Universitat de Barcelona

Master in Pure and Applied Logic

Master thesis

On a game-theoretic semantics for the Dialectica interpretation of analysis

The constructive content of Ramsey's theorem

Student:
Mireia González Bedmar

Advisor:
Joost J. Joosten

Academic year 2017-2018

Universitat de Barcelona

Abstract

Gödel's Dialectica interpretation is a tool of practical interest within proof theory. Although it was initially conceived in the realm of Hilbert's program, after Kreisel's fundamental work in the 1950's it has become clear that Dialectica, as well as other popular interpretations, can be used to extract explicit bounds and approximations from classical proofs in analysis. The program that was then started, consisting of using methods of proof theory to analyse and extract new information from classical proofs, is called proof mining.

The first extension of the Dialectica interpretation to analysis was achieved by Spector by means of a principle called bar recursion. Recently, Escardó and Oliva presented a new extension using a principle called *product of selection functions*, which provides a game-theoretic semantics to the interpreted theorems of analysis. This eases the task of understanding the constructive content and meaning of classical proofs, instead of only extracting quantitative information from them.

In this thesis we present the Dialectica interpretation and its extensions to analysis, both using bar recursion and the product of selection functions. A whole chapter is thus devoted to exposing the theory of sequential games by Escardó and Oliva.

In [38], Oliva and Powell gave a constructive proof of the Dialectica interpretation of the Infinite Ramsey Theorem for pairs and two colours using the product of selection functions. This yields an algorithm, which can be understood in game-theoretic terms, computing arbitrarily good approximations to the infinite monochromatic set. In this thesis we revisit this paper, extending all the results for the case of r colours, with $r \geq 2$.

Acknowledgements

I would like to thank all the people who have contributed directly or indirectly to this thesis. First of all, I am deeply grateful to my supervisor, Joost Joosten, for his time, patience, and help, which has gone far beyond academic collaboration.

A special thanks to Thomas Powell, for his involvement and guidance, and for the several ideas and clarifications that facilitated my understanding and improved this thesis.

I would like to thank all the professors who guided me when I had doubts about my academic future, including José Luis Balcázar, Joan Bagaria, and Dolors Herbera.

I also wish to thank my colleagues and friends Juan and Miguel for the support they provided, for the laughter we did and will share, and for inspiring the characters of one of the examples of this thesis. I thank Ana Borges for her advice, her time and ideas, and for the nice afternoons working by my side. I am also thankful to Martín Sócola for his help with English grammar. And, of course, I thank Pablo Llueca for being always there for me.

I am grateful to my mother and sisters: without their work, support, and help, this thesis, and so many other important things, would not exist.

Finally, thanks to J.C.C., to whom I will be grateful for everything I write.

Contents

Introduction

Kurt Gödel presented his functional interpretation in a lecture in 1941 [21] within the context of Hilbert's program. His recent incompleteness theorems had ended the search for finitistic proofs of the consistency of arithmetic, but had started the search for relative consistency proofs.

It was known, as Kolmogorov [28], Gödel [19], and Gentzen [16,17] had proved independently, that the consistency of Peano arithmetic (PA) reduces to the consistency of its intuitionistic counterpart, known as Heyting arithmetic (HA), via a negative translation assigning to each formula of arithmetic A, another formula A^N, in such a way that PA $\vdash A \leftrightarrow A^N$ and furthermore, if PA $\vdash A$ then HA $\vdash A^N$. The relative consistency proof comes from the fact that the translation of falsum is falsum, and hence if PA proves a contradiction, so does HA.

The Dialectica interpretation —as Gödel's functional interpretation was called after the name of the journal where it was first published [20]— defines, for each formula A of the language of arithmetic, a formula A^D of the form $\exists x \forall y A_D(x, y)$, where A_D is a formula of a quantifier-free system in all finite types, called $\mathbf{T}$. The so-called soundness theorem for the Dialectica interpretation guarantees that if HA $\vdash A$, then there is a term t of system $\mathbf{T}$, which can be effectively extracted from the proof of A in HA, such that $\mathbf{T} \vdash A_D(t, y)$. Again, the interpretation of falsum is falsum; therefore, if HA is inconsistent, so is $\mathbf{T}$. The benefit lies, in Troelstra's words [45], in the fact that "if $\mathbf{T}$ is regarded as embodying evident principles, this can be regarded as a consistency proof for HA". By composing a negative translation from PA to HA with the Dialectica interpretation, a consistency proof for classical arithmetic is obtained relative to $\mathbf{T}$.

Gödel already suggested in [20] that system $\mathbf{T}$ could be extended to a system with quantifiers in all finite types, and a proof of relative consistency for analysis could be achieved by extending his interpretation by means of "the sort of inference that Brouwer used in proving the fan theorem".

Indeed, system $\mathbf{T}$ extends to various systems of arithmetic with quantifiers in all finite types. Among them, the most convenient for the Dialectica interpretation is a system called WE-HA$^\omega$ (WE stands for *weakly extensional*, due to the treatment of equality), to which one can add the law of excluded middle to get a classical counterpart WE-PA$^\omega$. Adjoining some non-constructive principles to WE-PA$^\omega$, such as comprehension over numbers and choice for quantifier-free formulas, one gets a system capable of formalizing most of classical analysis.

Clifford Spector, using a principle of bar recursion BR inspired by Brouwer's fan theorem, achieved in [43] an extension of the ND-interpretation, i.e., the interpretation that consists of applying N first and D second, to a system of analysis as described above. That is, he proved that from a proof of a theorem from classical analysis, formalized in the system above, one can effectively extract a term t of $\text{WE-HA}^\omega + \mathsf{BR}$ such that $\text{WE-HA}^\omega + \mathsf{BR} \vdash (A^N)_D(t, y)$.

Meanwhile, in a series of articles [29–33] of the 1950's, Kreisel advocated for a "shift of emphasis" from relative consistency proofs to term extraction. He intuited that mathematical proofs carry hidden information besides the truth of their corresponding theorem, and that the application of methods of proof theory, not only Gödel's functional interpretation but also others like Hilbert's ε-substitution method and his own no-counterexample interpretation, to classical proofs from analysis, would help to extract this information. He launched a program which he called "unwinding of proofs", and he actually obtained the first primitive recursive bounds for Artin's solution to Hilbert's 17th problem using these methods [8, 36].

Although a number of term extractions were achieved, the program became less and less popular due to the lack of other applications. However, in the 1990's and 2000's, due to the refinement of some interpretations by U. Berger and H. Schwichtenberg [4,5], and to the monotone variant of the Dialectica interpretation proposed and successfully used by U. Kohlenbach [23–25], Kreisel's ideas were reborn. The program that started since this rebirth is called *proof mining*, after a suggestion of Dana Scott to Kohlenbach [26, 37].

Recently, the proof mining program began to focus not only on term extraction but also on the constructive meaning of the extracted proof. This is especially difficult when an instance of Spector's bar recursion appears, due to the abstruse nature of this principle. Thus, the effort of some proof theorists has focused on providing a clear and comprehensible semantics to bar recursion. In this context, Martín Escardó and Paulo Oliva [10–13] came out with a new theory of sequential games. They envisaged a principle of recursion called *product of selection functions*, or EPS, which allows for computing optimal strategies in their sequential games and, moreover, is equivalent to bar recursion.

This new principle has been used to replace bar recursion and to study the constructive content of several classical theorems, such as the Bolzano-Weierstrass theorem in [39] and the Infinite Ramsey Theorem for pairs in [38].

The aim of this thesis is to give an insight into proof mining, concretely into the combination of the negative translation with the Dialectica interpretation, and its extension to analysis by means of the new game-theoretic principle of EPS. To this end, besides the exposition of all the theoretic content needed, we devote a great effort to a practical case study: we revisit [38], extending the results for the Infinite Ramsey Theorem for pairs and two colours to the case of r colours, with $r \geq 2$, and laying emphasis on the significance of the constructive content of the classical proof.

Outline of the thesis

Chapter 1 establishes some preliminary results and conventions. Section 1.1 presents the basic definitions of classical and intuitionistic first-order logic, while Section 1.2 defines the systems of classical and intuitionistic arithmetic PA and HA. Section 1.3 gives a general idea of what a proof interpretation is, while Section 1.4 presents one of the variants of the negative translation between classical and intuitionistic first-order logic and arithmetic. Finally, Section 1.5 establishes some notational conventions to be used in the sequel.

Chapter 2 is devoted to presenting the Dialectica interpretation of arithmetic. Section 2.1 presents Gödel's system **T**. Section 2.2 gives a motivated definition of the Dialectica interpretation. The original version of the soundness theorem is moreover stated. Then, we move to Section 2.3 to present some extensions of HA and PA to systems with quantifiers in all finite types, and we state some important results about our preferred intuitionistic extension, WE-HA$^\omega$, to which an extension of Dialectica is possible. Finally, Section 2.4 presents again the definition of the Dialectica interpretation, this time over WE-HA$^\omega$ and emphasizing the types of the resulting variables, and gives soundness and characterization theorems.

In Chapter 3, we discuss what principles should be added to our system of arithmetic in all finite types in order to get a system capable of formalizing analysis. Section 3.1 shows that it is possible to reduce the *ND*-interpretation of those principles to the *D*-interpretation of DNS, the principle of the double negation shift. We also give a sufficient condition for three terms to witness DNS^D. In Section 3.2, we present the definition of Spector's bar recursion, and we prove two basic lemmas about it. Finally, in Section 3.3 we show how bar recursion can be used to construct terms satisfying the sufficient condition of Section 3.1. We end the chapter with a theorem stating the extension of the *ND*-interpretation to our system of analysis.

Then, we move to Escardó and Oliva's new theory of sequential games in Chapter 4. Section 4.1 gives the definition of the most simple kind of sequential games, those whose length is finite and fixed. Section 4.2 presents the fundamental notion of selection functions and their product. This product, when extended to the case of finite games with unbounded length, will be strong enough to replace bar recursion in the extension of Dialectica to analysis. In Section 4.3 we show how the product of selection functions can be used to compute optimal strategies in sequential games. Section 4.4 presents some applications of these notions to different areas of standard mathematics. Section 4.5 is devoted to explaining how the product of selection functions can be extended to a dependent version of sequential games, in which the goal of the game or the possible moves at each round may depend on the development of the previous rounds. In Section 4.6 we present the variant of sequential games in which the number of rounds is not fixed at the beginning of the play. Finally, Section 4.7 gives the definition of EPS and shows how it can be applied to find optimal strategies of the unbounded variant of sequential games.

Chapter 5 continues the discussion from Chapter 3, this time extending the Dialectica interpretation using EPS instead of bar recursion. Section 5.1 gives the main theorems on EPS, which will be extensively used in the sequel. In Section 5.2 the extension of Dialectica using EPS is achieved. Section 5.3 states the equivalence between EPS and bar recursion. In Section 5.4, we present the game-theoretic intuition behind the interpretation of DNS, and we explain what the advantages of EPS over bar recursion are.

Finally, in Chapter 6 we follow [38] to extract constructive content from the proof of Ramsey's theorem for pairs. We extend the proof given in [38], which is restricted to the case of 2 colours, to the case of r colours for $r \geq 2$. The structure of Chapter 6 is as follows. Section 6.1 gives preliminary well-known definitions and theorems. In Section 6.2 we present a classical proof formalized to a certain extent. In Section 6.3 we show how the statements of Ramsey's theorem and other principles used translate via the *ND*-interpretation. Section 6.4 presents the proof of Ramsey's theorem using several instances of EPS to interpret intermediate steps. In Section 6.5, we present a summary of the construction, which is actually an algorithm for approximating the infinite set whose existence is claimed by Ramsey's theorem. Finally, Section 6.6 explains the game-theoretic intuition behind each use of EPS.

What is new?

For the sake of transparency, we state here which parts of this thesis are new contributions.

The main original component is the extension of the interpretation and constructive proof of Ramsey's theorem from the 2-colour case of [38] to the case of r colours for $r \geq 2$. In particular, the proofs of Proposition 6.15 and of Theorem 6.30 are new. Of course, the rest of Chapter 6 has been adapted to this case.

Other small contributions are meant to make the exposition more accessible. In this direction, we have adapted and worked out some of the contents of this thesis. For instance, in Section 2.4 we explicitly give the types of the variables resulting from the Dialectica interpretation, a detail that is not usual in the literature. In Chapter 4, we present several examples in order to clarify and motivate the concepts to introduce. Finally, in Chapter 6 we work out all the details. In Section 6.3 we show how the *ND*-interpretation is computed and simplified in practice, something that was done, but not explained, in [38]. The structure of Subsection 6.4.2 is also different from the corresponding part of the original paper, focusing on clarifying what the order of the steps of the construction is at the expense of a bit of redundancy. The summary of Section 6.5 differs from [38] as well, in an attempt to give a clearer glimpse of the algorithmic nature of the construction.

Chapter 1

Preliminaries

1.1 Intuitionistic and classical first-order logic

In this section we present the basic definitions of first-order logic, and the axioms and rules of systems IL and CL, i.e., intuitionistic logic and classical logic. Our presentation here follows closely [26].

The language of first-order logic consists of the logical constants $\wedge$, $\vee$, $\to$, $\perp$, $\exists$, $\forall$; countably-many variables $x, y, z, \ldots$; for any arity $n \geq 0$ a possibly empty countable set of function symbols (with arity 0 they are called constants), and for any arity $n \geq 1$ a possibly empty countable set of predicate symbols. There is also a symbol for equality, $=$.

Definition 1.1. Terms are recursively defined as:

(i) Variables and constants are terms;

(ii) If $t_0, \ldots, t_{n-1}$ are terms and f is an n-ary function symbol, then $ft_0 \ldots t_{n-1}$ is a term.

Terms not containing variables are called **closed**.

Now we can use equality and predicate symbols to combine terms into atomic formulas.

Definition 1.2. Atomic formulas are defined as:

(i) $\perp$ is an atomic formula;

(ii) If s, t are terms, then $(s = t)$ is an atomic formula;

(iii) If $t_0, \ldots, t_{n-1}$ are terms and P is an n-ary predicate symbol, then $Pt_0 \ldots t_{n-1}$ is an atomic formula.

Finally, we define formulas.

Definition 1.3. Formulas are recursively defined as:

(i) Atomic formulas are formulas;

(ii) If A, B are formulas, then $(A \wedge B), (A \vee B), (A \to B)$ are formulas;

(iii) If A is a formula and x is a variable, then $(\exists x A)$, $(\forall x A)$ are formulas.

The outermost parentheses are usually dropped. We will use the abbreviations

$$\neg A := A \to \bot, \quad A \leftrightarrow B := (A \to B) \wedge (B \to A).$$

We use the following convention on parentheses: negation and quantifiers bind stronger than $\vee, \wedge$, which bind stronger than $\to, \leftrightarrow$. Moreover, $\to$ associates to the right, that is, $A \to B \to C$ means $\big(A \to (B \to C)\big)$.

The definition of *free* and *bound* variables, and of free and bound occurrences of variables, is as usual. Given a term t, a variable x, and a formula A, we define $A[t/x]$ as the formula obtained by substituting all free occurrences of x by t in A, also as usual. Moreover, we say that t is free for x in A, or that the substitution of t for x in A is free, if there is no free occurrence of x in A inside the scope of a quantifier bounding a variable of t. We assume that all the substitutions that we perform are free, via renaming of bound variables if necessary.

The axioms and rules of intuitionistic first-order logic IL are now presented. Most of them are to be understood as axiom schemata over all formulas A, B or all function or predicate symbols f, P. We write EFQ for *ex falso quodlibet*.

$$\begin{aligned}
\vee\text{-contraction} &: A \vee A \to A; \\
\wedge\text{-contraction} &: A \to A \wedge A; \\
\vee\text{-weakening} &: A \to A \vee B; \\
\wedge\text{-weakening} &: A \wedge B \to A; \\
\vee\text{-permutation} &: A \vee B \to B \vee A; \\
\wedge\text{-permutation} &: A \wedge B \to B \wedge A; \\
\text{EFQ} &: \bot \to A; \\
\exists\text{-quantifier} &: A[t/x] \to \exists x A \qquad \text{(where } t \text{ is free for } x \text{ in } A); \\
\forall\text{-quantifier} &: \forall x A \to A[t/x] \qquad \text{(where } t \text{ is free for } x \text{ in } A); \\
=\text{-reflexivity} &: x = x; \\
=\text{-symmetry} &: x = y \to y = x; \\
=\text{-transitivity} &: x = y \wedge y = z \to x = z; \\
=\text{-functional} &: x_0 = y_0 \wedge \ldots \wedge x_{n-1} = y_{n-1} \to f x_0 \ldots x_{n-1} = f y_0 \ldots y_{n-1}; \\
=\text{-relational} &: x_0 = y_0 \wedge \ldots \wedge x_{n-1} = y_{n-1} \to P x_0 \ldots x_{n-1} \to P y_0 \ldots y_{n-1}.
\end{aligned}$$

The rules are the following:

$$\text{MP} \ \frac{A \quad A \to B}{B}\,;$$

$$\text{Syllogism} \ \frac{A \to B \quad B \to C}{A \to C}\,;$$

$$\text{Exportation} \ \frac{A \wedge B \to C}{A \to B \to C}\,;$$

$$\text{Importation } \frac{A \to B \to C}{A \wedge B \to C};$$

$$\text{Expansion } \frac{A \to B}{C \vee A \to C \vee B};$$

$$\forall\text{-rule } \frac{B \to A}{B \to \forall x A}, \text{ where } x \text{ is not free in } B;$$

$$\exists\text{-rule } \frac{A \to B}{\exists x A \to B}, \text{ where } x \text{ is not free in } B.$$

This ends the definition of system IL.

Classical first-order logic CL is obtained from IL by adding the schema of the **law of excluded middle**:

$$\mathsf{LEM} : A \vee \neg A.$$

1.2 Heyting and Peano arithmetic

Heyting arithmetic HA is defined over intuitionistic first-order logic with a constant 0 (zero), a function symbol S (successor), and function symbols for all (derivations of) the other primitive recursive functions. The axioms added are the successor axioms:

$$Sx \neq 0, \; Sx = Sy \to x = y,$$

the axiom schema of complete induction:

$$\mathsf{IND} : A(0) \to \forall x\big(A(x) \to A(Sx)\big) \to \forall x A(x),$$

and the defining equations for the primitive recursive functions. The concrete way to do this last step is not relevant here and we omit it. For further details we refer the reader to [44, Section 1.3]. We emphasize, however, that all these defining axioms are quantifier-free formulas.

Peano arithmetic, also known as classical arithmetic, is obtained from HA by adding the law of excluded middle schema

$$\mathsf{LEM} : A \vee \neg A.$$

1.3 Proof interpretations

Proof interpretation is the name received by a wide range of different notions, with different goals and applications, from proving relative consistency of theories to extracting explicit bounds from a proof of classical analysis. Here we try to explain what one can expect when facing a so-called proof interpretation. This section does not present any formal definitions, the vocabulary used is intended just for intuition, and the only technical or official terms are those which appear in boldface.

Given two formal systems $\mathcal{S}$ and $\mathcal{T}$, a proof interpretation is a (usually recursive) translation from formulas of $\mathcal{S}$ to formulas of $\mathcal{T}$. A bit more precisely, it is a way of uniformly obtaining a formula A' of system $\mathcal{T}$ given a formula A of system $\mathcal{S}$. The formula A' is usually defined by induction on the logical structure of A.

For a proof interpretation to be meaningful, the translation has to relate both systems in some deep way. When defining a proof interpretation, two kinds of theorems are usually presented: the **soundness** theorem and the **characterization** theorem.

Soundness theorems have a form similar to:

$$\textit{For all formulas } A \textit{ of the language of } \mathcal{S}, \\ \textit{if } \mathcal{S} + \mathcal{C} + \Gamma \vdash A, \textit{ then } \mathcal{T} + \Gamma' \vdash A'.$$

Here $\mathcal{C}$ and Γ are collections of formulas of the language of $\mathcal{S}$, which we call principles, because they are not derivable from $\mathcal{S}$, but they are accepted in some contexts of interest. The collection Γ' consists of all formulas B' for $B \in \Gamma$. The reason for having two different sets of principles is that the translations of the principles of $\mathcal{C}$ are not needed in $\mathcal{T}$ in order to prove A'. In fact, they usually vanish under the proof interpretation, that is, for every $C \in \mathcal{C}$, C' is a valid formula in $\mathcal{T}$.

Characterization theorems have a form similar to:

$$\textit{For all formulas } A \textit{ of the language of } \mathcal{S}, \\ \mathcal{U} + \mathcal{P} \vdash A \leftrightarrow A'.$$

Here $\mathcal{U}$ is a formal system that includes the language of $\mathcal{S}$ and the language of $\mathcal{T}$, and $\mathcal{P}$ is a collection of principles.

