A higher order perspective on computational analysis

Dag Normann
The University of Oslo
Department of Mathematics

(based on jww Sam Sanders)

08.09.2023

My start

- In March 2015 Sam Sanders visited Oslo. In the e-mail exchange following this visit he sent me the specifications for a set of type 3 functionals, weird-looking functionals named Θ, and he asked me if I could say anything sensible about such functionals Θ.
- After a while we realised (after a rewriting of the specifications) that each Θ will be a realiser of the Heine Borel Theorem for the Cantor space $\mathbf{C} = 2^{\mathbb{N}}$, that no Θ can be computable in a type 2 functional, and that they are not of a kind that had been systematically studied by people working with computability aspects of higher type functionals until then.

My motivation

The existence of a vast unexplored territory of functionals of type 3 between the explored territories of the normal functionals and of the continuous (countable) functionals, but potentially containing objects with a natural appearance in ordinary mathematics, appealed to me.

Structure of the talk

The talk will consist of three parts

Part 1: Aims and tools

Part 2: “Old" results

Part 3: Partial subfunctions of $\exists^3$

Aims and tools

Nornal/continuous

- Normal functionals F of type 3 are functionals that consider the continuum $\mathbb{N}^{\mathbb{N}}$ as finite, i.e. consider the quantifier $\exists f \in \mathbb{N}^{\mathbb{N}}$ as computable.
- The countable or continuous functionals are functionals that, in a natural way, have codes that are elements of $\mathbb{N}^{\mathbb{N}}$.
- They come in two forms, the extensional type structure introduced by Kreisel (1959) and the non-extensional substructure of the full structure of pure types, as introduced by Kleene (1959).

True CA - a disclaimer

- The coding of objects appearing in ordinary mathematics as subsets of $\mathbb{N}$ or as functions from $\mathbb{N}$ to $\mathbb{N}$ is essential if one wants to study how digital computers, or Turing machines with oracles, can deal with these objects.
- Thus, even though someone might think that what is known as the Normann-Sanders (N.- S.) project is challenging the current view on RM, no one should think the same about CA.
- The computability aspects in the N.- S.-project represent applications of generalized computability theory, and not of the original Turing concept.

From object to code, an example

- If O is an open set in $\mathbb{R}$, O will be the union of a set of open intervals with rational endpoints, and thus, definable from an enumerated set that we call the code.
- O itself is an object of type 2, while the code is an object of type one.
- Thus there is a partial functional of type 3 mapping an open set to its code.
- Towards the end of the talk we will say a few sensible words about the complexity of this functional, in terms of higher order computability theory.
- There is actually an open problem related to those sensible words.

Computing with functionals

- The set Tp(n) of functionals of pure finite type n are defined by recursion as follows
  1. Tp(0) = $\mathbb{N}$
  2. Tp($n+1$) is the sets of all functions mapping Tp(n) into $\mathbb{N}$.
- In 1959 Kleene published a computability model involving algorithms

$$\{e\}(\Phi_1, \ldots, \Phi_n)$$

  with possible values in $\mathbb{N}$.
- His motivation seems to have been to develop a tool for understanding a transfinite use of higher order quantifiers, generalising the step from arithmetical to hyperarithmetical.

Choice of computability model

- In contrast to computations involving integers, finite words or streams of such, there is no Church-Turing thesis for computing with higher order inputs.
- Following Kreisel, generalising computability theory involves to generalise the concept of being finite.
- In order to use a generalised concept of computability to study relative complexity, we must agree on what is to be considered as non-complex.
- Using Kleene's model this means to decide which oracles we are free to use. The other dimension will be how much of Kleene's model we accept as unproblematic.

The Kleene schemes

- Kleene's definition is a grand monotone inductive definition of the relation

$$\{e\}(\Phi_1, \ldots, \Phi_n) = a$$

via nine clauses known as the schemes S1 - S9.

