Classical descriptive set theory as a refinement of effective descriptive set theory

Yiannis N. Moschovakis

Department of Mathematics, University of California, Los Angeles, Ca, USA
Department of Mathematics, University of Athens, Athens, Greece

Abstract

The (effective) Suslin-Kleene Theorem is obtained as a corollary of a standard proof of the classical Suslin Theorem, by noticing that it is mostly constructive and applying to it a naive realizability interpretation.

Effective Descriptive Set Theory is advertized as a *refinement* of the classical theory of definability (on Polish spaces) developed in the first half of the 20th century, for example in the introduction to Moschovakis (2009a). Consider the following paradigmatic case, where

$$\mathcal{X} = X_1 \times \cdots \times X_n \tag{1}$$

is a product of copies of the natural numbers $\mathbb{N} = \{0, 1, \ldots\}$ and $\mathcal{N} = (\mathbb{N} \to \mathbb{N})$, the classical Baire space of all infinite sequences from $\mathbb{N}$:

Suslin's Theorem (Suslin (1917))**.** *If $A \subseteq \mathcal{X}$ and $\mathcal{X} \setminus A$ are both analytic, then A is Borel measurable.*

Suslin-Kleene Theorem. *There is a recursive function $u : \mathcal{N} \times \mathcal{N} \to \mathcal{N}$, such that if α is a code of an analytic set $A \subseteq \mathcal{X}$ and β is a code of its complement $\mathcal{X} \setminus A$, then $u(\alpha, \beta)$ is a Borel code of A.*

Even without precise definitions of the notions and the codings used in these results (which will be given in the sequel), their statements suggest that the second theorem refines the first, as it provides a *uniformity*, an effective method to transform an "analytic-coanalytic" definition of a set $A \subseteq \mathcal{X}$ into a "Borel construction" of A. In fact, it is a much stronger result with wider applicability: Suslin's Theorem is vacuous when $\mathcal{X} = \mathbb{N}$, since every set of natural numbers is (trivially) Borel measurable, while the Suslin-Kleene Theorem yields in this case (very easily) one of the most celebrated results of Kleene:

The basic idea for this article was presented in Moschovakis (1971). The full paper was never written up, but I thought that it would fit well in a volume honoring Prof. N. A. Shanin.

Kleene's Theorem (Kleene (1955a,b)). *Every Δ^1_1 set $A \subseteq \mathbb{N}$ is hyperarithmetical.*

This simple analysis, however, does not do justice to the classical theory, because it fails to take into account the "constructive bent" of the analysts who developed it: the standard proof of Suslin's Theorem in Kuratowski (1966) or Moschovakis (2009a) is, in fact, constructive, and if we apply to it the sort of *realizability* analysis pioneered by Kleene, which is well-understood today, *it yields the Suslin-Kleene Theorem.* From this point of view, the classical work is a refinement of the modern theory, since it yields the uniformities which refine the statements of the classical results, and it also provides constructive proofs that they do.

This is the main point that I want to make in this article, and it basically amounts to an observation about the work of Stephen Cole Kleene: his deepest result in what we now call effective descriptive set theory is a direct corollary of classical work and his independently developed (and to the innocent eye unrelated) work in the foundations of intuitionism. Kleene's main technical tool is his *Second Recursion Theorem*, which he applies in both legs of his work: one might say that the observation we will make here simply reduces these crucial applications of the Recursion Theorem from two to one.[1]

I have included in the last section a discussion of the generality of the method in the article and whether it justifies the title.

Since there are very few researchers who are familiar with both descriptive set theory and intuitionism and I would like to make these ideas accessible more broadly, I am including below precise definitions of all the notions I need, as well as (condensed) outlines of the required arguments.

1. Recursion in Baire space

We summarize here the basic facts about recursive partial functions[2] with variables ranging over $\mathbb{N}$ or $\mathcal{N} = (\mathbb{N} \to \mathbb{N})$ and values in $\mathbb{N}$ or $\mathcal{N}$. To simplify notation, we reserve the Latin letters e, m, s, t, u, v, w (perhaps with subscripts) for variables over $\mathbb{N}$; the Greek letters $\alpha, \beta, \gamma, \delta$ for variables over $\mathcal{N}$; and x, y, z for variables over *points*, i.e., members of *product spaces* as in (1). By definition $(X_1 \times \cdots \times X_n) \times (Y_1 \times \cdots \times Y_m) = X_1 \times \cdots \times X_n \times Y_1 \times \cdots \times Y_m$, and if $x = (x_1, \ldots, x_n), y = (y_1, \ldots, y_m)$, then

$$(x, y) = x \star y = (x_1, \ldots, x_n, y_1, \ldots, y_m).$$

Subsets of these product spaces are called *pointsets*.

[1] Cf. the companion articles Moschovakis (2009b, 2010).

[2] A partial function $f : \mathcal{X} \rightharpoonup \mathcal{Y}$ is a (total) function $f : D_f \to \mathcal{Y}$ defined on some subset $D_f \subseteq \mathcal{X}$, its *domain of convergence*, and for $x \in \mathcal{X}$, we write $f(x)\downarrow \iff x \in D_f$. Partial functions compose *strictly*, so that $f(g_1(x), \ldots, g_m(x))\downarrow \implies g_1(x)\downarrow, \ldots, g_m(x)\downarrow$.

We assume all the elementary facts and standard notations from elementary recursion (computability) theory on $\mathbb{N}$.[3]

To allow some of the arguments of relations and partial functions to vary over $\mathcal{N}$, we set

$$\overline{\alpha}(t) = \langle \alpha(0), \dots, \alpha(t-1) \rangle, \quad \overline{s}(t) = s,$$

and for $x = (x_1, \dots, x_n)$, we put $\overline{x}(t) = (\overline{x}_1(t), \dots, \overline{x}_n(t)) \in \mathbb{N}^n$.

Definition. Fix some $\delta_0 \in \mathcal{N}$. A relation $P \subseteq \mathcal{X} = X_1 \times \cdots \times X_n$ is *semirecursive in* δ_0, if there is a recursive relation $R \subseteq \mathbb{N}^{n+1}$ such that

$$P(x) \iff (\exists t) R(\overline{\delta_0}(t), \overline{x}(t)) \quad (x \in \mathcal{X});$$

a partial function $f : \mathcal{X} \rightharpoonup \mathbb{N}$ is *recursive in* δ_0 if its graph

$$\mathrm{Graph}_f(x, w) \iff f(x) = w$$

is semirecursive in δ_0; a partial function $f : \mathcal{X} \rightharpoonup \mathcal{N}$ is *recursive in* δ_0, if

$$f(x) = \lambda t f^*(x, t),$$

with some $f^* : \mathcal{X} \times \mathbb{N} \rightharpoonup \mathbb{N}$ which is recursive in δ_0, so that, in particular,

$$f(x)\downarrow \iff (\forall t) f^*(x, t)\downarrow;$$

and $f : \mathcal{X} \rightharpoonup \mathbb{N}$ or $f : \mathcal{X} \rightharpoonup \mathcal{N}$ is *continuous* if it is recursive in some δ_0.[4]

We naturally skip the relativizing "in δ_0" clause if δ_0 is recursive.

The basic properties of these relations and partial functions on product spaces are quite easily deduced from the properties of the corresponding number-theoretic notions. We list some of them, skipping the relativized versions for simplicity:

(1) The class of semirecursive relations is closed under (total) recursive substitutions and the operations $\&$, $\vee$, $(\forall t < s)$, $(\exists t)$, $(\exists \alpha)$.

(2) The class of recursive partial functions with values in $\mathbb{N}$ is closed under substitution of recursive partial functions with values in $\mathbb{N}$ and substitution of recursive total functions with values in $\mathcal{N}$.[5]

[3] In particular, $\langle x \rangle$ is the code of the sequence $x = (x_0, \dots, x_{n-1}) \in \mathbb{N}^n$, so that the empty sequence has code 1 and for suitable recursive relations and functions,

$$\langle x \rangle = f_n(x_0, \dots, x_{n-1}), \ \mathrm{Seq}(w) \iff w \text{ is a sequence code}, \ \mathrm{lh}(\langle x \rangle) = n, \ (\langle x \rangle)_i = x_i,$$
$$\langle x \rangle * \langle y_0, \dots, y_{m-1} \rangle = \langle (x, y) \rangle, \ \langle x \rangle \restriction j = \langle x_0, \dots, x_{j-1} \rangle \quad (j \le n).$$

[4] It is easy to check that $f : \mathcal{X} \rightharpoonup \mathbb{N}$ is continuous (by this definition) if its domain of convergence D_f is an open set and f is continuous (in the topological sense) on D_f; and $f : \mathcal{X} \rightharpoonup \mathcal{N}$ is continuous if D_f is a G_δ-set (a countable intersection of open sets) and f is continuous on D_f.

[5] This class is not closed under substitution of recursive partial functions with values in $\mathcal{N}$: for example, if

$$g(\alpha) = 1, \ \ h(e) = \lambda t \mu y T_1(e, t, y), \text{ and } f(e) = g(h(e)),$$

then the relation $f(e)\downarrow \iff (\forall t)(\exists y) T_1(e, t, y)$ is not semirecursive, and so $f : \mathbb{N} \rightharpoonup \mathbb{N}$ is not recursive.

(3) The class of recursive partial functions with values in $\mathcal{N}$ is closed under substitutions of recursive partial functions of both kinds.

(4) If $f : \mathcal{X} \rightharpoonup \mathcal{N}$ is recursive and $f(x) = \beta$, then β is recursive in x.[6]

Following (mostly) standard notation, we set

$$\alpha \mapsto \alpha^* = \lambda t \alpha(t+1),$$
$$(\alpha_0, \ldots, \alpha_{n-1}) \mapsto \langle \alpha_0, \ldots, \alpha_{n-1} \rangle = \lambda t \langle \alpha_0(t), \ldots, \alpha_{n-1}(t) \rangle,$$
$$(\alpha, i) \mapsto (\alpha)_i = \lambda t (\alpha(t))_i, \text{ so that for } i < n,\ (\langle \alpha_0, \ldots, \alpha_{n-1} \rangle)_i = \alpha_i;$$

the *shift* α^* and these coding and decoding functions for tuples in $\mathcal{N}$ are clearly recursive.