In this thesis we will present a so-called negative translation, which gives an interpretation of classical logic into intuitionistic logic, and where the languages of both theories are equal, so no extra theory $\mathcal{U}$ will be needed. Moreover, we develop the Dialectica interpretation in two settings, but the characterization theorem is presented only in a context where the languages of both theories are also equal, so again no mix of languages will be needed.

1.4 The negative translation

A negative translation is a way of obtaining from a theorem A of classical logic, a theorem A^N of intuitionistic logic such that, moreover, in classical logic we can prove $A \leftrightarrow A^N$. This is inspired by the fact that there exists such a translation for propositional logic. If CPL is the system of classical propositional logic and IPL is the system of intuitionistic propositional logic, Glivenko's theorem [18] tells us that if $\mathsf{CPL} \vdash A$, then $\mathsf{IPL} \vdash \neg\neg A$, and of course $\mathsf{CPL} \vdash A \leftrightarrow \neg\neg A$. Actually, Glivenko's theorem is stronger, as it states that $\mathsf{CPL} \vdash A$ if and only if $\mathsf{IPL} \vdash \neg\neg A$. For a detailed presentation of CPL and IPL we refer the reader to [46].

However, in first-order logic the trick of just double-negating all formulas does not work. A simple counterexample is:

$$\neg\neg\forall x\big(A(x) \vee \neg A(x)\big),$$

which is not intuitionistically valid (for a proof see [42, Proposition 8.5.3]).

There are several variants of the negative translation. The first one is due to Kolmogorov [28], and it consists of double-negating all subformulas of the original formula. A few years later, Gödel [19] and Gentzen [16,17] independently presented similar translations. Later others appeared, due for instance to Kuroda [35]. We use here one of the variants proposed by Kuroda, which is used in [26] as well. All variants turn out to be equivalent, not only in CL, which is obvious, but in IL, in the sense that, given a formula A of classical logic, if A' is one of the translations and $\breve{A}$ is another one, then $\mathsf{IL} \vdash A' \leftrightarrow \breve{A}$. For a proof of this fact see [15].

Definition 1.4. Let A be a formula in (an extension of) the language of CL. Define $A^N := \neg\neg A^*$, where A^* is defined by induction on the logical structure of A as:

(i) $A^* := A$ for atomic A;

(ii) $(A \diamond B)^* := A^* \diamond B^*$ for $\diamond \in \{\wedge, \vee, \to\}$;

(iii) $(\exists x A)^* := \exists x A^*$;

(iv) $(\forall x A)^* := \forall x \neg\neg A^*$.

The soundness and characterization theorems for this translation are well-known. A detailed proof can be found in [15].

Theorem 1.5 (Soundness)**.** *Let Γ be a set of principles in the language of* CL *and A a formula in the same language. If* $\mathsf{CL} + \Gamma \vdash A$, *then* $\mathsf{IL} + \Gamma^N \vdash A^N$.

In particular, we have that:

Corollary 1.6 (Soundness for arithmetic)**.** *If* $\mathrm{PA} \vdash A$, *then* $\mathrm{HA} \vdash A^N$.

Proof. Notice that PA is $\mathsf{CL} + \mathsf{S} + \Delta + \mathsf{IND}$, where S stands for the axioms for the successor, and Δ is the set of defining axioms of all primitive recursive functions; and HA is $\mathsf{IL} + \mathsf{S} + \Delta + \mathsf{IND}$. Given A such that $\mathrm{PA} \vdash A$, we know that $\mathsf{IL} + \mathsf{S}^N + \Delta^N + \mathsf{IND}^N \vdash A^N$. To see that $\mathrm{HA} \vdash A^N$, it will be sufficient to show that S^N, Δ^N and IND^N are provable in HA. This is clear for S^N and Δ^N, since all axioms in S and Δ are quantifier-free. Let us see how IND translates:

$$\mathsf{IND}^N : \neg\neg\Big(A^*(0) \to \forall x \neg\neg\big(A^*(x) \to A^*(Sx)\big) \to \forall x \neg\neg A^*(x)\Big).$$

Since $\mathsf{IL} \vdash B \to \neg\neg B$ and $\mathsf{IL} \vdash \neg\neg(B \to C) \to (\neg\neg B \to \neg\neg C)$, it suffices to prove:

$$\mathrm{HA} \vdash \neg\neg A^*(0) \to \forall x\big(\neg\neg A^*(x) \to \neg\neg A^*(Sx)\big) \to \forall x \neg\neg A^*(x).$$

And this is true simply because HA contains the induction schema, in particular HA contains the induction axiom for $\neg\neg A^*(x)$. $\square$

Moreover, we have the characterization theorem:

Theorem 1.7 (Characterization). *For all formulas A in the language of* CL, $\mathsf{CL} \vdash A \leftrightarrow A^N$.

This yields directly:

Corollary 1.8 (Characterization for arithmetic). *For all formulas A in the language of* PA, $\mathrm{PA} \vdash A \leftrightarrow A^N$.

1.5 Notational conventions

Throughout the following chapters, we will work with finite and (countably) infinite sequences. Depending on the framework, sequences may be differently formalized.

When working in the framework of standard mathematics, we will make use of the Cartesian product. Given a natural number n and sets $X_0, \dots, X_{n-1}$, we consider the elements $s : \prod_{i=0}^{n-1} X_i$ as finite sequences. An infinite sequence may be $\alpha : \prod_{i=0}^{\infty} X_i$ for sets X_i for each $i \in \mathbb{N}$, or, in the case where all elements of the sequence are elements of the same set X, $\alpha : \mathbb{N} \to X$. Moreover, we use the notation X^* for the set of all finite sequences of elements of X.

When working in a formal system in all finite types, as the ones presented in the following chapter, we will not have product types, but we will still use $\mathbb{N} \to X$ as a type of infinite sequences (which will actually be written $0 \to \sigma$, as we will see), and sometimes the informal type σ^* of finite sequences over type σ, which will be in fact an abbreviation of an encoding of finite sequences using an actual type of our formal system. See Remark 5.2.

In both frameworks, we write sequences in angle brackets, as in $\langle a_0, \dots, a_{n-1} \rangle$. The letter s is usually used for sequences, especially finite, while α is usually used for infinite sequences. Other letters are in boldface when they denote sequences, for instance, $\boldsymbol{x}$ or $\boldsymbol{a}$. To denote the ith element of a sequence s, we write s_i or $s(i)$.

Let s be a finite sequence of length n, let t be a finite sequence of length m and α an infinite sequence. We use the following notations:

$$
\begin{aligned}
\langle\rangle &\equiv \text{the empty sequence.} \\
|s| &\equiv \text{the length of } s. \\
s * t &\equiv \text{the concatenation of } s \text{ and } t, \text{ i.e.,} \langle s_0, \dots, s_{n-1}, t_0, \dots, t_{m-1} \rangle. \\
s * \alpha &\equiv \text{the concatenation of } s \text{ and } \alpha, \text{i.e.,} \\
&\qquad (s * \alpha)(i) = s(i) \text{ for } i < n, (s * \alpha)(i) = \alpha(i - n) \text{ for } i \geq n, \\
\overline{s} &\equiv s * \mathbf{0}.
\end{aligned}
$$

Moreover, for any sequence s of length at least n, infinite or finite:

$$
\begin{aligned}
[s](n) &\equiv \langle s_0, \dots, s_{n-1} \rangle. \\
\overline{s, n} &\equiv [s](n) * \mathbf{0}.
\end{aligned}
$$

Here $\mathbf{0}$ is an infinite sequence of elements that will depend on the context: it may be the zero of the corresponding type, or it may be a canonical element of the corresponding set, as we will announce in each case.

By an abuse of notation, when s is a finite sequence of elements of some set or type X and a is an element of X, we write $s * a$ instead of $s * \langle a \rangle$.

Furthermore, if we have a function $f : X \to (Y \to Z)$ and $x : X$, we sometimes write f_x instead of $f(x)$. If $f : \prod_{i=0}^{n-1} X_i \to Y$ and $s : \prod_{i=0}^{k-1} X_i$ for $k < n$, then f_s is defined by $f_s(t) = f(s * t)$ for all $t : \prod_{i=k}^{n-1} X_i$, i.e., f_s is f partially evaluated on s. The same applies if $f : \prod_{i=0}^{\infty} X_i \to Y$ and $s : \prod_{i=0}^{k-1} X_i$. We define $f_s(t) = f(s * t)$ for all $t : \prod_{i=k}^{\infty}$.

Chapter 2

The Dialectica interpretation

Gödel's functional interpretation of arithmetic, also called Dialectica interpretation after the name of the journal where it was first published [20] in 1958, was in fact conceived years before, as two lectures by Gödel in 1938 [22] and 1941 [21] make clear.

The Dialectica interpretation takes arbitrary formulas A of Heyting arithmetic to formulas of the form $\exists x \forall y A_D(x, y)$, where A_D is quantifier-free, of a system with higher types. As we shall see in this chapter, the original complexity of the quantifiers of the formula is converted into complexity of the types of x and y.

The original goal of the functional interpretation was to achieve a proof of consistency for arithmetic. As Gödel's second incompleteness theorem had shown, a finitistic proof of the consistency of PA is impossible, since even PA is incapable of proving its own consistency. It was known, however, that the negative translation reduces the consistency of PA to that of HA. Gödel's intention with Dialectica was to provide a consistency proof for arithmetic relative to the consistency of $\mathbf{T}$, his system in all finite types without quantifiers. To this end, it is shown that if $\mathrm{HA} \vdash A$, then there is a term t such that $\mathbf{T} \vdash A_D(t, y)$. Since the interpretation of $\perp$ is $\perp$, this shows that if HA is inconsistent, then so is $\mathbf{T}$.

Spector [43] extended the Dialectica interpretation to an interpretation of a system capable of formalizing classical analysis, by means of a principle called bar recursion that we present in Chapter 3. A relative consistency proof for analysis was thus achieved too.

However, Kreisel, in a series of publications [29–33] in the 1950's, promoted a shift of emphasis. He realized that the extraction of a term witnessing, in some sense, a classical theorem, was of practical interest. He started a program of term extraction from classical proofs which he called "unwinding of proofs" and which involved the use of several interpretations, Dialectica among them.

These ideas experimented a rebirth in the 1990's and 2000's, due to the work of U. Berger and H. Schwichtenberg [4, 5], and U. Kohlenbach [23–25]. This rebirth has been called proof mining.

In Section 2.1 we present the formal system $\mathbf{T}$. Section 2.2 presents the definition of the Dialectica interpretation and gives a motivating discussion. Finally, Sections 2.3 and 2.4 extend Dialectica to a system of arithmetic in all finite types and with quantifiers, which will be useful for our purposes within proof mining.

2.1 Gödel's system T

The set of all finite types T is generated inductively as:

(i) $0 \in T$.

(ii) If $\sigma, \tau \in T$, then $(\sigma \to \tau) \in T$.

There are many alternative notations for $\sigma \to \tau$ around in the literature, such as (σ, τ), τ^σ, $(\sigma)\tau$, $(\tau)\sigma$, etc. The intended reading is that 0 represents the type of the natural numbers, while $(\sigma \to \tau)$ represents a type of some kind of functions from σ to τ.

As usual, we omit parentheses when no confussion is possible, always understanding that $\to$ associates to the right.

Some presentations of system **T** add a third way of defining types:

(iii) If $\sigma, \tau \in T$, then $(\sigma \times \tau) \in T$.

As one may expect, $(\sigma \times \tau)$ is intended to be the cartesian product of σ and τ. However, this condition can be eliminated by *currying*, that is, interpreting any type $(\sigma \times \tau) \to \rho$ as $\sigma \to \tau \to \rho$. Officially, we do not take condition (iii), but sometimes it is used as notational abbreviation of the longer curried version.

The **pure types** are defined recursively as:

- $(0) = 0$.
- $(n+1) = (n) \to 0$.

We drop parentheses when the context makes clear that we are dealing with pure types.

Gödel's system **T** is a quantifier-free theory over the finite types T. The language of **T** includes countably-many variables $x^\sigma, y^\sigma, z^\sigma, \ldots$ for each type $\sigma \in T$. When it is clear from the context, we will omit the superscript. The logical constants are the usual connectives $\wedge, \vee, \to, \bot$. Regarding equality, there are several different options that can be adopted. Here we take the intensional decidable original version from [20]. For each type σ, there is an equality symbol $=_\sigma$. We will discuss other options in Section 2.3.

We now define inductively the set of terms of **T** together with the relation $t : \sigma$, read "term t is of type σ":

(i) Every variable x^σ is a term of type σ.

(ii) If $s : \sigma$ and $t : \sigma \to \tau$, then $\big(t(s)\big) : \tau$.

(iii) There are the following constants:

- $\mathbf{0} : 0$.
- $\mathbf{S} : 0 \to 0$.
- For all types σ, τ, $\mathbf{\Pi}_{\sigma,\tau} : \sigma \to \tau \to \sigma$.
- For all types ρ, σ, τ, $\mathbf{\Sigma}_{\rho,\sigma,\tau} : (\rho \to \sigma \to \tau) \to (\rho \to \sigma) \to \rho \to \tau$.
- For all types σ, $\mathbf{R}_\sigma : \sigma \to (0 \to \sigma \to \sigma) \to 0 \to \sigma$.

The intended interpretation is that $\mathbf{0}$ denotes the number zero, $\mathbf{S}$ denotes the successor function and $\mathbf{\Pi}, \mathbf{\Sigma}$ and $\mathbf{R}$ are type combinators and recursors that will be defined later. Moreover, if $s : \sigma$ and $t : \sigma \to \tau$, then $\big(t(s)\big)$ is to be understood as the result of applying the function t to the argument s. For the sake of brevity, we sometimes drop parentheses and write $t(s)$ or ts. When application is iterated, for instance, if we have $t : \rho \to \sigma \to \tau$, $r : \rho$, $s : \sigma$, we write $t(r)(s)$, $t(r, s)$ or even $trs : \tau$ meaning $\big(t(r)\big)(s)$, that is, we understand that application associates to the left.

Each type σ can be written as $\sigma_0 \to \ldots \to \sigma_{n-1} \to 0$ for some natural number n and some types $\sigma_0, \ldots, \sigma_{n-1}$. This can be verified by a straightforward induction.

The formulas of **T** are inductively defined as:

(i) $\perp$ is a formula.

(ii) For each type σ and terms $t, s : \sigma$, $t =_\sigma s$ is a formula.

(iii) If A, B are formulas, then $A \wedge B$, $A \vee B$, $A \to B$ are formulas.

The axioms of **T** are those of *classical* propositional logic, together with the equality axioms:

$$t =_\sigma t,$$
$$t =_\sigma s \wedge A[t/x] \to A[s/x],$$

and the following defining equations for the constants:

$$\mathbf{S}x^0 \neq_0 \mathbf{0},$$
$$\mathbf{S}x^0 =_0 \mathbf{S}y^0 \to x^0 =_0 y^0,$$
$$\mathbf{\Pi}_{\sigma,\tau} x^\sigma y^\tau =_\sigma x^\sigma,$$
$$\mathbf{\Sigma}_{\rho,\sigma,\tau} x^{\rho\to\sigma\to\tau} y^{\rho\to\sigma} z^\rho =_\tau xz(yz),$$
$$\mathbf{R}_\sigma x^\sigma y^{0\to\sigma\to\sigma} \mathbf{0} =_\sigma x^\sigma,$$
$$\mathbf{R}_\sigma x^\sigma y^{0\to\sigma\to\sigma} (\mathbf{S}z^0) =_\sigma yz(\mathbf{R}_\sigma xyz).$$

Notice that $\mathbf{\Pi}_{\sigma,\tau}$ can be understood as a first projection, while $\mathbf{\Sigma}_{\rho,\sigma,\tau}$ combines its arguments in a way that will allow for defining λ-abstraction in Section 2.3. Finally, the recursor $\mathbf{R}_\sigma$ performs the usual definition by recursion:

$$h(n) = \begin{cases} f & \text{if } n = 0, \\ g(k, h(k)) & \text{if } n = k + 1. \end{cases}$$

The rules of **T** are the induction rule:

$$\mathsf{IND\text{-}R} \ \frac{A(0) \quad A(x) \to A(\mathbf{S}x)}{A(t)}$$

and a rule allowing for the substitution of arbitrary terms for variables of the same type:

$$\frac{A}{A[t/x]}, \text{ where } t \text{ and } x \text{ have the same type.}$$

2.2 The Dialectica interpretation

This section is devoted to defining an interpretation (in the sense of Section 1.3) from HA into system $\mathbf{T}$. Its main aim is to provide the reader with the intuition on why the Dialectica interpretation is defined as it is, and so sometimes the treatment is not entirely formal. The following sections will approach this topic in our preferred setting, namely a system of arithmetic with quantifiers over all finite types.

From a formula A in the language of HA, we will obtain a formula:

$$A^D = \exists \boldsymbol{x} \forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y}),$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are (possibly empty) tuples of variables of any type, and A_D is a (quantifier-free) formula of $\mathbf{T}$. The free variables of A^D are exactly the free variables of A. The variables in $\boldsymbol{x}, \boldsymbol{y}$ and their types only depend on the logical structure of A. Moreover, A^D does not contain disjunctions.

Definition 2.1. Given a formula A in the language of HA, A^D and A_D are inductively defined as follows:

(i) $A^D := A_D := A$ for atomic A.

If $A^D = \exists \boldsymbol{x} \forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y})$ and $B^D = \exists \boldsymbol{u} \forall \boldsymbol{v} B_D(\boldsymbol{u}, \boldsymbol{v})$, then:

(ii) $(A \wedge B)^D := \exists \boldsymbol{x}, \boldsymbol{u}\, \forall \boldsymbol{y}, \boldsymbol{v} \big(A_D(\boldsymbol{x}, \boldsymbol{y}) \wedge B_D(\boldsymbol{u}, \boldsymbol{v})\big)$.

(iii) $(A \vee B)^D := \exists z, \boldsymbol{x}, \boldsymbol{u}\, \forall \boldsymbol{y}, \boldsymbol{v} \Big(\big(z = \mathbf{0} \to A_D(\boldsymbol{x}, \boldsymbol{y})\big) \wedge \big(z \neq \mathbf{0} \to B_D(\boldsymbol{u}, \boldsymbol{v})\big)\Big)$.

(iv) $(A \to B)^D := \exists \boldsymbol{U}, \boldsymbol{Y}\, \forall \boldsymbol{x}, \boldsymbol{v} \big(A_D(\boldsymbol{x}, \boldsymbol{Y}\boldsymbol{x}\boldsymbol{v}) \to B_D(\boldsymbol{U}\boldsymbol{x}, \boldsymbol{v})\big)$.

(v) $(\exists z A(z))^D := \exists z, \boldsymbol{x} \forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y}, z)$.

(vi) $(\forall z A(z))^D := \exists \boldsymbol{X} \forall \boldsymbol{y}, z A_D(\boldsymbol{X} z, \boldsymbol{y}, z)$.

The interpretation of $\exists z A(z)$ is clear, while $(\forall z A(z))^D$ is intuitive if one skolemizes the formula:

$$\forall z \exists \boldsymbol{x} \forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y}, z).$$

Similarly, the following skolemization gives intuition about $(A \to B)^D$.

$$\exists \boldsymbol{x} \forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y}) \to \exists \boldsymbol{u} \forall \boldsymbol{v} B_D(\boldsymbol{u}, \boldsymbol{v}) \quad \Leftrightarrow \tag{2.1}$$

$$\forall \boldsymbol{x} \big(\forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y}) \to \exists \boldsymbol{u} \forall \boldsymbol{v} B_D(\boldsymbol{u}, \boldsymbol{v})\big) \quad \Leftrightarrow \tag{2.2}$$

$$\forall \boldsymbol{x} \exists \boldsymbol{u} \big(\forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y}) \to \forall \boldsymbol{v} B_D(\boldsymbol{u}, \boldsymbol{v})\big) \quad \Leftrightarrow \tag{2.3}$$

$$\forall \boldsymbol{x} \exists \boldsymbol{u} \forall \boldsymbol{v} \big(\forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y}) \to B_D(\boldsymbol{u}, \boldsymbol{v})\big) \quad \Leftrightarrow \tag{2.4}$$

$$\forall \boldsymbol{x} \exists \boldsymbol{u} \forall \boldsymbol{v} \exists \boldsymbol{y} \big(A_D(\boldsymbol{x}, \boldsymbol{y}) \to B_D(\boldsymbol{u}, \boldsymbol{v})\big) \quad \Leftrightarrow \tag{2.5}$$

$$\forall \boldsymbol{x} \exists \boldsymbol{u}, \boldsymbol{Y}' \forall \boldsymbol{v} \big(A_D(\boldsymbol{x}, \boldsymbol{Y}'(\boldsymbol{v})) \to B_D(\boldsymbol{u}, \boldsymbol{v})\big) \quad \Leftrightarrow \tag{2.6}$$

$$\exists \boldsymbol{U}, \boldsymbol{Y} \forall \boldsymbol{x} \forall \boldsymbol{v} \big(A_D(\boldsymbol{x}, \boldsymbol{Y}(\boldsymbol{x}, \boldsymbol{v})) \to B_D(\boldsymbol{U}(\boldsymbol{x}), \boldsymbol{v})\big).$$

The order of the steps above is important because it tries to minimize the use of non-constructive principles. A detailed discussion on how the other choices are worse can be found in [26].