- The schemes introduce basic computations relative to type 1 oracles, together with higher order application and enumeration:

S8 $\{e\}(\Phi^{n+2}, \vec{\Psi}) \simeq \Phi(\lambda\phi^n.\{d\}(\phi, \Phi, \vec{\Psi}))$

S9 $\{e\}(d, \vec{\Psi}, \vec{\Phi}) \simeq \{d\}(\vec{\Psi})$

Modified Kleene

- In N.- S. '22 we modify Kleene's definition to cater for the needs of our project.
- We abandon the restriction to pure types, and we allow partial functionals of type 3 as oracles, cfr. the coding functional only defined for (characteristic functions of) open sets.
- We express the computable functionals as terms in a kind of lambda-calculus with a least fixed point operator instead of using S9.
- We do NOT add anything to the computational power of S1 - S9, inputs must be total while oracles may be partial.

Alternative models

- Gödel's T, for higher order primitive recursion, is a less powerful computability model, so using Gödel instead of Kleene for positive results is better.
- In CiE 2019 Welch introduced Infinite Time Turing Machines with oracles of type 2 (ITTMs). These are more powerful than Kleene's algorithms, so using Welch instead of Kleene for negative results is better.
- We have observed a kind of dichotomy in our results: For positive results we mostly need a tiny fragment of the Kleene model, while negative results can be proved for the full model.

Coding the reals

- We still need to represent mathematical objects as some objects our computability model can handle, but preferably by adding as little information as possible.
- We represent rational numbers as integers in some standard way, and we represent reals as fast converging sequences of rationals, with the canonical Π^0_1 equivalence relation on the set of representations.
- A function $\Phi : \mathbb{R} \to \mathbb{R}$ is then represented by a function $F : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ respecting this equivalence.
- We leave out the details.

Back to Θ and Heine -Borel

In 1895 Borel published a paper where he proved the following:

Theorem

Given a way to associate an open neighbourhood to each point in a closed interval $[a, b]$*, we can construct a finite subcovering of the induced covering.*

His proof made use of Cantor's second number system, i.e. the countable ordinals, and can be seen as an early instance of a non-monotone inductive definition.

- As we see it today, he defined, using set theory, a functional taking a point-wise covering as an argument and providing a finite subcovering as the value.
- We have named the corresponding functional defined for point-wise coverings of the Cantor space, for Borel's Θ.

Back to Θ and Heine - Borel

- In general, the specification for a functional Θ is as follows, for inputs F of type 2:

1. If $f : \mathbb{N} \to \{0, 1\}$, then $F(f) = n$ defines a neighbourhood

$$\mathbf{C}_{F,f} = \{g \in \mathbf{C} : \forall m < n(g(m) = f(m))\}.$$

2. $\Theta(F)$ will be a coded finite sequence $\langle f_1, \ldots, f_n \rangle$ such that the open neighbourhoods $\mathbf{C}_{F,f_i}$ cover $\mathbf{C}$.

- No Θ will be computable in a functional of type 2.

More on Θ

- A function $F : \mathbf{C}^2 \to \mathbb{N}$ can be seen as a parameterised family of functions F_g, and thus as a parameterised family of open coverings.

Theorem

There is a Borel-measurable function $F : \mathbf{C}^2 \to \mathbb{N}$ such that there is no Borel-measurable way to select a finite sub-covering of the covering induced from each F_g as a function of $g \in \mathbf{C}$.

Continuous vs. discontinuous functionals

- It does not make sense to restrict our attention to continuous functions, in particular since the early researchers made a point of considering *functions in the most general sense of the term* (Cfr. Pincherle 1882.)
- The characteristic function of an open subset of $[0,1]$ is not, with two exceptions, continuous.
- The prototype of a discontinuous function is

$$\exists^2(f) = \begin{cases} 1 & \text{if} \quad \exists n(f(n) > 0) \\ 0 & \text{if} \quad \forall n(f(n) = 0) \end{cases}$$

- $\exists^2$ is the characteristic function of a complete Σ^0_1-subset of $\mathbb{N}^{\mathbb{N}}$.

$\exists^2$ and discontinuous functionals

- If F is effectively discontinuous, then $\exists^2$ is computable in F.
- Relativisation gives us that if F is discontinuous, then $\exists^2$ is computable in F and some $f \in \mathbb{N}^{\mathbb{N}}$.
- As a consequence, we use $\exists^2$ freely as an oracle in our investigations, often without mentioning.
- Intuitively we consider quantification over enumerable sets as unproblematic, while when quantifiers over the continuum are used, a closer analysis of the complexity is due.
- An extra bonus is that we often do not need to distinguish between **C** and e.g. $[0, 1]$.