Next comes the basic representation result for these partial functions which follows quite easily from the corresponding fact about recursion on $\mathbb{N}$:

Theorem 1 (Normal Form and Parametrization)**.** *There is a recursive function $U : \mathbb{N} \to \mathbb{N}$ and for each product space $\mathcal{X} = X_1 \times \cdots \times X_n$, a recursive relation $T^{\mathcal{X}} \subseteq \mathbb{N}^{n+2}$, so that the following hold, with*

$$\varphi^{\mathcal{X},0}(\varepsilon, x) = \{\varepsilon\}^0(x) = U(\mu s T^{\mathcal{X}}(\varepsilon(0), \overline{\varepsilon^*}(s), \overline{x}(s))),$$
$$\varphi^{\mathcal{X}}(\varepsilon, x) = \{\varepsilon\}(x) = \lambda t \varphi^{\mathcal{X} \times \mathbb{N},0}(x, t).$$

(1) *A partial function $f : \mathcal{X} \rightharpoonup \mathbb{N}$ is recursive in δ_0 if and only if there exists some ε, recursive in δ_0 and such that*

$$f(x) = \{\varepsilon\}^0(x) \quad (x \in \mathcal{X});$$

and $f : \mathcal{X} \rightharpoonup \mathcal{N}$ is recursive in δ_0 if and only if there exists some ε, recursive in δ_0 and such that

$$f(x) = \{\varepsilon\}(x) \quad (x \in \mathcal{X}).$$

(2) *For every product space $\mathcal{Y} = Y_1 \times \cdots \times Y_m$, there is a recursive, total function $S = S^{\mathcal{Y}}_{\mathcal{X}} : \mathcal{N} \times \mathcal{Y} \to \mathcal{N}$ such that for all $x \in \mathcal{X}, y \in \mathcal{Y}$,*

$$\{S(\varepsilon, y)\}(x) = \{\varepsilon\}(y, x).$$

We have favored the partial functions into $\mathcal{N}$ by skipping the superscript in the notation because these are the main objects that we will be using, as in the next, crucial result:

Theorem 2 (Kleene's Second Recursion Theorem)**.** *For each recursive partial $f : \mathcal{N} \times \mathcal{X} \rightharpoonup \mathcal{N}$, there is a recursive $\varepsilon \in \mathcal{N}$ such that*

$$\{\varepsilon\}(x) = f(\varepsilon, x).$$

Proof. The composition $f(S(\alpha, \alpha), x)$ is recursive, and so there is a recursive ε_0 such that

$$\{\varepsilon_0\}(\alpha, x) = f(S(\alpha, \alpha), x);$$

so with $\varepsilon = S(\varepsilon_0, \varepsilon_0)$, we have

$$\{\varepsilon\}(x) = \{S(\varepsilon_0, \varepsilon_0)\}(x) = \{\varepsilon_0\}(\varepsilon_0, x) = f(S(\varepsilon_0, \varepsilon_0), x) = f(\varepsilon, x). \qquad \square$$

[6] We say that β is recursive in $x = (n_1, \ldots, n_k, \alpha_1, \ldots, \alpha_m)$, if β is recursive in $\alpha_1, \ldots, \alpha_m$.

2. The category of (inhabited) Baire-coded sets

A *Baire-coded* (or just *coded*) *set* is a pair $(\mathcal{A}, c_\mathcal{A})$ of a set $\mathcal{A}$ and a surjection (the *coding map*) $c_\mathcal{A} : |\mathcal{A}| \twoheadrightarrow \mathcal{A}$ of a *set of codes* $|\mathcal{A}| \subseteq \mathcal{N}$ onto $\mathcal{A}$, such that $|\mathcal{A}|$ has at least one recursive member. We think of any $\alpha \in |\mathcal{A}|$ such that $c_\mathcal{A}(\alpha) = a \in \mathcal{A}$ as a name of a, and we will generally refer to a "coded set $\mathcal{A}$" and write

$$\overline{\overline{\alpha}} = c_\mathcal{A}(\alpha) \quad (\alpha \in |\mathcal{A}|), \tag{2}$$

when the coding map $c_\mathcal{A}$ is clear from the context or has already been specified.

Consider the following examples of coded sets and "coded versions" of some familiar operations of set theory on them:

(CS1) We view $\mathbb{N}$ and $\mathcal{N}$ as coded sets, taking

$$|\mathbb{N}| = |\mathcal{N}| = \mathcal{N}; \quad c_\mathbb{N}(\alpha) = \alpha(0), \; c_\mathcal{N}(\alpha) = \alpha.$$

(CS2) *Products.* The product $\mathcal{A} \times \mathcal{B}$ of two coded sets is coded by

$$|\mathcal{A} \times \mathcal{B}| = \{\alpha \mid (\alpha)_0 \in |\mathcal{A}|, (\alpha)_1 \in |\mathcal{B}|\}, \quad c_{\mathcal{A}\times\mathcal{B}}(\alpha) = (c_\mathcal{A}((\alpha)_0), c_\mathcal{B}((\alpha)_1)),$$

and similarly for n-fold products; so every product space $\mathcal{X}$ is coded.

(CS3) *Subsets.* Every non-empty subset $\mathcal{B} \subseteq \mathcal{A}$ of a coded set $\mathcal{A}$ is naturally coded by

$$|\mathcal{B}| = \{\alpha \in |\mathcal{A}| \mid c_\mathcal{A}(\alpha) \in \mathcal{B}\}; \quad c_\mathcal{B} = c_\mathcal{A} \restriction |\mathcal{B}|.$$

(CS4) *Continuous and recursive operations on coded sets.* A map $f : \mathcal{A} \to \mathcal{B}$ on one coded set to another is *continuous*, if there is a continuous, partial $f^* : \mathcal{N} \rightharpoonup \mathcal{N}$ which *computes* f, i.e., such that the following diagram commutes:

$$\begin{tikzcd}
\mathcal{A} \arrow[r, "f"] & \mathcal{B} \\
|\mathcal{A}| \arrow[u, "c_\mathcal{A}"] \arrow[r, "f^*"] & |\mathcal{B}| \arrow[u, "c_\mathcal{B}"']
\end{tikzcd}$$

stating pedantically the convergence conditions, this means that for all $\alpha \in \mathcal{N}$,

$$\alpha \in |\mathcal{A}| \implies [f^*(\alpha)\!\downarrow \;\&\; f^*(\alpha) \in |\mathcal{B}| \;\&\; c_\mathcal{B}(f^*(\alpha)) = f(c_\mathcal{A}(\alpha))]. \tag{3}$$

If (3) holds with a recursive f^*, we say that $f : \mathcal{A} \to \mathcal{B}$ is *recursive*; and if $f^* = \{\varepsilon\} : \mathcal{N} \rightharpoonup \mathcal{N}$, we call ε a code of f, so that the sets $(\mathcal{A} \to_c \mathcal{B}), (\mathcal{A} \to_r \mathcal{B})$ of all continuous and recursive maps on $\mathcal{A}$ to $\mathcal{B}$ are also naturally coded.[7] We write

$$f : \mathcal{A} \to_c \mathcal{B} \iff f \in (\mathcal{A} \to_c \mathcal{B}), \quad f : \mathcal{A} \to_r \mathcal{B} \iff f \in (\mathcal{A} \to_r \mathcal{B}).$$

[7] It is natural to think of the class of all Baire-coded sets as a category, with either the continuous or the recursive maps as morphisms. This certainly motivates some of the subsequent definitions, but we will not use it here in any significant way. (See also Footnote 10.)

An interesting special case is $(\mathbb{N} \to_c \mathcal{B})$, which comprises all sequences from $\mathcal{B}$.

(CS5) *Strings, continuous (clopen) subsets and trees.* The set $\mathcal{A}^{<\omega}$ of all finite sequences from a coded set $\mathcal{A}$ is coded by

$$|\mathcal{A}^{<\omega}| = \{\alpha \mid (\forall i < \alpha(0))[(\alpha^*)_i \in |\mathcal{A}|]\}, \quad c^*_{\mathcal{A}}(\alpha) = (\overline{\overline{(\alpha^*)_0}}, \dots, \overline{\overline{(\alpha^*)_{\alpha(0)-1}}})$$

where $\overline{\overline{\beta}} = c_{\mathcal{A}}(\beta)$. The length function $\mathrm{lh} : \mathcal{A}^{<\omega} \to_r \mathbb{N}$, the imbedding $a \mapsto (a)$ of $\mathcal{A}$ into $\mathcal{A}^{<\omega}$, and the restriction $(u, j) \mapsto u \restriction j$ and concatenation $(u, v) \mapsto u \star v$ maps are all recursive operations. For the *initial segment* relation, we write

$$u \sqsubseteq v \iff (\exists j < \mathrm{lh}(v))[u = v \restriction j].$$

A subset $X \subseteq \mathcal{A}$ is *continuous* (clopen) or *recursive* if its characteristic function is continuous or recursive, and the continuous and recursive powersets $\mathcal{P}_c(\mathcal{A}), \mathcal{P}_r(\mathcal{A})$ are coded as subsets of the corresponding coded function spaces $(\mathcal{A} \to_c \mathbb{N})$ and $(\mathcal{A} \to_r \mathbb{N})$:

$$t \in X \iff X(t) = 1, \qquad (X : \mathcal{A} \to_c \mathbb{N}).$$

Note that all subsets of $\mathbb{N}$, $\mathbb{N}^k$ and $\mathbb{N}^{<\omega}$ are continuous.