Although we do not include this discussion here, we analyse however which non-constructive principles are used by the above. First of all, we notice that stating the needed principles requires the use of quantifiers over higher types. So for the moment we will suppose that we have some extension of $\mathbf{T}$ that allows such quantifiers. We will see in the next section that there are in fact several such extensions, and in Section 2.4 we will study the behaviour of D over one of these systems.

Steps (2.1) and (2.3) are constructively valid. Steps (2.5) and (2.6) make use of the *axiom of choice*:

$$\mathsf{AC} : \forall x \exists y A(x, y) \to \exists Y \forall x A(x, Yx),$$

where A ranges over arbitrary formulas, and x, y are variables of arbitrary types. Moreover, AC justifies the skolemization in the case $(\forall z A(z))^D$.

Step (2.2) requires a form of a principle called *independence of premise*:

$$\mathsf{IP} : \big(A \to \exists x B(x)\big) \to \exists x \big(A \to B(x)\big),$$

where A and B range over arbitrary formulas and x is of any type, not free in A.

We take this principle applied only to purely universal premises, that is:

$$\mathsf{IP}_\forall : \big(\forall u A_0(u) \to \exists x B(x)\big) \to \exists x \big(\forall u A_0(u) \to B(x)\big),$$

where A_0 is quantifier-free and u is of any type.

Finally, step (2.4) is justified if we accept *Markov's principle*:

$$\mathsf{MP} : \neg\neg \exists x A_0(x) \to \exists x A_0(x),$$

where A_0 is quantifier-free and x is of arbitrary type. Indeed, we have:

$$\forall \boldsymbol{x} \exists \boldsymbol{u} \forall \boldsymbol{v} \big(\forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y}) \to B_D(\boldsymbol{u}, \boldsymbol{v})\big). \tag{2.7}$$

Using that A_D, B_D are quantifier-free, it is known (see [26]) that, over intuitionistic logic, (2.7) is equivalent to:

$$\forall \boldsymbol{x} \exists \boldsymbol{u} \forall \boldsymbol{v} \neg\neg \exists \boldsymbol{y} \big(A_D(\boldsymbol{x}, \boldsymbol{y}) \to B_D(\boldsymbol{u}, \boldsymbol{v})\big),$$

and then by MP we complete step (2.4).

The next natural step would be to prove the soundness and characterization theorems for this interpretation. However, it is convenient for our purposes to postpone this step until a formal system of arithmetic with quantifiers over higher types is presented. The considerations on this section about AC, $\mathsf{IP}_\forall$, and MP will justify the characterization theorem for Dialectica over that system.

For an extensive treatment of Dialectica over HA, the reader is referred to [1]. In any case, the soundness theorem in this setting is as follows:

Theorem 2.2 (Gödel [20])**.** *Let A be a formula in the language of* HA. *If* HA *proves A, then there is a tuple of terms $\boldsymbol{t}$ such that $\mathbf{T}$ proves $A_D(\boldsymbol{t}, \boldsymbol{y})$. Moreover, $\boldsymbol{t}$ can be effectively extracted from the proof of A in* HA.

From PA, we can go first to HA via the negative translation, and then apply the Dialectica interpretation. Theorem 2.2 together with Corollary 1.6 yields:

Theorem 2.3 (Gödel [20])**.** *Let A be a formula in the language of* PA. *If* PA *proves A, then there is a tuple of terms $\boldsymbol{t}$ such that $\mathbf{T}$ proves $(A^N)_D(\boldsymbol{t}, \boldsymbol{y})$.*

2.3 Systems of arithmetic in all finite types

Here we present systems of arithmetic extending HA to languages with all finite types. In fact, these systems are different versions of $\mathbf{T}$ adding quantifiers over types. Our presentation follows [44].

We first describe system N-HA$^\omega$, where the N is for neutral, as it has no treatment of equality. The language of N-HA$^\omega$ is very similar to that of $\mathbf{T}$, but adding quantifiers. For the sake of clarity, we present it again in detail.

The language of N-HA$^\omega$ includes countably-many variables $x^\sigma, y^\sigma, z^\sigma, \ldots$ for each type $\sigma \in T$. For each type σ, there is an equality symbol $=_\sigma$. The logical constants are the usual connectives $\wedge, \vee, \to, \perp$ and quantifiers $\forall x^\sigma, \exists x^\sigma$ for each variable x^σ, type $\sigma \in T$. Finally, we have constants like above, $\mathbf{0} : 0$; $\mathbf{S} : 0 \to 0$; $\mathbf{\Pi}_{\sigma,\tau} : \sigma \to \tau \to \sigma$; $\mathbf{\Sigma}_{\rho,\sigma,\tau} : (\rho \to \sigma \to \tau) \to (\rho \to \sigma) \to \rho \to \tau$; and $\mathbf{R}_\sigma : \sigma \to (0 \to \sigma \to \sigma) \to 0 \to \sigma$, for all types σ, τ, ρ.

In N-HA$^\omega$ we have the same terms as in $\mathbf{T}$.

Definition 2.4. The formulas of N-HA$^\omega$ are inductively defined as:

(i) $\perp$ is a formula;

(ii) For each type σ and $t, s : \sigma$, $t =_\sigma s$ is a formula;

(iii) If A, B are formulas, then $A \wedge B$, $A \vee B$, $A \to B$, $(\forall x^\sigma A)$, $(\exists x^\sigma A)$ are formulas.

The axioms and rules of N-HA$^\omega$ are:

(a) All the axioms and rules of IL presented in Section 1.1, except the equality axioms, ranging over formulas of N-HA$^\omega$.

(b) The equality axioms:

$$
\begin{aligned}
& x^\sigma =_\sigma x^\sigma, \\
& x^\sigma =_\sigma y^\sigma \to y^\sigma =_\sigma x^\sigma, \\
& x^\sigma =_\sigma y^\sigma \wedge y^\sigma =_\sigma z^\sigma \to x^\sigma =_\sigma z^\sigma, \\
& x^\sigma =_\sigma y^\sigma \to z^{\sigma\to\tau} x^\sigma =_\tau z^{\sigma\to\tau} y^\sigma, \\
& x^{\sigma\to\tau} =_{\sigma\to\tau} y^{\sigma\to\tau} \to x^{\sigma\to\tau} z^\sigma =_\tau y^{\sigma\to\tau} z^\sigma.
\end{aligned}
$$

(c) The induction schema IND, now ranging over all formulas of the language.

(d) The defining axioms for $\mathbf{S}, \mathbf{\Pi}, \mathbf{\Sigma}$ and $\mathbf{R}$ that we have seen in $\mathbf{T}$.

System N-HA$^\omega$ is a basic system that can be extended in several different ways depending on the desired treatment of equality. In general, equality between objects of higher types (for instance, functions) can be interpreted as *intensional* or *extensional*. The idea is that two functions are extensionally equal if they yield equal outputs from equal inputs; they are intensionally equal if their definitions are syntactically equal in some sense depending on the particular setting. Intensional equality is usually taken to be decidable.

What follows is a description of systems I-HA$^\omega$ (*intensional*), E-HA$^\omega$ (*extensional*) and WE-HA$^\omega$ (*weakly extensional*). System WE-HA$^\omega$ is the most usual in contexts where the Dialectica interpretation appears since it is a suitable extension for the proof of soundness to work.

I-HA$^\omega$

From N-HA$^\omega$ we define a system I-HA$^\omega$ where the intended meaning of equality is intensional. As discussed above, intensional equality should be decidable. We add for each type σ a constant $\mathbf{E}_\sigma : \sigma \to \sigma \to 0$, and axioms:

$$\mathbf{E}_\sigma x^\sigma y^\sigma =_0 \mathbf{0} \leftrightarrow x^\sigma =_\sigma y^\sigma,$$
$$\mathbf{E}_\sigma x^\sigma y^\sigma =_0 \mathbf{0} \vee \mathbf{E}_\sigma x^\sigma y^\sigma =_0 \mathbf{S0}.$$

This implies decidability of equality at all types.

E-HA$^\omega$ and E-PA$^\omega$

In E-HA$^\omega$ equality is interpreted as extensional. A reasonable axiom of extensionality is:

$$\forall x^{\sigma\to\tau} y^{\sigma\to\tau}\big(x =_{\sigma\to\tau} y \leftrightarrow \forall z^\sigma(xz =_\tau yz)\big).$$

But, for the purposes for which E-HA$^\omega$ is used, it is more convenient to have the equality of type 0 as the only primitive one, and equality for all other types as a defined notion:

$$x^{\sigma\to\tau} =_{\sigma\to\tau} y^{\sigma\to\tau} \quad :\equiv \quad \forall z^\sigma(xz =_\tau yz).$$

We keep all axioms of N-HA$^\omega$, including the equality ones, now interpreted as referring to our defined equality (some of them become redundant). Now extensionality is expressed by the axiom:

$$x^\sigma =_\sigma y^\sigma \to z^{\sigma\to\tau} x^\sigma =_\tau z^{\sigma\to\tau} y^\sigma \tag{2.8}$$

Finally, E-PA$^\omega$ is obtained from E-HA$^\omega$ by adding the law of excluded middle LEM for arbitrary formulas.

WE-HA$^\omega$ and WE-PA$^\omega$

It turns out that the soundness theorem for the Dialectica interpretation does not hold over system E-HA$^\omega$ because of the full extensionality axiom. For a discussion on this see [26, pp. 126–127]. We will thus weaken extensionality. System WE-HA$^\omega$ is obtained replacing the axiom of extensionality (2.8) by the rule schema (with $s^\sigma, t^\sigma, r^\tau \in T$):

$$\frac{A_0 \to s =_\sigma t}{A_0 \to r[s/x^\sigma] =_\tau r[t/x^\sigma]}\ \text{, where } A_0 \text{ is quantifier-free.}$$

Finally, WE-PA$^\omega$ is obtained from WE-HA$^\omega$ adding the law of excluded middle LEM for arbitrary formulas.

Now we give propositions stating that system WE-HA$^\omega$ is strong enough to define λ-abstraction and some terms that will be useful later on. For proofs of these facts the reader is referred to [26].

Proposition 2.5. *For every term $t : \tau$ of* WE-HA$^\omega$*, there is a term $(\lambda x^\sigma . t) : \sigma \to \tau$, whose free variables are exactly those of t except for x, such that for every term $s : \sigma$,*

$$\text{WE-HA}^\omega \vdash (\lambda x^\sigma . t)(s) =_\tau t[s/x].$$

The term $(\lambda x.t)$ is called a λ-abstraction of t. We use the notation $\lambda x, y \,.\, t$ to refer to $\big(\lambda x.(\lambda y.t)\big)$, and analogously for any number of nested λ-abstractions. For each type $\sigma = \sigma_0 \to \ldots \to \sigma_{n-1} \to 0$, we define the following term:

$$\mathbf{0}^\sigma := \lambda x_0^{\sigma_0}, \ldots, x_{n-1}^{\sigma_{n-1}} . \mathbf{0}^0.$$

Proposition 2.6. *Let $A_0(\boldsymbol{x})$ be a quantifier-free formula of the language of* WE-HA$^\omega$ *whose free variables are among $\boldsymbol{x} = \langle x_0, \ldots, x_{n-1} \rangle$. Then there is a closed term t_{A_0} such that:*

$$\text{WE-HA}^\omega \vdash \forall \boldsymbol{x} \big(t_{A_0} \boldsymbol{x} =_0 \mathbf{0} \leftrightarrow A_0(\boldsymbol{x}) \big).$$

The quantifier-free law of excluded middle is defined as:

$$\mathsf{QF\text{-}LEM} : A_0 \vee \neg A_0,$$

where A_0 ranges over quantifier-free formulas.

Corollary 2.7. WE-HA$^\omega \vdash$ $\mathsf{QF\text{-}LEM}$*, that is, for every quantifier-free formula A_0 of* WE-HA$^\omega$*,*

$$\text{WE-HA}^\omega \vdash A_0 \vee \neg A_0.$$

We also can do definition by cases:

Proposition 2.8. *For every type $\sigma \in T$, there is a closed term t of* WE-HA$^\omega$ *such that:*

$$\text{WE-HA}^\omega \vdash \forall z^0, x^\sigma, y^\sigma \big((z =_0 \mathbf{0} \to tzxy =_\sigma x) \wedge (z \neq_0 \mathbf{0} \to tzxy =_\sigma y) \big).$$

Moreover, the negative translation as presented in Section 1.4 is also sound in WE-HA$^\omega$. We understand now that the quantifiers of Definition 1.4 are over any type. Then we have:

Theorem 2.9 ([15, Theorem 2.9])**.** *Let Γ be a set of principles in the language of* WE-PA$^\omega$ *and A a formula in the same language. If* WE-PA$^\omega + \Gamma \vdash A$*, then* WE-HA$^\omega + \Gamma^N \vdash A^N$*.*

2.4 The Dialectica interpretation on WE-HA$^\omega$

As we have seen, the Dialectica interpretation of a formula A has the form $\exists \boldsymbol{x} \forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y})$. The boldface $\boldsymbol{x}$ and $\boldsymbol{y}$ indicate that they are tuples, that is, the form of A^D is actually

$$\exists x_0 \ldots \exists x_{n-1} \forall y_0 \ldots \forall y_{m-1} A_D(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{m-1}).$$

Each of these variables is possibly of higher type, so in fact we have:

$$\exists x_0^{\sigma_0} \dots \exists x_{n-1}^{\sigma_{n-1}} \forall y_0^{\tau_0} \dots \forall y_{m-1}^{\tau_{m-1}} A_D(x_0, \dots, x_{n-1}, y_0, \dots, y_{m-1}).$$

In this case, we will write, just as a metamathematical abbreviation, $\boldsymbol{x} : \boldsymbol{\sigma}$ and $\boldsymbol{y} : \boldsymbol{\tau}$, and the formula becomes $\exists \boldsymbol{x}^{\boldsymbol{\sigma}} \forall \boldsymbol{y}^{\boldsymbol{\tau}} A_D(\boldsymbol{x}, \boldsymbol{y})$. Moreover, if ρ is a type, we will also write things like $\boldsymbol{x}^{\rho \to \boldsymbol{\sigma}}$ abbreviating $x_0^{\rho \to \sigma_0}, \dots, x_{n-1}^{\rho \to \sigma_{n-1}}$; and $\boldsymbol{x}^{\rho \to \boldsymbol{\sigma}} z^\rho$ abbreviating $x_0(z), \dots, x_{n-1}(z)$. Finally, $\boldsymbol{z}^{\boldsymbol{\sigma} \to \boldsymbol{\tau}}$ means:

$$z_0^{\sigma_0 \to \dots \to \sigma_{n-1} \to \tau_0}, \dots, z_{m-1}^{\sigma_0 \to \dots \to \sigma_{n-1} \to \tau_{m-1}},$$

while $\boldsymbol{x}^{(\boldsymbol{\sigma} \to \boldsymbol{\tau}) \to \boldsymbol{\sigma}}$ means $x_0, \dots, x_{n-1}$ where for $i = 0, \dots, n-1$:

$$x_i : \big(\sigma_0 \to \dots \to \sigma_{n-1} \to \tau_0\big) \to \dots \to \big(\sigma_0 \to \dots \to \sigma_{n-1} \to \tau_{m-1}\big) \to \sigma_i.$$

Remark 2.10. The awkward definitions for tuples introduced above can be avoided just by allowing product types, as Section 2.1 discusses. However, we choose here the alternative that keeps the definition of our systems the simplest. It is well-known that, using pairing functions, our version of system WE-HA$^\omega$ is capable of coding tuples of variables into a single one and we will make use of this fact in some future cases in order to avoid even the tuple notation above when it becomes so tedious that it makes proofs incomprehensible. For details see [44].

Now we present the definition of the Dialectica interpretation over system WE-HA$^\omega$. The definition is as Definition 2.1, but here we are explicit about the types of the variables.

Definition 2.11. Given a formula A in the language of WE-HA$^\omega$, we inductively define another formula in the same language $A^D := \exists \boldsymbol{x} \forall \boldsymbol{y} A_D(\boldsymbol{x}, \boldsymbol{y})$, where A_D is a quantifier-free formula without $\vee$. The free variables of A^D are that of A. The variables in $\boldsymbol{x}, \boldsymbol{y}$ and their types only depend on the logical structure of A.

(i) $A^D := A_D := A$ for atomic A.

If $A^D = \exists \boldsymbol{x}^{\boldsymbol{\sigma}} \forall \boldsymbol{y}^{\boldsymbol{\tau}} A_D(\boldsymbol{x}, \boldsymbol{y})$ and $B^D = \exists \boldsymbol{u}^{\boldsymbol{\mu}} \forall \boldsymbol{v}^{\boldsymbol{\nu}} B_D(\boldsymbol{u}, \boldsymbol{v})$, then:

(ii) $(A \wedge B)^D := \exists \boldsymbol{x}^{\boldsymbol{\sigma}}, \boldsymbol{u}^{\boldsymbol{\mu}}\ \forall \boldsymbol{y}^{\boldsymbol{\tau}}, \boldsymbol{v}^{\boldsymbol{\nu}} \big(A_D(\boldsymbol{x}, \boldsymbol{y}) \wedge B_D(\boldsymbol{u}, \boldsymbol{v})\big)$.

(iii) $$(A \vee B)^D := \exists z^0, \boldsymbol{x}^{\boldsymbol{\sigma}}, \boldsymbol{u}^{\boldsymbol{\mu}}\ \forall \boldsymbol{y}^{\boldsymbol{\tau}}, \boldsymbol{v}^{\boldsymbol{\nu}} \Big(\big(z = \mathbf{0} \to A_D(\boldsymbol{x}, \boldsymbol{y})\big) \wedge \big(z \neq \mathbf{0} \to B_D(\boldsymbol{u}, \boldsymbol{v})\big)\Big).$$

(iv) $(A \to B)^D := \exists \boldsymbol{U}^{\boldsymbol{\sigma} \to \boldsymbol{\mu}}, \boldsymbol{Y}^{\boldsymbol{\sigma} \to \boldsymbol{\nu} \to \boldsymbol{\tau}}\ \forall \boldsymbol{x}^{\boldsymbol{\sigma}}, \boldsymbol{v}^{\boldsymbol{\nu}} \big(A_D(\boldsymbol{x}, \boldsymbol{Y}\boldsymbol{x}\boldsymbol{v}) \to B_D(\boldsymbol{U}\boldsymbol{x}, \boldsymbol{v})\big)$.

(v) $(\exists z^\rho A(z))^D := \exists z^\rho, \boldsymbol{x}^{\boldsymbol{\sigma}} \forall \boldsymbol{y}^{\boldsymbol{\tau}} A_D(\boldsymbol{x}, \boldsymbol{y}, z)$.

(vi) $(\forall z^\rho A(z))^D := \exists \boldsymbol{X}^{\rho \to \boldsymbol{\sigma}} \forall \boldsymbol{y}^{\boldsymbol{\tau}}, z^\rho A_D(\boldsymbol{X}z, \boldsymbol{y}, z)$.

Remark 2.12. We notice that for any formula A in the language of WE-HA$^\omega$, we have $(A^D)^D = A^D$ (up to renaming of bound variables). Hence, by the definition of D, when computing A^D we can apply D to some subformula first; for example, for any formulas A, B, we have $(A \to B)^D = (A^D \to B^D)^D$.

Since our final aim is to interpret theorems of classical analysis, we will need to apply the negative translation before the Dialectica interpretation. Therefore, the Dialectica interpretation of a double-negated formula will be often needed, so we compute it in general terms:

Remark 2.13. The Dialectica interpretation of the negation and double negation is as follows:

(a) $(\neg A)^D = \exists \boldsymbol{Y}^{\boldsymbol{\sigma}\to\boldsymbol{\tau}}\, \forall \boldsymbol{x}^{\boldsymbol{\sigma}} \neg A_D(\boldsymbol{x}, \boldsymbol{Y}\boldsymbol{x})$.

