some predicate we write $\overset{\infty}{\forall}_x \; [P(x)]$ for "$P(x)$ holds for all x except finitely many" and $\overset{\infty}{\exists}_x \; [P(x)]$ for "there exist infinitely many x such that $P(x)$". Using these notations we let $f(x) \preceq g(x)$ denote $\overset{\infty}{\forall}_x \; [f(x) \leq g(x)]$. This later notation can be relativized moreover to a subset $A \subseteq \mathbf{N}$: we let $f(x) \preceq g(x)(A)$ denote$\overset{\infty}{\forall}_x$ $[x \in A \to f(x) \leq g(x)]$. We let $\mu z[P(z)]$ denote "the least z such that $P(z)$".

We use a fixed recursive *pairing function* $< x, y >$ with coordinate projection functions π_1 and π_2. We have $\pi_1(< x, y >) = x, \pi_2(< x, y >) = y$ and $< \pi_1 z, \pi_2 z >= z$. Moreover $< x, y >$ is increasing in both arguments and consequently $< 0, 0 >= 0$. We let ϵ (**zero**) denote the function which is everywhere undefined (zero). According to our convention $\epsilon^2(\mathbf{zero}^2)$ denote the two argument function which is everywhere undefined (zero). Using this pairing function we can interpret one-variable functions as being many-variable functions; an occasional superindex like for example in $\varphi_i^2(x, y) = \varphi_i(< x, y >)$ indicates the use of this interpretation.

By $(\varphi_i)_i$ we denote a fixed *Gödel numbering* of partial recursive functions [7]. The *universal* function $u(i, x) = \varphi_i(x)$ is recursive and there exists a total function s, called the *s-n-m function* satisfying $\varphi_i^2(x, y) = \varphi_{s(i,x)}(y)$. Using the interpretation $\varphi_i^2(x, y) = \varphi_i(< x, y >)$ many variable functions are included in our enumeration. The functions φ_i are also called *programs*.

We extend the enumeration $(\varphi_i)_i$ to a *Complexity measure* by means of a sequence of *step counting functions* $(\Phi_i)_i$; this sequence satisfies the two *Blum axioms*: for each i, $\mathcal{D}\varphi_i = \mathcal{D}\Phi_i$ and the relation $\Phi_i(x) = y$ is decidable. Again we write $\Phi_i^2(x, y)$ for $\Phi_i(< x, y >)$.

A *transformation of programs* σ is a total recursive function operating on the indices of programs. In general these transformations are defined intensionally by implicit invocation of the s-n-m axiom and the universal machine axiom (possibly in combination with the recursion theorem) by writing a formula like

$$\varphi_{s(i)}(x) \Leftarrow P(i, x)$$

where $P(i, x)$ denotes some expression recursive in i and x.

A *measured set* is a sequence of functions $(\gamma_i)_i$ such that the predicate $\gamma_i(x) = y$ is decidable. The sequence of run-times $(\Phi_i)_i$ is an example. A transformation τ such that $(\varphi_{\tau(i)})_i$ is measured is called a *measured transformation of programs*.

For a (partial) function t we define:

$$F_t = \{\varphi_i | \overset{\infty}{\forall}_x \; [x \in \mathcal{D}t \Leftarrow \Phi_i(x) \leq t(x)]\}$$

$$C_t = \{f | \exists_i [f = \varphi_i \wedge \varphi_i \in F_t]\}$$

The *complexity class of functions* C_t contains all functions computed by programs in the *complexity class of programs* F_t. Note that in our definition both F_t and C_t contain partial functions even if the *name* of the class t is total.