A *tree* on $\mathcal{A}$ is any set $T \subseteq \mathcal{A}^{<\omega}$ of finite sequences from $\mathcal{A}$ closed under initial segments, i.e., such that $u \sqsubseteq v \in T \implies u \in T$. For any $u \in A^{<\omega}$, we let

$$T_u = \{v \in T \mid u \sqsubseteq v\},$$

so that T_u is also a tree (albeit the empty tree if $u \notin T$) and $T_\emptyset = T$.

The *body* of a tree T on $\mathcal{A}$ is the set of all *infinite paths* in T,

$$[T] = \{f : \mathbb{N} \to_c \mathcal{A} \mid (\forall n \in \mathbb{N})[f \restriction n \in T]\} \subseteq (\mathbb{N} \to \mathcal{A}),$$

where $f \restriction n = (f(0), \dots, f(n-1))$. The *projection* of a tree T on $\mathcal{A} \times \mathcal{A}$ is

$$\mathfrak{p}[T] = \{f : \mathbb{N} \to_c \mathcal{A} \mid (\exists g)(f \star g) \in [T]\} \subseteq (\mathbb{N} \to \mathcal{A}),$$

where $(f \star g)(i) = (f(i), g(i))$. The set $\mathcal{T}_c(\mathcal{A})$ of all continuous trees is naturally coded as a subset of $\mathcal{P}_c(\mathcal{A})$. We will be especially interested in trees on $\mathbb{N}$ (or $\mathbb{N}^k$ for some k), and they are all continuous.

Next we introduce the natural coded versions of the most basic *pointclasses* of Descriptive Set Theory.[8]

(CS6) *Open and semirecursive sets.* By Footnote 4, a set $G \subseteq \mathcal{X}$ is topologically open if it is the domain of a continuous, partial $f : \mathcal{X} \rightharpoonup \mathbb{N}$, so that for some ε,

$$x \in G \iff \{\varepsilon\}^0(x)\downarrow; \tag{4}$$

[8] A *pointclass* is an operation Γ which assigns to each product space $\mathcal{X}$ a family $\Gamma \restriction \mathcal{X}$ of subsets of $\mathcal{X}$; and so a coded pointclass assigns to each $\mathcal{X}$ a coded set $\Gamma \restriction \mathcal{X}$ of subsets of $\mathcal{X}$.

and so the family $\underset{\sim}{\mathbf{\Sigma}}^0_1 \restriction \mathcal{X}$ of open (or $\underset{\sim}{\mathbf{\Sigma}}^0_1$) subsets of $\mathcal{X}$ is coded, with code set $\mathcal{N}$ and coding map

$$\varepsilon \mapsto \{x \in \mathcal{X} \mid \{\varepsilon\}^0(x)\downarrow\}.$$

The class $\Sigma^0_1 \restriction \mathcal{X}$ of semirecursive subsets of $\mathcal{X}$ is the (coded) subset of $\underset{\sim}{\mathbf{\Sigma}}^0_1 \restriction \mathcal{X}$ of open sets with recursive codes.

Lemma 3. *The* (coded) *pointclass* $\underset{\sim}{\mathbf{\Sigma}}^0_1$ *of open pointsets is recursively closed under the operations* $\&, \vee, (\forall t < s), (\exists t), (\exists \alpha)$.

Proof. The precise meaning of this for the case (for example) of $(\exists t)$ is that for each $\mathcal{X}$, there is a recursive map

$$\exists^{\mathbb{N}} : \underset{\sim}{\mathbf{\Sigma}}^0_1 \restriction (\mathcal{X} \times \mathbb{N}) \to \underset{\sim}{\mathbf{\Sigma}}^0_1 \restriction \mathcal{X}$$

such that

$$\exists^{\mathbb{N}}(P)(x) \iff (\exists t \in \mathbb{N}) P(x,t) \quad (P \in \underset{\sim}{\mathbf{\Sigma}}^0_1, x \in \mathcal{X}).$$

To prove this, we choose a recursive ε_0 such that

$$\{\varepsilon_0\}(\alpha, x)\downarrow \iff (\exists t)[\{\alpha\}(x,t)\downarrow]$$

and check that the function $f^* = \{S(\varepsilon_0, \alpha)\} : \mathcal{N} \to \mathcal{N}$ computes $\exists^{\mathbb{N}}$. The same sort of familiar argument proves all the claims in the lemma. $\square$

If (4) holds, we can think of ε as also coding the complement of G and so get codings for the classes $\underset{\sim}{\mathbf{\Pi}}^0_1 \restriction \mathcal{X}, \Pi^0_1 \restriction \mathcal{X}$ of closed and co-semirecursive subsets of $\mathcal{X}$; and then, inductively, we can code the classical finite Borel and projective pointclasses for the product spaces we have been considering, as well as their effective subclasses. We do this explicitly only for the analytic and coanalytic sets with which we are especially concerned.

(CS7) *Analytic and coanalytic sets.* A set $A \subseteq \mathcal{X}$ is *analytic* if it is the projection of a closed set $F \subseteq \mathcal{X} \times \mathcal{N}$, i.e.,

$$x \in A \iff (\exists \alpha) F(x, \alpha).$$

By the coding of closed sets above, this means that there is an $\varepsilon \in \mathcal{N}$ such that

$$x \in A \iff (\exists \alpha)[\{\varepsilon\}^0(x,\alpha)\uparrow]; \tag{5}$$

and so we can code the family $\underset{\sim}{\mathbf{\Sigma}}^1_1 \restriction \mathcal{X}$ of analytic subsets of $\mathcal{X}$ taking $\mathcal{N}$ as the set of codes and

$$\varepsilon \mapsto A^{\mathcal{X}}_{\varepsilon} = \{x \mid (\exists \alpha)[\{\varepsilon\}^0(x,\alpha)\uparrow]\}$$

as the coding map. The class $\underset{\sim}{\mathbf{\Pi}}^1_1 \restriction \mathcal{X}$ of *coanalytic* (complements of analytic) subsets of $\mathcal{X}$ is naturally coded with code set $\mathcal{N}$ again and coding map

$$\varepsilon \mapsto P^{\mathcal{X}}_{\varepsilon} = \{x \mid (\forall \alpha)[\{\varepsilon\}^0(x,\alpha)\downarrow]\};$$

and the class $\underset{\sim}{\mathbf{\Delta}}^1_1 \restriction \mathcal{X}$ of subsets of $\mathcal{X}$ which are both analytic and coanalytic is coded by $\varepsilon \mapsto P^{\mathcal{X}}_{(\varepsilon)_0}$ on the code set $\{\varepsilon \in \mathcal{N} \mid P^{\mathcal{X}}_{(\varepsilon)_0} = \mathcal{X} \setminus P^{\mathcal{X}}_{(\varepsilon)_1}\}$.

The *effective classes* $\Sigma^1_1{\restriction}\mathcal{X}, \Pi^1_1{\restriction}\mathcal{X}, \Delta^1_1{\restriction}\mathcal{X}$ comprise the analytic, coanalytic and $\underset{\sim}{\mathbf{\Delta}}^1_1$ subsets of $\mathcal{X}$ which have recursive codes.

Finally, we give the coding for the Borel sets, which is the most complex one that we need.

(CS8) *Borel sets.* The class $\mathbf{B} \restriction \mathcal{X}$ of Borel (measurable) subsets of $\mathcal{X}$ is the smallest class of subsets of $\mathcal{X}$ which contains all the open sets and is closed under complementation and countable unions.

To code $\mathbf{B} \restriction \mathcal{X}$, we first define by recursion on the countable ordinals the following subsets of $\mathcal{N}$:[9]

$$\mathrm{BC}_0 = \{\alpha \mid \alpha(0) = 0\},$$
$$\mathrm{BC}_\xi = \mathrm{BC}_0 \cup \{\alpha \mid \alpha(0) = 1 \ \&\ (\forall t \in \mathbb{N})[\{\alpha^*\}(t){\downarrow}\ \ \&\ \{\alpha^*\}(t) \in \textstyle\bigcup_{\eta<\xi} \mathrm{BC}_\eta]\}.$$

Next we define, by recursion again, for each ξ, a mapping $c^{\mathcal{X}}_\xi : \mathrm{BC}_\xi \to \mathbf{B}{\restriction}\mathcal{X}$:

$$c^{\mathcal{X}}_0(\alpha) = \{x \in \mathcal{X} \mid \{\alpha^*\}^0(x){\downarrow}\} \quad (\alpha \in \mathrm{BC}_0),$$
$$c^{\mathcal{X}}_\xi(\alpha) = \textstyle\bigcup_t \Big(\mathcal{X} \setminus c^{\mathcal{X}}_{\eta(t)}(\{\alpha^*\}(t))\Big),$$
$$\text{where } \eta(t) = \text{least } \eta \text{ such that } \{\alpha^*\}(t) \in \mathrm{BC}_\eta, \quad (\alpha \in \mathrm{BC}_\xi \setminus \mathrm{BC}_0),$$

and we let

$$A \in \underset{\sim}{\mathbf{\Sigma}}^0_\xi \iff (\exists \alpha \in \mathrm{BC}_\xi)[A = c^{\mathcal{X}}_\xi(\alpha)] \quad (A \subseteq \mathcal{X}). \tag{6}$$

It is easy to show (by induction on ξ) that

$$\eta \le \xi \implies [\mathrm{BC}_\eta \subseteq \mathrm{BC}_\xi \ \&\ c^{\mathcal{X}}_\eta \subseteq c^{\mathcal{X}}_\xi],$$

so that $c^{\mathcal{X}} = \bigcup_\xi c^{\mathcal{X}}_\xi$ defines a function on $\mathrm{BC} = \bigcup_\xi \mathrm{BC}_\xi$; and then, similarly, that each $c^{\mathcal{X}}(\alpha)$ is a Borel subset of $\mathcal{X}$ and every Borel set is $c^{\mathcal{X}}(\alpha)$ for some $\alpha \in \mathrm{BC}$, so that $c^{\mathcal{X}}$ is a coding of $\mathbf{B}{\restriction}\mathcal{X}$.