(b) $(\neg\neg A)^D = \exists \boldsymbol{X}^{(\boldsymbol{\sigma}\to\boldsymbol{\tau})\to\boldsymbol{\sigma}}\, \forall \boldsymbol{Y}^{\boldsymbol{\sigma}\to\boldsymbol{\tau}} \neg\neg A_D\big(\boldsymbol{X}\boldsymbol{Y}, \boldsymbol{Y}(\boldsymbol{X}\boldsymbol{Y})\big)$, which in WE-HA$^\omega$ is equivalent to $\exists \boldsymbol{X}^{(\boldsymbol{\sigma}\to\boldsymbol{\tau})\to\boldsymbol{\sigma}}\, \forall \boldsymbol{Y}^{\boldsymbol{\sigma}\to\boldsymbol{\tau}} A_D\big(\boldsymbol{X}\boldsymbol{Y}, \boldsymbol{Y}(\boldsymbol{X}\boldsymbol{Y})\big)$.

Now we are ready to give soundness and characterization theorems for the Dialectica interpretation over WE-HA$^\omega$. A purely universal sentence is a formula of the form $\forall \boldsymbol{x} A_0$, where A_0 is a quantifier-free formula whose free variables are among the ones in $\boldsymbol{x}$.

Theorem 2.14 (Soundness of the Dialectica interpretation, Gödel [20])**.** *Let $\mathcal{P}$ be a set of purely universal sentences in the language of* WE-HA$^\omega$*. Let $A(\boldsymbol{a})$ be a formula in the language of* WE-HA$^\omega$ *containing only $\boldsymbol{a}$ free. Then, if*

$$\text{WE-HA}^\omega + \mathsf{AC} + \mathsf{IP}_\forall + \mathsf{MP} + \mathcal{P} \vdash A(\boldsymbol{a}),$$

then there is a tuple of terms $\boldsymbol{t}$ of WE-HA$^\omega$ *such that*

$$\text{WE-HA}^\omega + \mathcal{P} \vdash \forall \boldsymbol{y} A_D(\boldsymbol{t}\boldsymbol{a}, \boldsymbol{y}, \boldsymbol{a}).$$

Moreover, $\boldsymbol{t}$ can be effectively extracted from a proof of the assumption.

Proof. The proof proceeds by induction on the length of the derivation. Using Propositions 2.5, 2.6 and 2.8, the proof goes along as mostly routine.

For instance, assuming that $A^D = \exists \boldsymbol{x}^{\boldsymbol{\sigma}} \forall \boldsymbol{y}^{\boldsymbol{\tau}} A_D(\boldsymbol{x}, \boldsymbol{y})$, the first axiom of IL is treated as follows:

$\vee$**-contraction** : $A \vee A \to A$**.**

$$\begin{aligned}(A \vee A \to A)^D = {} & \exists \boldsymbol{X}'', \boldsymbol{Y}\boldsymbol{Y}'\, \forall z^0, \boldsymbol{x}, \boldsymbol{x}', \boldsymbol{y}'' \\ & \Big(\big(z = \boldsymbol{0} \to A_D(\boldsymbol{x}, \boldsymbol{Y} z \boldsymbol{x}\boldsymbol{x}'\boldsymbol{y}'')\big) \wedge \big(z \neq \boldsymbol{0} \to A_D(\boldsymbol{x}', \boldsymbol{Y}' z \boldsymbol{x}\boldsymbol{x}'\boldsymbol{y}'')\big) \to \\ & \qquad A_D(\boldsymbol{X}'' z \boldsymbol{x}\boldsymbol{x}', \boldsymbol{y}'')\Big).\end{aligned}$$

Here the following terms do the job (recall that $\boldsymbol{a}$ are the free variables of A):

$$\boldsymbol{t}_{\boldsymbol{X}''} := \lambda \boldsymbol{a} z \boldsymbol{x}\boldsymbol{x}'. \begin{cases} \boldsymbol{x} & \text{if } z = \boldsymbol{0} \\ \boldsymbol{x}' & \text{otherwise,} \end{cases}$$
$$\boldsymbol{t}_{\boldsymbol{Y}} := \lambda \boldsymbol{a} z \boldsymbol{x}\boldsymbol{x}'\boldsymbol{y}''.\boldsymbol{y}'' =: \boldsymbol{t}_{\boldsymbol{Y}'}.$$

The only difficulty arises in the treatment of the seemingly harmless axiom of $\wedge$-contraction:

$\wedge$-contraction : $A \to A \wedge A$.

$$(A \to A \wedge A)^D = \exists \boldsymbol{Y}, \boldsymbol{X}', \boldsymbol{X}'' \, \forall \boldsymbol{x}, \boldsymbol{y}', \boldsymbol{y}'' \\ \Big(A_D(\boldsymbol{x}, \boldsymbol{Y}\boldsymbol{x}\boldsymbol{y}'\boldsymbol{y}'') \to A_D(\boldsymbol{X}'\boldsymbol{x}, \boldsymbol{y}') \wedge A_D(\boldsymbol{X}''\boldsymbol{x}, \boldsymbol{y}'') \Big).$$

Take:

$$\boldsymbol{t}_{\boldsymbol{X}'} := \lambda \boldsymbol{a}\boldsymbol{x}.\boldsymbol{x} =: \boldsymbol{t}_{\boldsymbol{X}''},$$

$$\boldsymbol{t}_{\boldsymbol{Y}} := \lambda \boldsymbol{a}\boldsymbol{x}\boldsymbol{y}'\boldsymbol{y}''. \begin{cases} \boldsymbol{y}' & \text{if } t_{A_D}\boldsymbol{x}\boldsymbol{y}'\boldsymbol{a} \neq \boldsymbol{0} \\ \boldsymbol{y}'' & \text{if } t_{A_D}\boldsymbol{x}\boldsymbol{y}'\boldsymbol{a} = \boldsymbol{0}. \end{cases}$$

The idea is that $\boldsymbol{t}_{\boldsymbol{Y}}$ takes witnesses of the failure of $A_D(\boldsymbol{x}, \boldsymbol{y}') \wedge A_D(\boldsymbol{x}, \boldsymbol{y}'')$ to a witness of the failure of $A_D(\boldsymbol{x}, \boldsymbol{t}_{\boldsymbol{Y}}\boldsymbol{a}\boldsymbol{x}\boldsymbol{y}'\boldsymbol{y}'')$. We are witnessing, therefore, the contrapositive of the axiom, and that is allowed because the law of the excluded middle holds for quantifier-free formulas.

Another remarkable case is the induction schema, which is easily interpreted using the recursors $\mathbf{R}$. The principles AC, $\mathsf{IP}_\forall$ and MP are mapped to tautologies by the Dialectica interpretation, more precisely to formulas of the form $(A \to A)^D$.

The complete proof can be found in [26]. □

Theorem 2.15 (Characterization of the Dialectica interpretation, [1, 26, 44])**.** *For every formula A in the language of* WE-HA$^\omega$*, we have:*

$$\text{WE-HA}^\omega + \mathsf{AC} + \mathsf{IP}_\forall + \mathsf{MP} \vdash A \leftrightarrow A^D$$

Proof. An easy induction. The ideas have been presented in Section 2.2. □

As we have discussed, we can combine the negative translation with the Dialectica interpretation. We first need another version of Theorem 2.9. Let $\mathsf{QF\text{-}AC}$ be the axiom schema of choice, AC, restricted to quantifier-free formulas:

$$\mathsf{QF\text{-}AC} : \forall x \exists y A_0(x, y) \to \exists Y \forall x A_0(x, Yx),$$

where A_0 ranges over quantifier-free formulas (of the language of WE-HA$^\omega$), and x, y are variables of arbitrary type.

Proposition 2.16 ([26, Proposition 10.6])**.** *Let $\mathcal{P}$ be a set of purely universal principles in the language of* WE-PA$^\omega$ *and A a formula in the same language. If* WE-PA$^\omega$ $+ \mathsf{QF\text{-}AC} + \mathcal{P} \vdash A$*, then* WE-HA$^\omega$ $+ \mathsf{QF\text{-}AC} + \mathcal{P} + \mathsf{MP} \vdash A^N$.

Proof. It is known (see [26]) that for $P \in \mathcal{P}$, using that P is purely universal, we get WE-HA$^\omega \vdash P \leftrightarrow P^N$. If we prove that WE-HA$^\omega + \mathsf{QF\text{-}AC} + \mathsf{MP} \vdash (\mathsf{QF\text{-}AC})^N$, then by Theorem 2.9 the result follows.

Given a quantifier-free formula A_0, we have that the negative translation of $\forall x \exists y A_0(x, y) \to \exists Y \forall x A_0(x, Yx)$ is:

$$\neg\neg\big(\forall x \neg\neg\exists y A_0(x,y) \to \exists Y \forall x \neg\neg A_0(x, Yx)\big),$$

which, since A_0 is quantifier-free, is equivalent to:

$$\neg\neg\big(\forall x \neg\neg\exists y A_0(x,y) \to \exists Y \forall x A_0(x, Yx)\big),$$

so it is sufficient to prove:

$$\text{WE-HA}^\omega + \mathsf{QF\text{-}AC} + \mathsf{MP} \vdash \forall x \neg\neg\exists y A_0(x,y) \to \exists Y \forall x A_0(x, Yx),$$

and this is clear by $\mathsf{QF\text{-}AC}$ and MP. □

The above proposition allows for combining N and D.

Theorem 2.17 ([26, Theorem 10.7])**.** *Let $\mathcal{P}$ be a set of purely universal sentences in the language of* WE-HA$^\omega$*. Let $A(\boldsymbol{a})$ be a formula in the language of* WE-HA$^\omega$ *containing only $\boldsymbol{a}$ free. Then, if*

$$\text{WE-PA}^\omega + \mathsf{QF\text{-}AC} + \mathcal{P} \vdash A(\boldsymbol{a}),$$

then there is a tuple of terms $\boldsymbol{t}$ of WE-HA$^\omega$ *such that*

$$\text{WE-HA}^\omega + \mathcal{P} \vdash \forall \boldsymbol{y} (A^N)_D(\boldsymbol{t}\boldsymbol{a}, \boldsymbol{y}, \boldsymbol{a}).$$

Moreover, $\boldsymbol{t}$ can be effectively extracted from a proof of the assumption.

Proof. It follows from Proposition 2.16 and Theorem 2.14. □

Chapter 3

Interpretation of analysis: an extension of Dialectica with bar recursion

It is known that most parts of classical analysis can be formalized in full second order arithmetic. It is thus palpable why an extension of the Dialectica interpretation into a functional interpretation of a system containing second order arithmetic was presented shortly after the first publication of Gödel's paper [20]. This was achieved by Spector in [43], following an idea already present in Gödel's paper. The extension of Dialectica for classical analysis was then seen as a means for a relative consistency proof, but has since shown to be very useful for the proof mining program, as its proof of soundness gives an algorithm for extracting bounds from proofs.

Our system of classical analysis will be $\text{WE-PA}^\omega + \mathsf{QF\text{-}AC} + \mathsf{CA}^0$, where CA^0 denotes the schema of full comprehension over numbers:

$$\mathsf{CA}^0 : \exists f^1 \forall x^0 \big(fx =_0 \mathbf{0} \leftrightarrow A(x) \big),$$

where A is an arbitrary formula and f is not free in A.

In this chapter we will see an interpretation of system $\text{WE-PA}^\omega + \mathsf{QF\text{-}AC} + \mathsf{CA}^0$ using Spector's bar recursion, a principle which, although capable of this task, has been regarded as an obscure way to make the proof of soundness work. This is in part the reason why the proof mining program has been mainly focused on explicit term extraction and not on the meaning of the interpreted theorem and proof. In following chapters, we present a recent alternative to bar recursion, given in several papers by Escardó and Oliva [10–13], which tries to fix this issue by means of a game-theoretic reading of the functional interpretation, opening the possibility of understanding the constructive content of classical proofs in analysis.

Our presentation closely follows [26].

3.1 Principles for analysis

Instead of giving a direct interpretation of CA^0, in Proposition 3.1 we observe that, in the presence of classical logic and the restricted schema of choice:

$$\mathsf{QF\text{-}AC}^{0,0} : \forall x^0 \exists y^0 A_0(x,y) \to \exists f^1 \forall x^0 A_0(x, fx),$$

where A_0 ranges over quantifier-free formulas, the schema CA^0 is equivalent to:

$$\mathsf{AC}^{0,0} : \forall x^0 \exists y^0 A(x,y) \to \exists f^1 \forall x^0 A(x, fx),$$

where A ranges over arbitrary formulas.

Proposition 3.1. *Over* WE-PA$^\omega$ + $\mathsf{QF\text{-}AC}^{0,0}$, *the schemata* CA^0 *and* $\mathsf{AC}^{0,0}$ *are equivalent.*

Proof. Assume CA^0. To prove $\mathsf{AC}^{0,0}$, let $A(x,y)$ be any formula. We apply CA^0 to $A(x,y)$ (using a pairing function):

$$\forall x^0, y^0 \big(g(x,y) =_0 \mathbf{0} \leftrightarrow A(x,y) \big).$$

Therefore, $\forall x^0 \exists y^0 A(x,y)$ is equivalent to $\forall x^0 \exists y^0 \big(g(x,y) = \mathbf{0} \big)$, and hence, using $\mathsf{QF\text{-}AC}^{0,0}$, we get $\exists f^1 \forall x^0 \big(g(x, fx) = \mathbf{0} \big)$, which is equivalent to the required:

$$\exists f^1 \forall x^0 A(x, fx).$$

Conversely, assume $\mathsf{AC}^{0,0}$. Let $A(x)$ be any formula and f^1 a variable not free in $A(x)$. We shall prove $\exists f^1 \forall x^0 \big(fx = \mathbf{0} \leftrightarrow A(x) \big)$. In WE-PA$^\omega$ we have:

$$\forall x^0 \exists n^0 \big(n = \mathbf{0} \leftrightarrow A(x) \big),$$

since we have LEM. Applying $\mathsf{AC}^{0,0}$ to this formula we get:

$$\exists f^1 \forall x^0 \big(fx = \mathbf{0} \leftrightarrow A(x) \big).$$

□

What we will prove, as Spector did, is that the axiom schema of countable choice AC^0 can be *ND*-interpreted using bar recursion. The schema AC^0 is defined as the union of the schemata $\mathsf{AC}^{0,\sigma}$ for every type σ, where:

$$\mathsf{AC}^{0,\sigma} : \forall x^0 \exists y^\sigma A(x,y) \to \exists f^{0\to\sigma} \forall x^0 A(x, fx).$$

The negative translation of this reads:

$$\neg\neg \big(\forall x^0 \neg\neg \exists y^\sigma A^*(x,y) \to \exists f^{0\to\sigma} \forall x^0 \neg\neg A^*(x, fx) \big).$$

Using that, intuitionistically, $\neg\neg(B \to C)$ is equivalent to $B \to \neg\neg C$, we get that over WE-HA$^\omega$ the above is equivalent to:

$$\forall x^0 \neg\neg \exists y^\sigma A^*(x,y) \to \neg\neg \exists f^{0\to\sigma} \forall x^0 \neg\neg A^*(x, fx).$$

The *double negation shift*, which is intuitionistically underivable, is the schema ranging over arbitrary formulas A:

$$\mathsf{DNS} : \forall x^0 \neg\neg A(x) \to \neg\neg \forall x^0 A(x).$$

Proposition 3.2. WE-HA$^\omega$ + AC^0 + DNS *proves* $(\mathsf{AC}^0)^N$.

Proof. Given a formula $A(x^0, y^\sigma)$, we have to prove:

$$\forall x^0 \neg\neg \exists y^\sigma A^*(x,y) \to \neg\neg\exists f^{0\to\sigma}\forall x^0 \neg\neg A^*(x, fx). \tag{3.1}$$

Applying AC^0 to the formula A^*, since $(B \to C) \to (\neg\neg B \to \neg\neg C)$ holds intuitionistically, we obtain:

$$\neg\neg\forall x^0 \exists y^\sigma A^*(x,y) \to \neg\neg\exists f^{0\to\sigma}\forall x^0 A^*(x, fx). \tag{3.2}$$

By DNS, $\forall x^0 \neg\neg\exists y^\sigma A^*(x,y)$ implies $\neg\neg\forall x^0\exists y^\sigma A^*(x,y)$, which by (3.2) implies $\neg\neg\exists f^{0\to\sigma}\forall x^0 A^*(x, fx)$. Since intuitionistically it holds that $B \to \neg\neg B$, we get (3.1) as required. □

Therefore, we have reduced the functional interpretation of $(\mathsf{AC}^0)^N$ to that of DNS.

Proposition 3.3 (Spector, [43]). *Given $A(x)$, let $\exists u \forall v A_D(x,u,v)$ be the formula obtained after computing $\big(A(x)\big)^D$ and contracting the existential tuple of variables into u and the universal tuple into v, as explained in Remark 2.10. Let $\mathsf{DNS}(A)$ be the statement of DNS applied to the formula A, i.e.,*

$$\forall x^0 \neg\neg A(x) \to \neg\neg\forall x^0 A(x).$$

Assume that t_x, t_W and t_V are terms containing only U, Y, Z as free variables, and that for all U, Y, Z, the following are satisfied:

$$\begin{cases} t_x = Y(t_W) \\ U(t_x, t_V) = t_W(Y(t_W)) \\ t_V(U(t_x, t_V)) = Z(t_W). \end{cases} \tag{3.3}$$

Then, t_x, t_W and t_V are witnesses of $\mathsf{DNS}(A)^D$.

Proof. First we compute $\mathsf{DNS}(A)^D$. Note that by Remark 2.12 we have that $\mathsf{DNS}(A)^D = \mathsf{DNS}(A^D)^D$.

$$\begin{aligned}
\mathsf{DNS}^D = & \big(\forall x \neg\neg\exists u \forall v A_D(x,u,v) \to \neg\neg\forall y \exists w \forall z A_D(y,w,z)\big)^D = \\
& \big(\forall x \exists U \forall V \neg\neg A_D(x, UV, V(UV)) \to \neg\neg\exists W \forall y, z A_D(y, Wy, z)\big)^D = \\
\Big(& \exists U\, \forall x, V \neg\neg A_D\big(x, UxV, V(UxV)\big) \to \\
& \exists W \forall Y, Z \neg\neg A_D\big(Y(WYZ), WYZ(Y(WYZ)), Z(WYZ)\big)\Big)^D = \\
\bigg(& \forall U \exists W \forall Y, Z \exists x, V \Big(\neg\neg A_D\big(x, UxV, V(UxV)\big) \to \\
& \neg\neg A_D\big(Y(WYZ), WYZ(Y(WYZ)), Z(WYZ)\big)\Big)\bigg)^D
\end{aligned}$$

We do not write the last step because it is almost incomprehensible for a human reader. But, recalling the justification for the definition of $(A \to B)^D$ given in Section 2.2, we know that the witnesses for this interpretation are terms t_W, t_x, t_V with only U, Y, Z free satisfying:

$$\forall U, Y, Z\Big(\neg\neg A_D\big(t_x, U(t_x, t_V), t_V(U(t_x, t_V)\big) \to \\ \neg\neg A_D\big(Y(t_W), t_W(Y(t_W)), Z(t_W))\big)\Big).$$

Therefore, it is sufficient for the terms to satisfy (3.3). $\square$

3.2 Spector's bar recursion

We extend system WE-HA$^\omega$ by new constants

$$\mathbf{B}_{\sigma,\tau} : \ 0 \to (0 \to \sigma) \to \big((0 \to \sigma) \to 0\big) \to \big(0 \to (0 \to \sigma) \to \tau\big) \to \\ \big(0 \to (0 \to \sigma) \to (\sigma \to \tau) \to \tau\big) \to \tau.$$

for any types σ and τ, together with the following defining axioms:

$$\mathsf{BR}_{\sigma,\tau} : \begin{cases} \omega(\overline{s,n}) <_0 n \to \mathbf{B}_{\sigma,\tau} n s \omega g \phi =_\tau gn(\overline{s,n}), \\ \omega(\overline{s,n}) \geq_0 n \to \mathbf{B}_{\sigma,\tau} n s \omega g \phi =_\tau \\ \qquad\qquad \phi\big(n, (\overline{s,n}), \lambda x^\sigma.\mathbf{B}_{\sigma,\tau}(\mathbf{S}n)(\overline{[s](n) * x})\omega g \phi\big), \end{cases}$$

where:

$$\begin{aligned} &n : 0, \\ &s : 0 \to \sigma, \\ &\omega : (0 \to \sigma) \to 0, \\ &g : 0 \to (0 \to \sigma) \to \tau, \\ &\phi : 0 \to (0 \to \sigma) \to (\sigma \to \tau) \to \tau. \end{aligned}$$

Recall that here $\overline{s,n}$ refers to $[s](n) * \mathbf{0}^{0 \to \sigma}$, i.e., sequence s cut at n and extended with zeros.