In this definition complexity is measured in terms of the running time of the machines only. If we take into consideration that larger function values may require longer running times for being evaluated we arrive at the concept of *honesty*. Honesty classes have two-argument functions as names:

$$G_R = \{\varphi_i | \overset{\infty}{\forall}_x [\varphi_i(x) \leq \infty \wedge < x, \varphi_i(x) > \in \mathcal{D}R \Leftarrow \Phi_i(x) \leq R(x, \varphi_i(x))]\}$$

$$H_R = \{f | \exists_i [f = \varphi_i \wedge \varphi_i \in G_R]\}$$

$G_R(H_R)$ is called a *Honesty class of programs (functions)*. Note that the condition enforced in the definition of G_R holds vacuously if $\varphi_i(x)$ diverges; consequently each honesty class contains all functions with a finite domain, whereas it can be shown that no complexity class except for the trivial class $C_\epsilon = \mathcal{P}$ contains any such function.

Special honesty classes with single variable names are obtained by considering honesty bounds R of the form $R(x, y) = t(x)$; the resulting classes are called *weak complexity classes*: $F_t^W = G_R$ and $C_t^W = H_R$. Note that C_t^W and C_t contain the same total functions. An alternative special type of honesty classes with single variable names is obtained by taking names R of the form $R(x, y) = t(\max(x, y))$; these classes are called *modified honesty classes* in [9].

There exists a close connection between the notions of a measured set and a honesty class. By a well known theorem Mc Creight [4] every measured set is included in some honesty class with a total name; conversely every honesty class with a total name can be enumerated in such a way that the enumerating sequence represents a measured set. More formally:

THEOREM 1 (MC CREIGHT & MEYER) *If $(\gamma_i)_i$ is a measured set the there exists a total function R, such that as a set of functions $(\gamma_i)_i \subseteq H_R$; moreover an index for R can be obtained uniformly from an index for the decision predicate for $\gamma_i(x) = y$. Conversely, if R is a total function then H_R is enumerated by some measured set $(\gamma_i)_i$ and indices for both the enumerating sequence and the decision predicate for $\gamma_i(x) = y$ are obtained uniformly in the index of R.*

The above theorem has led to the feeling that the two concepts are more or less equivalent. This is certainly not the whole truth. The above equivalence is lost as soon as the name of the honesty class is partial. Moreover, it is not hard to construct a presentation of a honesty class with a total name such that this presentation as a sequence is not a measured set.

We now formulate the *naming theorem* of Mc Creight and Meyer [5] and our full *non-renameability result*:

THEOREM 2 (NAMING THEOREM) *There exists a measured transformation of programs σ such that for each i the classes F_{φ_i} and $F_{\varphi_{\sigma(i)}}$ are equal (and consequently $C_{\varphi_i} = C_{\varphi_{\sigma(i)}}$ as well).*

THEOREM 3 (NON-RENAMEABILITY THEOREM) *For every measured transformation σ there exists an index i of a total function such that $H_{\varphi_i^2} \cap \mathcal{R} \neq H_{\varphi_{\sigma(i)}^2} \cap \mathcal{R}$.*

This result resembles the results proven in [9]; however if inspected in more details all the results published in this paper are weaker: in theorem 6 the result claimed reads $H_{\varphi_i^2} \neq H_{\varphi^2_{\sigma(i)}}$, i.e., the classes may turn out to be different due to the presence of partial functions in the classes; in theorem 7 there is shown a difference on the subclasses of total functions within the honesty classes, but the result is proven for the modified honesty classes only, i.e. the names are of a special form.

Expressions describing functions and/or transformations of programs in this paper are defined in terms of the hybrid language introduced in my thesis which combines elements from standard recursion theory and the (by now archaic) programming language ALGOL68. The resulting expressions may have in general several plausible computational interpretations which may differ with respect to convergence; the intended computational meaning is uniquely determined according to the guidelines as indicated in [8], section 1. The reader should keep in mind that according to this intended interpretation inequalities involving either a step-counting function or an element of some other measured set are evaluated using the decision predicate instead of a brute-force evaluation.