Lemma 4. *The operations of complementation $A \mapsto A^c = (\mathcal{X} \setminus A)$, countable union $\{A_i \mid i \in \mathbb{N}\} \mapsto \bigcup_i A_i$ and countable intersection $\{A_i \mid i \in \mathbb{N}\} \mapsto \bigcap_i A_i$ are recursive on $\mathbf{B}$.*

Proof. For complementation, choose a recursive ε_0 such that $\{\varepsilon_0\}(\alpha, t) = \alpha$, and let $u(\alpha) = (1) * S(\varepsilon_0, \alpha)$, defined so that

$$u(\alpha)(0) = 1 \text{ and for all } t,\ \{u(\alpha)^*\}(t) = \alpha.$$

This is a recursive function and it computes the complementation operation on $\mathbf{B}$, since for $\alpha \in \mathrm{BC}$,

$$c^{\mathcal{X}}(u(\alpha)) = \textstyle\bigcup_t(\mathcal{X} \setminus c^{\mathcal{X}}(\alpha)) = \mathcal{X} \setminus c^{\mathcal{X}}(\alpha).$$

The rest follows by applying the De Morgan rules and the idempotence of the complementation operation, $(A^c)^c = A$. ☐

[9] I am using here almost exactly the coding in Section 7B of Moschovakis (2009a).

3. Effective truth and realizability on $\mathcal{S}$

Suppose $(\mathcal{A}, c_{\mathcal{A}}), (\mathcal{B}, c_{\mathcal{B}})$ are fixed coded sets and $P \subseteq \mathcal{A} \times \mathcal{B}$. The $\forall$-$\exists$ proposition[10]

$$\varphi \equiv (\forall x \in \mathcal{A})(\exists y \in \mathcal{B})P(x, y) \tag{7}$$

is *effectively* (or *uniformly*) *true* (for the given codings) if there is a recursive partial function $u : \mathcal{N} \rightharpoonup \mathcal{N}$ such that for all α,

$$\alpha \in |\mathcal{A}| \implies [u(\alpha)\downarrow \ \& \ u(\alpha) \in |\mathcal{B}| \ \& \ P(c_{\mathcal{A}}(\alpha), c_{\mathcal{B}}(u(\alpha)))].$$

Our aim in the section is to show that *constructive* $\forall$-$\exists$ *consequences of effectively true* $\forall$-$\exists$ *propositions are also effectively true*—in fact something a little stronger than this; and then the Suslin-Kleene Theorem will follow directly from a constructive proof of this kind of the Suslin Theorem, which is expressed by the $\forall$-$\exists$ proposition

$$(\forall A \in \underset{\sim}{\mathbf{\Delta}}{}^1_1)(\exists B \in \mathbf{B})[A = B] \quad (A, B \subseteq \mathcal{X}).$$

The proof is given by formalizing constructive reasoning about coded sets and then applying a very simple realizability interpretation on this intuitionistic theory. It uses very well understood ideas and techniques—so we will be brief and we will skip the formal details. We will, however, be quite precise in the formulation of definitions and results.

The language $\mathcal{L}$. We fix a family $\mathcal{S}$ of coded sets which includes all the coded sets we care about, and a set $\mathbb{M}$ which includes all the members of every $\mathcal{A} \in \mathcal{S}$. The idea is to view each $\mathcal{A}$ as a *type* (or *sort*) of member of $\mathbb{M}$ and to introduce a rich, typed first order language $\mathcal{L}$ on this set of types.

The *terms* of type $\mathcal{A}$ comprise a sequence $\mathsf{v}_0, \mathsf{v}_1, \ldots$ of variables of type $\mathcal{A}$, and a constant c for each *recursive member* c of $\mathcal{A}$, i.e., each $c = c_{\mathcal{A}}(\alpha)$ with a recursive $\alpha \in |\mathcal{A}|$. (So formally, we should write $\mathsf{v}_i^{\mathcal{A}}$ and $\mathsf{c}^{\mathcal{A}}$.)

The *formulas* of $\mathcal{L}$ are defined by the following recursion, where P is any n-ary relation over $\mathbb{M}$, viewed as a constant naming itself and $(\mathcal{A}, c_{\mathcal{A}})$ is any $\mathcal{N}$-coded set with $\mathcal{A} \subseteq \mathbb{M}$:

$$\begin{aligned}\varphi :\equiv s = t \mid s \in t \mid P(s_1, \ldots, s_n) \\ \mid \neg(\varphi_1) \mid (\varphi_1) \ \& \ (\varphi_2) \mid (\varphi_1) \vee (\varphi_2) \mid (\varphi_1) \rightarrow (\varphi_2) \\ \mid (\exists \mathsf{v}_i^{\mathcal{A}})\varphi_1 \mid (\forall \mathsf{v}_i^{\mathcal{A}})\varphi_1\end{aligned}$$

It is important here that the relation symbols are not typed, e.g., $s = t$ is well formed even when s and t are of different type.

[10] It is natural (and useful) to think of the class of all coded sets as a category, with an arrow $\mathcal{A} \xrightarrow{u} \mathcal{B}$ signifying that the proposition $(\forall x \in \mathcal{A})(\exists y \in \mathcal{B})[x = y]$ is effectively true via u.

The symbols for identity and membership are superfluous since we have allowed names for all relations on $\mathbb{M}$, but their explicit mention here suggests how we intend to use the language. We will also use heavily the *continuous application* relations $\mathrm{Ap}^{\mathcal{A},\mathcal{B}} \subseteq \mathbb{M}^3$,

$$\mathrm{Ap}^{\mathcal{A},\mathcal{B}}(f,x,y) \iff f : \mathcal{A} \to_c \mathcal{B} \text{ and } f(x) = y. \tag{8}$$

We will "abbreviate and misspell" these formal expressions, as usual, skipping (or adding) parentheses, writing $(\exists x \in \mathcal{A})\varphi$, $(\forall x \in \mathcal{A})\varphi$ for the last two clauses, and identifying a recursive element $c \in \mathcal{A}$ with its formal name $\mathsf{c}^{\mathcal{A}}$, letting the context determine $\mathcal{A}$. We also set

$$x \in \mathcal{A} :\equiv (\exists x' \in \mathcal{A})[x = x'], \quad f(x) = y :\equiv \mathrm{Ap}^{\mathcal{A},\mathcal{B}}(f,x,y),$$

letting again the context determine the specific $\mathcal{A}, \mathcal{B}$ in the second of these.

Classical semantics. The *satisfaction relation* $\pi \models \chi$ between an assignment to the variables and a formula is defined by the usual recursive clauses and disregards the codings, for example:[11]

$$\pi \models s = t \iff \pi(s) = \pi(t), \quad \pi \models s \in t \iff \pi(s) \in \pi(t)$$
$$\models (\exists x \in \mathcal{A})\varphi \iff \text{ there is some } a \in \mathcal{A} \text{ such that } \pi\{x := a\} \models \varphi$$

Realizability. The *C-realization relation* $\varepsilon, \pi \Vdash^{\mathrm{cr}} \chi$ between $\varepsilon \in \mathcal{N}$, an assignment π and a formula χ is defined by the following recursive clauses:

$$\begin{aligned}
\varepsilon, \pi \Vdash^{\mathrm{cr}} \chi &\iff \pi \models \chi, \text{ if } \chi \text{ is } s = t \text{ or } s \in t \text{ or } P(s_1, \ldots, s_n) \\
\varepsilon, \pi \Vdash^{\mathrm{cr}} \neg\varphi &\iff \text{for every } \alpha \in \mathcal{N},\ \alpha, \pi \not\Vdash^{\mathrm{cr}} \varphi \\
\varepsilon, \pi \Vdash^{\mathrm{cr}} \varphi \;\&\; \psi &\iff (\varepsilon)_0, \pi \Vdash^{\mathrm{cr}} \varphi \text{ and } (\varepsilon)_1, \pi \Vdash^{\mathrm{cr}} \psi \\
\varepsilon, \pi \Vdash^{\mathrm{cr}} \varphi \vee \psi &\iff [\varepsilon(0) = 0 \text{ and } \varepsilon^*, \pi \Vdash^{\mathrm{cr}} \varphi] \text{ or } [\varepsilon(0) = 1 \text{ and } \varepsilon^*, \pi \Vdash^{\mathrm{cr}} \psi] \\
\varepsilon, \pi \Vdash^{\mathrm{cr}} \varphi \to \psi &\iff \text{for every } \alpha, \\
&\qquad \text{if } \alpha, \pi \Vdash^{\mathrm{cr}} \varphi, \text{ then } [\{\varepsilon\}(\alpha)\!\downarrow \text{ and } \{\varepsilon\}(\alpha), \pi \Vdash^{\mathrm{cr}} \psi] \\
\varepsilon, \pi \Vdash^{\mathrm{cr}} (\exists x \in \mathcal{A})\varphi &\iff (\varepsilon)_0 \in |\mathcal{A}| \text{ and } (\varepsilon)_1, \pi\{x := (c_{\mathcal{A}}((\varepsilon)_0))\} \Vdash^{\mathrm{cr}} \varphi \\
\varepsilon, \pi \Vdash^{\mathrm{cr}} (\forall x \in \mathcal{A})\varphi &\iff (\text{for every } \alpha \in |\mathcal{A}|) \\
&\qquad [\{\varepsilon\}(\alpha)\!\downarrow \text{ and } \{\varepsilon\}(\alpha), \pi\{x := (c_{\mathcal{A}}(\alpha))\} \Vdash^{\mathrm{cr}} \varphi.
\end{aligned}$$

We skip the π in the notation when χ is a sentence and (as usual) π is irrelevant.

