Friedberg numberings in the Ershov hierarchy

Serikzhan A. Badaev*, Mustafa Manat† and Andrea Sorbi‡

Abstract

We show that for every $n \geq 1$, there exists a Σ_n^{-1}-computable family which up to equivalence has exactly one Friedberg numbering which does not induce the least element of the corresponding Rogers semilattice.

1 Introduction

Minimal numberings became a fashionable research topic in the classical theory of numberings at the end of the sixties.The study of minimal numbering beginning from the famous theorem of Friedberg [10] on the existence of one-to-one computable numbering of the family $\mathcal{C}$ of all c.e. sets. One of the main questions on minimal numberings, that is the problem of finding, up to equivalence of numberings, the possible number of minimal numberings, was settled by Yu.L. Ershov[3]. A Friedberg numbering is a special but very important case of minimal numbering. The theory of minimal numberings, and in particular Friedberg numberings, has many successful applications in classical recursion theory, recursive model theory ([9], [14]), and theoretical computer science ([15]).The main powerful methods for constructing families of c.e. sets with a finite number of Friedberg numberings, due to Goncharov [9],to show that numbers of spectrum of the nonautoequivalent constructivizations of recursive models is equal to $\{\omega, 0, 1, 2 \ldots\}$ ([12]). It was the starting point of some of the most important researches on algorithmic dimension of recursive models. Another application of this results was found by Kummer ([15]).

We refer to Kleene's system O of ordinal notations for computable ordinals: for details, see [18]. In particular, for $a \in O$, the symbol $|a|_O$ represents the

*Al-Farabi Kazakh National University, Al-Farabi ave., 71, Almaty, 050038, Kazakhstan. Part of the research contained in this paper was carried out while the first author was GNSAGA-INDAM Visiting Professor at the Department of Mathematics and Computer Science "Roberto Magari" of the University of Siena, Italy, July 2011. The first author wishes to thank INDAM-GNSAGA for supporting the visiting professorship.

†Al-Farabi Kazakh National University, Al-Farabi ave., 71, Almaty, 050038, Kazakhstan. Part of the research contained in this paper was carried out while the second author was visiting the Department of Mathematics and Computer Science "Roberto Magari" of the University of Siena, Italy. The second author wishes to thank the Al-Farabi University for supporting the visit,and the Department of Mathematics and Computer Science "Roberto Magari" of siena for its hospitality.

‡Dipartimento di Scienze Matematiche ed Informatiche "Roberto Magari", Università di Siena, 53100 Siena, Italy.

ordinal of which a is a notation; the symbol $<_O$ denotes Kleene's partial ordering relation on O; moreover, the symbol $+_0$ denotes a partial computable function, defined on O, such that $|a +_O b|_O = |a|_O + |b|_O$, and $a \leq_O a +_O b$.

Definition 1.1. Any surjective mapping α of the set ω of natural numbers onto a nonempty set A is called a *numbering* of A. Let α and β be numberings of A. We say that numbering α is *reducible* to numbering β (in symbols, $\alpha \leqslant \beta$) if there exists a computable function f such that $\alpha(n) = \beta f(n)$ for any $n \in \omega$. We say that the numberings α and β are *equivalent* (in symbols, $\alpha \equiv \beta$) if $\alpha \leqslant \beta$ and $\beta \leqslant \alpha$.

Definition 1.2. Let $\theta_\alpha \rightleftharpoons \{< x, y >| \ \alpha x = \alpha y\}$. Numbering α is called *decidable (positive)* if θ_α is decidable (correspondingly, c.e.) set.Numbering α is called *Fridberg* if it is one to one.

It is obvious that if α and β are equivalent numberings then α is decidable (positive) if and only if β is decidable (positive). Every decidable numbering of infinite family is equivalent to one-to-one numbering or single-valued numbering [7], [8].

Rogers semilattice $\mathcal{R}^i_a(\mathcal{A})$ of a family $\mathcal{A} \subseteq \Sigma^i_a$ is a quotient structure of all Σ^i_a-computable numberings of the family $\mathcal{A}$ modulo equivalence of the numberings ordered by the relation induced by reducibility of the numberings. $\mathcal{R}^i_a(\mathcal{A})$ allows one to measure the different computations of a given family $\mathcal{A}$ and used also as a tool to classify the properties of Σ^i_a-computable numberings for the different families $\mathcal{A}$.

A numbering α of a set A is said to be *minimal* if $\beta \leq \alpha$ implies $\alpha \leq \beta$ for every numbering β of A. The minimal numberings are just those ones which induce the minimal elements in $\mathcal{R}^i_a(\mathcal{A})$.

We now briefly review the basic notions concerning Ershov hiearchy. There is several equivalent definitions of the Ershov hierarchy, introduced in [4, 5, 6]. But our presentation is based on [17].

Definition 1.3. If a is a notation for a computable ordinal, then a set of numbers A is said to be Σ^{-1}_a if there are a computable function $f(z,t)$ and a partial computable function $\gamma(z,t)$ such that, for all z,

1. $A(z) = \lim_t f(z,t)$, with $f(z,0) = 0$; (here, given a set X, and a number z, the symbol $X(z)$ denotes the value of the characteristic function of X on z);
2. (a) $\gamma(z,t)\downarrow \Rightarrow \gamma(z,t+1)\downarrow \& \gamma(z,t+1) \leq_O \gamma(z,t) <_O a$;
   (b) $f(z,t+1) \neq f(z,t) \Rightarrow \gamma(z,t+1)\downarrow \neq \gamma(z,t)$.

We call the partial function γ the *mind–change function for A, relatively to f*.

A Σ^{-1}_a*–approximation* to a Σ^{-1}_a–set A, is a pair $\langle f, \gamma\rangle$, where f and γ are respectively a computable function and a partial computable function satisfying 1. and 2., above, for A.

Following [13], we give the following:

Definition 1.4. A Σ_a^{-1}–*computable numbering* of a family $\mathcal{A}$ of Σ_a^{-1}–sets is an onto function $\pi : \omega \longrightarrow \mathcal{A}$, such that

$$\{\langle y, x\rangle : x \in \pi(y)\} \in \Sigma_a^{-1}.$$

Hence there exist a computable function $f(z,t)$ and a partial computable function $\gamma(z,t)$, such that $\pi(y)(x) = \lim_t f(\langle y, x\rangle, t)$, with $f(z,0) = 0$ for all z; and γ is the mind-change function for $\{\langle y, x\rangle : x \in \pi(y)\}$ relatively to f.