The Suslin functional

- The characteristic function of a complete Σ^1_1 subset of $\mathbb{N}^{\mathbb{N}}$, defined as follows

$$\mathbf{S}^2_1(f) = \begin{cases} 1 & \text{if} \quad \exists g \forall n (f(\bar{g}(n)) = 0) \\ 0 & \text{if} \quad \forall g \exists n (f(\bar{g}(n)) > 0) \end{cases}$$

  is known as the *Suslin functional*.
- In general, we may consider the corresponding functionals $\mathbf{S}^2_k$, but we only need those for $k > 2$ in constructing two models for Kohlenbach's type theory satisfying the axioms of full second order arithmetic, but failing a number of easy mathematical facts involving objects of order three.

$\exists^3$

- $\exists^3$ is the functional that makes all constructions in analysis computable.
- $$\exists^3(F) = \begin{cases} 1 & \text{if} \quad \exists f(F(f) > 0) \\ 0 & \text{if} \quad \forall f(F(f) = 0) \end{cases}$$
- When we discover a functional, appearing in nature, that computes $\exists^3$, there is no need for a further analysis of its complexity.

PART 2

“Old" results

Non-monotone induction

- We mentioned that Borel's proof from 1895 used a kind of non-monotone induction.
- We view non-monotone induction as a functional $\mathcal{I}$ of type 3 as follows:
- If $F : 2^{\mathbb{N}} \to 2^{\mathbb{N}}$, the iteration of F is defined to be the sequence f_α of functions where

$$f_\alpha(n) = \sup\{F(f_\beta)(n) : \beta < \alpha\}$$

- $\mathcal{I}(F) = f_\alpha$ for the least α such that $f_{\alpha+1} = f_\alpha$.

Non-monotone induction

Theorem

a) $\mathcal{I}$ *is computable in an* ITTM, *but does not compute* ITTM*s in general.*

b) $\mathcal{I}$ *is computable in* $\mathbf{S}^2_1$ *and any* Θ.

c) $\mathbf{S}^2_1$ *is computable in* $\exists^2$ *and Borel's* Θ.

Consequently, $\mathcal{I}$ *and Borel's* Θ *are equivalent, given* $\exists^2$.

d) *For each* Θ, *there is a function* f *of type 1 that is computable in* Θ *and* $\exists^2$, *but that is not hyperarithmetical.*

e) *If a function* f *is uniformly computable in* $\exists^2$ *and all instances of* Θ, *then* f *is computable in* $\exists^2$, *i.e. hyperarithmetical.*

The power of Non-monotone induction

$\mathcal{I}$ can handle many computational tasks that will require genuinely type 3 operations. Some examples are

- $\Phi_{\mathrm{NIN}}(F) = \langle f_1, f_2 \rangle$ where $f_1 \neq f_2$ while $F(f_1) = F(f_2)$. (NIN-realisers).
- $\Phi_{\mathrm{Baire}}(\langle O_n \rangle_{n\in\mathbb{N}}) \in \bigcap_{n\in\mathbb{N}} O_n$ whenever each O_n is open and dense.
- If $F : \mathbf{C} \to \mathbb{N}$ and $\mathbf{C}_{F,f}$ is the neighbourhood of $f \in \mathbf{C}$ as before, then $\Lambda(F)$ will be a sequence $\langle f_n \rangle_{n\in\mathbb{N}}$ where $\bigcup_{n\in\mathbb{N}} \mathbf{C}_{F,f_n}$ has measure 1.

Λ captures exactly the complexity of the Vitali Covering Theorem, modulo $\exists^2$, while Θ captures the complexity of the Vitali Covering Lemma (for **C**).

The power of Non-monotone induction

- Each Θ computes a Λ, but the converse is not the case.
- Each Λ computes a $\Phi_{\rm NIN}$. We conjecture that the converse is not the case, but have no proof.
- In honor of Pincherle's Boundedness Theorem from 1882 we also considered functionals M where $M(F)$ will be a uniform upper bound of all functions $\phi : \mathbf{C} \to \mathbb{R}$ satisfying that ϕ is bounded by $2^{F(f)}$ on $\mathbf{C}_{F,f}$ for each f.
  Each Θ computes an M, and there is a Λ that cannot compute an M, but the remaining relations are open.