3 Proof of the non-renameability result

The proof of the improved result uses the same technique used in the earlier results: we obtain a suitable version of the *mirror lemma*, which shows that a measured transformation σ eventually will "reflect" some name $\varphi_e^2(x,y)$ with respect to some suitably large function $R(x,y)$ in the sense that $\varphi_e^2(x,y)$ is large compared to $R(x,y)$ if and only if $\varphi^2_{\sigma(e)}(x,y)$ is small; subsequently we show that the set of arguments where the reflected name is small supports the graph of a total diagonal function which is included in the honesty class with the original name but not in the transformed class. This diagonal then separates the original class from its renamed version.

We start with a function R which is sufficiently large in order that there exists an R-honest *odd-valued* function which is not $\mathbf{zero}^2$-honest. We define the transformation α by:

$$\begin{aligned}\varphi^2_{\alpha(i,j)}(x,y) \Leftarrow\ & \mathbf{if\ even}\, y\, \mathbf{then}\ \varphi_j^2(x,y) + R(x,y) + 1\\ & \mathbf{elif}\ \varphi^2_{\sigma(i)}(x,y) \leq R(x,y)\ \mathbf{then}\ \varphi_j^2(x,y) + R(x,y) + 1\\ & \mathbf{else}\ 0\ \mathbf{fi}\end{aligned}$$

By the recursion theorem there exists a transformation ρ satisfying

Claim 1 (Mirror Lemma)

$$\begin{aligned}\varphi^2_{\rho(j)}(x,y) =\ & \mathbf{if\ even}\, y\, \mathbf{then}\ \varphi_j^2(x,y) + R(x,y) + 1\\ & \mathbf{elif}\ \varphi^2_{\sigma(\rho(j))}(x,y) \leq R(x,y)\ \mathbf{then}\ \varphi_j^2(x,y) + R(x,y) + 1\\ & \mathbf{else}\ 0\ \mathbf{fi}\end{aligned}$$

We next define (implicitly using the recursion theorem once again) the transformation κ:

$$\varphi_{\kappa(j)}(n) = \textbf{if } n = 0 \textbf{ then } \mu m[\varphi^2_{\sigma(\rho(j))}(\pi_1 m, \pi_2 m) \leq R(\pi_1 m, \pi_2 m) \textbf{ and odd } \pi_2 m] \textbf{ else } \mu m[\textbf{ odd } \pi_2 m \textbf{ and } \pi_1 m > \pi_1 \varphi_{\kappa(j)}(n-1) \textbf{ and } \varphi^2_{\sigma(\rho(j))}(\pi_1 m, \pi_2 m) \leq R(\pi_1 m, \pi_2 m)] \textbf{ fi}$$

Hence $\varphi_{\kappa(j)}$ enumerates pairs $< x, y >$ with x increasing and y odd such that $\varphi_{\sigma(\rho(j))}(\leq)R(x,y)$. A "partial inverse" of $\varphi_{\kappa(j)}$ is obtained by the transformation:

$$\varphi_{\beta(j)}(x) \Leftarrow \textbf{if } \pi_1 \varphi_{\kappa(j)}(\mu n[\pi_1 \varphi_{\kappa(j)}(n) \geq x]) = x \textbf{ then } \mu n[\pi_1 \varphi_{\kappa(j)}(n)] \textbf{ else false fi}$$

The function $\varphi_{\beta(j)}$ computes in fact a partial inverse to $\pi_1 \varphi_{\kappa(j)}$. If for some input x some pair $< x, y >$ is enumerated then $\varphi_{\beta(j)}(x)$ yields the index of this pair in this enumeration; if no such pair is enumerated but if eventually some pair $< x', y' >$ is enumerated with $x' > x$ the the value is **false** . Otherwise $\varphi_{\beta(j)}$ is undefined.