Note that $\{\langle x, m\rangle : x \in \alpha(m)\} \in \Sigma_a^{-1}$ if and only if $\{\langle x, m\rangle : x \in \alpha(m)\}$ is $a-$ computably enumerable in a sense of Putnam [16].
We recall (see e.g. [5]) that there is an indexing $\{A_z\}_{z\in\omega}$ of the family of all Σ_a^{-1} sets, such that $\{\langle x, z\rangle : x \in A_z\} \in \Sigma_a^{-1}$. From this, it is possible (for more details, see [17]) to define an indexing $\{\pi_k\}_{k\in\omega}$ of all computable numberings of families of Σ_a^{-1} sets, for which

$$\{\langle k, y, x\rangle : x \in \pi_k(y)\} \in \Sigma_a^{-1},$$

i.e. the set $\{\langle k, y, x\rangle : x \in \pi_k(y)\}$ has a Σ_a^{-1}–approximation $\langle f, \gamma\rangle$: an indexing satisfying this property is called a Σ_a^{-1}–*computable indexing* of all Σ_a^{-1}–computable numberings. Clearly, from k, y one has an effective way of getting a Σ_a^{-1}–approximation $\langle f_{\pi_k(y)}, \gamma_{\pi_k(y)}\rangle$ to the set $\pi_k(y)$.

2 The theorem

In [11], S.S. Goncharov showed that there exist classes of recursively enumerable sets admitting up to equivalence exactly one Friedberg numbering which does not induce the least element in the corresponding Rogers semilattice. Later, a simple example of a such a class was found by M. Kummer: This example appears in the paper of S.A. Badaev and S.S. Goncharov ([1]). We generalize this result to all successor ordinal levels of the Ershov hierarchy.

Theorem 2.1. *for every ordinal notation a of a nonzero successor ordinal, there exists a Σ_a^{-1}-computable family $\mathcal{A}$ whose Rogers semilattice has exactly one Friedberg degree which is not the least element of the semilattice.*

Proof. Given a, with $|a|_O \geq 1$ and $|a|_O$ successor. We will construct a Σ_a^{-1}-computable Friedberg numbering α and a Σ_a^{-1}-computable numbering β such that:

1. $\alpha(\omega) = \beta(\omega)$.
2. If π is a computable Friedberg numbering of $\mathcal{A} = \alpha(\omega)$ then $\pi \leqslant \alpha$.
3. $\alpha \nleqslant \beta$.

Requirements. We will define numberings α and β so that, for every e, k, the following requirements are satisfied:

$$\begin{aligned}
\mathcal{F}: &\quad \alpha \text{ is a Friedberg numbering,}\\
\mathcal{B}: &\quad \alpha(\omega) = \beta(\omega) = \mathcal{A},\\
\mathcal{C}: &\quad \alpha \text{ and } \beta \text{ are } \Sigma_a^{-1}\text{-computable,}\\
\mathcal{P}_k: &\quad \text{if } \pi_k \text{ is a Friedberg numbering of } \mathcal{A} \text{ then } \pi_k = \alpha \circ g_k,\\
\mathcal{D}_e: &\quad \text{if } \varphi_e \text{ is total then } \alpha \neq \beta \circ \varphi_e,
\end{aligned}$$

where g_k is a computable function built by us.

Strategy for $\mathcal{F}$. We fix three one-to-one computable functions

$$a(k, m), b(k, m), c(k, m),$$

with pairwise disjoint ranges and construct the numbering α so that for every m there exist at least one k_m with the following properties, where for simplicity we let $a = a(k_m, m)$, $b = b(k_m, m)$, $c = c(k_m, m)$:

$$\alpha(2m) \cap \{a, b, c\} = \{a, b, c\},$$
$$\alpha(2m + 1) \cap \{a, b, c\} = \{b, c\}$$

and, for every $m' \neq m$,

$$\alpha(2m') \cap \{a, b, c\} = \alpha(2m' + 1) \cap \{a, b, c\} = \emptyset.$$

Strategy for $\mathcal{B}$. For every m, we force β to satisfy the following equalities: either, for all s,

$$\beta^s(3m) = \alpha^s(2m)$$
$$\beta^s(3m + 1) = \beta^s(3m + 2) = \alpha^s(2m + 1);$$

or there exists s_0 such that for all $s \leq s_0$

$$\beta^s(3m) = \alpha^s(2m)$$
$$\beta^s(3m + 1) = \beta^s(3m + 2) = \alpha^s(2m + 1)$$

and for all $s > s_0$

$$\beta^s(3m) = \alpha^s(2m)$$
$$\beta^s(3m + 1) = \alpha^s(m')$$
$$\beta^s(3m + 2) = \alpha^s(m'').$$

Here $\{m', m''\} = \{2m, 2m + 1\}$.

Strategy for $\mathcal{C}$. To ensure Σ_a^{-1}-computability of the numberings α and β we do not explicitly construct suitable corresponding changing functions, but in all the strategies and the construction we implicitly ensure the correct behavior of the approximations to α and β.

Strategy for $\mathcal{P}_k$ in isolation. This strategy aims, for every m, at finding by some uniform procedure, two π_k-indices $x \neq y$ such that $\alpha(2m) = \pi_k(x)$ and $\alpha(2m+1) = \pi_k(y)$. In the case when π_k is a Friedberg numbering of $\mathcal{A}$, this will give us a reduction of π_k to α.

Initially, for all $k \in \omega$, we put $a(k,m), b(k,m), c(k,m)$ into $\alpha(2m)$, and we put $b(k,m), c(k,m)$ into $\alpha(2m+1)$. Note that, for every m, k, we never remove the numbers $b(k,m), c(k,m)$ from the sets $\alpha(2m)$ and $\alpha(2m+1)$.

Due to injectiveness of the functions $a(k,m), b(k,m), c(k,m)$, we can split the strategy for $\mathcal{P}_k$ into independent substrategies $\mathcal{P}_{k,m}$, with $m \in \omega$. Henceforth, we write "Substrategy $\mathcal{P}_{k,m}$" to denote the substrategy for $\mathcal{P}_{k,m}$.

[Note that at any stage of the construction below we use a uniform approximation to the numbering π_k in which at most one change might happen in $\pi_k(x)$ at any stage.]

Substrategy for $\mathcal{P}_{k,m}$ in isolation.

1. Search for a π_k-index x such that

   $$\pi_k(x) \cap \{a(k,m), b(k,m), c(k,m)\} = \{a(k,m), b(k,m), c(k,m)\}.$$

   From now on let a, b, c stand for the numbers $a(k,m), b(k,m), c(k,m)$, respectively.