A sentence is *C-realizable* if some ε C-realizes it,

$$\Vdash^{\mathrm{cr}} \chi \iff \text{ for some } \varepsilon \in \mathcal{N}, \varepsilon \Vdash^{\mathrm{cr}} \chi, \tag{9}$$

and *recursively C-realizable* if it is C-realized by a recursive ε,

$$\Vdash^{\mathrm{rcr}} \chi \iff \text{ for some recursive } \varepsilon \in \mathcal{N}, \varepsilon \Vdash^{\mathrm{cr}} \chi. \tag{10}$$

[11] Assignments respect types and are assumed extended to give the correct values to the constants. The *update* $\pi\{\mathsf{v} := a\}$ changes π only at v, to which it assigns a.

A formula χ is *C-realizable* if its universal closure is C-realizable, and similarly for recursive C-realizability.

Note. We have allowed (almost) arbitrary codings for the sets in $\mathcal{S}$, as well as names in $\mathcal{L}$ for all (typed) recursive points and all relations on $\mathbb{M}$, and we have identified realizability with satisfaction on the prime formulas. These non-constructive features—and the fact that we argue classically about it—make C-realizability a useful tool for dealing effectively (if not always constructively) with complex families of sets, but they also make it quite unusual as a realizability notion.[12] Still, the general form of the definition is sufficiently close to Kleene's original "number realizability" to make the next result routine, once we fix a formulation of intuitionistic logic which is appropriate for $\mathcal{L}$:[13]

Theorem 5 (Kleene)**.** *If a formula χ is a formal consequence in intuitionistic predicate calculus of recursively C-realizable sentences $\chi_1, \dots, \chi_n$, then χ is also recursively C-realizable.*

Realizability, truth and effective truth. Directly from the definition, a $\forall$-$\exists$ sentence is recursively C-realizable exactly when it is effectively true. On the other hand, not all realizable sentences are true: for example

$$(\forall \alpha \in \mathcal{N})[(\exists t)[\alpha(t) = 0] \vee (\forall t)[\alpha(t) \neq 0]]$$

is true but not realizable, and hence its (false) negation is realizable. This is a well-understood feature of all realizability interpretations which gets in the way of using realizability methods to derive classical results. We can bypass it, up to a point, by utilizing the richness of $\mathcal{L}$ to express classically interesting propositions by "essentially" $\forall$-$\exists$ sentences, using the next proposition.

A formula $\varphi(x_1, \dots, x_n)$ with (no more than) the indicated free variables ranging over $\mathcal{A}_1, \dots, \mathcal{A}_n$ is *robust* if the following two conditions hold:

(R1) If $\varepsilon, \pi \Vdash^{\mathrm{cr}} \varphi(x_1, \dots, x_n)$, then $\pi \models \varphi(x_1, \dots, x_n)$.

(R2) If $\pi \models \varphi(x_1, \dots, x_n)$ and for each i, $\alpha_i \in |\mathcal{A}_i|$ and $\overline{\overline{\alpha_i}} = \pi(x_i)$, then there is some ε recursive in $\alpha_1, \dots, \alpha_n$ such that $\varepsilon, \pi \Vdash^{\mathrm{cr}} \varphi(x_1, \dots, x_n)$.

[12] I thank Peter Aczel for pointing me to the "Kleene realizability" in Definition 8.4 of Rathjen (2005) which shares many important features with the present notion, including having code sets (types) for each set on which the realizers operate and (basically) the exact, same inductive clauses for the propositional and the bounded quantification constructs. The main differences are that Rathjen's types are defined "internally", within his constructive set theory, and so he can refer to and operate on them—which we cannot do in $\mathcal{L}$; and (more significantly) that for membership $x \in y$ and identity $x = y$, Rathjen's realization relationship is defined inductively, as in a classical forcing model, so that we cannot simply "import classical facts" about sets, as we can do in $\mathcal{L}$, in which realizability coincides with (classical) truth on prime formulas. It is very difficult to compare the two notions, but it would (obviously) be very interesting if Rathjen can derive the Suslin-Kleene Theorem using his more constructive notion.

[13] The technical assumption that each code set $|\mathcal{A}|$ has a recursive member is needed here to make $(\exists x \in \mathcal{A})[x = x]$ recursively realizable.

Proposition 6. (1) *Every prime formula is robust.*

(2) *The class of robust formulas is closed under the propositional operations.*

(3) *Suppose $\varphi(\vec{z}, y)$ is robust, $f : \mathcal{A} \to_r \mathcal{B}$, and let*

$$\varphi(\vec{z}, f(x)) :\equiv (\exists y \in \mathcal{B})[\mathrm{Ap}(f, x, y) \ \& \ \varphi(\vec{z}, y)] \tag{11}$$

where f is a constant naming itself; then $\varphi(\vec{z}, f(x))$ is also robust.

In particular, if f, g are recursive operations, then the formula $f(x) = g(y)$ is robust.

Proof (classical). (1) is trivial and (2) follows by an easy induction on formulas.

To prove (R2) for (3), assume for simplicity that y is the only free variable in $\varphi(y)$ and suppose that $\pi \models (\exists y \in \mathcal{B})[\mathrm{Ap}(f, x, y) \ \& \ \varphi(y)]$, so that for some $b \in \mathcal{B}$,

$$\pi\{y := b\} \models \mathrm{Ap}(f, x, y) \ \& \ \varphi(y).$$

If $a = \pi(x)$, this means that $f(a) = b$; and if $f^* : \mathcal{N} \rightharpoonup \mathcal{N}$ is recursive and computes f and $a = \overline{\overline{\alpha}}$ for any $\alpha \in |\mathcal{A}|$, then $\beta = f^*(\alpha)$ is a code of b which is recursive in α. By the robustness of $\varphi(y)$, there is some ε recursive in β (and hence recursive in α) such that $\varepsilon, \pi\{y := b\} \Vdash^{\mathrm{cr}} \varphi(y)$; and then, directly from the definitions, for any γ,

$$\langle \beta, \langle \gamma, \varepsilon \rangle \rangle \Vdash^{\mathrm{cr}} (\exists y \in \mathcal{B})[\mathrm{Ap}(f, x, y) \ \& \ \varphi(y)],$$

so we get a recursive realizer by plugging any recursive γ in the expression on the left.

The last claim follows by applying (3) twice to the prime formula $u = v$. $\square$

To see how we will use this Proposition in the next section, consider the following:

Corollary 7. *If $\mathcal{A}, \mathcal{B}$ are coded families of subsets of a space $\mathcal{X}$ and they are both recursively closed under countable unions, then the proposition*

"If $(i \mapsto A_i)$ is a countable sequence of sets in $\mathcal{A}$ and each $A_i \in \mathcal{B}$,
then $\cup_i A_i \in \mathcal{B}$" (12)

is recursively C-realizable.

Proof. The displayed proposition is formalized in $\mathcal{L}$ by

$$\chi :\equiv (\forall f : \mathbb{N} \to_c \mathcal{A})\Big[(\forall i \in \mathbb{N})(\exists B \in \mathcal{B})[f(i) = B] \to (\exists C \in \mathcal{B})[\cup^{\mathcal{A}}(f) = C\Big],$$

where $\cup^{\mathcal{A}} : (\mathbb{N} \to \mathcal{A}) \to \mathcal{A}$ is the assumed, recursive countable union operation on $\mathcal{A}$. To compute a realizer for it, suppose we are given a code of some sequence $f : \mathbb{N} \to_c \mathcal{A}$ and a realizer of $(\forall i \in \mathbb{N})(\exists B \in \mathcal{B})[f(i) = B]$; we can compute from these a code of some $g : \mathbb{N} \to \mathcal{B}$ such that for each i, $g(i) = f(i)$; and then $\cup^{\mathcal{B}}(g)$ is a code of the set $C \in \mathcal{B}$ which is needed to realize the conclusion of χ, with $\cup^{\mathcal{B}} : (\mathbb{N} \to_c \mathcal{B}) \to \mathcal{B}$ the given, recursive countable union operation on $\mathcal{B}$. $\square$

Notice that the argument would not work for χ' obtained from χ by the replacement

$$\cup^{\mathcal{A}}(f) = C :\equiv (\forall x \in \mathcal{X})[x \in C \leftrightarrow (\exists i \in \mathbb{N})[x \in f(i)]]$$

which is not (in general) realizable, even when true for the given values of C and f.

Next we consider some sentences which express natural and useful propositions and determine whether they are (recursively) C-realizable.

The Continuous Axiom of Choice is the scheme

$$(\forall x \in \mathcal{A})(\exists y \in \mathcal{B})\varphi(x, y) \rightarrow (\exists f : \mathcal{A} \rightarrow_c \mathcal{B})(\forall x \in A)\varphi(x, f(x)) \qquad (\mathbf{AC}_c)$$

and it is not in general C-realizable, even with prime formulas φ. For example, the sentence

"every $\utilde{\mathbf{\Sigma}}^1_1$ subset of $\mathcal{N}$ is the projection of a tree on $\mathbb{N}^2$"

is effectively true since from a $\utilde{\mathbf{\Sigma}}^1_1$-code of A we can easily compute a code of a tree T on $\mathbb{N}^2$ which projects onto A; so

$$\Vdash^{\text{rcr}} (\forall A \in \utilde{\mathbf{\Sigma}}^1_1)(\exists T \in \mathcal{T}(\mathbb{N}^2))[A = \mathfrak{p}[T]] \quad (A \subseteq \mathcal{N}). \tag{13}$$

On the other hand, there is no $f : \utilde{\mathbf{\Sigma}}^1_1 {\restriction} \mathcal{N} \rightarrow_c \mathcal{T}(\mathbb{N}^2)$ such that for every $A \in \utilde{\mathbf{\Sigma}}^1_1$, $A = \mathfrak{p}[f(A)]]$; because this implies

$$A_\alpha = A_\beta \iff T_{f^*(\alpha)} = T_{f^*(\beta)} \quad (\alpha, \beta \in \mathcal{N})$$

with a continuous f^* in the notation of (CS7), which is impossible, since the relation on the left is complete $\utilde{\mathbf{\Pi}}^1_2$ while that on the right is Π^0_1. This is not the simplest counterexample to the Continuous Axiom of Choice, but it is interesting here because (13) is relevant to our project.[14]

A coded set $\mathcal{A}$ is *extensional* if there is a recursive $g : \mathcal{A} \rightarrow_r \mathcal{N}$ such that for every $a \in \mathcal{A}$, $c_{\mathcal{A}}(g(a)) = a$. The class of extensional coded sets contains $\mathbb{N}$ and $\mathcal{N}$ and it is closed under products and the operations $\mathcal{A} \mapsto \mathcal{A}^{<\omega}, \mathcal{T}_c(\mathcal{A})$.