Finally we define a diagonal transformation $\varphi_{\delta(j)}$ by:

$$\varphi_{\delta(j)}(x) \Leftarrow \textbf{if } \varphi_{\beta(j)}(x) = \textbf{ false then } 2\Phi_{\beta(j)}(x) \textbf{ elif } \Phi_{\pi_1 \varphi_{\beta(j)}(x)}(x) \leq R(x, \pi_2 \varphi_{\kappa(j)}(\varphi_{\beta(j)}(x))) \textbf{ and } \varphi_{\pi_1 \varphi_{\beta(j)}(x)}(x) = \pi_2 \varphi_{\kappa(j)}(\varphi_{\beta(j)}(x)) \textbf{ then } \pi_2 \varphi_{\kappa(j)}(\varphi_{\beta(j)}(x)) - 1 \textbf{ else } \pi_2 \varphi_{\kappa(j)}(\varphi_{\beta(j)}(x)) \textbf{ fi}$$

Informally, in order to evaluate $\varphi_{\delta(j)}(x)$ one first must evaluate $\varphi_{\beta(j)}(x)$. If this computation diverges then $\varphi_{\delta(j)}(x)$ is undefined. If the computation converges but yields the value **false** then output twice the time it has taken to compute this value **false** . Otherwise we diagonalize: we know that for some value y a pair $< x, y >$ is enumerated by $\varphi_{\kappa(j)}$, say $\varphi_{\kappa(j)}(m) =< x, y >$. Test whether $\Phi_{\pi_1 m}(x) \leq R(x, y)$ and if so whether $\varphi_{\pi_1 m}(x) = y$; if both conditions are satisfied then output $y - 1$ and output y otherwise.

Note that this computation diverges when $\varphi_{\beta(j)}(x)$ diverges, and this will only happen if no pair $< x', y' >$ with $x' > x$ is enumerated by $\varphi_{\kappa(j)}$, i.e., when $\varphi_{\kappa(j)}$ is partial. Hence in case $\varphi_{\kappa(j)}$ is total then so is $\varphi_{\delta(j)}$

CLAIM 2 *The sequence $(\varphi_{\delta(j)})_j$ is a measured set.*

This can be seen as follows.

For a given pair $< x, y >$ it can be decided whether $< x, y > \in \Re \varphi_{\kappa(j)}$: if y is even or if $\varphi^2_{\sigma(\rho(j))}(x, y) > R(x, y)$ then $< x, y >$ is no candidate for being enumerated so we can answer "no". Otherwise we know that some pair $< x', y' >$ with $x' \geq x$ will eventually be enumerated and we can wait and see whether $< x, y >$ is enumerated by that time.

Using this observation we can describe the following decision procedure for $\varphi_{\delta(j)}(x) = y$?

If y is even then test whether $\Phi_{\beta(j)}(x) = y/2$ and $\varphi_{\beta(j)}(x) =$ **false**; if so the answer is "yes". Otherwise test whether $< x, y+1 > \in \Re\varphi_{\kappa(j)}$; if not then the answer is "no". If $< x, y+1 >= \varphi_{\kappa(j)}(m)$ test whether $\Phi_{\pi_1 m}(x) \leq R(x, y+1)$ and $\varphi_{\pi_1 m}(x) = y+1$; if so the answer is "yes" and otherwise the answer is "no".

If y is odd then test directly whether $< x, y > \in \Re\varphi_{\kappa(j)}$. If not the answer is "no". Otherwise let m be the argument such that $< x, y >= \varphi_{\kappa(j)}(m)$, and test whether $\Phi_{\pi_1 m}(x) \leq R(x, y)$ and $\varphi_{\pi_1 m}(x) = y$; if so the answer is "no" and otherwise the answer is "yes".

The correctness proof for this decision procedure is left to the reader.