Countably based functionals

- All functionals we have considered so far, included those that are computable using an ITTM, are, what is called, countably based.
- The countably based functionals are almost of type 2, they can be coded by type 2 objects in a natural way.
- In contrast, John Hartley showed, assuming the continuum hypothesis, that if a functional Φ is NOT countably based, then $\exists^3$ is computable in Φ and some type 2 functional F.
- So, to decide if a functional Φ, that grows out of some natural construction in ordinary mathematics, is countably based or not is like deciding if one can do with less than the full continuum in constructing the functional.

Countably based functionals

We slightly generalise the definition originally due to Stan Wainer:

Definition

A partial functional Φ of type $(\mathbb{N}^{\mathbb{N}} \to \mathbb{N}) \to \mathbb{N}$ is countably based if whenever F is total and $\Phi(F)$ is defined there is a countable set $X \subset \mathbb{N}^{\mathbb{N}}$ (the support of $\Phi(F)$) such that if G is total, $F(f) = G(f)$ for all $f \in X$ and $\Phi(G)$ is defined, then $\Phi(F) = \Phi(G)$.

We view all total functionals of type 2 to be countably based as well. The class of countably based functionals is closed under relative computability with respect to Kleene's model.

Originally, the countably based functionals were seen as a generalisation of the continuous functionals of finite type, cfr. the contributions by John Hartley.

Partial subfunctions of $\exists^3$

The Jordan decomposition theorem

- A function $\phi : [0, 1] \to \mathbb{R}$ is of bounded variation if the set of the sums of the absolute values of the ups and downs along any partition is bounded. In that case, the least upper bound of this set is called the variation.
- The Jordan decomposition theorem states that if ϕ is of bounded variation, then ϕ can be expressed as the difference $\phi = \phi^+ - \phi^-$ of two increasing functions.
- Our question will be

  Given that we know that ϕ is of bounded variation, how hard is it to find a Jordan decomposition?

A first analysis

- If we look at the standard proof, we see that $\phi^+(x)$ is defined by taking $\phi(0) +$ the sup of the sums of the ups over all partitions of $[0, x]$ (and $\phi^-(x) = \phi^+(x) - \phi(x)$), so at first glance we seem to need full quantification over the continuum.
- Moreover, the map $\phi \mapsto \phi^+(1)$ cannot be countably based, because we could always modify ϕ to a ψ of bounded variation at one point outside any given support of $\phi^+(1)$, destroying $\phi^+(1)$ as a possible value for $\psi^+(1)$.

A further analysis I

- Let ϕ be of bounded variation. If $A \subset [0, 1]$ is dense, and contains all points where ϕ is discontinuous, we may restrict our partitions to those with elements in A when defining ϕ^+.
- When ϕ is of bounded variation, then ϕ is regulated. This means that the one-sided limits of ϕ at any point exist. The reason is that a one-sided oscillation will have an infinite variation as a consequence.

A further analysis II

- When ϕ is regulated, the one-sided limits can be calculated from the restriction of ϕ to the rationals.
- Consequently, the set D_ϕ of points where ϕ is discontinuous is computable from ϕ and $\exists^2$.
- If we can find an enumeration of D_ϕ, then (trivially) computable in ϕ, $\exists^2$ and this enumeration, we can decide if ϕ is of bounded variation, and in case it is, calculate ϕ^+.

A further analysis III

- The maximal deviation of $\phi(x)$ from its two one-sided limits gives a measure of how discontinuous ϕ is at x. Let $D_{\phi,n}$ be the set of points where this deviation is at least 2^{-n}.
- A cluster point in $D_{\phi,n}$ would cause the lack of a one-sided limit, so we know that $D_{\phi,n}$ must be finite, and thus that D_ϕ can be enumerated.
- The problem is that $D_{\phi,n}$ is an object of type 2, while a listing of $D_{\phi,n}$ will be an object of type 1.
- It turns out that it is in the writing of each $D_{\phi,n}$ as a finite set $\{x_{1,n}, \ldots, x_{k_n,n}\}$ the complexity of the Jordan decomposition, in terms of higher order computability theory, lies.