   Furthermore, we check whether b is in $\pi_k(x)$ every time we start Substrategy $\mathcal{P}_{k,m}$.

   If b is in $\pi_k(x)$ then go to item 3, otherwise go to item 2. In the latter case wait until b comes back to $\pi_k(x)$ and only after that continue Substrategy $\mathcal{P}_{k,m}$ in item 3.

2. So, what should we do when $b \notin \pi_k(x)$?

   Enumerate b into $\alpha(z)$ for all $z \notin \{2m, 2m+1\}$ and wait until b appears in $\pi_k(x)$, and only when this happens, remove b from $\alpha(z)$ for all $z \notin \{2m, 2m+1\}$.

   These movements of b from, and into, $\pi_k(x)$ eventually stop, and the corresponding synchronized changes of b for $\alpha(z)$ are compatible with having $\alpha(z)$ Σ_a^{-1}-computable. When π_k is a numbering of $\mathcal{A}$, we have that $b \in \pi_k(x) \cap \alpha(2m) \cap \alpha(2m+1)$ and $b \notin \alpha(z)$ for all $z \notin \{2m, 2m+1\}$. Thus $\pi_k(x) \in \{\alpha(2m), \alpha(2m+1)\}$. (Note that if π_k is not a numbering of $\mathcal{A}$ then the option with $b \in \alpha(z)$ for all $z \in \omega$ and $b \notin \pi_k(x)$ is possible too.)

3. [$b \in \pi_k(x)$ and either $a \in \alpha(2m)$ or $a \in \alpha(2m+1)$: The construction guarantees that at each stage, at the beginning of the current item, $a \in \alpha(2m)$ if and only if $a \notin \alpha(2m+1)$] Then $\pi_k(x) \cap \{a\}$ is equal to either $\alpha(2m) \cap \{a\}$ or $\alpha(2m+1) \cap \{a\}$. Go to item 4 if $\pi_k(x) \cap \{a\} = \alpha(2m) \cap \{a\}$, and go to item 5 otherwise.

4. [$\pi_k(x) \cap \{a\} = \alpha(2m) \cap \{a\}$] Check whether a has exhausted all possible changes in $\pi_k(x)$ (i.e. $a \in \pi_k(x)$ and a can not be extracted from $\pi_k(x)$ anymore, or $a \notin \pi_k(x)$ and a can not be put into $\pi_k(x)$ anymore). If so then define $g_k(x) = 2m$ and go to item 6, otherwise go to 4a or to 4b according to whether the question "Is $a \in \pi_k(x)$?" has positive or negative answer.

   (a) [$a \in \pi_k(x)$] Extract a from $\alpha(2m)$ and wait until a leaves $\pi_k(x)$ (if a never leaves $\pi_k(x)$ then π_k is not a numbering of $\mathcal{A}$ since a currently does not belong to any set of $\mathcal{A}$ and we prevent a from being put into any $\alpha(z)$ in the future).

   When a leaves $\pi_k(x)$, we put a into $\alpha(2m+1)$ and go to 3.

   (b) [$a \notin \pi_k(x)$] Put a into $\alpha(2m)$ (notice that already $a \in \alpha(2m+1)$), and wait until a is enumerated in $\pi_k(x)$ (if a never appears in $\pi_k(x)$ then π_k is not a numbering of $\mathcal{A}$.

   When a is enumerated in $\pi_k(x)$, we remove a from $\alpha(2m+1)$ and from $\alpha(z)$ for all $z \neq 2m$, and go to (3).

5. [$\pi_k(x) \cap \{a\} = \alpha(2m+1) \cap \{a\}$] Check whether a has exhausted all possible changes in $\pi_k(x)$. If so then define $g_k(x) = 2m+1$ and go to 6, otherwise go to 5a or to 5b according to whether the question "Is $a \in \pi_k(x)$?" has positive or negative answer.

   (a) [$a \in \pi_k(x)$] Extract a from $\alpha(2m+1)$ and wait until a leaves $\pi_k(x)$.

   When a leaves $\pi_k(x)$, we put a into $\alpha(2m)$, and go to 3.

   (b) [$a \notin \pi_k(x)$] Put a into $\alpha(2m+1)$ (notice that already $a \in \alpha(2m)$), and wait until a is enumerated in $\pi_k(x)$.

   When a is enumerated in $\pi_k(x)$, we remove a from $\alpha(2m)$ and from $\alpha(z)$ for all $z \neq 2m+1$, and go to 3.

6. Let $\tilde{m} \in \{2m, 2m+1\}$ be such that $g_k(x) \neq \tilde{m}$. Our goal now is to find a π_k-index $y \neq x$ such that $\pi_k(y) = \alpha(\tilde{m})$ if π_k is a Friedberg numbering of $\mathcal{A}$.

   Search for a π_k-index y such that $y \neq x$ and

   $$\pi_k(y) \cap \{a(k,m), b(k,m), c(k,m)\} = \alpha(\tilde{m}) \cap \{a(k,m), b(k,m), c(k,m)\}.$$

   Define $g_k(y) = \tilde{m}$.

   After this, every time $c(k,m)$ leaves $\pi_k(y)$, put $c(k,m)$ into $\alpha(z)$ for all $z \notin \{2m, 2m+1\}$, and wait until $c(k,m)$ is enumerated in $\pi_k(y)$. Whenever the number $c(k,m)$ is enumerated in $\pi_k(y)$ remove it from $\alpha(z)$ for all $z \notin \{2m, 2m+1\}$.

As a result of item 6, $\pi_k(y) \in \{\alpha(2m), \alpha(2m+1)\}$ if π_k is a numbering of $\mathcal{A}$ since in this case, for every $z \notin \{2m, 2m+1\}$,

$$c(k,m) \in \pi_k(y) \Leftrightarrow c(k,m) \notin \alpha(z).$$

Moreover, if π_k is a Friedberg numbering of $\mathcal{A}$ then $\pi_k(y) = \alpha(g_k(y))$ because $\pi_k(x) = \alpha(g_k(x))$, $x \neq y$, $g_k(x) \neq g_k(y)$ and $\{\pi_k(x), \pi_k(y)\} = \{\alpha(2m), \alpha(2m+1)\}$.