BR is the schema formed by $\mathsf{BR}_{\sigma,\tau}$ for all types σ, τ. In WE-HA$^\omega$ + BR, the following is derivable:

$$\forall \omega^{(0 \to \sigma) \to 0}, s^{0 \to \sigma} \exists n^0 \big(\omega(\overline{s,n}) < n\big). \tag{3.4}$$

For a discussion on models of bar recursion, see [26]. Here, it is enough to observe that (3.4) actually makes sense if we think of ω as a computable functional, since then ω just sees a finite part of s, say $[s](m)$ for some natural number m. Therefore, $\omega(s) = \omega(\overline{s,m}) = \omega(\overline{s,k})$ for any $k \geq m$, and hence taking $n := \max\{m, \omega(s)\} + 1$, we get:

$$\omega(\overline{s,n}) = \omega(\overline{s,m}) = \omega(s) < n.$$

In order to satisfy (3.3), we actually will only use a particular form of bar recursion defined by:

$$\mathsf{SBR}_\sigma ns\omega\varepsilon := \begin{cases} \overline{s,n} & \text{if } \omega(\overline{s,n}) < n \\ \mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{[s](n) * a})\omega\varepsilon & \text{otherwise,} \end{cases}$$

where $a := \varepsilon n\big(\lambda x^\sigma.\mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{[s](n) * x})\omega\varepsilon\big)$.

This can be defined from $\mathbf{B}_{\sigma,0\to\sigma}$ by:

$$\mathsf{SBR}_\sigma ns\omega\varepsilon :=_{0\to\sigma} \mathbf{B}_{\sigma,0\to\sigma} ns\omega g\check{\varepsilon},$$

where $gns := \overline{s,n}$ and for each $i : 0$,

$$\check{\varepsilon}nspi := \begin{cases} s(i) & \text{if } i < n \\ p(\varepsilon np)i & \text{otherwise.} \end{cases}$$

That is, $\check{\varepsilon}nsp$ is the sequence $[s](n)$ concatenated with $\langle p(\varepsilon np)\rangle_{i\geq n}$.

Notice that, by definition, if $i < n$ we have that:

$$\mathsf{SBR}_\sigma ns\omega\varepsilon i = s(i). \tag{3.5}$$

Notation 3.4. Sometimes we write $\mathsf{SBR}_\sigma^{\omega,\varepsilon} ns$ instead of $\mathsf{SBR}_\sigma ns\omega\varepsilon$.

The following two lemmas, also due to Spector, will be useful to give solutions to (3.3). Let us write n' for $\mathbf{S}n$.

Lemma 3.5. *Given $\omega : (0 \to \sigma) \to 0$ and $\varepsilon : 0 \to \big(\sigma \to (0 \to \sigma)\big) \to \sigma$, let $\alpha := \mathsf{SBR}_\sigma^{\omega,\varepsilon}\mathbf{0}^0\mathbf{0}^{0\to\sigma}$. Then, for all $n : 0$,*

$$\alpha = \mathsf{SBR}_\sigma^{\omega,\varepsilon} n(\overline{\alpha,n}).$$

Proof. We proceed by induction on n. The case $n = 0$ is clear by definition. Now assume $n \geq 0$ and let us prove it for n'. The inductive hypothesis says:

$$\alpha = \mathsf{SBR}_\sigma^{\omega,\varepsilon} n(\overline{\alpha,n}).$$

We consider two cases.

Case 1. $\omega(\overline{\alpha,n}) < n$.

Then $\alpha = \mathsf{SBR}_\sigma^{\omega,\varepsilon} n(\overline{\alpha,n}) = \overline{\alpha,n}$, hence $\alpha n = \mathbf{0}$, so $\overline{\alpha,n} = \overline{\alpha,n'}$ and:

$$\omega(\overline{\alpha,n'}) = \omega(\overline{\alpha,n}) < n < n'.$$

Therefore,

$$\mathsf{SBR}_\sigma^{\omega,\varepsilon} n'(\overline{\alpha,n'}) = \overline{\alpha,n'} = \overline{\alpha,n} = \alpha.$$

Case 2. $\omega(\overline{\alpha,n}) \geq n$.

Then:

$$\alpha = \mathsf{SBR}_\sigma^{\omega,\varepsilon} n(\overline{\alpha,n}) = \mathsf{SBR}_\sigma^{\omega,\varepsilon} n'(\overline{[\alpha](n) * a}),$$

where $a = \varepsilon n\big(\lambda x^\sigma.\mathsf{SBR}_\sigma^{\omega,\varepsilon} n'(\overline{[\alpha](n) * x})\big)$. Thus $\alpha n = a$, whence:

$$\alpha = \mathsf{SBR}_\sigma^{\omega,\varepsilon} n'(\overline{\alpha,n'}). \qquad \square$$

Lemma 3.6. *Given* $\omega : (0 \to \sigma) \to 0$ *and* $\varepsilon : 0 \to \big(\sigma \to (0 \to \sigma)\big) \to \sigma$*, let* $\alpha := \mathsf{SBR}^{\omega,\varepsilon}_{\sigma} \mathbf{0}^0 \mathbf{0}^{0\to\sigma}$ *and* $n := \omega(\alpha)$*. Then,* $\omega(\overline{\alpha, n}) \geq n$ *and moreover:*

$$\alpha n = \varepsilon n\big(\lambda x^{\sigma}.\mathsf{SBR}^{\omega,\varepsilon}_{\sigma} n'(\overline{[\alpha](n) * x})\big).$$

Proof. Assume towards a contradiction that $\omega(\overline{\alpha, n}) < n$. Then, using Lemma 3.5,

$$\alpha = \mathsf{SBR}^{\omega,\varepsilon}_{\sigma} n(\overline{\alpha, n}) = \overline{\alpha, n}.$$

But then $n = \omega(\alpha) = \omega(\overline{\alpha, n}) < n$, a contradiction. Hence $\omega(\overline{\alpha, n}) \geq n$. Now,

$$\alpha n = \mathsf{SBR}^{\omega,\varepsilon}_{\sigma} n(\overline{\alpha, n}) n = \mathsf{SBR}^{\omega,\varepsilon}_{\sigma} n'(\overline{[\alpha](n) * a}) n = a,$$

which by definition is $\varepsilon n\big(\lambda x^{\sigma}.\mathsf{SBR}^{\omega,\varepsilon}_{\sigma} n'(\overline{[\alpha](n) * x})\big)$. □

3.3 Interpretation of DNS

We are ready to give witnesses for the interpretation of DNS by means of bar recursion. Recall that we only need solutions t_x, t_W, t_V of the system of equations (3.3), i.e.,

$$\begin{cases} t_x = Y(t_W) \\ U(t_x, t_V) = t_W(Y(t_W)) \\ t_V(U(t_x, t_V)) = Z(t_W). \end{cases}$$

In order to make clearer the relation with the previous section, we rename variables as follows: $t_x \mapsto n$, $Y \mapsto \omega$, $t_W \mapsto \alpha$, $U \mapsto \varepsilon$, $t_V \mapsto p$ and $Z \mapsto q$. Therefore, the system reads:

$$\begin{cases} n = \omega(\alpha) \\ \varepsilon n p = \alpha(\omega\alpha) \\ p(\varepsilon n p) = q\alpha. \end{cases} \tag{3.6}$$

Theorem 3.7 (Spector, [43])**.** *Given* $\omega : (0 \to \sigma) \to 0$*,* $q : (0 \to \sigma) \to (0 \to \sigma)$*, and* $\varepsilon : 0 \to \big(\sigma \to (0 \to \sigma)\big) \to \sigma$*, define* $\tilde{\varepsilon} m v := \varepsilon m\big(\lambda x.q(vx)\big)$*. Then,*

$$\begin{aligned} \alpha &:= \mathsf{SBR}^{\omega,\tilde{\varepsilon}}_{\sigma} \mathbf{0}^0 \mathbf{0}^{0\to\sigma}, \\ n &:= \omega\alpha, \\ p &:= \lambda x^{\sigma}.q(E_n x), \end{aligned}$$

where $E_m := \lambda x^{\sigma}.\mathsf{SBR}^{\omega,\tilde{\varepsilon}}_{\sigma} m'(\overline{[\alpha](m) * x})$*, satisfy* (3.6).

Proof. The first equation of (3.6) holds by definition. For the second one, notice that by Lemma 3.6,

$$\alpha n = \tilde{\varepsilon} n E_n =: a,$$

and using Lemma 3.5,

$$\alpha = \mathsf{SBR}^{\omega,\tilde{\varepsilon}}_{\sigma} n'(\overline{[\alpha](n) * a}) = E_n a.$$

Therefore, $\varepsilon np = \tilde{\varepsilon} n E_n = \alpha n$. Finally,

$$p(\varepsilon np) = q\big(E_n(\varepsilon np)\big) = q\big(E_n(\alpha n)\big) = q(E_n a) = q\alpha.$$

□

Thus, we have achieved an extension of Dialectica (actually of *ND*) to analysis.

Theorem 3.8 (Spector, [43]). *Let $A(\boldsymbol{a})$ be a formula in the language of* WE-HA$^\omega$ *containing only $\boldsymbol{a}$ free. If*

$$\text{WE-PA}^\omega + \mathsf{QF\text{-}AC} + \mathsf{AC}^0 \vdash A(\boldsymbol{a}),$$

then there is a tuple of closed terms $\boldsymbol{t}$ of WE-HA$^\omega + \mathsf{BR}$ *such that*

$$\text{WE-HA}^\omega + \mathsf{BR} \vdash \forall \boldsymbol{y} (A^N)_D(\boldsymbol{ta}, \boldsymbol{y}, \boldsymbol{a}).$$

Moreover, $\boldsymbol{t}$ can be effectively extracted from a proof of the assumption.

Proof. The discussion in Section 3.1, together with Proposition 2.16, shows that WE-PA$^\omega + \mathsf{QF\text{-}AC} + \mathsf{AC}^0$ is interpreted by N in WE-HA$^\omega + \mathsf{QF\text{-}AC} + \mathsf{AC}^0 + \mathsf{MP} + \mathsf{DNS}$, and hence in WE-HA$^\omega + \mathsf{AC} + \mathsf{MP} + \mathsf{DNS}$. Hence, we only need to extend the proof of soundness for Dialectica to DNS. This is done in Theorem 3.7. □

Chapter 4

Sequential games

In a series of recent papers [10–13], Martín Escardó and Paulo Oliva presented a new mathematical definition of game, whose primary purpose was to supply a meaningful semantics to bar recursion, but which was so general that its wide range and interest quickly became clear. This chapter is devoted to presenting this theory of sequential games, and to give the reader a glimpse of the generality of its semantics and applications. Chapter 5 explains its relation with bar recursion and the interpretation of analysis by its means.

Sequential games, in our context, are meant to represent situations where several choices must be made one after another. Let us suppose that the number of choices is fixed as $n \in \mathbb{N}$. We will have a set of possible moves (choices) X_i at each round $i < n$, and a set R of possible outcomes of the game. A play will be a sequence of elements $\boldsymbol{x} : \prod_{i=0}^{n-1} X_i$, intended to mean the moves that have been actually picked. The definition of game will also require an outcome function $p : \prod_{i=0}^{n-1} X_i \to R$ taking plays to its result. The number of players and the order as they alternate is left implicit in functions $\phi_i : (X_i \to R) \to R$ for each round i that say which is, at round i, the most desirable among the outcomes that are possible given the conditions of the game. We note that ϕ_i takes as an argument a function $q : X_i \to R$, i.e., a function that can be considered a 'local outcome function' for round i. The next sections will clarify how these functions behave.

For instance, assume that we want to express, through this definition of game, a game between two players that alternate, where the possible outcomes are $R = \{-1, 0, 1\}$ and Player A plays at even rounds and aims for a 1, Player B plays at odd rounds and aims for a -1, and 0 stands for a draw. Our way to encode this is to have functions ϕ_i expressing, for even i, that the goal is to obtain a 1 if possible, a 0 if not, while for odd i, ϕ_i expresses that the goal is to obtain a -1 if possible, a 0 if not. More concretely, if we assume that we have a function $q : X_i \to R$ assigning, to each possible move at round i, the outcome of the game if that move is picked, then $\phi_i(q)$ is the best possible outcome for the current player: e.g., if i is even, $\phi_i(p) = 1$ if Player A has a winning strategy (i.e., can obtain a 1 no matter how B plays), $\phi_i(p) = 0$ if A has not a winning strategy but can obtain a draw no matter how B plays, and $\phi(p) = -1$ if there is no way for A to avoid losing (assuming that B does the best he or she can do).

Therefore, we think that our players always want to win, that they are informed about the preferences of the other players, and that they always assume that the other players will do the best they can.

Above, we have assumed that the number of rounds of the game is a fixed natural number n, but in Section 4.6 we will define a variant of games whose number of rounds is not determined at the beginning, as it depends on the development of the play.

This chapter is carried out in the framework of standard mathematics. Our presentation combines content from [10–13].

4.1 Finite sequential games

Let us assume that we are given a sequence of non-empty sets $\langle X_i : i < \omega \rangle$. We understand each X_i as a set of possible moves at the ith round of a sequential game.

Definition 4.1. Given a natural number n and non-empty sets $X_0, \ldots, X_{n-1}$, an n-**play** is an element $s \in \prod_{i=0}^{n-1} X_i$. A **play** is an n-play for some $n \in \mathbb{N}$.

In the style of type theory, we shall write $s : \prod_{i=0}^{n-1} X_i$. When $n = 0$, the set $\prod_{i=0}^{n-1} X_i$ is interpreted as the set that only contains the empty sequence.

Defining a game also requires deciding what the outcomes of the possible moves and the intentions of the players are. Thus, let us further suppose that we are given a non-empty set R of possible outcomes of the game.

Definition 4.2. Given a natural number n and non-empty sets $X_0, \ldots, X_{n-1}$ and R, an **outcome function** for n-plays, or simply outcome function, is any function $p : \prod_{i=0}^{n-1} X_i \to R$.

Thus, an outcome function is a function taking each play to its result. Now we have to take care of the crucial step of defining what kind of functions will express the intentions of the players in our games. The notion that captures this will be that of *generalized quantifier*.

Definition 4.3. Given non-empty sets X and R, a **generalized quantifier** is a function $\phi : (X \to R) \to R$. We abbreviate this type as:

$$K_R X := (X \to R) \to R.$$

When no confusion is possible, we will omit the subscript R, and thus write simply KX.

Now we can give our definition of finite game.

Definition 4.4. Let n be a natural number. An n**-round game** is a tuple $\big(\langle X_i\rangle_{i=0}^{n-1}, p, \phi\big)$, where $R, X_0, \ldots, X_{n-1}$ are non-empty sets,

$$p : \prod_{i=0}^{n-1} X_i \to R$$

is an outcome function and

$$\phi : \prod_{i=0}^{n-1} KX_i$$

is a finite sequence of quantifiers, one for each round.

Following the notation by Escardó and Oliva, we do not write the set R explicitly in the tuple.

Below we give several examples in order to clarify the concept of finite game.

Example 4.5. Let us consider a game with a single round, set of possible moves X and outcome function $p : X \to R$, where $R = \{0, 1\}$ and the goal of the player is to get a 1. Then, the quantifier ϕ for the only round expresses the best outcome that the player can obtain. That is, if there is a move $x : X$ such that $p(x) = 1$, then $\phi(p) = 1$; otherwise, $\phi(p) = 0$. Notice that, if we think of R as the boolean values, where 0 stands for false and 1 stands for true, then p can be understood as a predicate on X, and then:

$$\phi(p) \equiv \exists x p(x).$$

That is where the name "quantifier" comes from.

Example 4.6. The usage of the word "generalized" is justified by the fact that R can be different from the booleans. Let $X = [0, 1]$ and $R = \mathbb{R}$. Define for $p : [0, 1] \to \mathbb{R}$:

$$\phi(p) := \begin{cases} \max_{x\in[0,1]} p(x) & \text{if the maximum exists} \\ p(0) & \text{otherwise.} \end{cases}$$

Then $\phi : K_{\mathbb{R}}[0, 1]$ is a generalized quantifier.[1]

Example 4.7. Let again $X = [0, 1]$ and $R = \mathbb{R}$. For $p : [0, 1] \to \mathbb{R}$ define:

$$\phi(p) := \begin{cases} \int_0^1 p & \text{if the integral exists} \\ p(0) & \text{otherwise.} \end{cases}$$

Then $\phi : K_{\mathbb{R}}[0, 1]$ is a generalized quantifier.

[1] We could avoid the case distinction if we restrict ourselves to, for instance, p continuous, so that Weierstrass' extreme value theorem ensures the existence of the maximum. But if we want to make this restriction formal, we would need some flexible definition of the type $[0, 1] \to \mathbb{R}$. A category theoretical reading is suitable in this context. For details on this see [11].

Example 4.8. In the informal example of a 2-player game at the beginning of this chapter, we had $R = \{-1, 0, 1\}$. The goal of Player A is to maximize (with respect the order $-1 < 0 < 1$) the outcome: to get a 1 if possible; if not, a 0; and only if this is again not possible, a -1. Similarly, the goal of player B is to minimize the outcome. Let us assume that there are only two rounds. Here, our outcome function p is of type $X \times Y \to R$, where $X := X_0$, $Y := X_1$ are the non-empty sets of possible moves for A and B respectively. What about the quantifiers? Let us first analyse ψ, the quantifier of B. Player B aims for a minimal outcome, and to B the move of A is given. So, if the move chosen by A is $x : X$, then we have that B should aim for the outcome:

$$\min\{p(x, y) : y \in Y\}.$$

Therefore, we can write $\psi := \min_Y$, and then the best possible outcome for B given the move $x : X$ of A is:

$$\psi(p_x) = \min_Y p_x = \min\{p(x, y) : y \in Y\}.$$

On the other hand, A must maximize the outcome of the game, knowing that B will take the move minimizing it. So the best possible outcome for A is:

$$\max\{\min\{p(x, y) : y \in Y\} : x \in X\}.$$

If we write $\phi := \max_X$ for the generalized quantifier of A, then the outcome of the game obtained if all players do the best they can is:

$$\phi\big(\lambda x.\psi(p_x)\big) = \max_X\big(\lambda x.\psi(p_x)\big) = \max\{\min\{p(x, y) : y \in Y\} : x \in X\}.$$

This motivates the definition of the product of quantifiers below, as a means for obtaining this 'best possible outcome of the game' through a single quantifier of type $K_R(X \times Y)$.

Definition 4.9. Let R, X, Y be non-empty sets. Given generalized quantifiers $\phi : K_R X$ and $\psi : K_R Y$, we define a quantifier $(\phi \otimes \psi) : K_R(X \times Y)$ as:

$$(\phi \otimes \psi)(p^{X \times Y \to R}) := \phi\big(\lambda x.\psi(p_x)\big) = \phi\big(\lambda x.\psi(\lambda y.p(x, y))\big).$$

Of course, we can iterate this construction any finite number of times, and we obtain:

Definition 4.10. Let $R, X_0, \ldots, X_{n-1}$ be non-empty sets. Given quantifiers $\phi_i : K_R X_i$ for $i = 0, \ldots, n-1$, we define recursively for each $k < n - 1$:

$$\bigotimes_{i=k}^{n-1} \phi_i := \phi_k \otimes \bigotimes_{i=k+1}^{n-1} \phi_i,$$

and:

$$\bigotimes_{i=n-1}^{n-1} \phi_i := \phi_{n-1}.$$

That is, $\bigotimes_{i=0}^{n-1} \phi_i = \phi_0 \otimes \ldots \otimes \phi_{n-1}$, where we understand that $\otimes$ associates to the right.

4.2 Selection functions

Now we shall introduce the notion of selection function. We have seen that a quantifier tells us what the outcome of the game is, assuming that the players pick the best possible move for them at each round. We are interested in functions of type $\varepsilon : (X \to R) \to X$, whose intended reading is that, given an outcome function $p : X \to R$, $\varepsilon(p)$ will tell us which move is best, that is, $p(\varepsilon p) = \phi p$.

This is not always possible, as we show in Example 4.14, but there are quantifiers that admit an associated selection function. These quantifiers are called attainable and, in fact, given any selection function, we can define the associated attainable quantifier.