Our next claim holds only in the case that $\varphi_{\kappa(j)}$ is a total functions, i.e., $\overset{\infty}{\exists} x \exists y [\varphi^2_{\sigma(\rho(j))}(x, y) \leq R(x, y)]$:

Claim 3 *If $\varphi_{\kappa(j)}$ is total then $\varphi_{\delta(j)} \notin H_{\varphi^2_{\sigma(\rho(j))}}$.*

Consider an index i for $\varphi_{\delta(j)}$ and a value m with $\pi_1 m = i$. Let $\varphi_{\kappa(j)}(m) =< x, y >$ then we have for this particular argument x:

$$\begin{aligned}\varphi_i(x) = \varphi_{\delta(j)}(x) = \; & \textbf{if } m = \textbf{ false then } 2\Phi_{\beta(j)}(x) \\ & \textbf{elif } \Phi_i(x) \leq R(x,y) \textbf{ and } \varphi_i(x) = y \textbf{ then } y - 1 \\ & \textbf{else } y \textbf{ fi}\end{aligned}$$

The first condition is evidently false; since the then-part for the second condition is contradictory we conclude that $\varphi_i(x) = y$ and $\Phi_i(x) > R(x, y)$; since also for this pair $< x, y >$ it holds that $\varphi^2_{\sigma(\rho(j))}(x, y) \leq R(x, y)$ this shows that φ_i violates the honesty condition at $< x, y >$. From the fact that $\varphi_{\kappa(j)}$ is total we infer that there exist infinitely many violations of this type; since also i was an arbitrary index for $\varphi_{\delta(j)}$ this proves our claim.

Claim 4 *For every pair $< x, y >$ such that $\varphi_{\delta(j)}(x) = y$ one has $\varphi^2_{\rho(j)}(x, y) \geq \varphi^2_j(x, y)$.*

For even y this claim is a direct consequence of the definition of ρ, whereas for odd y the definition of δ implies that $< x, y >$ is a pair enumerated by $\varphi_{\kappa(j)}$ and therefore the condition $\varphi^2_{\sigma(\rho(j))}(x, y) \leq R(x, y)$ is satisfied. However, according to our use of the mirror lemma this means that $\varphi^2_{\rho(j)}(x, y) = R(x, y) + \varphi^2_j(x, y) + 1 \geq \varphi^2_j(x, y)$.

The theorem now can be derived using the above claims.

Since $(\varphi_{\delta(j)})_j$ is a measured set there exists an index j_0 of some total function $\varphi^2_{j_0}$ such that $(\varphi_{\delta(j)})_j \subseteq H_{\varphi^2_{j_0}}$.

If for this index j_0 the function $\varphi^2_{\kappa(j_0)}$ is total then $\varphi^2_{\delta(j_0)}$ is a total function in $H_{\varphi^2_{\rho(j_0)}} \setminus H_{\varphi^2_{\sigma(\rho(j_0))}}$.

In the alternative case that $\varphi^2_{\delta(j_0)}$ is a partial function then for almost all x it holds that $\varphi^2_{\rho(j_0)}(x,y) = 0$ for all odd values of y. So the odd-valued functions in $H_{\varphi^2_{\rho(j_0)}}$ are **zero**^{2}-honest functions. By the mirror lemma it follows that for almost all x one has $\varphi^2_{\sigma(\rho(j_0))}(x,y) \geq R(x,y)$ for all odd values of y. Since there exists by assumption an odd-valued R-honest function f which is not **zero**^{2}-honest, one concludes that $f \in H_{\varphi^2_{\sigma(\rho(j_0))}} \setminus H_{\varphi^2_{\rho(j_0)}}$.

Having shown that in both cases the honesty classes are different, the proof is complete.

4 Looking backwards

With hindsight one may ask why this result is not included in the earlier presentations of the non-renameability theorem. There is just one additional technique involved in the proof which was not present in the proofs in [9]: the use of parity. The problem in the earlier proofs is how to obtain "escape values" for the diagonalization, in such a way that the choice for this escape value won't lead to a violation against the original honesty bound. The earliest proof of the non-renameability uses the undefined escape value, since this choice will never violate any honesty condition. The consequence is that the diagonal function becomes partial.