Observations on Substrategy $\mathcal{P}_{k,m}$. It is easy to see that whenever x has been determined, Substrategy $\mathcal{P}_{k,m}$ proceeds through several cycles with starting point at item 3 and possible interruptions of these cycles at item 2. Let (u,v,w) stand for the triple of current total number of changes (enumerations and extractions) of $a = a(k,m)$ relative to the sets $\alpha(2m)$, $\alpha(2m+1)$, and $\pi_k(x)$ respectively. A closer look at the evolution in time of this triple shows

Lemma 2.2. *The following hold*

(i) if π_k is a numbering of $\mathcal{A}$ then eventually $w = n$;

(ii) at the end of every cycle 3 (even when the cycle is not completed because we define g_k), $u \leq w$ and $v < w$.

Proof. Statement (i) follows from the instructions of items 4a-5b. In each of these four cases, $\pi_k(x)$ is forced to move a, otherwise π_k can not be a numbering of $\mathcal{A}$.

In proving (ii), we can ignore item 2 at all, since instructions of item 2 do not change u and v and can not force w to decrease. By analyzing items 4a-5b, it is easy to see that, if at the beginning of any cycle at 3 we have a triple (u,v,w) then a complete cycle, before returning to 3, may be described by one the following two series of actions, the former due to 4, and the latter due to 5:

$$(u,v,w) \longrightarrow (u+1,v,w) \longrightarrow (u+1,v,w+1) \longrightarrow (u+1,v+1,w+1),$$

$$(u,v,w) \longrightarrow (u,v+1,w) \longrightarrow (u,v+1,w+1) \longrightarrow (u+1,v+1,w+1).$$

So, if at the beginning of the cycle, (u,v,w) satisfies (ii), then so does at the end of the cycle. The claim then follows from the fact that initially, when we start 3 for the first time, we have $u = 1, v = 0, w \geq 1$. Notice that 3 may stop before completion of the cycle, if $w = n$ at the beginning of the cycle. □

Strategy for $\mathcal{D}_e$ in isolation. If φ_e is total then we diagonalize against the reduction $\alpha = \beta \circ \varphi_e$ at the argument $x = 2e+1$. If $\varphi_e(2e+1) \in \{3e+1, 3e+2\}$ then we define $\beta(\varphi_e(2e+1)) = \alpha(2e)$ from the moment when the computation $\varphi_e(2e+1)$ converges. So, up to the stage when $\varphi_e(2e+1)$ becomes defined, both $\beta(3e+1)$ and $\beta(3e+2)$ behave like $\alpha(2e+1)$, but after that stage the set $\beta(\varphi_e(2e+1))$ behaves as $\alpha(2e)$ while the second one continues to behave as $\alpha(2e+1)$.

The main idea here is to exploit the possibility to transform

$$\alpha(2e+1) \cap \{a(k,e), b(k,e), c(k,e)\}$$

into

$$\alpha(2e) \cap \{a(k,e), b(k,e), c(k,e)\},$$

for all k, uniformly. This means that, at the moment when $\beta(\varphi_e(2e+1))$ switches from behaving like $\alpha(2e+1)$ to behave like $\alpha(2e)$, every $x \in \alpha(2e+1)$ must have at its disposal the possibility of changing its membership status from $\alpha(2e+1)(x)$ to $\alpha(2e)(x)$. In isolation, $\mathcal{D}_e$ can easily achieve this: To this end, notice also, that the total number of membership changes of every $x \neq a(k,e)$, relative to $\alpha(2e+1)$ and $\alpha(2e)$, are the same. So, we only need to control the changes of $a(k,e)$, with $k \in \omega$.

In details, the strategy for $\mathcal{D}_e$ (henceforth referred to also as "Strategy $\mathcal{D}_e$") in isolation proceeds as follows.

1. Wait for the computation $\varphi_e(2e+1)$ to be defined. If $\varphi_e(2e+1) \notin \{3e+1, 3e+2\}$ then do nothing, since in this case, evidently, $\alpha(2e+1) \neq \beta(\varphi_e(2e+1))$. Otherwise,

2. Wait until, for every k, if Substrategy $\mathcal{P}_{k,e}$ has acted relatively to $a(k,e)$ then at least one of $2e$ or $2e+1$ has been already put into the range of g_k.

3. If we successfully stop waiting for every k, make $\beta(\varphi_e(2e+1))$ equal to the current $\alpha(2e)$ as follows:

   (a) if $a(k,e) \in \alpha(2e) \setminus \alpha(2e+1)$ then enumerate $a(k,e)$ into $\beta(\varphi_e(2e+1))$;

   (b) if $a(k,e) \in \alpha(2e+1) \setminus \alpha(2e)$ then remove $a(k,e)$ from $\beta(\varphi_e(2e+1))$.

4. After 3 is done, do not touch anymore $a(k,e)$, $k \in \omega$, in any of the sets $\alpha(2e)$, $\alpha(2e+1)$, $\beta(3e)$, $\beta(3e+1)$, $\beta(3e+2)$.

Interactions between strategies. Obviously, there is no interference between the various substrategies $\mathcal{P}_{k,m}$, for $k, m \in \omega$. Strategies $\mathcal{D}_e$, with $e \in \omega$, are pairwise independent too since, if $e \neq e'$, then we diagonalize against reductions of α to β via φ_e and $\varphi_{e'}$, respectively, on different α-indices.

No substrategy $\mathcal{P}_{k,m}$, $k \in \omega$, conflicts with Strategy $\mathcal{D}_e$, if $m \neq e$, since they deal with disjoint pairs of sets, namely, with the pair $\alpha(2m)$, $\alpha(2m+1)$, and the pair $\alpha(2e)$, $\alpha(2e+1)$, respectively.

Strategy $\mathcal{D}_e$ can conflict with Substrategy $\mathcal{P}_{k,e}$ for an isolated k, or meet an infinite series of conflicts with the sequence consisting of Substrategies $\mathcal{P}_{k,e}$, $k \in \omega$. The sequence $\mathcal{P}_{k,e}$, $k \in \omega$, might prevent Strategy $\mathcal{D}_e$ from succeeding, because it might cause $\mathcal{D}_e$ to wait forever in item 3 because of the following reasons:

- some substrategy $\mathcal{P}_{k',e}$ has acted, using $a(k', e)$, before the moment when $\varphi_e(2e+1)$ has converged, but $\mathcal{P}_{k',e}$ does not achieve its goal, i.e. neither of $2e, 2e+1$ becomes a value of $g_{k'}$ (so, $\mathcal{P}_{k',e}$ might want to move again $a(k', e)$, conflicting with 4 of $\mathcal{D}_e$); or

- each $\mathcal{P}_{k,e}$, $k \in \omega$, achieves its goal, but at any stage after convergence of the computation $\varphi_e(2e+1)$ there is at least one $\mathcal{P}_{k,e}$ which has already acted with $a(k, e)$, but is still in progress, i.e. has not as yet contributed to the definition of g_k.