The functional Ω_b

- Ω_b is an innocent looking partial functional of type 3, $\Omega_b(X)$ is defined exactly when $X \subseteq 2^{\mathbb{N}}$ has at most one element, and then $\Omega_b(X) = 1$ if and only if $X \neq \emptyset$.
- Since we consider $\exists^2$ as given, it does not matter if we consider subsets of $2^{\mathbb{N}}$, $\mathbb{N}^{\mathbb{N}}$, $[0, 1]$ or $\mathbb{R}$.
- If $X \subseteq [0, 1]$ has at most one element, the characteristic function ϕ_X of X is of bounded variation.
- Then $X = \emptyset$ if and only if $\phi_X^+(1) = \phi_X^-(1) = 0$.
- So, modulo $\exists^2$, Jordan decomposition computes Ω_b.
- We will see that the converse is true as well.

Computing with Ω_{b}

Lemma

There is a partial functional Ω, defined on all sets with at most one element, computably equivalent to Ω_{b}, and selecting the element in X whenever $X \neq \emptyset$.

Theorem

There is a partial functional Ω_{fin}, computably equivalent to Ω_{b}, defined for all finite sets $X \subset 2^{\mathbb{N}}$, and outputting the elements of X organised as a sequence.

Remark

Ω_{fin} is defined using the fixed point facility of Kleene computing, and it is not known if the theorem is true using Gödel's T instead of Kleene's model.

Back to Jordan decomposition

- Recall the sets $D_{\phi,n}$ from our discussion of Jordan decomposition.
- These sets are finite, and Ω_{b} provides us with a listing of each of those.
- Given these listings, the Jordan decomposition is actually arithmetically definable, and thus computable in $\exists^2$.

The Ω-cluster

- The Ω-cluster consists of all partial functionals of type 3 that are equivalent to Ω_{b} modulo $\exists^2$.
- Informally, the Ω-cluster contains the operators of type 3 that are genuinely of type 3 and easy to construct given the abilities to
  1. quantify over an enumerated set
  2. write a set, known to be finite, as $\{x_0, \dots, x_{n-1}\}$ for some $n \geq 0$.

Another example

- What makes Ω_b worth understanding from a foundational point of view is that it captures a simple step in many mathematical arguments, a step that is mathematically unproblematic, but with a dubious computational content.
- We say that a non-empty set $X \subseteq \mathbb{R}$ is countable if there is an injection F from X to $\mathbb{N}$, and enumerable if there is a surjective map G from $\mathbb{N}$ to X.
- As we all know, the two concepts are equivalent. It is easy to see that the operator transforming an injection to the corresponding surjection is in the Ω-cluster.

A third example

- Let X be a collection of pairwise disjoint open intervals of reals, given by the set of pairs of endpoints.
- Allowing for quantifiers over the continuum, it is easy to define an injective map from X to $\mathbb{Q}$, and thus show that X must be enumerable.
- The partial functional “computing" this enumeration will be in the Ω-cluster.

The computational strength of Ω_{b}

On its own, or just in conjunction with $\exists^2$, Ω_{b} does not show much strength:

Theorem

If $f \in \mathbb{N}^{\mathbb{N}}$ is computable in Ω_{b} and $\exists^2$, then f is computable in just $\exists^2$, i.e. f is hyperarithmetical (Δ^1_1).

We say that Ω_{b} is lame.

The computational strength of Ω_b

- On the other hand, Ω_b is not computable in any reasonable sense.
- There can be no countable support for $\Omega_b(\emptyset) = 0$, so Ω_b is not countably based.
- So, for instance, Ω_b, and the other elements in the Ω-cluster, can not be computed by an ITTM with type two oracles.
- AND, with a little bit of HELP, Ω_b can be immensely strong.