Definition 4.11. Let R and X be non-empty sets. A **selection function** is a function of type $\varepsilon : (X \to R) \to X$. We abbreviate this type as:

$$J_R X := (X \to R) \to X.$$

When no confusion is possible, as before, we will omit the subscript R, and thus write simply JX.

If a quantifier is associated to a selection function, we call it attainable:

Definition 4.12. Let R and X be non-empty sets. A quantifier $\phi : K_R X$ is **attainable** if there is a selection function $\varepsilon : J_R X$ such that for all $p : X \to R$, $p(\varepsilon p) = \phi p$.

Given a selection function $\varepsilon : J_R X$, we define its **associated** (attainable) quantifier $\overline{\varepsilon}$ as:

$$\overline{\varepsilon}(p) := p(\varepsilon p)$$

for every $p : X \to R$.

Example 4.13. We recall Example 4.5, where $R = \{0, 1\}$ is interpreted as the boolean values and we define $\phi(p) \equiv \exists x p(x)$. Let us assume, for instance, that X is finite and that in our context all $p : X \to R$ are computable. Let $x_0 : X$ be some fixed element. We have a computable selection function defined as:

$$\varepsilon(p) := \begin{cases} x & \text{for some } x : X \text{ such that } p(x) = 1 \\ x_0 & \text{if there is no such } x. \end{cases}$$

Note that $p(\varepsilon p) = \phi(p)$. This selection function says which move we should pick: if there is an $x : X$ such that $p(x) = 1$, then pick that x and win; if not, then pick x_0, since we will lose anyway. Thus, ϕ is an attainable quantifier.

Example 4.14. Let $X = (0, 1)$ and $R = \mathbb{R} \cup \{\infty\}$. For $p : (0, 1) \to \mathbb{R} \cup \{\infty\}$, define:

$$\phi(p) := \sup_{x \in (0,1)} p(x),$$

where we define $\sup_{x \in (0,1)} p(x) = \infty$ either if $p(x) = \infty$ for some $x \in (0, 1)$ or if the image of p is contained and unbounded in $\mathbb{R}$. Then, ϕ is a non-attainable generalized quantifier, since there are functions p such that $\phi(p) = \infty$ but there is no $x \in (0, 1)$ with $p(x) = \infty$. Take, for instance, $p(x) = \frac{1}{x}$.

Example 4.15. I propose to my friends John and Michaela the following 2-round game. Consider the arithmetical expression:

$$a^2 + b + 2ab.$$

The set of moves for both players is $X = Y = \{-1, 0, 1\}$. We understand that when John chooses a move $x \in X$, he is setting a value for a, while Michaela sets a value for b. This is reflected by our outcome function, which is defined, for all $(x, y) : X \times Y$, as $p(x, y) = x^2 + y + 2xy$. Notice that the set of possible outcomes here is then:

$$R = \{-2, -1, 0, 1, 2, 4\}.$$

We conceive R as a linear order as expected with $-2 < -1 < 0 < 1 < 2 < 4$. John's goal is to maximize the outcome, while Michaela's is to minimize it. If the final outcome is positive, John wins, and if it is negative, Michaela does. A zero outcome means a draw.

So John's quantifier is $\phi = \max_X$, while Michaela's is $\psi = \min_Y$. But they are not satisfied with a quantifier: they want a way of selecting their moves. It is John's turn and, since he is a careful player, he studies his possibilities. But, even when it is not her turn yet, Michaela is already thinking about what she will do. She thinks:

- If John gives a -1, then the expression simplifies as $1 + b - 2b = 1 - b$. So Michaela shall work with the outcome function:

  $$p_{-1}(b) = 1 - b.$$

  In this case the minimum is:

  $$\psi(p_{-1}) = \min_Y\{1 - b : b \in Y\} = 0.$$

  But that is not the only thing that Michaela wants to know. She wants to know which b she should pick in order to attain this minimum. In this case, this value of b is 1.

- If John gives a 0, then we get:

  $$p_0(b) = b.$$

  In this case

  $$\psi(p_0) = \min_Y\{b : b \in Y\} = -1$$

  and it is realized by $b = -1$.

- Finally, if John gives a 1, then:

  $$p_1(b) = 1 + b + 2b = 1 + 3b.$$

  Then,

  $$\psi(p_1) = \min_Y\{1 + 3b : b \in Y\} = -2$$

  and it is realized by $b = -1$.

So, even before John selects his move, Michaela knows how she is going to reply. In case John takes $x : X$, Michaela takes:

$$\delta(p_x) := \text{the } y \in Y \text{ such that } p_x(y) \text{ is minimum.}$$

(Of course this function is recursive, since all sets involved are finite.) In this case, we have seen that $\delta(p_x) = b(x)$ given by:

$$b(x) := \begin{cases} 1 & \text{if } x = -1 \\ -1 & \text{if } x = 0 \text{ or } x = 1 \end{cases}$$

Now, of course, John has also thought of all this. His quantifier (his goal) suggests him that he should take the move that maximizes the minimum that Michaela will try to get. He knows that if he takes $x : X$, Michaela will take $b(x) : Y$. Hence, he knows that the outcome of the game will be $p(x, b(x))$. His selection function, $\varepsilon : JX$, is defined by:

$$\varepsilon(q^{X \to R}) = \text{the } x \in X \text{ such that } q(x) \text{ is maximal.}$$

And so the move he takes is:

$$a = \varepsilon\big(\lambda x.p(x, b(x))\big) = -1,$$

which gives the optimal outcome for him:

$$(\phi \otimes \psi)(p) = \max_X \min_Y(p) = 0.$$

So, in this game, if we assume that the second player does the best she can do, the first one can get at most a draw.

Now we define the product of selection functions, which will make us able to select two moves at once.

Definition 4.16. Given selection functions $\varepsilon : J_R X$ and $\delta : J_R Y$, we define a selection function $(\varepsilon \otimes \delta) : J_R(X \times Y)$ as:

$$(\varepsilon \otimes \delta)(p^{X \times Y \to R}) := \langle a, b(a) \rangle,$$

where:

$$b(x) := \delta(p_x) = \delta\big(\lambda y.p(x, y)\big),$$
$$a := \varepsilon\big(\lambda x.p(x, b(x))\big).$$

As in the case of the quantifiers, we can iterate this construction.

Definition 4.17. Given selection functions $\varepsilon_i : J_R X_i$ for $i = 0, \ldots, n-1$, we define recursively for each $k < n-1$:

$$\bigotimes_{i=k}^{n-1} \varepsilon_i := \varepsilon_k \otimes \bigotimes_{i=k+1}^{n-1} \varepsilon_i,$$

and:

$$\bigotimes_{i=n-1}^{n-1} \varepsilon_i := \varepsilon_{n-1}.$$

That is, $\bigotimes_{i=0}^{n-1} \varepsilon_i = \varepsilon_0 \otimes \ldots \otimes \varepsilon_{n-1}$, where we understand again that $\otimes$ associates to the right.

Our first theorem states that the attainable quantifier associated to the product of two selection functions coincides with the product of the attainable quantifiers associated to each one of them.

Theorem 4.18 ([11])**.** *Let $\varepsilon : JX$ and $\delta : JY$. Then:*

$$\overline{\varepsilon \otimes \delta} = \overline{\varepsilon} \otimes \overline{\delta}.$$

Proof. Just unfolding the definitions, we get that both quantifiers encode the optimal outcome $p(a, b(a))$.

$$\overline{\varepsilon \otimes \delta}(p^{X \times Y \to R}) = p((\varepsilon \otimes \delta)p) = p(a, b(a)),$$

$$(\overline{\varepsilon} \otimes \overline{\delta})(p) = \overline{\varepsilon}(\lambda x.\overline{\delta}(p_x)) = \overline{\varepsilon}(\lambda x.p_x(\delta p_x)) = \overline{\varepsilon}(\lambda x.p(x, b(x))) = p(a, b(a)).$$

□

Using the above theorem and a straightforward induction, we obtain:

Theorem 4.19. *Let $\varepsilon : \prod_{i=0}^{n-1} JX_i$ be a sequence of selection functions. Then,*

$$\overline{\bigotimes_{i=0}^{n-1} \varepsilon_i} = \bigotimes_{i=0}^{n-1} \overline{\varepsilon_i}.$$

4.3 Optimal strategies

This section is devoted to computing optimal strategies in our finite sequential games. The definition of optimal outcome and move will rely on the product of quantifiers, while the computation of explicit optimal strategies will be possible using the product of selection functions.

First, we need some definitions. Let $\mathcal{G} = \left(\langle X_i \rangle_{i=0}^{n-1}, p, \phi\right)$ be an n-round game.

(a) A **partial play** is a sequence $\boldsymbol{a} : \prod_{i=0}^{k-1} X_i$ with $k \leq n$. Given a partial play $\boldsymbol{a} : \prod_{i=0}^{k-1} X_i$, we can define a **subgame** of $\mathcal{G}$, which is an $(n-k)$-round game, as:

$$\big(\langle X_i \rangle_{i=k}^{n-1}, p_{\boldsymbol{a}}, \langle \phi_i \rangle_{i=k}^{n-1}\big).$$

This game is like the original one but it starts at the position determined by the partial play.

(b) The **optimal outcome** of the game is:

$$w := \left(\bigotimes_{i=0}^{n-1} \phi_i \right) (p).$$

Given a partial play $\boldsymbol{a} : \prod_{i=0}^{k-1} X_i$, with $k < n$, we define:

$$w_{\boldsymbol{a}} := \left(\bigotimes_{i=k}^{n-1} \phi_i \right) (p_{\boldsymbol{a}}).$$

If $\boldsymbol{a}$ is an n-play, then $w_{\boldsymbol{a}} := p(\boldsymbol{a})$. We note that $w = w_{\langle\rangle}$. If $k = n-1$, then:

$$w_{\boldsymbol{a}} = \phi_{n-1}\big(\lambda x_{n-1}.p_{\boldsymbol{a}}(x_{n-1})\big) = \phi_{n-1}(\lambda x_{n-1}.w_{\boldsymbol{a} * x_{n-1}}),$$

and if $k < n-1$, then:

$$w_{\boldsymbol{a}} = \phi_k \left(\lambda x_k. \left(\bigotimes_{i=k+1}^{n-1} \phi_i \right) (p_{\boldsymbol{a} * x_k}) \right) = \phi_k(\lambda x_k.w_{\boldsymbol{a} * x_k}).$$

(c) Given a partial play $\boldsymbol{a} : \prod_{i=0}^{k-1} X_i$ for $k < n$, an **optimal move** at round $k < n$ is a move a_k such that $w_{\boldsymbol{a}} = w_{\boldsymbol{a} * a_k}$, that is, a move that does not make worse the optimal outcome of the subgame.

(d) A play $\boldsymbol{a} = \langle a_0, \ldots, a_{n-1} \rangle$ is **optimal** if for each $k < n$, a_k is an optimal move with respect to the subgame determined by $\langle a_0, \ldots, a_{k-1} \rangle$. Therefore, the play $\boldsymbol{a}$ is optimal if and only if:

$$w_{\langle\rangle} = w_{\langle a_0 \rangle} = \ldots = w_{\langle a_0, \ldots, a_{n-1} \rangle}.$$

(e) A **strategy** is a family of functions:

$$\mathrm{next}_k : \prod_{i=0}^{k-1} X_i \to X_k,$$

for $k < n$, computing a move to be played at round k. That is, if the game is at position $\boldsymbol{a}$ and we are following the above strategy, then the next move selected is $a_k = \mathrm{next}_k(\boldsymbol{a})$.

(f) A strategy is **optimal** if for every $k < n$ and every $\boldsymbol{a} : \prod_{i=0}^{k-1} X_i$, the move $\mathrm{next}_k(\boldsymbol{a})$ is optimal at round k. More concisely, if $w_{\boldsymbol{a}} = w_{\boldsymbol{a} * \mathrm{next}_k(\boldsymbol{a})}$. Given an optimal strategy, we can define by induction an optimal play:

$$a_0 := \mathrm{next}_0(\langle\rangle), \qquad a_k := \mathrm{next}_k(\langle a_0, \dots, a_{k-1}\rangle)$$

Remark 4.20. Our games as defined above cannot model the situation where the set of allowed moves at round i depends on the moves played at previous rounds. We will see a way of solving this problem in Section 4.5.

For the remainder of the section we assume that each quantifier ϕ_i in our game is associated to a selection function ε_i.

Lemma 4.21 ([11])**.** *Let $\left(\langle X_i\rangle_{i=0}^{n-1}, p, \phi\right)$ be an n-round game. For $i = 0, \dots, n-1$, let us assume that ϕ_i is the associated quantifier of a selection function ε_i. For each $k < n$, define a function $\mathrm{next}_k : \prod_{i=0}^{k-1} X_i \to X_k$ as, for all $\boldsymbol{x} : \prod_{i=0}^{k-1} X_i$:*

$$\mathrm{next}_k(\boldsymbol{x}) := \varepsilon_k(\lambda x_k . w_{\boldsymbol{x} * x_k}).$$

Then, the strategy $\langle \mathrm{next}_k\rangle_{k=0}^{n-1}$ is an optimal strategy.

Proof. Fix $k < n$ and $\boldsymbol{a} : \prod_{i=0}^{k-1} X_i$. We have to show that $w_{\boldsymbol{a}} = w_{\boldsymbol{a} * \mathrm{next}_k(\boldsymbol{a})}$.

$$\begin{aligned} w_{\boldsymbol{a}} &= \phi_k(\lambda x_k . w_{\boldsymbol{a} * x_k}) = (\lambda x_k . w_{\boldsymbol{a} * x_k})(\varepsilon_k(\lambda x_k . w_{\boldsymbol{a} * x_k})) \\ &= w_{\boldsymbol{a} * \varepsilon_k(\lambda x_k . w_{\boldsymbol{a} * x_k})} = w_{\boldsymbol{a} * \mathrm{next}_k(\boldsymbol{a})}. \end{aligned}$$

□

Theorem 4.22 ([11])**.** *Let $\left(\langle X_i\rangle_{i=0}^{n-1}, p, \phi\right)$ be an n-round game. For $i=0, \dots, n-1$, let us assume that ϕ_i is the associated quantifier of a selection function ε_i. The optimal strategy of Lemma 4.21 can be computed as, for all $\boldsymbol{x} : \prod_{i=0}^{k-1} X_i$:*

$$\mathrm{next}_k(\boldsymbol{x}) = \left(\left(\bigotimes_{i=k}^{n-1} \varepsilon_i\right)(p_{\boldsymbol{x}})\right)(0). \tag{4.1}$$

Proof. Recall that for an n-play $\boldsymbol{a}$ we have that $w_{\boldsymbol{a}} = p(\boldsymbol{a})$. So for $k = n-1$ we have that:

$$\mathrm{next}_{n-1}(\boldsymbol{x}) = \varepsilon_{n-1}(\lambda x_{n-1} . w_{\boldsymbol{x} * x_{n-1}}) = \varepsilon_{n-1}(\lambda x_{n-1} . p(\boldsymbol{x} * x_{n-1})) = \varepsilon_{n-1}(p_{\boldsymbol{x}}),$$

which is equal to the right-hand side of (4.1) in case $k = n-1$.

If $k < n-1$, notice that by definition of the product:

$$\left(\bigotimes_{i=k}^{n-1} \varepsilon_i\right)(p_{\boldsymbol{x}}) = a_k * \left(\bigotimes_{i=k+1}^{n-1} \varepsilon_i\right)(p_{\boldsymbol{x} * a_k}),$$

where:

$$a_k := \varepsilon_k\left(\lambda x_k . p_{\boldsymbol{x}}\left(x_k, \left(\bigotimes_{i=k+1}^{n-1} \varepsilon_i\right)(p_{\boldsymbol{x} * x_k})\right)\right).$$

Now, using Theorem 4.19,

$$\mathrm{next}_k(\boldsymbol{x}) = \varepsilon_k(\lambda x_k.w_{\boldsymbol{x}*x_k}) = \varepsilon_k\left(\lambda x_k.\left(\bigotimes_{i=k+1}^{n-1}\phi_i\right)(p_{\boldsymbol{x}*x_k})\right) =$$

$$\varepsilon_k\left(\lambda x_k.p_{\boldsymbol{x}*x_k}\left(\left(\bigotimes_{i=k+1}^{n-1}\varepsilon_i\right)(p_{\boldsymbol{x}*x_k})\right)\right) = a_k = \left(\left(\bigotimes_{i=k}^{n-1}\varepsilon_i\right)(p_{\boldsymbol{x}})\right)(0).$$

□

Corollary 4.23 ([11])**.** *Let $(\langle X_i\rangle_{i=0}^{n-1}, p, \phi)$ be an n-round game. For $i=0,\ldots,n-1$, let us assume that ϕ_i is the associated quantifier of a selection function ε_i. Then, the play:*

$$\boldsymbol{a} := \left(\bigotimes_{i=0}^{n-1}\varepsilon_i\right)(p)$$

is optimal. In particular, $p(\boldsymbol{a}) = w$.

4.4 Applications

In this section, we present several applications of sequential games to different parts of mathematics. The aim is to illustrate the generality of the construction.

The examples have been extracted and adapted from [11, 12].

4.4.1 Nash equilibrium

In standard game theory, a game of n players is defined by a tuple:

$$(\langle X_i\rangle_{i=0}^{n-1}, \langle p_i\rangle_{i=0}^{n-1}),$$

where X_i is the set of possible moves for player i, and $p_i : \prod_{i=0}^{n-1} X_i \to R$ is the payoff function for player i. The set R comes with an order relation, and greater values are understood as better payoffs. A play, also known as a *strategy profile*, is an element $\boldsymbol{x} = \langle x_0,\ldots,x_{n-1}\rangle \in \prod_{i=0}^{n-1} X_i$. A Nash equilibrium is a strategy profile such that no player can improve her payoff by changing her move (assuming the other players keep their moves).

The product of selection functions can compute a Nash equilibrium. Our game according to Definition 4.4 has n rounds and the set of possible moves at round i is X_i. The set of possible outcomes is R^n, and the outcome function is $p : \prod_{i=0}^{n-1} X_i \to R^n$ defined as $p(\boldsymbol{x}) := \langle p_0(\boldsymbol{x}),\ldots,p_{n-1}(\boldsymbol{x})\rangle$.

Assume we have selection functions $\varepsilon_0,\ldots,\varepsilon_{n-1}$ such that for every i:

$$\varepsilon_i(q^{X_i\to R^n}) = x_i : X_i \text{ such that } q(x_i)(i) = \max_{x_i:X_i}\{q(x_i)(i)\}. \tag{4.2}$$

Then we can compute a Nash equilibrium for the game.

Theorem 4.24 ([12]). *Given a game* $\left(\langle X_i\rangle_{i=0}^{n-1}, \langle p_i\rangle_{i=0}^{n-1}\right)$ *of standard game theory, let* $p : \prod_{i=0}^{n-1} X_i \to R^n$ *be defined as, for all* $\boldsymbol{x} : \prod_{i=0}^{n-1} X_i$,

$$p(\boldsymbol{x}) := \langle p_0(\boldsymbol{x}), \ldots, p_{n-1}(\boldsymbol{x})\rangle.$$

Let us suppose that we have selection functions $\varepsilon_i : J_{R^n} X_i$, *for every* $i < n$, *satisfying* (4.2). *Consider a game* $\left(\langle X_i\rangle_{i=0}^{n-1}, p, \langle \overline{\varepsilon_i}\rangle_{i=0}^{n-1}\right)$. *The optimal play:*

$$\boldsymbol{x} := \left(\bigotimes_{i=0}^{n-1} \varepsilon_i\right)(p)$$

is a Nash equilibrium.

4.4.2 Fixed point theory

A *fixed point operator* is a map fix : $(R \to R) \to R$ such that for every $p : R \to R$:

$$\mathrm{fix}(p) = p(\mathrm{fix}(p)).$$

These operators do not exist if we interpret $R \to R$ as the set of all functions from R to R, but they do exist in some categories of domains.

Since, in this case, $J_R R = K_R R = (R \to R) \to R$, a fixed point operator can be considered both a quantifier and a selection function. Indeed, a function $f : (R \to R) \to R$ is a fixed point operator if and only if it is its own selection function:

$$f(p) = p(f(p)).$$

Bekič's Lemma [2] states that if X and Y have fixed point operators, then so does $X \times Y$. We now prove this theorem constructing the fixed point operator of the product as a product of selection functions.