Theorem 8 (The Extensional Continuous Axiom of Choice).[15] *If $\mathcal{A}$ is extensional, then for every $\mathcal{B}$ and every formula $\varphi(x, y)$,*[16]

$$\Vdash^{\text{rcr}} (\forall x \in \mathcal{A})(\exists y \in \mathcal{B})\varphi(x, y) \rightarrow (\exists f : \mathcal{A} \rightarrow_c \mathcal{B})(\forall x \in \mathcal{A})\varphi(x, f(y)]. \tag{14}$$

[14] If we code the set $\mathcal{R}$ of real numbers in any natural way, then the Archimedean Principle

$$(\forall x \in \mathcal{R})(\exists n \in \mathbb{N})[x < \iota(n)]$$

(with $\iota : \mathbb{N} \rightarrow \mathcal{R}$ the natural imbedding) is effectively true; but every function $f : \mathcal{R} \rightarrow_c \mathbb{N}$ which is continuous in the present sense is also continuous relative to the usual topology of $\mathcal{R}$, and hence constant, so that $(\exists f : \mathcal{R} \rightarrow_c \mathbb{N})(\forall x \in \mathcal{R})[x < \iota(f(x))]$ is not C-realizable.

[15] The converse of this result is also true: if

$$\varepsilon \Vdash^{\text{cr}} (\forall x \in \mathcal{A})(\exists \alpha \in \mathcal{N})[c_{\mathcal{A}}(\alpha) = x] \rightarrow (\exists f : \mathcal{A} \rightarrow \mathcal{N})(\forall x \in \mathcal{A})[c_{\mathcal{A}}(f(x)) = x]$$

and $\{\hat{g}\}(\alpha) = \langle \alpha, \alpha \rangle$, then (easily) $g^*(\alpha) = \{(\{\varepsilon\}(\hat{g}))_0\}(\alpha)$ has the required property.

[16] With $\mathcal{A} = \mathbb{N}$, this is the *Countable Axiom of Choice*, and with $\mathcal{A} = \mathcal{N}$, it expresses various choice-versions of the so-called *Continuity Principle*.

Proof. Suppose (for simplicity) that $\varphi(x, y)$ has no free variables other than x, y, and let $g^* : \mathcal{N} \rightharpoonup \mathcal{N}$ compute some $g : \mathcal{A} \to_r \mathcal{N}$ which witnesses the extensionality of $\mathcal{A}$, so that

$$\alpha \in |\mathcal{A}| \implies g^*(\alpha)\downarrow \ \& \ c_{\mathcal{A}}(\alpha) = c_{\mathcal{A}}(g^*(\alpha)).$$

If β C-realizes the hypothesis of ($\mathbf{AC}_c$), then for each $\alpha \in |\mathcal{A}|$, if $a = c_{\mathcal{A}}(\alpha)$ and $b = c_{\mathcal{B}}((\{\beta\}(\alpha))_0)$, then $(\{\beta\}(\alpha))_1 \Vdash^{\mathrm{cr}} \varphi(a, b)$; and so also,

$$(\{\beta\}(g^*(\alpha)))_1 \Vdash^{\mathrm{cr}} \varphi(c_{\mathcal{A}}(g^*(\alpha)), c_{\mathcal{B}}((\{\beta\}(g^*(\alpha)))_0),$$

but by the given property of g^* now,

$$\begin{aligned} c_{\mathcal{A}}(\alpha) = c_{\mathcal{A}}(\alpha') \implies g^*(\alpha) = g^*(\alpha') \\ \implies c_{\mathcal{B}}(\{\beta\}(g^*(\alpha))_0) = c_{\mathcal{B}}(\{\beta\}(g^*(\alpha'))_0). \end{aligned}$$

This means that the map $\alpha \mapsto (\{\beta\}(g^*(\alpha))_0$ computes a function $f : \mathcal{A} \to \mathcal{B}$ with the required property, and that any ε which satisfies

$$\{\{\varepsilon\}(\beta)\}(\alpha) = \langle(\{\beta\}(g^*(\alpha))_0), (\{\beta\}(g^*(\alpha)))_1\rangle,$$

C-realizes ($\mathbf{AC}_c$). It is easy to get a recursive such ε using (2) of Theorem 1. □

The Extensional Continuous Axiom of Choice is not generally true, and so it is worth putting down the axioms of *Countable* and *Dependent Choices* which are; the first of these is a corollary of Theorem 8, and the second is easy.

Theorem 9 ($\mathbf{AC}_{\mathbb{N}}$, $\mathbf{DC}$)**.** *For every coded set $\mathcal{A}$ and every $\varphi(n, x)$,*

$$\Vdash^{\mathrm{rcr}} (\forall n \in \mathbb{N})(\exists x \in \mathcal{A})\varphi(n, x) \to (\exists f : \mathbb{N} \to_c \mathcal{A})(\forall n \in \mathbb{N})\varphi(n, f(n))$$
$$\Vdash^{\mathrm{rcr}} (\forall x \in \mathcal{A})(\exists y \in \mathcal{A})\varphi(x, y) \to (\exists f : \mathbb{N} \to_c \mathcal{A})(\forall n \in \mathbb{N})\varphi(f(n), f(n+1))$$

The next result is also easy and we will skip the proof, but it is important:

Theorem 10 (Markov's Principle)**.** *For every formula $\varphi(n)$,*

$$\Vdash^{\mathrm{rcr}} \Big\{(\forall n \in \mathbb{N})[\varphi(n) \vee \neg\varphi(n)] \ \& \ \neg(\forall n \in \mathbb{N})\varphi(n)\Big\} \implies (\exists n \in \mathbb{N})\neg\varphi(n).$$

Finally we verify the recursive C-realizability of a very general principle of proof by induction on a wellfounded relation, which is the main tool that we will use in the next section:

Theorem 11 (Proof by wellfounded induction)**.** *For each wellfounded relation $\prec \ \subseteq \mathcal{A} \times \mathcal{A}$ and each formula $\varphi(x)$,*

$$\Vdash^{\mathrm{rcr}} (\forall x \in \mathcal{A})\Big[(\forall y \in \mathcal{A})[y \prec x \to \varphi(y)] \to \varphi(x)\Big] \to (\forall x \in \mathcal{A})\varphi(x). \tag{15}$$

Proof. Assume again that $(\forall x \in \mathcal{A})\varphi(x)$ is a sentence. Using the convention (2) to simplify notation, we need to define a recursive $\varepsilon \in \mathcal{N}$ such that for every γ,

$$(\forall \alpha \in |\mathcal{A}|)\Big[\{\gamma\}(\alpha) \Vdash^{\mathrm{cr}} (\forall y \in \mathcal{A})[y \prec \overline{\overline{\alpha}} \rightarrow \varphi(y)] \rightarrow \varphi(\overline{\overline{\alpha}})\Big] \\ \Longrightarrow (\forall \alpha \in |\mathcal{A}|)\Big[\{\{\varepsilon\}(\gamma)\}(\alpha) \Vdash^{\mathrm{cr}} \varphi(\overline{\overline{\alpha}})\Big]. \quad (16)$$

Fix some γ which satisfies the hypothesis of (16). We will prove its conclusion by induction on $\overline{\overline{\alpha}}$, for an ε that we will define at the end of the argument (by appealing to the Second Recursion Theorem 2) when we will need it to justify the last step of the argument.

By the induction hypothesis, if $\overline{\overline{\beta}} \prec \overline{\overline{\alpha}}$, then, for any δ,

$$f_0(\varepsilon, \gamma, \beta, \delta) = \{\{\varepsilon\}(\gamma)\}(\beta) \Vdash^{\mathrm{cr}} \varphi(\overline{\overline{\beta}});$$

and so if $\hat{f}_0$ is any (recursive) code of f_0 and

$$f_1(\varepsilon, \gamma, \beta) = S(\hat{f}_0, \varepsilon, \gamma, \beta),$$

then $\{f_1(\varepsilon, \gamma, \beta)\}(\delta) \Vdash^{\mathrm{cr}} \varphi(\overline{\overline{\beta}})$ for every $\overline{\overline{\beta}} \prec \overline{\overline{\alpha}}$, so that

$$f_1(\varepsilon, \gamma, \beta) \Vdash^{\mathrm{cr}} \overline{\overline{\beta}} \prec \overline{\overline{\alpha}} \rightarrow \varphi(\overline{\overline{\beta}}).$$

It follows that if $\hat{f}_1$ is a recursive code of f_1 then

$$f_2(\varepsilon, \gamma) = S(\hat{f}_1, \varepsilon, \gamma) \Vdash^{\mathrm{cr}} (\forall y \in \mathcal{A})[y \prec \overline{\overline{\alpha}} \rightarrow \varphi(y)],$$

and so by the hypothesis on γ,

$$f_3(\varepsilon, \gamma, \alpha) = \{\{\gamma\}(\alpha)\}(f_2(\varepsilon, \gamma)) \Vdash^{\mathrm{cr}} \varphi(\overline{\overline{\alpha}}).$$

So far we have not used any properties of ε, except for the induction hypothesis. Now fix a (recursive) code $\hat{f}_3$ of f_3 and choose ε by the Second Recursion Theorem so that it is recursive and

$$\{\varepsilon\}(\gamma) = S(\hat{f}_3, \varepsilon, \gamma);$$

so $\{\{\varepsilon\}(\gamma)\}(\alpha) = f_3(\varepsilon, \gamma, \alpha) \Vdash^{\mathrm{cr}} \varphi(\overline{\overline{\alpha}})$, which is what we needed to show. ∎

The proof did not depend on any assumptions on the relation $\prec$ other than that it is wellfounded, and so its constructiveness (and consequently the constructive truth of (15)) may be in dispute.[17] In any case, what we need is an "internal" version of this result which justifies proof by backward (bar) induction on a tree with no infinite paths, as follows:

[17]It seems to me that the proof of Theorem 11 is constructive, provided that by "$\prec$ is wellfounded" we understand precisely that propositions can be proved by induction along $\prec$.