$\Omega_{\mathfrak{b}}$ and the Suslin function

- The Suslin function can decide any Π^1_1-subset of $(\mathbb{N}^{\mathbb{N}})^2$.
- The Kondo-Addison theorem, stating Π^1_1-uniformisation, then turns a Π^1_1-relation to one where $\Omega_{\mathfrak{b}}$ is defined on each section, deciding if it is empty or not.
- Consequently, $\Omega_{\mathfrak{b}}$ computes $\mathbf{S}^2_2$ from $\mathbf{S}^2_1$.
- Due to the theorem on the next slide, it is consistent with **ZF** that $\Omega_{\mathfrak{b}} + \mathbf{S}^2_1$ computes $\exists^3$.

Ω_b and well orderings

For $X \subseteq \mathbb{N}^{\mathbb{N}}$, let

$$\exists^X(F) = \begin{cases} 1 & \text{if} \quad \exists f \in X(F(f) > 0) \\ 0 & \text{if} \quad \forall f \in X(F(f) = 0) \end{cases}$$

Theorem

Let $X \subseteq \mathbb{N}^{\mathbb{N}}$ and let R be a well ordering of X. Then $\exists^X$ is computable from Ω_b, $\exists^2$, X and R.

This theorem requires the full strength of Kleene's model.

$\Omega_\mathfrak{b}$ is intrinsically partial

Theorem

There is no total functional in the Ω-cluster.

The proof takes a large portion of Section 3.4 in N.- S. ’22.

Conjecture

There is a total extension of $\Omega_\mathfrak{b}$ in which $\exists^3$ is not computable.

The weaker Ω_1

- Ω_1 is the countably based shadow of Ω_{b}.
- $\Omega_1(X)$ is defined exactly when X is a singleton $\{f\}$, and then $f = \Omega_1(X)$.
- The Ω_1-cluster is the set of partial functionals computably equivalent to Ω_1 modulo $\exists^2$.
- Even if it is countably based, Ω_1 cannot be computed by an ITTM, since the halting problem for ITTMs is computable in Ω_1 and some ITTM.
- There is no total functional in the Ω_1-cluster.

The Ω_1-cluster

A set $X \subseteq \mathbb{R}$ is strongly countable, witnessed by F, if $F : \mathbb{R} \to \mathbb{N}$ is a bijection when restricted to X.

The following functionals are in the Ω_1-cluster:

- Find an enumeration of a strongly countable set X from X and the witness F.
- Evaluate the least upper bound of a strongly countable and bounded set X from X and the witness F.
- Find the Jordan decomposition of a function with bounded variation from the function and the variation.
- Find the supremum of a function with bounded variation from the function and its variation.

The stronger(?) Ω_{C}

- The following question led to the discovery of a seemingly stronger alternative to Ω_{b}:

  How hard is it to find the least upper bound of a bounded upper semicontinuous function ϕ on $[0, 1]$?

- The answer turned out to be Ω_{C}, where $\Omega_{\mathrm{C}}(X)$ is defined exactly when X is a closed subset of $2^{\mathbb{N}}$, and then $\Omega_{\mathrm{C}}(X) = 0$ exactly when $X = \emptyset$.
- So Ω_{C} is a partial subfunction of $\exists^3$.
- Ω_{C} is also lame, and Ω_{b} is obviously computable in Ω_{C}.

Equivalences to Ω_C

We define the Ω_C-cluster in analogy with the previous clusters. The following will be in the Ω_C-cluster:

- Find the supremum of an upper semicontinuous function on $[a, b]$ or of a bounded one on $\mathbb{R}$.
- Decide if an upper semicontinuous function on $\mathbb{R}$ is bounded.
- Find the RM-code of an open subset O of $\mathbb{R}$ or **C**.
- Find a selector function for the class of closed subsets of $\mathbb{R}$ or of **C**.

WARNING: We cannot replace $\mathbb{R}$ or **C** with $\mathbb{N}^{\mathbb{N}}$, as that would blow up the complexity considerably. To define Ω_C on compact subsets of $\mathbb{N}^{\mathbb{N}}$ will work, though.

An open problem, and a reference

Are Ω_b and Ω_C equivalent modulo $\exists^2$?

The conjecture is that this will not be the case.
We also conjecture that Ω_C is not equivalent to any total object, but this is of less foundational interest.

Reference

N.- S. ’22. D. Normann and S. Sanders, *On the computational properties of basic mathematical notions*, Journal of Logic and Computation 32(8), (2022), pp.1747 - 1795.

Thank You