The question whether the non-renameability result extends to the case that only the total functions in a honesty class are considered originates with Albert Meyer. Evidently, considering the simple case of the weak complexity classes (which are non-renameable if partial functions are considered; see theorem 4 in [9]), will yield no answer since the weak and the strong complexity classes contain the same total functions, and the strong classes can be renamed. Thus the need for finite escape values arose.

In order that the choice of the escape value y does not lead to a violation of a honesty condition at argument x, the pair $< x, y >$ should be located at a place where the original bound S is large. If the transformed bound S' is obtained using the mirror lemma, then these places can be detected by deciding whether $S'(x,y)$ is small; however, since existence of such a value y is in general undecidable finding one may be too hard. Only because of the special structure of the names for the modified complexity classes this hurdle could be overcome.

Using the parity of the y value as a dividing condition our new proof in fact constructs bounds S and S' where the mirror effect only is enforced on half of the plane (i.e., for odd values of y only). The diagonal tries to produce violations against the honesty bound $S'(x,y)$ for odd values of y for which $S(x,y)$ is large and $S'(x,y)$ is consequently small. The escape value is chosen to be even. By a standard combining lemma argument the complexity of this diagonalization can be estimated, and it suffices to choose $S(x,y)$ being sufficiently large for even y and pairs $< x, y >$ where S should be large.

Ultimately there are two cases; either the diagonalization succeeds and a member of $H_S \setminus H_{S'}$ is obtained or S' becomes so small that some odd-valued member of H_S gets excluded. Evidently this idea does not reach far beyond the original techniques, so the result could have been obtained already in 1973 with the others.

A more interesting question is whether the whole field of Abstract Complexity Theory should be looked at at all at this stage in the development of theoretical computer science. The subject disappeared from the battlefields of theoretical computer science since the axioms of the theory failed to put any constraint of naturalness on the models; all sort of pathologies were possible, and any attempt to further constrain the theory by enforcing naturalness conditions was doomed to failure [3]. Also the theory failed to provide any insight in the core problems of the field: the relation between time and space and the power of nondeterminism.

I claim however that some sort of a positive revival today is possible; the gap between recursion theory and complexity theory is being narrowed these years, both because of the nowadays frequent use of recursion theoretical techniques in structural complexity theory, but also since researchers in recursion theory once more become interested in complexity issues. So there still might be a market for the lost textbook on Abstract Complexity Theory.

REFERENCES

1. Bennison, V.L., *On the computational complexity of recursively enumerable sets*, ph.d. thesis, Univ. of Chicago, 1976.
2. Blum, M.,*A machine-independent theory of the complexity of recursive functions*, J. Assoc. Comput. Mach. 14 (1967) 322–336.
3. Lischke, G., *Erhaltungssätze in der Theorie der Blumschen Kompiziertheitsmaße*, ph.d. thesis, Fr. Schiller Univ. Jena, 1976.
4. Mc Creight, E.M., *Classes of computable functions defined by bounds on computation*, ph.d. thesis, Carnegy Mellon Univ., 1969.
5. Mc Creight, E.M. & Meyer, A., *Classes of computable functions defined by bounds on computation*, Proc. SIGACT STOC 1, 1969, 79–88.
6. Odifreddi, P., *Classical Recursion Theory, part 1*, North-Holland, Studies in Logic and the Foundations of Mathematics, vol. 125, 1989; part 2, to appear.
7. Rogers, H., jr.,*Gödel numbering of partial recursive functions*, J.S.L. 23 (1958) 331–341.
8. van Emde Boas, P., *Abstract Resource Bound Classes*, ph.d. thesis, Univ. of Amsterdam, 1974.
9. van Emde Boas, P., *The non-renameability of Honesty Classes*, Computing 14 (1975) 183–193.
10. van Emde Boas, P., *Some applications of the Mc Creight-Meyer algorithms in Abstract Complexity Theory*, Theor. Computer Science 7 (1978) 79–98.