Theorem 12 (Proof by bar induction)**.** *For each formula* $\varphi(u)$*,*

$$\Vdash^{\mathrm{rcr}} \Big\{T \in \mathcal{T}_c(\mathcal{A})\ \&\ \neg(\exists f : \mathbb{N} \to \mathcal{A})(\forall n)[f \restriction n \in T] \\ \&\ (\forall u \in T)\Big[(\forall s)[u \star (s) \in T \to \varphi(u \star (s))] \to \varphi(u)\Big]\Big\} \\ \to (\forall u \in T)\varphi(u),$$

where n *varies over* $\mathbb{N}$ *and* $(\forall u \in T)\psi :\equiv (\forall u \in)^{<\omega}\mathcal{A}[u \in T \to \psi]$*.*

Proof. The idea is that if the hypothesis of the implication to be proved is C-realizable for a given (value of the variable) T, then there are no infinite paths through T and so the relation

$$u \prec v \iff u \in T\ \&\ (\exists s \in \mathcal{A})[u = v \star (s)]$$

is wellfounded—which then allows us to construct a γ' that C-realizes the conclusion by a notational variant of the proof of Theorem 11.[18] □

4. The Separation Theorem for $\underset{\sim}{\mathbf{\Sigma}}{}^1_1$ and Suslin's Theorem

Let $\mathbb{T}^r$ be the intuitionistic theory with all true, recursively C-realizable sentences as axioms. A proof of a sentence χ in $\mathbb{T}^r$ guarantees that χ is true, but also recursively C-realizable by the basic Theorem 5. It is often much easier to construct than a direct, messy definition of some ε which C-realizes χ—especially as we can often give informal constructive proofs skipping the (well understood) process of their formalization.

Proofs in $\mathbb{T}^r$ are a bit hard to explain, because they include "subroutines" where we need to show, perhaps classically, that certain key propositions are recursively realizable. We will try to mark clearly the points when we switch between these two, different kinds of argument. In any case, we can appeal to all the numbered theorems in the preceding section except for 8, the Extensional Axiom of Continuous Choice, as well as to Proposition 6, which is an important tool.

Let us first prove in $\mathbb{T}^r$ the "easy part" of Suslin's Theorem.

Theorem (in $\mathbb{T}^r$) 13. *Every Borel subset of a product space* $\mathcal{X}$ *is* $\underset{\sim}{\mathbf{\Delta}}{}^1_1$*.*

Proof. With the notation of (CS8), let for $A, B \in \mathbf{B}\restriction\mathcal{X}$:

$$A \prec B \iff (\exists\xi)[A \in \underset{\sim}{\mathbf{\Sigma}}{}^0_\xi\ \&\ B \notin \underset{\sim}{\mathbf{\Sigma}}{}^0_\xi] \quad (A, B \subseteq \mathcal{X})$$

[18] The inference from $\neg(\exists f : \mathbb{N} \to \mathcal{A})(\forall n)[f \restriction n \in T]$ to "*we can prove propositions by induction along* $\prec$" is not obviously valid constructively. Some intuitionists accept it if T is a tree on $\mathbb{N}$ and we replace $\neg(\exists f)(\forall n)[f \restriction n \in T]$ by $(\forall f : \mathbb{N} \to \mathcal{A})(\exists n)[f \restriction n \notin T]$, which is how we will apply this result. The question is moot in the presence of Markov's Principle which implies the equivalence of these two properties of T when $T \in \mathcal{T}(\mathbb{N})$, and which we can use, since it is recursively C-realizable. In any case, we only claim a classical proof of Theorem 12.

and verify (outside $\mathbb{T}^r$) that this is a wellfounded relation on the Borel subsets of $\mathcal{X}$. So Theorem 11 applies, and it is enough to prove (in $\mathbb{T}^r$) that

$$(\forall A \in \mathbf{B})\Big[(\forall B \in \mathbf{B})[B \prec A \rightarrow B \in \underset{\sim}{\mathbf{\Delta}}{}^1_1] \rightarrow [A \in \underset{\sim}{\mathbf{\Delta}}{}^1_1]\Big]. \tag{17}$$

The key observation is that

$$\begin{aligned}\Vdash^{\mathrm{rcr}} &\text{“for each } A \in \mathbf{B}\text{, either } A \text{ is open,}\\ &\quad \text{or } A = \textstyle\bigcup_i(\mathcal{N} \setminus A_i) \text{ for a sequence of Borel sets such that } A_i \prec A\text{”;}\end{aligned} \tag{18}$$

this is true because if α is a Borel code of A, then A is open if $\alpha(0) = 0$ (and α^* gives us an open code for it), or A is a union of Borel sets of “lower order” if $\alpha(0) = 1$, and then α^* gives us a code of a sequence $i \mapsto A_i$ with the required property. Using this, we can verify (17) (in $\mathbb{T}^r$) by taking cases on A and using the closure properties of $\underset{\sim}{\mathbf{\Delta}}{}^1_1$ of Lemma 4 as in Corollary 7. $\square$

For the converse, more difficult direction of Suslin's Theorem, it is convenient to deal only with subsets of $\mathcal{N}$, from which the general result can be easily derived by standard methods. We show first the following, stronger result:[19]

Theorem (in $\mathbb{T}^r$) 14 (Separation Theorem)**.** *If A, B are disjoint, $\underset{\sim}{\mathbf{\Sigma}}{}^1_1$ subsets of $\mathcal{N}$, then there exists a Borel set $C \subseteq \mathcal{N}$ which separates A from B, i.e.,*

$$(\forall\alpha \in \mathcal{N})[\alpha \in A \implies \alpha \in C] \text{ and } \neg(\exists\alpha \in \mathcal{N})[\alpha \in C \ \&\ \alpha \in B]. \tag{19}$$

Proof.[20] It is perhaps simplest to treat the separation relation as a primitive,

$$\mathrm{Sep}(A, B, C) \iff C \text{ separates } A \text{ from } B,$$

so we do not need to go into its definition in the constructive part of the proof, but notice that

$$\Vdash^{\mathrm{rcr}} \mathrm{Sep}(A, B, C) \leftrightarrow (\forall\alpha \in \mathcal{N})[\alpha \in A \rightarrow \alpha \in C] \ \&\ \neg(\exists\alpha \in \mathcal{N})[\alpha \in C \ \&\ \alpha \in B].$$

With this understanding, the key is the following simple

Lemma. If for all $i, j \in \mathbb{N}$, A_i, B_j are $\underset{\sim}{\mathbf{\Sigma}}{}^1_1$ subsets of $\mathcal{N}$ and $C_{i,j}$ is a Borel set which separates A_i from B_j, then $\bigcup_i \bigcap_j C_{i,j}$ separates $\bigcup_i A_i$ from $\bigcup_j B_j$.

It is quite easy and we will skip its proof.[21]

[19] Aczel (2009) proves a version of the Separation Theorem in constructive set theory by analyzing the same proof of Lusin with which I am working here. The result is both interesting and obviously relevant, but a precise comparison is difficult.

[20] This argument is somewhat different from the proof of Theorem 2E.1 in Moschovakis (2009a), which uses definition by *bar recursion* rather than proof by *bar induction.* (That argument can also be formalized in $\mathbb{T}^r$, but it takes some extra work to formulate the principle of definition by bar recursion and show that it is recursively C-realizable.)

[21] The Lemma holds for arbitrary A_i, B_j, of course. The restriction to $\underset{\sim}{\mathbf{\Sigma}}{}^1_1$ subsets of $\mathcal{N}$ is put in only so that we can formulate it in $\mathcal{L}$, using the recursive operations of countable union and countable intersection in $\underset{\sim}{\mathbf{\Sigma}}{}^1$ and $\underset{\sim}{\mathbf{\Delta}}{}^1_1$, as in Corollary 7.

Now given two disjoint $\underset{\sim}{\mathbf{\Sigma}}^1_1$ sets $A, B \subseteq \mathcal{N}$, choose by (13) trees T, S on $\mathbb{N}^2$ such that

$$A = \mathfrak{p}[T], \quad B = \mathfrak{p}[S],$$

and let

$$J = \{((t_0, \xi_0, \eta_0), \ldots, (t_{n-1}, \xi_{n-1}, \eta_{n-1})) \\ \mid ((t_0, \xi_0), \ldots, (t_{n-1}, \xi_{n-1})) \in T \ \& \ ((t_0, \eta_0), \ldots, (t_{n-1}, \eta_{n-1})) \in S\}.$$

There is no infinite branch through J: because if $(f \star g \star h) \in [J]$ with

$$(f \star g \star h)(i) = (f(i), g(i), h(i)),$$

then $(f \star g) \in [T]$ and $(f \star h) \in [S]$, so that $f \in A \cap B$, contradicting the hypothesis. So we can prove propositions by bar induction on J.