Theorem 4.25 (Bekič's Lemma). *Let* $\pi_X : X \times Y \to X$ *and* $\pi_Y : X \times Y \to Y$ *denote the usual projections. Let* $\mathrm{fix}_X : J_X X$ *and* $\mathrm{fix}_Y : J_Y Y$ *be fixed point operators. Define* $\varepsilon_X : J_{X\times Y} X$ *and* $\varepsilon_Y : J_{X\times Y} Y$ *as, for every* $p : X \to X \times Y$ *and* $q : Y \to X \times Y$:

$$\varepsilon_X(p) := \mathrm{fix}_X(\pi_X \circ p), \quad \varepsilon_Y(q) := \mathrm{fix}_Y(\pi_Y \circ q).$$

Then, $X \times Y$ *has a fixed point operator* $\mathrm{fix}_{X\times Y} : J_{X\times Y}(X \times Y)$ *defined by:*

$$\mathrm{fix}_{X\times Y} := \varepsilon_X \otimes \varepsilon_Y.$$

Proof. Given $r : X \times Y \to X \times Y$, we need to show that $\mathrm{fix}_{X\times Y}(r) = r(\mathrm{fix}_{X\times Y}(r))$. Let $s : X \times Y \to X$ and $t : X \times Y \to Y$ be the components of r, that is, for all $x \in X$ and $y \in Y$, $r(x, y) = \langle s(x, y), t(x, y)\rangle$. Notice that:

$$\mathrm{fix}_{X\times Y}(r) = (\varepsilon_X \otimes \varepsilon_Y)(r) = \langle a, b(a)\rangle,$$

where:

$$b(x) = \varepsilon_Y(r_x) = \mathrm{fix}_Y(t_x),$$

$$a = \varepsilon_X(\lambda x.r(x, b(x))) = \mathrm{fix}_X(\lambda x.s(x, b(x))).$$

Then,

$$r(a, b(a)) = \langle s(a, b(a)), t(a, b(a))\rangle = \langle a, b(a)\rangle,$$

using that fix_X and fix_Y are fixed point operators. □

4.4.3 Algorithmics: backtracking

Backtracking algorithms can be expressed via the product of selection functions. Here we show the example of deciding the satisfiability of a propositional formula in classical logic. For other examples on this, see [11].

Let A be a propositional formula with n variables, $\boldsymbol{r} = \langle r_0, \dots, r_{n-1}\rangle$. Then we can define a game with the goal of finding a model satisfying the formula. The set of possible outcomes is $R = \{0, 1\}$, understood as the boolean values. We will have n rounds, each one deciding the value of one variable. So the set of allowed moves at round i is $X_i = \{0, 1\}$, with the intended reading that we are assigning the chosen value to the variable r_i. The outcome function:

$$p_A : \{0, 1\}^n \to \{0, 1\}$$

is defined, given $\boldsymbol{x} : \{0, 1\}^n$, as the evaluation of A under the interpretation where each r_i has value x_i.

We interpret all functions $f : \{0, 1\}^m \to \{0, 1\}$ for $m > 0$ as propositional formulas, and we say that f is satisfiable if there is $\boldsymbol{x} : \{0, 1\}^m$ such that $f(\boldsymbol{x}) = 1$.

For each round i, the quantifier $\phi_i : (\{0, 1\} \to \{0, 1\}) \to \{0, 1\}$ expresses that the goal is to get a 1, so for each $q : \{0, 1\} \to \{0, 1\}$,

$$\phi_i(q) := \phi(q) := \begin{cases} 1 & \text{if } q \text{ is satisfiable} \\ 0 & \text{otherwise.} \end{cases}$$

This quantifier has associated selection function:

$$\varepsilon_i(q) := \varepsilon(q) := \begin{cases} 0 & \text{if } q(0) = 1 \\ 1 & \text{otherwise.} \end{cases} \tag{4.3}$$

Theorem 4.26 ([11]). *Let $\left(\{0, 1\}^n, p_A, \langle\phi_i\rangle_{i=0}^{n-1}\right)$ be the game described above, and for each $i = 0, \dots, n-1$, let ε_i be as in (4.3). Then, if A is satisfiable,*

$$\boldsymbol{x} := \left(\bigotimes_{i=0}^{n-1} \varepsilon_i\right)(p_A)$$

is an assignment for $\boldsymbol{r}$ satisfying A.

4.5 Dependent products

The allowed moves at each round of a game can depend on several factors, such as the number of the current round or the development of each concrete play.

First of all, we observe that in the version of sequential games above the set of possible moves depends just on the round, since for each round i we have a different set of possible moves X_i. Our first observation is that this dependency can be easily avoided. We can have just a set of possible moves of the game, X, and encode which moves are forbidden at round i via the outcome function p and the quantifier ϕ_i. For instance, if a move $x : X$ is forbidden at round i, and if $r : R$ is a bad outcome for round i (in the sense that it is contrary to the corresponding quantifier ϕ_i), then we define p in such a way that, if move x is taken at round i, then the goal of round i is not attained: $p(\boldsymbol{x}) = r$ for every $\boldsymbol{x} : X^n$ with $\boldsymbol{x}(i) = x$. To do this, we need to have available more than one possible outcome, so $|R| > 1$. If several forbidden moves are taken at the same play, the outcome function must penalize the first one: once a player takes a forbidden move, she loses. For further details see [11].

The remainder of this section is devoted to treating another important case: when the intentions of the players, the way they alternate, or the possible moves for round i depend not only on the number of the round but also on the current state of the game, i.e., the moves taken at previous rounds. We encode this dependency by means of a dependency of the quantifiers, that will result in a dependency of the selection functions.

Here we reproduce previous definitions, taking now into account this new dependency.

Definition 4.27. Let R and $X_0, \dots, X_{n-1}$ be non-empty sets, and let $i < n$. A **dependent quantifier** is any function $\phi_i : \prod_{j=0}^{i-1} X_j \to K_R X_i$.

We notice that, given a dependent quantifier $\phi_i : \prod_{j=0}^{i-1} X_j \to KX_i$, we need a partial play $s : \prod_{j=0}^{i-1} X_j$ in order to have an ordinary quantifier $\phi_{i,s} : KX_i$.

By an abuse of notation, we sometimes write $\phi_0 : KX_0$ and $\phi_1 : X_0 \to KX_1$, instead of $\phi_0 : \{\langle\rangle\} \to KX_0$ and $\phi_1 : \prod_{j=0}^{0} X_j \to KX_1$.

Definition 4.28. Let R, X and Y be non-empty sets. Given dependent quantifiers $\phi : K_R X$ and $\psi : X \to K_R Y$, we define a quantifier $(\phi \otimes \psi) : K_R(X \times Y)$:

$$(\phi \otimes \psi)(p^{X\times Y \to R}) := \phi\big(\lambda x.\psi_x(p_x)\big) = \phi\big(\lambda x.\psi(x, \lambda y.p(x,y))\big).$$

The iterated product generalizes as expected, although the notation is awkward here.

Definition 4.29. Let R and $X_0, \dots, X_{n-1}$ be non-empty sets. Given (dependent) quantifiers $\phi_i : \prod_{j=0}^{i-1} X_j \to KX_i$ for $i = 0, \dots, n-1$, we recursively define for each $k < n$ a (dependent) quantifier:

$$\left(\bigotimes_{i=k}^{n-1} \phi_i\right) : \prod_{i=0}^{k-1} X_i \to K\left(\prod_{i=k}^{n-1} X_i\right)$$

as, for $k < n-1$ and $s : \prod_{i=0}^{k-1} X_i$:

$$\left(\bigotimes_{i=k}^{n-1} \phi_i\right)_s := \phi_{k,s} \otimes \left(\bigotimes_{i=k+1}^{n-1} \phi_i\right)_s$$

and for $k = n-1$:

$$\bigotimes_{i=n-1}^{n-1} \phi_i := \phi_{n-1}.$$

That is, $\bigotimes_{i=0}^{n-1} \phi_i = \phi_0 \otimes \ldots \otimes \phi_{n-1}$, where we understand that $\otimes$ associates to the right.

Selection functions have corresponding dependent versions as well.

Definition 4.30. Let R and $X_0, \ldots, X_{n-1}$ be non-empty sets, and let $i < n$. A **dependent selection function** is any function $\varepsilon_i : \prod_{j=0}^{i-1} X_j \to J_R X_i$.

Again, given a dependent selection function $\varepsilon_i : \prod_{j=0}^{i-1} X_j \to JX_i$, we need a partial play $s : \prod_{j=0}^{i-1} X_j$ in order to have an ordinary selection function $\varepsilon_{i,s} : JX_i$.

Definition 4.31. Let R, X and Y be non-empty sets. Given dependent selection functions $\varepsilon : J_R X$ and $\delta : X \to J_R Y$, we define a selection function $(\varepsilon \otimes \delta) : J_R(X \times Y)$ as:

$$(\varepsilon \otimes \delta)(p^{X \times Y \to R}) := \langle a, b(a) \rangle,$$

where:

$$b(x) = \delta_x(p_x) = \delta\big(x, \lambda y.p(x,y)\big),$$
$$a = \varepsilon\big(\lambda x.p(x, b(x))\big).$$

As in the case of the quantifiers, we can iterate this construction.

Definition 4.32. Let R and $X_0, \ldots, X_{n-1}$ be non-empty sets. Given (dependent) selection functions $\varepsilon_i : \prod_{j=0}^{i-1} X_j \to JX_i$ for $i = 0, \ldots, n-1$, we recursively define for each $k < n$ a (dependent) selection function:

$$\left(\bigotimes_{i=k}^{n-1} \varepsilon_i\right) : \prod_{i=0}^{k-1} X_i \to J\left(\prod_{i=k}^{n-1} X_i\right)$$

as, for $k < n-1$ and $s : \prod_{i=0}^{k-1} X_i$:

$$\left(\bigotimes_{i=k}^{n-1} \varepsilon_i\right)_s = \varepsilon_{k,s} \otimes \left(\bigotimes_{i=k+1}^{n-1} \varepsilon_i\right)_s$$

and for $k = n-1$:

$$\bigotimes_{i=n-1}^{n-1} \varepsilon_i = \varepsilon_{n-1}.$$

That is, $\bigotimes_{i=0}^{n-1} \varepsilon_i = \varepsilon_0 \otimes \ldots \otimes \varepsilon_{n-1}$, where we understand again that $\otimes$ associates to the right.

Theorem 4.18 still holds for this version of the product, as a quick look at the proof reveals.

The remaining sections of this chapter use dependent products to define unbounded games and the product of selection functions for them.

4.6 Unbounded games

As we previously announced, we are interested in games that have an unbounded, although finite, number of rounds. This property will allow for the emulation of bar recursion within our game-theoretic context.

Now we need to define plays as infinite sequences of moves:

Definition 4.33. Given non-empty sets X_i for each $i \in \mathbb{N}$, an **infinite play**, or simply a **play**, is an element $\alpha : \prod_{i=0}^{\infty} X_i$.

Thus, the outcome function needs to work over infinite plays.

Definition 4.34. Given non-empty sets X_i for each $i \in \mathbb{N}$, and a non-empty set R, an **outcome function** is any function $p : \prod_{i=0}^{\infty} X_i \to R$.

Again, X_i is the set of possible moves at round i. We will get rid of the subscript i later on, when we need to use unbounded games inside a formal system as WE-HA$^\omega$, since there the product type is not defined. Thus, the infinite plays will become of type $0 \to X$. For the moment, however, we assume that in each X_i there is a canonical move 0_i, and $\mathbf{0}$ means the sequence of the canonical elements of appropriate type (for instance, $\mathbf{0} : \prod_{i=n}^{\infty} X_i$ is the sequence $\langle 0_i \rangle_{i=n}^{\infty}$). This is done in order to have a canonical way of extending a finite play into an infinite play.

Although the games we will consider in this section shall be unbounded, we do require that they end. One way to ensure that plays end is imposing that the relevant part of the play, that is, the initial segment of the play that is used to determine the outcome of the game, is finite. Formally, this means that for every sequence $\alpha : \prod_{i=0}^{\infty} X_i$, there is an n such that $p(\alpha) = p([\alpha](n) * s)$ for every $s : \prod_{i=n}^{\infty} X_i$. We also can state this as:

$$\forall \alpha^{\prod_{i=0}^{\infty} X_i} \exists n^{\mathbb{N}} \forall \beta^{\prod_{i=0}^{\infty} X_i} \Big(\forall i < n \big(\alpha(i) = \beta(i) \big) \to p(\alpha) = p(\beta) \Big). \tag{4.4}$$

Of course this n now depends on the particular play, and so the number of rounds of the game depends on how it develops.

An option is to impose that R is discrete (as a topological space) and p is continuous, considering $\prod_{i=0}^{\infty} X_i$ with the usual product topology. This is a sufficient condition for (4.4). Details on this option can be found in [11].

However, we do not take condition (4.4). Another way to ensure that plays end is having an explicit control, by what we mean a function $\omega : \prod_{i=0}^{\infty} X_i \to \mathbb{N}$. The idea is that, given a play $\alpha : \prod_{i=0}^{\infty} X_i$, if $n = \omega(\alpha)$ we consider that the game is ended at round n, and so the relevant part of the play is $[\alpha](n)$. We notice that this does not guarantee that, given a play α, $p(\alpha) = p([\alpha](n) * \mathbf{0})$, nor that $\omega(\alpha) = \omega([\alpha](n) * \mathbf{0})$. That is, in principle ω determines the relevant part of the play based on the complete infinite play, and so the relevant part, and the outcome, of α and $[\alpha](n) * \mathbf{0}$ (and of any other extension of $[\alpha](n)$) may be different. In some contexts, however, we can impose further conditions[2]. Anyway, these conditions may be desirable but are not necessary. That the relevant part of the play according to ω is not conclusive to determine the outcome of the game may be an inconvenience (at least from a semantical viewpoint), but, as we shall see, it is not an impediment to define the unbounded version of our games. Therefore, our definition below does not include any conditions to ensure (4.4).

Definition 4.35. Given non-empty sets X_i for each $i \in \mathbb{N}$, an **explicit control**, or **control function**, is an object $\omega : \prod_{i=0}^{\infty} X_i \to \mathbb{N}$.

That is the last ingredient for the definition of unbounded games.

Definition 4.36. Let R and X_i for each $i \in \mathbb{N}$ be non-empty sets. An **unbounded game with explicit control**, or unbounded game for short, is a tuple $\left(\langle X_i\rangle_{i=0}^{\infty}, p, \phi, \omega\right)$, where:

$$p : \prod_{i=0}^{\infty} X_i \to R$$

is an outcome function;

$$\phi : \prod_{i=0}^{\infty} \left(\prod_{j=0}^{i-1} X_j \to K_R X_i \right)$$

is an infinite sequence of dependent quantifiers, one for each round, and:

$$\omega : \prod_{i=0}^{\infty} X_i \to \mathbb{N}$$

is a control function.

Remark 4.37. Given an outcome function $p : \prod_{i=0}^{\infty} X_i \to R$, a control function $\omega : \prod_{i=0}^{\infty} X_i \to \mathbb{N}$ and selection functions $\varepsilon : \prod_{i=0}^{\infty} \left(\prod_{j=0}^{i-1} X_j \to J_R X_i \right)$, we already can define an unbounded game $\left(\langle X_i\rangle_{i=0}^{\infty}, p, \overline{\varepsilon}, \omega\right)$, where $\overline{\varepsilon} ks := \overline{\varepsilon ks}$. In some cases, we will use the notation (p, ε, ω) to refer to this unbounded game.

Given an unbounded game $\mathcal{G} = \left(\langle X_i\rangle_{i=0}^{\infty}, p, \phi, \omega\right)$, we define the following versions of the finite-case concepts.

---

[2]If R is discrete and p is continuous, ω can be chosen as the implicit control function stated above, that is, for each α, $\omega(\alpha)$ is the least k such that $p([\alpha](k) * s) = p(\alpha)$ for every $s : \prod_{i=k}^{\infty}$. This definition of ω uses the axiom of choice.

Also, in contexts where the functionals allowed are computable in some sense, usually $p(\alpha)$ uses only a finite part of α to compute the result, and so (4.4) holds and an ω can be defined as before.

(a) A **partial play** is a sequence $\boldsymbol{a} : \prod_{i=0}^{k-1} X_i$ with $k \in \mathbb{N}$. Given a partial play $\boldsymbol{a} : \prod_{i=0}^{k-1} X_i$, we can define a **subgame** of $\mathcal{G}$, which will be still an unbounded game, as:

$$\big(\langle X_i \rangle_{i=k}^{\infty}, p_{\boldsymbol{a}}, \langle \phi_i \rangle_{i=k}^{\infty}, \omega_{\boldsymbol{a}}\big).$$

This game is like the original one but it starts at the position determined by the partial play.

(b) A **strategy** is a family of functions:

$$\mathrm{next}_k : \prod_{i=0}^{k-1} X_i \to X_k,$$

for $k \in \mathbb{N}$, computing a move to be played at round k. That is, if the game is at position $\boldsymbol{a}$ and we are following the above strategy, then the next move selected is $a_k = \mathrm{next}_k(\boldsymbol{a})$. The strategy inductively defines a **strategic extension** of $\boldsymbol{a}$ as:

$$\beta^{\boldsymbol{a}}(j) := \mathrm{next}_j(a_0, \ldots, a_{k-1}, \beta^{\boldsymbol{a}}(k), \ldots, \beta^{\boldsymbol{a}}(j-1))$$

for $j \geq k$. That is, $\boldsymbol{a} * \beta^{\boldsymbol{a}}$ is the play starting with $\boldsymbol{a}$ and completed following the strategy next.

(c) A strategy is **optimal** if for every partial play $\boldsymbol{a} : \prod_{i=0}^{k-1} X_i$ such that $k \leq \omega(\boldsymbol{a} * \beta^{\boldsymbol{a}})$ we have:

$$p(\boldsymbol{a} * \beta^{\boldsymbol{a}}) = \phi_{k,\boldsymbol{a}}(\lambda x_k . p(\boldsymbol{a} * x_k * \beta^{\boldsymbol{a} * x_k})).$$

4.7 Explicitly controlled product

The goal of this section is to define a product of selection functions that takes into account the fact that the game is unbounded. This product is defined by means of a functional named EPS, the acronym used in the literature after "explicitly controlled product of selection functions". As the following chapter discusses, this instance of the product is powerful enough to replace bar recursion on the task of interpreting analysis.

As in the case of bar recursion, EPS will be well-defined if we are in a setting of computable functionals, where the control function ω only sees a finite part of the sequence that takes as an argument. A condition like (3.4), which in the current context (with Cartesian products) can be stated as:

$$\forall \omega^{\prod_{i=0}^{\infty} X_i \to \mathbb{N}}, s^{\prod_{i=0}^{\infty} X_i} \exists n^{\mathbb{N}} \big(\omega(\overline{s, n}) < n\big),$$

is sufficient. For the complete discussion see Section 3.2. From now on we assume that we are in a setting where all functionals are computable. This will be more precise when we study EPS as a principle for the formal system WE-HA$^{\omega}$.

Definition 4.38. Given non-empty sets R and X_i for each $i \in \mathbb{N}$, we define, for each $k \in \mathbb{N}$, the functional EPS_k, which has type:

$$\prod_{i=0}^{k-1} X_i \to \left(\prod_{i=0}^{\infty} X_i \to \mathbb{N}\right) \to \prod_{i=0}^{\infty}\left(\prod_{j=0}^{i-1} X_j \to J_R X_i\right) \to J_R\left(\prod_{i=k}^{\infty} X_i\right),$$

and is defined as:

$$\mathsf{EPS}_k s\omega\varepsilon := \begin{cases} \lambda p^{\prod_{i=k}^{\infty} X_i \to R}.\mathbf{0} & \text{if } \omega(s * \mathbf{0}) < k \\ \varepsilon_{k,s} \otimes \lambda x^{X_k}.\mathsf{EPS}_{k+1}(s * x)\omega\varepsilon & \text{otherwise,} \end{cases}$$

where $s : \prod_{i=0}^{k-1} X_i$ is a partial play, $\omega : \prod_{i=0}^{\infty} X_i \to \mathbb{N}$ is a control function, and $\varepsilon : \prod_{i=0}^{\infty}\left(\prod_{j=0}^{i-1} X_j \to JX_i\right)$ is a sequence of (dependent) selection functions.

We use the name EPS to refer to the functional $\lambda k.\mathsf{EPS}_k$. The following lemma states how EPS behaves when applied to an outcome function.