To resolve these conflicts we use, in Substrategy $\mathcal{P}_{k,m}$, two triples of functions $a_i(k, m)$, $b_i(k, m)$, $c_i(k, m)$, with $i \in \{0, 1\}$ instead of just a single triple of functions $a(k, m), b(k, m), c(k, m)$ as above. The subscript i is considered as a switch for using one triple or another.

For a given k, m, we start with option $i = 0$. This means that we implement the instructions of Substrategy $\mathcal{P}_{k,m}$, with the functions $a(k, m)$, $b(k, m)$, $c(k, m)$ replaced by the functions $a_0(k, m)$, $b_0(k, m)$, $c_0(k, m)$, respectively. We carry out this option until $\varphi_m(2m+1)$ is defined (if ever) at a stage s_m. We continue option $i = 0$ forever if $\varphi_m(2m+1) \notin \{2m, 2m+1\}$. Otherwise, we go to the end of stage s_m and transform $\beta(\varphi_m(2m+1))$ into $\alpha(2m)$ as in item 3 of Strategy $\mathcal{D}_m$; but *for all* $k \in \omega$, starting from stage $s_m + 1$ we switch to option $i = 1$, i.e. we carry out the instructions of Substrategy $\mathcal{P}_{k,m}$ operating with $a_1(k, m)$, $b_1(k, m)$, $c_1(k, m)$ from the very beginning but *only for those* k *such that by stage* s_m *the range of the function* g_k *is disjoint from* $\{2m, 2m+1\}$: In this case, from stage $s_m + 1$ the functions $a_0(k, m)$, $b_0(k, m)$, $c_0(k, m)$ play a passive role while proceeding under option $i = 1$.

We assume that all these six functions above are injective and have pairwise disjoint ranges.

Given m, we build approximations to the sets $\alpha(2m), \alpha(2m+1)$ uniformly in m by a stage construction. Approximations to the sets $\beta(3m)$, $\beta(3m+1)$, $\beta(3m+2)$ are built essentially from the approximations for $\alpha(2m)$, $\alpha(2m+1)$. We define simultaneously the sequence g_k, with $k \in \omega$, of partial computable functions, or, to be more precise, the preimages of g_k on the set $\{2m, 2m+1\}$. We denote the option used for constructing the sets $\alpha(2m), \alpha(2m+1)$ at stage s, by $i^s(m)$.

The construction. The construction is by stages. At each stage, if not explicitly redefined, every parameter is understood to retain the same value as at the previous stage.

Stage **0.** Let $i^0(m) = 0$,

$$f_\alpha(2m, x, 0) = 1 \text{ with } \gamma_\alpha(2m, x, 0) = a; \quad f_\alpha(2m+1, a_0(k, m), 0) = 0 \text{ with } \gamma_\alpha(2m+1, a_0(k, m), 0) \uparrow;$$

$$f_\alpha(2m+1, \varrho, 0) = 1 \text{ with } \gamma_\alpha(2m+1, \varrho, 0) = a$$

$$\textit{where } x \in \{a_0(k, m), b_0(k, m), c_0(k, m)\} \textit{ and } \varrho \in \{b_0(k, m), c_0(k, m)\}$$

So, we put a_0, b_0, c_0 to $\alpha(2m)$ and put b_0, c_0 to $\alpha(2m+1)$ with maximal change a . Go to stage 1.

Stage $s+1$**.** Let $s = \langle k, m, t\rangle$. At the beginning of stage $s+1$ we decide on the option $i = i^{s+1}(m)$, and execute the instructions of Substrategy $\mathcal{P}^i_{k,m}$, i.e. Substrategy $\mathcal{P}_{k,m}$, but relative to option i.We split the instructions of Substrategy $\mathcal{P}_{k,m}$ into mutually exclusive parts, named Procedure $\mathcal{I}^i_{k,m}$ and Procedure $\mathcal{P}^{i,j}_{k,m}$, where i is the current option, and $j \leq 2$ is a parameter relative to the procedure, connected with the cardinality of range$(g^s_k) \cap \{2m, 2m+1\}$.

Procedure $\mathcal{I}^i_{k,m}$ corresponds to the instructions of item 2 of Substrategy $\mathcal{P}_{k,m}$, while Procedures $\mathcal{P}^{i,j}_{k,m}$, with $j \leq 2$, correspond to items 3–6 of Substrategy $\mathcal{P}_{k,m}$. We complete stage $s+1$ with the procedure *End of Stage* which aims at constructing the sets $\beta(3m)$, $\beta(3m+1)$, $\beta(3m+2)$. Thus Strategy $\mathcal{D}_m$ is in fact implemented in *End of Stage*.

In order to decide on the option i, we go through stage $s+1$ by checking whether $\varphi^{s+1}_m(2m+1)$ is defined, and $\varphi_m(2m+1)$ is in $\{3m+1, 3m+2\}$. If this is not the case then we let $i^{s+1}(m) = 0$ and go to Procedure $\mathcal{I}^0_{k,m}$, otherwise we let $i^{s+1}(m) = 1$. If $i^s(m) = 0$ then we go to *End of Stage*, otherwise we go to Procedure $\mathcal{I}^1_{k,m}$.

For the sake of simplicity, let $a_i = a_i(k,m)$, $b_i = b_i(k,m)$, and $c_i = c_i(k,m)$; we denote by x_i, y_i the π_k-indices possibly determined by Procedures $\mathcal{P}^{i,j}_{k,m}$.

Procedure $\mathcal{I}^i_{k,m}$**.** If x_i and b_i have not been determined by Procedure $\mathcal{P}^{i,0}_{k,m}$ then go to Procedure $\mathcal{P}^{i,0}_{k,m}$, otherwise execute the instructions of one of the following cases.

1. If $f_\pi(k, x_i, b_i, s{+}1) = 1$ with $\gamma_\pi(k, b_i, x_i, s{+}1)\downarrow$ and $f_\alpha(2m{+}2, b_i, s{+}1) = 0$ then go to Procedure $\mathcal{P}^{i,j}_{k,m}$, for the relevant j.

2. If $f_\pi(k, x_i, b_i, s+1) = 0$ and $f_\alpha(2m+2, b_i, s+1) = 0$ then enumerate b_i into $\alpha(z)$ for all $z \notin \{2m, 2m+1\}$ with $\gamma_\alpha(z, b_i, s+1) = \gamma_\pi(k, x_i, b_i, s+1)$ and go to the next stage.