For each $u = ((t_0, \xi_0, \eta_0), \ldots, (t_{n-1}, \xi_{n-1}, \eta_{n-1})) \in \mathbb{N}^n$, let

$$\sigma(u) = ((t_0, \xi_0), \ldots, (t_{n-1}, \xi_{n-1})), \quad \tau(u) = ((t_0, \eta_0), \ldots, (t_{n-1}, \eta_{n-1})),$$

so that

$$u \in J \iff \tau(u) \in T \ \& \ \sigma(u) \in S,$$

and, in the notation of (CS5), set

$$A_u = \mathfrak{p}[T_{\tau(u)}], \quad B_u = \mathfrak{p}[S_{\sigma(u)}],$$

and check (easily) that

$$A_u = \bigcup_{t,\xi} \mathfrak{p}[T_{\tau(u)\star(t,\xi)}], \quad B_u = \bigcup_{s,\eta} \mathfrak{p}[S_{\sigma(u)\star(s,\eta)}].$$

The proposition that we will prove by bar induction is

$$(\forall u \in J)[\text{there exists a Borel set } C_u \text{ which separates } A_u \text{ from } B_u].$$

In view of the Lemma, it is then enough to show that if $u \in J$, then

> *for each tuple (t, ξ, s, η), there is a Borel set D which separates $\mathfrak{p}[T_{\tau(u)\star(t,\xi)}]$ from $\mathfrak{p}[S_{\sigma(u)\star(s,\eta)}]$;*

if we can do this, then the Countable Axiom of Choice will give us a function

$$(t, \xi, s, \eta) \mapsto D_{t,\xi,s,\eta}$$

such that $D_{t,\xi,s,\eta}$ separates $\mathfrak{p}[T_{\tau(u)\star(t,\xi)}]$ from $\mathfrak{p}[S_{\sigma(u)\star(s,\eta)}]$, and then the set

$$C_u = \bigcup_{t,\xi} \bigcap_{s,\eta} D_{t,\xi,s,\eta}$$

will separate A_u from B_u as required.

To show the claim in italics, given (t, ξ, s, η):

(1) If $t \neq s$, take $D = \{\alpha \mid \alpha(n) = t\}$, where $n = \mathrm{lh}(u)$.

(2) If $t = s$ and $\tau(u) \star (t, \xi) \notin T$, take $D = \emptyset$.

(3) If $s = t$ and $\sigma(u) \star (s, \eta) \notin S$, take $D = \mathcal{N}$.

(4) If $s = t$ and $\tau(u)\star(t, \xi) \in T$ and $\sigma(v)\star(s, \eta) \in S$, then $u\star(t, \xi, \eta) \in J$, and the induction hypothesis guarantees the existence of some D with the required property. $\square$

Corollary (in $\mathbb{T}^r$) 15 (Suslin's Theorem)**.** *Every $\underset{\sim}{\mathbf{\Delta}}^1_1$ subset of $\mathcal{N}$ is Borel.*

Proof. The sentence

$$(\forall A \in \underset{\sim}{\mathbf{\Delta}}^1_1)[A \in \underset{\sim}{\mathbf{\Sigma}}^1_1 \ \& \ (\mathcal{N} \setminus A) \in \underset{\sim}{\mathbf{\Sigma}}^1_1]$$

is recursively C-realizable, because of the coding of $\underset{\sim}{\mathbf{\Delta}}^1_1 \restriction \mathcal{N}$. By the Separation Theorem, there is a Borel set C such that $A \subseteq C$ and $C \cap (\mathcal{N} \setminus A) = \emptyset$, and so the proof will be complete if we can show

$$\Vdash^{\text{rcr}} \Big[(\forall \alpha \in \mathcal{N})(\alpha \in A \to \alpha \in C) \ \& \ \neg(\exists \alpha \in \mathcal{N})(\alpha \in C \ \& \ \alpha \in (\mathcal{N} \setminus A))\Big] \to A = C.$$

The (classical) proof of this is direct. ☐

5. Comments and questions

Briefly, what we have done is to

(1) introduce a formal language $\mathcal{L}$ in which many of the propositions of classical Descriptive Set Theory can be naturally expressed;

(2) define a realizability interpretation for $\mathcal{L}$ which is respected by intuitionistic deductive reasoning; and

(3) derive the (effective) Suslin-Kleene Theorem by verifying that the (classical) Suslin Theorem can be proved intuitionistically from true, recursively C-realizable hypotheses.

It has been argued, moreover, that the same method can be used to prove the effective (uniform) versions of many classical results of descriptive set theory.

Is this a useful technical tool? The answer is clearly negative for those who are not already familiar with intuitionistic logic—there is just too much overhead. For those who know enough of it, however, so that they can recognize whether an informal argument can be formalized in intuitionistic logic, it can, indeed, be a powerful tool. At least for discovering results—which can then be proved by *effective transfinite induction*, as Kleene called it, meaning that the relevant applications of Theorem 11 are checked one-by-one, by separate applications of the Second Recursion Theorem. This is the method used for many, basic results in Chapter 7 of Moschovakis (2009a) and in much of Kleene's (and many others') work on constructive ordinals and hyperarithmetical sets, cf. Moschovakis (2010).

Is $\mathbb{T}^r$ a refinement of effective descriptive set theory? This was the claim in the introduction, that the classical theory, properly understood, "yields the uniformities which refine the statements of the classical results, and it also provides constructive proofs that they do". Well, a proof in $\mathbb{T}^r$ is exactly as constructive as the proofs of the recursive C-realizability of the axioms it uses, and these axioms fall (roughly) in three categories.

First, there are statements like (18) which are blatantly non-constructive on their face, but whose recursive C-realizability is a direct (and evidently constructive) consequence of our choice of codings. I think that these are innocuous;

and the choice of codings which make these basic propositions obvious is at the heart of the applicability of the method.

At the other extreme are statements like those in Lemmas 3, 4 and Proposition 6, where the proofs of C-realizability are blatantly classical. For example, the claim that $(A^c)^c = A$ in the proof of Lemma 4 is (ultimately) justified because

$$\Vdash^{\text{rcr}} (\forall x \in \mathcal{A})(\forall y \in \mathcal{B})[x \in y \leftrightarrow \neg\neg(x \in y)]. \tag{20}$$

This is a (classically) trivial consequence of our identifying C-realizability with satisfiability for prime formulas, and there is no getting around it without changing the method radically.

Somewhere between these two extremes are statements like Theorem 11, the basic Principle of Proof by Wellfounded Induction, where reasonable people may reasonably differ on whether the proof of its C-realizability is constructive. (And they are likely to fall in several camps, according to the specific wellfounded relation $\prec$.)

Taken with all its axioms no matter how they are established, $\mathbb{T}^r$ is obviously not a constructive theory. It has, however, the "constructive feature" that it can only prove effectively true $\forall$-$\exists$ sentences—but I don't have a good name for this property of a theory.

What were they thinking? Borel and Lebesgue and Lusin and the others. They thought of themselves as *constructivists*—and in some cases they say that, explicitly; they rejected (also explicitly) some blatant applications of the Axiom of Choice—like the existence of a function which assigns to each countable ordinal ξ a wellordering in $\mathbb{N}$ with order type ξ; but they accepted the Countable Axiom of Choice, the De Morgan rules for countable unions, and (I believe, though I do not have a reference) the stability of membership (20). Moreover, from reading them, one gets the feeling that their conception of *sets* and what one does with them was robust, not the object of introspection or doubt. (Lusin, for example, had apparently expressed the belief that co-analytic sets cannot be "constructively" uniformized, but he accepted Novikoff's proof of it even though it is by no means elementary. The Novikoff-Kondo proof of $\boldsymbol{\Pi}^1_1$ uniformization can be given in $\mathbb{T}^r$, of course, and so it yields its effective version about Π^1_1.)

I find the problem of understanding better the *universe* of classical descriptive set theory and the *logic* of its practitioners fascinating, both historically and (especially) mathematically, and it is my hope that the considerations of this article may be relevant to it.

Acknowledgents. I thank Peter Aczel, Jeremy Avigad, Thierry Coquand, Ulrich Kohlenbach, Andre Scedrov, Jaap van Oosten, Michael Rathjen and Wim Veldman, for useful comments on an early draft of this article, and I am especially grateful to my wife Joan Rand Moschovakis for listening patiently and responding usefully to many questions (some of them not very coherent) on a subject that she knows well and I do not. However, none of these kind people vetted the final version, and it should not be assumed that any of them is

sympathetic to any "foundational views" that are expressed in it, explicitly or implicitly.

References

Aczel, P., 2009. A constructive version of the Lusin Separation Theorem. In: Logicism, Intuitionism and Formalism – What has become of them? Vol. 341 of Synthese Library. Springer, pp. 129–151.

Kleene, S. C., 1955a. Arithmetical predicates and function quantifiers. Transactions of the American Mathematical Society 79, 312–340.

Kleene, S. C., 1955b. Hierarchies of number theoretic predicates. Bulletin of the American Mathematical Society 61, 193–213.

Kuratowski, K., 1966. Topology. Vol. 1. Academic Press, New York and London, translated from the French *Topologie*, vol. 1, PWN, Warsaw, 1958.

Moschovakis, Y. N., 1971. Classical descriptive set theory as a refinement of modern hierarchy theory, abstract #690-e10. Notices of the American Mathematical Society, 1088.

Moschovakis, Y. N., 2009a. Descriptive set theory, Second edition. Vol. 155 of Mathematical Surveys and Monographs. American Mathematical Society.

Moschovakis, Y. N., 2009b. Kleene's amazing second recursion theorem (extended abstract). In: Grädel, E., Kahle, R. (Eds.), CSL 2009. No. 5771 in Lecture Notes in Computer Science. Springer-Verlag, Berlin - Heidelberg, pp. 24–39.

Moschovakis, Y. N., 2010. Kleene's amazing second recursion theorem. The Bulletin of Symbolic Logic 16, 189 – 239.

Rathjen, M., 2005. Constructive Set Theory and Brouwerian Principles. Journal of Universal Computer Science 11, 2008–2033.

Suslin, M., 1917. Sur une definition des ensembles measurables B sans nombres transfinis. Comptes Rendus Acad. Sci. Paris 164, 88–91.