3. If $f_\pi(k, x_i, b_i, s{+}1) = 0$ and $f_\alpha(2m{+}2, b_i, s{+}1) = 1$ with $\gamma_\alpha(2m{+}2, b_i, s{+}1) = \gamma_\pi(k, x_i, b_i, s+1)$, then go to the next stage.

4. If $f_\pi(k, x_i, b_i, s+1) = 1$ and $f_\alpha(2m+2, b_i, s+1) = 1$ then extract b_i from $\alpha(z)$ for all $z \notin \{2m, 2m+1\}$ and go to Procedure $\mathcal{P}^{i,j}_{k,m}$, for the relevant j.

Procedure $\mathcal{P}^{i,0}_{k,m}$**.** This procedure is executed if a_i, b_i, c_i have been chosen but a π_k-index x_i has not been determined.

Search for x such that $f_\pi(k, x_i, q_i, s+1) = 1$ for all q_i, where $q_i \in \{a_i, b_i, c_i\}$.

If x exists then denote by x_i the least such x, go to Procedure $\mathcal{P}^{i,1}_{k,m}$; otherwise go to the next stage.

Procedure $\mathcal{P}^{i,1}_{k,m}$. This procedure is executed if neither $2m$ nor $2m+1$ is in the range of g_k and if a_i and a π_k-index x_i have been chosen in Procedure $\mathcal{P}^{i,0}_{k,m}$.

If a_i does exhaust all possible changes in $\pi_k(x_i)$ then define

$$g_k(x_i) = \begin{cases} 2m, & \text{if } f_\pi(k, x_i, a_i, s+1) = f_\alpha(2m, a_i, s), \\ 2m+1, & \text{if } f_\pi(k, x_i, a_i, s+1) = f_\alpha(2m+1, a_i, s), \end{cases}$$

and go to Procedure $\mathcal{P}^{i,2}_{k,m}$.

If a_i does not exhaust all possible changes in $\pi_k(x_i)$ then execute one of the following nine mutually exclusive cases and after that go to the next stage.

1. If $f_\alpha(2m, a_i, s) = f_\pi(k, x_i, a_i, s+1) = 1$ and $f_\alpha(2m+1, a_i, s) = 0$ then extract a_i from $\alpha(2m)$.

2. If $f_\alpha(2m, a_i, s) = f_\alpha(2m+1, a_i, s) = f_\pi(k, x_i, a_i, s+1) = 0$ then enumerate a_i into $\alpha(2m+1)$.

3. If $f_\alpha(2m, a_i, s) = f_\pi(k, x_i, a_i, s+1) = 0$ and $f_\alpha(2m+1, a_i, s) = 1$ then enumerate a_i into $\alpha(2m)$.

4. If $f_\alpha(2m, a_i, s) = f_\pi(k, x_i, a_i, s+1) = f_\alpha(2m+1, a_i, s) = f_\alpha(2m+2, a_i, s) = 1$ then remove a_i from $\alpha(2m+1)$.

5. If $f_\pi(k, x_i, a_i, s+1) = f_\alpha(2m+1, a_i, s) = 1$ and $f_\alpha(2m, a_i, s) = 0$ then extract a_i from $\alpha(2m+1)$.

6. If $f_\alpha(2m, a_i, s) = f_\pi(k, x_i, a_i, s+1) = f_\alpha(2m+1, a_i, s) = 0$ then enumerate a_i into $\alpha(2m)$.

7. If $f_\alpha(2m+1, a_i, s) = f_\pi(k, x_i, a_i, s+1) = 0$ and $f_\alpha(2m, a_i, s) = 1$ then enumerate a_i into $\alpha(2m+1)$.

8. If $f_\alpha(2m, a_i, s) = f_\pi(k, x_i, a_i, s+1) = f_\alpha(2m+1, a_i, s) = f_\alpha(2m+2, a_i, s) = 1$ then remove a_i from $\alpha(2m)$.

9. Cases 1–8 do not hold. Do nothing, just go to the next stage.

Procedure $\mathcal{P}^{i,2}_{k,m}$. This procedure is executed when a_i, b_i, c_i have been chosen and exactly one of numbers $2m$ or $2m+1$ is in the range of g_k. Let $\tilde{m} \in \{2m, 2m+1\}$ be the number which still is not in the range of g_k. The procedure includes $\tilde{m}$ into the range of g_k, and after that it continues to control correctness of $\tilde{m}$ as a value of g_k. This procedure corresponds to item 6 of Substrategy $\mathcal{P}_{k,m}$, and consists of the following 4 mutually exclusive cases.

1. $\tilde{m} \notin \text{range}(g^s_k)$ and there exists y such that $y \neq x$ and $f_\pi(k, y, q_i, s+1) = f_\alpha(\widetilde{m}, q_i, s)$ for all q_i, where $q_i \in \{a_i, b_i, c_i\}$. Let y_i stand for the least such y. Define $g_k(y_i) = \tilde{m}$ and go the next stage.

2. $g_k(y_i) = \tilde{m}$ and $f_\pi(k, y_i, c_i, s+1) = f_\alpha(2m+2, c_i, s) = 0$. Enumerate c_i into $\alpha(z)$ for all $z \notin \{2m, 2m+1\}$ and go to the next stage.

3. $g_k(y_i) = \tilde{m}$ and $f_\pi(k, y_i, c_i, s+1) = f_\alpha(2m+2, c_i, s) = 1$.
   Remove c_i from $\alpha(z)$ for all $z \notin \{2m, 2m+1\}$ and go to the next stage.

4. Cases 1–3 do not hold. Do nothing, just go to the next stage.

End of Stage. If $\varphi_m^{s+1}(2m+1)$ is defined and $\varphi_m(2m+1) \in \{3m+1, 3m+2\}$ then denote $\varphi_m(2m+1) = m'$, and denote the number from $\{3m+1, 3m+2\}$ different from m' by m''.

1. If $i^{s+1}(m) = 0$ then define

$$\begin{aligned} \beta^{s+1}(3m) &= \alpha^{s+1}(2m), \\ \beta^{s+1}(3m+1) = \beta^{s+1}(3m+2) &= \alpha^{s+1}(2m+1). \end{aligned}$$

2. If $i^{s+1}(m) = i^s(k, m) = 1$ then define

$$\begin{aligned} &\beta^{s+1}(3m) = \beta^{s+1}(m') = \alpha^{s+1}(2m), \\ &\beta^{s+1}(m'') = \alpha^{s+1}(2m+1). \end{aligned}$$

3. If $i^{s+1}(m) = 1$ but $i^s(m) = 0$ then define

$$\begin{aligned} &\beta^{s+1}(3m) = \beta^{s+1}(m') = \alpha^{s+1}(2m) \\ &\beta^{s+1}(m'') = \alpha^{s+1}(2m+1), \end{aligned}$$

   where

$$\begin{aligned} &\alpha^{s+1}(2m) = \\ &\alpha^s(2m) \cup \{a_1(l,m), b_1(l,m), c_1(l,m) \mid \text{range}(g_l^s) \cap \{2m, 2m+1\} = \emptyset\} \end{aligned}$$

   and

$$\begin{aligned} &\alpha^{s+1}(2m+1) = \\ &\qquad \alpha^s(2m+1) \cup \{b_1(l,m), c_1(l,m) \mid \text{range}(g_l^s) \cap \{2m, 2m+1\} = \emptyset\}. \end{aligned}$$

   (Notice that we never choose to operate with either of $a_1(l,m)$, $b_1(l,m)$, $c_1(l,m)$ if $\text{range}(g_l^s) \cap \{2m, 2m+1\} \neq \emptyset$. In other words, once the first definition has been made for g_l by Procedure $\mathcal{P}_{l,m}^{0,1}$, the procedure does not need to move again $a_0(l,m)$, which then can freely change its membership status in $\beta(m'')$ from that of $\alpha(2m+1)$ to that of $\alpha(2m)$).

Go to the next stage.

Verification. By Lemma 2.2, and Procedure *End of Stage*, we have that α and β are Σ^{-1}_a-computable numberings of the same family $\mathcal{A} = \alpha(\omega)$.

Lemma 2.3. *$\alpha \not\leq \beta$.*

Proof. Suppose that φ_m is total, and $\varphi_m(2m+1) = m' \in \{3m+1, 3m+2\}$. Let $s+1$ be the least stage at which $i^{s+1}(m) = 1$. Since up to this stage (i.e. at all stages $t \leq s$) we had $\beta^t(m') = \alpha^t(2m+1)$, by Lemma 2.2 we have that for every k such that $\text{range}(g^s_k) \cap \{2m, 2m+1\} \neq \emptyset$, the number v of changes of $a_0(k,m)$ in the approximation to $\beta(m')$ up to stage s, is $v < n$, so we can afford to change $\beta^{s+1}(m')(a_0(k,m))$ if needed, in order to switch $\beta(m'')$ to $\alpha(2m)$.

Since $\alpha \neq \beta \circ \varphi_m$, for every total φ, we have that $\alpha \not\leq \beta$. □

Lemma 2.4. *For every m, there exist k_m such that, letting $a = a_0(k_m, m)$, $b = b_0(k_m, m)$, $c = c_0(k_m, m)$, we have that $\{a, b, c\} \subseteq \alpha(2m)$, $\alpha(2m+1) \cap \{a, b, c\} = \{b, c\}$, and for every $m' \neq m$, $\alpha(2m') \cap \{a, b, c\} = \alpha(2m'+1) \cap \{a, b, c\} = \emptyset$.*

Proof. Without loss of generality, we can assume that π_0 is a numbering of the family $\{\emptyset\}$ and that $\pi^s_0(x) = \emptyset$ for all $s, x \in \omega$. Then, for every m, at stage 0, numbers $a_0(0,m), b_0(0,m), c_0(0,m)$ are enumerated into $\alpha(2m+1)$ while the numbers $b_0(0,m), c_0(0,m)$ are enumerated into $\alpha(2m)$. Since $\pi^s_0(x) = \emptyset$ for all $s, x \in \omega$ it follows that we will never deal with Procedures $\mathcal{P}^{0,j}_{k,m}$ for $j \geq 1$. Therefore, we will not operate with numbers $a_0(0,m), b_0(0,m), c_0(0,m)$ at all stages $s > 0$, so their mebership status relative to any $\alpha(z)$ never changes. □

In particular,

Corollary 2.5. *α is a Friedberg numbering.*

Proof. Immediate. □

Lemma 2.6. *For every k, if π_k is a numbering of $\mathcal{A}$ then $\text{range}(g_k) = \omega$.*

Proof. Assume that π_k is a numbering of $\mathcal{A}$, and let $m \in \omega$. We show in this case that $\{2m, 2m+1\} \subseteq \text{range}(g_k)$. We distinguish the following two cases.

Case 1: $i^s(m) = 0$ for every s. Thus we implement Substrategy $\mathcal{P}_{k,m}$, at each stage of the form $\langle k, m, t\rangle$, by executing only procedures relative to option $i = 0$, operating on the elements $a_0 = a_0(k,m)$, $b_0 = b_0(k,m)$, $c_0 = c_0(k,m)$. Since π_k is a numbering of the family, Procedure $\mathcal{P}^{0,0}_{k,m}$ gives eventually a successful number x_0: Otherwise, for every s, we would have $a_0, b_0, c_0 \in \alpha^s(2m)$ but for every x, s, $\pi^s_k(x) \cap \{a_0, b_0, c_0\} \neq \{a_0, b_0, c_0\}$. Similarly, Procedure $\mathcal{I}^0_{k,m}$ eventually exits, after its last execution, at item 1 or item 4, the other outcomes providing $b_0 \in \alpha(z) \setminus \pi_k(x)$, for all z: Notice that item 1 or item 4 give that $b_0 \in \pi_k(x) \cap \alpha(2m) \cap \alpha(2m+1)$ and $b_0 \notin \alpha(z)$ for every $z \notin \{2m, 2m+1\}$. After last execution of $\mathcal{I}^0_{k,m}$, at stages of the form $\langle k, m, t\rangle$ we execute procedure $\mathcal{P}^{0,1}_{k,m}$. Since π_k is a numbering of $\mathcal{A}$, by Lemma 2.2 the only possible exits for this procedure are when $\pi_k(x)(a_0)$ has made n changes, and we define $g_k(x_0) = 2m$, or $g_k(x_0) = 2m+1$. After this, again at stages of the form $\langle k, m, t\rangle$, we execute Procedure $\mathcal{P}^{0,2}_{k,m}$: Since π_k is a numbering of $\mathcal{A}$, the procedure exits with determining a number $y_0 \neq x_0$ such that $g_k(y_0) = \tilde{m}$, with $\tilde{m} \in \{2m, 2m+1\}$ such that $g_k(x_0) \neq \tilde{m}$: Such a number y_0 exists since $\alpha(2m) \neq \alpha(2m+1)$ by Corollary 2.5.

Case 2: There exists a least stage $s+1$ such that $i^{s+1}(m) = 1$. Up to, and including, stage s, we have already put some pairs $\{2m', 2m'+1\}$ in the range of g_k; or for some m'' we have put only one of $\{2m'', 2m''+1\}$ in the range of g_k: If, say, we have put only $2m'$ in the range of g_k, then arguing as in Case 1, and using the fact that π_k is a numbering of the family, we conclude that Procedure $\mathcal{P}^{0,2}_{k,m}$ eventually finishes off its job, by putting also $2m''+1$ in the range of g_k. A similar argument applies if by stage s we have put only $2m''+1$ in the range of g_k. For all other numbers m (those for which $\mathrm{range}(g^s_k) \cap \{2m, 2m+1\} = \emptyset$), starting from s+1 we stop executing Procedures $\mathcal{P}^{0,0}_{k,m}$, $\mathcal{I}^0_{k,m}$, $\mathcal{P}^{0,1}_{k,m}$, $\mathcal{P}^{0,2}_{k,m}$, and we execute instead $\mathcal{P}^{1,0}_{k,m}$, $\mathcal{I}^1_{k,m}$, $\mathcal{P}^{1,1}_{k,m}$, $\mathcal{P}^{1,2}_{k,m}$. An argument similar to Case 1 allows us to conclude that, by operating with the elements $a_1 = a_1(k,m)$, $b_1 = b_1(k,m)$, $c_1 = c_1(k,m)$, one eventually defines $\{g_k(x_1), g_k(y_1)\} = \{2m, 2m+1\}$ for suitable numbers x_1, y_1 determined by Procedures $\mathcal{P}^{1,0}_{k,m}$, $\mathcal{P}^{1,1}_{k,m}$, and $\mathcal{P}^{1,2}_{k,m}$. □

Corollary 2.7. *For every k, if π_k is a Friedberg numbering of $\mathcal{A}$ then g_k is a total function and g_k reduces π_k to α.*

Proof. Suppose that π_k is a Friedberg numbering of $\mathcal{A}$. If after executing for the last time Procedure $\mathcal{P}^{i,1}_{k,m}$, we define $g_k(x_i) = m'$, with $m' \in \{2m, 2m+1\}$, and after executing for the last time Procedure $\mathcal{P}^{i,2}_{k,m}$, we define $g_k(y_i) = m''$, with $y_i \neq x_i$, $m'' \neq m'$, and $m'' \in \{2m, 2m+1\}$, then we can argue that $\pi_k(y_i) = \alpha(m'')$ as follows. Operating on c_i (by extracting or enumerating c_i into $\alpha(z)$) we make sure that $\pi_k(y_i)(c_i) \neq \alpha(z)(c_i)$ for every $z \notin \{2m, 2m+1\}$. Thus $\pi_k(y_i) \in \{\alpha(2m), \alpha(2m+1)\}$: Since π_k is Friedberg and $y_i \neq x_i$, we conclude that $\pi_k(y_i) = \alpha(m'')$.

Finally let us show that g_k is total: for every x, there exists m such that $\pi_k(x) = \alpha(m)$, but since π_k is Friedberg there is no $y \neq x$ such that $\pi_k(x) = \alpha(m)$. So when we put $m \in \mathrm{range}(g_k)$ we in fact define $g_k(x) = m$. □

□

References

[1] S.A. Badaev, S.S. Goncharov. On computable minimal enumerations,.In Algebra. Proceedings of the Third Internation-al Conference on Algebra, Dedicated to the Memory of M.I. Kargopolov. Krasnoyarsk, August 23-28, 1993.- Walter de Gruyter, Berlin- New York, 1995, pp 21-32

[2] Badaev, Serikzhan A. and Goncharov, Sergey S., *The theory of numberings: open problems*, in: "Computability theory and its applications. Current trends and open problems" (eds. Cholak, Peter A.; Lempp, Steffen; Lerman,

Manuel; and Shore, Richard A.), Amer. Math. Soc., Providence, RI, 2000, 23–38.

[3] Yu.L. Ershov. Numberings of the families of total recursive functions, Sib. Math. J., 1967, v.8, n.5., pp.1015-1025 (Russian).

[4] Yu.L. Ershov,*On a hierarchy of sets, I.* Algebra i Logika, 1968, v.7, n.1, pp,47–74 (Russian).

[5] Yu.L. Ershov, *On a hierarchy of sets, II.* Algebra i Logika, 1968, v.7, n.4, pp,15–47 (Russian).

[6] Yu.L. Ershov, *On a hierarchy of sets, III.* Algebra i Logika, 1970, v.9, n.1, pp,20–31 (Russian).

[7] Yu.L. Ershov, *Theory of numberings.*—Nauka, Moscow, 1977 (Russian).

[8] Yu.L. Ershov, *Theorie der Numerierungen.* Z. Math. Logik Grundlagen Math., 1977, v.23, pp.289–371.

[9] , S.S. Goncharov.Computable single-valued numerations. Algebra and Logic, 1980, v.19, n.5, pp,325–356.

[10] R.M. Friedberg, *Three theorems on recursive enumeration.* J. Symbolic Logic, 1958, v.23, n.3, pp.309–316.

[11] S.S. Goncharov.The family with unique univalent but not the smallest enumeration. Trudy Inst. Matem. SO AN SSSR, v.8, pp.42-48, Nauka, Novosibirsk, 1988 (Russian).

[12] Goncharov, S. S. Problem of the number of non-self-equivalent constructivizations. Algebra and Logic, v.19, n.6(1980), 401-414.

[13] S.S. Goncharov, A. Sorbi, *Generalized computable numerations and non-trivial Rogers semilattices.* Algebra and Logic, 1997, v.36, n.4., pp.359–369.

[14] , Yu.L. Ershov,Goncharov, S. S..Constructive models. Transl. from the Russian. (English) (Siberian School of Algebra and Logic) Siberian School of Algebra and Logic. New York, NY: Consultants Bureau. xii, 293 p. (2000) 1980.

[15] , M. Kummer.Some applications of computable one-one numberings. Arch. Math. Log. v.30, n.4 (1990), 219-230.

[16] H. Putnam, *Trial and error predicates and the solution to a problem of Mostowski.* J. Symbolic Logic, 1965, v.30, no.1, 49–57.

[17] S. Ospichev. Computable family of Σ_a^{-1}-sets without Friedberg numberings. In *6th Conference on Computability in Eueope, CiE 2010*, 6th Conference on Computability in Europe, CiE 2010. Ponta Delgada, Azores, pages 311–315, 2010.

[18] H. Rogers, Jr. *Theory of Recursive Functions and Effective Computability.* McGraw-Hill, New York, 1967.