Lemma 4.39 ([10, 13])**.** *Let R and X_i for $i \in \mathbb{N}$ be non-empty sets, let $\omega : \prod_{i=0}^{\infty} X_i \to \mathbb{N}$ be a control function, and $\varepsilon : \prod_{i=0}^{\infty}\left(\prod_{j=0}^{i-1} X_j \to JX_i\right)$ a sequence of selection functions. Given $k \in \mathbb{N}$, for every partial play $s : \prod_{i=0}^{k-1} X_i$ and every outcome function $p : \prod_{i=k}^{\infty} X_i \to R$, we have that:*

$$\mathsf{EPS}_k s\omega\varepsilon p = \begin{cases} \mathbf{0} & \textit{if } \omega(s * \mathbf{0}) < k \\ a * \mathsf{EPS}_{k+1}(s * a)\omega\varepsilon p_a & \textit{otherwise,} \end{cases}$$

where $a := \varepsilon_{k,s}\big(\lambda x^{X_k}.p_x(\mathsf{EPS}_{k+1}(s * x)\omega\varepsilon p_x)\big)$.

Proof. Just unfolding the definitions. To compute $\mathsf{EPS}_k s\omega\varepsilon p$, we first check if $\omega(s * \mathbf{0}) < k$. If that is true, then we get $(\lambda p.\mathbf{0})(p) = \mathbf{0}$. If not, then we get:

$$\mathsf{EPS}_k s\omega\varepsilon p = \big(\varepsilon_{k,s} \otimes \lambda x^{X_k}.\mathsf{EPS}_{k+1}(s * x)\omega\varepsilon\big)(p) = a * b(a),$$

where $b(x) = \mathsf{EPS}_{k+1}(s * x)\omega\varepsilon p_x$ and $a = \varepsilon_{k,s}\big(\lambda x^{X_k}.p_x(b(x))\big)$, that is,

$$a = \varepsilon_{k,s}\big(\lambda x^{X_k}.p_x(\mathsf{EPS}_{k+1}(s * x)\varepsilon\omega p_x)\big),$$

and so we are done. $\square$

The intuition is that $\mathsf{EPS}_0\langle\rangle\omega\varepsilon$ is the iteration $\bigotimes_{i=0}^{\infty} \varepsilon_i$, and actually in the literature the latter is sometimes the notation used. However, we preserve here the heavy notation to keep in mind that EPS strongly depends on its control.

Remark 4.40. We notice that if ω is the constant function $\lambda s.n$, then we get the finite product, in the sense that $\mathsf{EPS}_0\langle\rangle\omega\varepsilon p = \left(\bigotimes_{i=0}^{n} \varepsilon_i\right)(p') * \mathbf{0}$, where p' is defined, for $s : \prod_{i=0}^{n} X_i$, by $p'(s) := p(s * \mathbf{0})$. See [14].

Lemma 4.41 ([10, 13]). *Let R and X_i for $i \in \mathbb{N}$ be non-empty sets, let $\omega : \prod_{i=0}^{\infty} X_i \to \mathbb{N}$ be a control function, and $\varepsilon : \prod_{i=0}^{\infty} \left(\prod_{j=0}^{i-1} X_j \to J_R X_i \right)$ a sequence of selection functions. Given $k \in \mathbb{N}$, a partial play $s : \prod_{i=0}^{k-1} X_i$ and an outcome function $p : \prod_{i=k}^{\infty} X_i \to R$, consider $\alpha = \mathsf{EPS}_k s \omega \varepsilon p$. For all $j \in \mathbb{N}$,*

$$\alpha = [\alpha](j) * \mathsf{EPS}_{k+j}\big(s * [\alpha](j)\big)\omega\varepsilon p_{[\alpha](j)}.$$

Proof. By induction on j. The case $j = 0$ is just the definition of α. Now assume it holds for $j \geq 0$, and let us prove it for $j + 1$. There are two cases:

Case 1. $\omega(s * [\alpha](j) * \mathbf{0}) < k + j$. Then:

$$\mathsf{EPS}_{k+j}\big(s * [\alpha](j)\big)\omega\varepsilon p_{[\alpha](j)} = \mathbf{0}.$$

So, using the IH,

$$\alpha = [\alpha](j) * \mathsf{EPS}_{k+j}\big(s * [\alpha](j)\big)\omega\varepsilon p_{[\alpha](j)} = [\alpha](j) * \mathbf{0} = [\alpha](j+1) * \mathbf{0}.$$

Hence, $\omega(s * [\alpha](j+1) * \mathbf{0}) = \omega(s * [\alpha](j) * \mathbf{0}) < k + j < k + j + 1$, so:

$$\mathsf{EPS}_{k+j+1}\big(s * [\alpha](j+1)\big)\omega\varepsilon p_{[\alpha](j+1)} = \mathbf{0}$$

and therefore:

$$\alpha = [\alpha](j+1) * \mathbf{0} = [\alpha](j+1) * \mathsf{EPS}_{k+j+1}\big(s * [\alpha](j+1)\big)\omega\varepsilon p_{[\alpha](j+1)}.$$

Case 2. $\omega(s * [\alpha](j) * \mathbf{0}) \geq k + j$. By the IH,

$$\begin{aligned}\alpha &= [\alpha](j) * \mathsf{EPS}_{k+j}\big(s * [\alpha](j)\big)\omega\varepsilon p_{[\alpha](j)} \\ &= [\alpha](j) * a * \mathsf{EPS}_{k+j+1}\big(s * [\alpha](j) * a\big)\omega\varepsilon p_{[\alpha](j)*a}.\end{aligned}$$

But then $\alpha(j) = a$, and so the expression above translates to

$$\alpha = [\alpha](j+1) * \mathsf{EPS}_{k+j+1}\big(s * [\alpha](j+1)\big)\omega\varepsilon p_{[\alpha](j+1)}.$$

□

We state the following corollary for future convenience:

Corollary 4.42. *Let R and X_i for $i \in \mathbb{N}$ be non-empty sets, let $\omega : \prod_{i=0}^{\infty} X_i \to \mathbb{N}$ be a control function, and $\varepsilon : \prod_{i=0}^{\infty} \left(\prod_{j=0}^{i-1} X_j \to J X_i \right)$ a sequence of selection functions. Given an outcome function $p : \prod_{i=0}^{\infty} X_i \to R$, consider $\alpha = \mathsf{EPS}_0 \langle\rangle \omega \varepsilon p$. For all $j \in \mathbb{N}$, we have:*

$$\alpha = [\alpha](j) * \mathsf{EPS}_j\big([\alpha](j)\big)\omega\varepsilon p_{[\alpha](j)}.$$

We define the associated quantifier to a selection function in this setting.

Definition 4.43. Given a dependent selection function $\varepsilon_i : \prod_{j=0}^{i-1} X_j \to J_R X_i$, we define its **associated** (attainable) dependent quantifier $\overline{\varepsilon_i} : \prod_{j=0}^{i-1} X_j \to K_R X_i$ as, for all $s : \prod_{j=0}^{i-1} X_j$,

$$(\overline{\varepsilon_i})_s := \overline{\varepsilon_{i,s}}.$$

Now we can compute optimal strategies for unbounded games.

Theorem 4.44 ([12]). *Let $\big(\langle X_i\rangle_{i=0}^{\infty}, p, \phi, \omega\big)$ be an unbounded game with explicit control, and for each $i \in \mathbb{N}$ let us assume that ϕ_i is the associated quantifier of a selection function ε_i. Then an optimal strategy for the game can be computed as:*

$$\mathrm{next}_k(\boldsymbol{x}) = \big(\mathsf{EPS}_k \boldsymbol{x} \omega \varepsilon p_{\boldsymbol{x}}\big)(0),$$

where $\boldsymbol{x} : \prod_{i=0}^{k-1} X_i$.

Proof. Note that, by Lemma 4.41, $\beta^{\boldsymbol{x}} = \mathsf{EPS}_k \boldsymbol{x} \omega \varepsilon p_{\boldsymbol{x}}$. Assume $\omega(\boldsymbol{x} * \beta^{\boldsymbol{x}}) \geq k$. Then, we also have that $\omega(\boldsymbol{x} * \mathbf{0}) \geq k$, since otherwise, by the definition of the strategy, we would have:

$$\mathsf{EPS}_k \boldsymbol{x} \omega \varepsilon p_{\boldsymbol{x}} = \mathbf{0} = \beta^{\boldsymbol{x}},$$

and that is a contradiction with the assumption.

Now, by definition, we have:

$$\beta^{\boldsymbol{x}}(k) = \mathrm{next}_k(\boldsymbol{x}) = \varepsilon_{k,\boldsymbol{x}}\big(\lambda x^{X_k}.p_{\boldsymbol{x}*x}(\mathsf{EPS}_{k+1}(\boldsymbol{x} * x)\omega\varepsilon p_{\boldsymbol{x}*x})\big).$$

Hence,

$$\beta^{\boldsymbol{x}}(k) = \varepsilon_{k,\boldsymbol{x}}(\lambda x^{X_k}.p_{\boldsymbol{x}}(x * \beta^{\boldsymbol{x}*x})).$$

Let $q = \lambda x^{X_k}.p_{\boldsymbol{x}}(x * \beta^{\boldsymbol{x}*x})$. Then:

$$\phi_{k,\boldsymbol{x}}(q) = q(\varepsilon_{k,\boldsymbol{x}} q),$$

and note that:

$$\beta^{\boldsymbol{x}}(k) = \mathrm{next}_k(\boldsymbol{x}) = \varepsilon_{k,\boldsymbol{x}}\big(\lambda x^{X_k}.p_{\boldsymbol{x}*x}(\mathsf{EPS}^{\omega}_{k+1}(\varepsilon)(p_{\boldsymbol{x}*x}))\big) = \varepsilon_{k,\boldsymbol{x}} q.$$

So:

$$q(\beta^{\boldsymbol{x}}(k)) = q(\varepsilon_{k,\boldsymbol{x}} q) = \phi_{k,\boldsymbol{x}}(q).$$

Since again $q(\beta^{\boldsymbol{x}}(k)) = p_{\boldsymbol{x}}(\beta^{\boldsymbol{x}}(k) * \beta^{\boldsymbol{x},\beta^{\boldsymbol{x}}(k)}) = p_{\boldsymbol{x}}(\beta^{\boldsymbol{x}})$, we are done. □

Notation 4.45. Notice that the notation $\phi_{k,s}$ and $\varepsilon_{k,s}$ above is redundant, since we have $k = |s|$, and therefore we sometimes write simply ϕ_s and ε_s.

Chapter 5

Interpretation of analysis: an extension of Dialectica with EPS

This chapter is devoted to showing how the product of selection functions with explicit control is capable of interpreting analysis, playing the role of bar recursion. The outline is as follows.

In Section 5.1 we present the main theorem on EPS that will allow for witnessing the interpretation. Picking up the discussion from Chapter 3, a well-suited system for the formalization of classical analysis is WE-PA$^\omega$ + QF-AC + CA^0, whose *ND*-interpretation reduces, as Propositions 2.16, 3.1, and 3.2 show, to the *D*-interpretation of WE-HA$^\omega$ + AC + MP + DNS. We witness this interpretation via EPS in Section 5.2.

Section 5.3 is devoted to showing the equivalence between EPS and bar recursion, and Section 5.4 explains the advantages of the former over the latter.

This chapter combines results by Escardó and Oliva [10–13] with our setting for the interpretation of analysis of Chapter 3, which recall that follows [26, Chapter 11].

5.1 Main theorem on EPS

In this section we prove the most general version of the main theorem about EPS, that will allow for interpreting several non-constructive steps in the proof of Ramsey's theorem in Chapter 6.

Since our presentation of sequential games occurs in a framework of standard mathematics and uses, for instance, Cartesian products, we also give here an adapted version for a system in all finite types as WE-HA$^\omega$.

The next theorem uses Notation 4.45.

Theorem 5.1 ([10]). *Let $(\langle X_i\rangle_{i=0}^{\infty}, q, \phi, \omega)$ be an unbounded game with attainable quantifiers, and so let $\varepsilon : \prod_{i=0}^{\infty}\left(\prod_{j=0}^{i-1} X_j \to J_R X_i\right)$ be such that for every $k \in \mathbb{N}$ and $s : \prod_{i=0}^{k-1} X_i$, $\phi_s = \overline{\varepsilon_s}$. For every partial play $s : \prod_{i=0}^{k-1} X_i$ and $x : X_k$, define:*

$$\begin{aligned} \alpha &:= \mathsf{EPS}_0\langle\rangle\omega\varepsilon q, \\ p_s(x) &:= \overline{\mathsf{EPS}_{k+1}(s * x)\omega\varepsilon}(q_{s*x}). \end{aligned}$$

Then, for all $n \leq \omega(\alpha)$ we have:

$$\begin{aligned} \alpha n &= \varepsilon_{[\alpha](n)}(p_{[\alpha](n)}), \\ q\alpha &= \overline{\varepsilon_{[\alpha](n)}}(p_{[\alpha](n)}). \end{aligned}$$

Proof. Assume that $n \leq \omega(\alpha)$. Then, we also have

$$n \leq \omega([\alpha](n) * \mathbf{0}), \tag{5.1}$$

since $n > \omega([\alpha](n) * \mathbf{0})$ together with Corollary 4.42 would imply $\alpha = [\alpha](n) * \mathbf{0}$, and hence:

$$n > \omega([\alpha](n) * \mathbf{0}) = \omega(\alpha) \geq n,$$

a contradiction. For the first equality,

$$\begin{aligned} \alpha n &\overset{4.42}{=} \mathsf{EPS}_n\big([\alpha](n)\big)\omega\varepsilon q_{[\alpha](n)}(0) \\ &\overset{(5.1)}{=} \varepsilon_{[\alpha](n)}\big(\lambda x^{X_n}.\overline{\mathsf{EPS}_{n+1}\big([\alpha](n) * x\big)\omega\varepsilon}(q_{[\alpha](n)*x})\big) \\ &= \varepsilon_{[\alpha](n)}(p_{[\alpha](n)}). \end{aligned}$$

For the second identity we use the first one:

$$\begin{aligned} q\alpha &\overset{4.42}{=} q\big([\alpha](n+1) * \mathsf{EPS}_{n+1}\big([\alpha](n+1)\big)\omega\varepsilon q_{[\alpha](n+1)}\big) \\ &= q_{[\alpha](n+1)}\big(\mathsf{EPS}_{n+1}\big([\alpha](n+1)\big)\omega\varepsilon q_{[\alpha](n+1)}\big) \\ &= \overline{\mathsf{EPS}_{n+1}\big([\alpha](n+1)\big)\omega\varepsilon}(q_{[\alpha](n+1)}) \\ &= p_{[\alpha](n)}(\alpha n) \\ &= \overline{\varepsilon_{[\alpha](n)}}(p_{[\alpha](n)}). \end{aligned}$$

□

The above theorem, together with Theorem 4.44, says that we can compute an optimal play α, and that moreover there are outcome functions 'local' for each round of the relevant part of the optimal play, in the following sense. For each $n \leq \omega(\alpha)$, we have a local outcome function $p_{[\alpha](n)} : X_n \to R$ such that the global outcome of the game is equal to the outcome predicted by the local function and the selection function of round n, i.e., $q\alpha = p_{[\alpha](n)}(\varepsilon_{[\alpha](n)} p_{[\alpha](n)})$.

Now we rephrase this theorem for adequate types. All sets of possible moves will be one and the same type, σ. Therefore, the type $\prod_{i=0}^{\infty} X_i$ becomes $0 \to \sigma$, or σ^0 for short. The type of the outcomes, R, is now written τ. Moreover, we make informal use of the type of finite sequences over σ, which is named σ^*. This type can be coded easily by the type σ^0. In some cases, however, we need a refinement of this encoding, since an increase of the complexity of the type is not desirable:

Remark 5.2. By σ^* we refer to the type of finite sequences over σ, encoded in any other type of our formal system (for instance, in σ^0). However, in the case of finite sequences over the natural numbers, 0^*, we assume that our encoding is done in type 0 itself. It is well-known that there are primitive recursive encodings of 0^* in 0, for instance, Gödel's encoding. We will also use this fact for finite binary sequences in the next chapter. More details on these encodings can be found in [44], [34].

Theorem 5.3. *Let $q : \sigma^0 \to \tau$, $\omega : \sigma^0 \to 0$, and let $\varepsilon : \sigma^* \to J_\tau\sigma$. For every $s : \sigma^*$ and $x : \sigma$, define:*

$$\begin{aligned} \alpha &:=_{\sigma^0} \mathsf{EPS}_0\langle\rangle\omega\varepsilon q, \\ p_s(x) &:=_\tau \overline{\mathsf{EPS}_{|s|+1}(s * x)\omega\varepsilon}(q_{s*x}). \end{aligned}$$

Then, for all $n \leq \omega(\alpha)$ we have:

$$\begin{aligned} \alpha n &=_\sigma \varepsilon_{[\alpha](n)}(p_{[\alpha](n)}), \\ q\alpha &=_\tau \overline{\varepsilon_{[\alpha](n)}}(p_{[\alpha](n)}). \end{aligned}$$

Remark 5.4. Sometimes, ε does not depend on the finite sequence but only on its length, as in our first presentation of the product. So its type can be $\varepsilon : 0 \to J_\tau\sigma$, and then we interpret the occurrences of ε_s in the definition of EPS simply as $\varepsilon_{|s|}$. Thus, we avoid defining each time selection functions ε_s as for all s, $\varepsilon_s := \varepsilon_{|s|}$. This can be understood as a notational matter.

5.2 Interpretation of DNS

In order to witness the interpretation of DNS, recall that Proposition 3.3 tells us that we only need to find terms t_x, t_W and t_V containing only U, Y, Z free satisfying:

$$(3.3) \begin{cases} t_x = Y(t_W) \\ U(t_x, t_V) = t_W(Y(t_W)) \\ t_V(U(t_x, t_V)) = Z(t_W). \end{cases}$$

After a renaming, this system of equations translates to:

$$(3.6) \begin{cases} n =_0 \omega(\alpha) \\ \varepsilon np =_\sigma \alpha(\omega\alpha) \\ p(\varepsilon np) =_\tau q\alpha, \end{cases}$$

where recall that, for the case of DNS, τ is σ^0.

Remark 5.5. There is a similarity between the results about bar recursion SBR obtained in Chapter 3 and the current results about EPS. Lemma 3.5 corresponds to Corollary 4.42; Theorem 3.7 corresponds to Theorem 5.6; and of course Theorem 3.8 corresponds to Theorem 5.8.

We now use EPS to give a solution to this system.

Theorem 5.6. *For all types $\sigma, \tau \in T$ and all $\omega : \sigma^0 \to 0$, $\varepsilon : 0 \to (\sigma \to \tau) \to \sigma$ and $q : \sigma^0 \to \tau$, define:*

$$\begin{aligned}
\alpha &:= \mathsf{EPS}_0\langle\rangle\omega\varepsilon q,\\
n &:= \omega\alpha,\\
p &:= \lambda x^\sigma.\big(\overline{\mathsf{EPS}_{n+1}\big([\alpha](n) * x\big)\omega\varepsilon}(q_{[\alpha](n)*x})\big).
\end{aligned}$$

We have that α, n and p satisfy (3.6).

Proof. It follows directly from Theorem 5.3 (and Remark 5.4). □

Thus we can extend Dialectica to analysis via EPS. For the extension, it is sufficient with the simplest of our definitions of EPS. The principle and the functional are both called EPS in the literature, and we follow this practice.

Definition 5.7. For all types σ, τ, let us consider a constant symbol $\mathsf{EPS}^{\sigma,\tau}$. We define the principle EPS as the schema of defining equations, for variables $s : \sigma^*$, $\omega : \sigma^0 \to 0$, $\varepsilon : 0 \to (\sigma \to \tau) \to \sigma$, and $p : \sigma^0 \to \tau$,

$$\mathsf{EPS}^{\sigma,\tau}_{|s|} s\omega\varepsilon p =_{\sigma^0} \begin{cases} \mathbf{0} & \text{if } \omega(s * \mathbf{0}) < |s| \\ a * \mathsf{EPS}^{\sigma,\tau}_{|s|+1}(s * a)\omega\varepsilon p_a & \text{otherwise,} \end{cases}$$

where $a := \varepsilon_{|s|}\big(\lambda x^\sigma.p_x(\mathsf{EPS}^{\sigma,\tau}_{|s|+1}(s * x)\omega\varepsilon p_x)\big)$.

System WE-HA$^\omega$+EPS is an extension of WE-HA$^\omega$ with new constant symbols $\mathsf{EPS}^{\sigma,\tau}$ for all types σ, τ, and the axiom schema EPS.

Theorem 5.8. *Let $A(\boldsymbol{a})$ be a formula in the language of* WE-HA$^\omega$ *containing only $\boldsymbol{a}$ free. If*

$$\text{WE-PA}^\omega + \mathsf{QF\text{-}AC} + \mathsf{AC}^0 \vdash A(\boldsymbol{a}),$$

then there is a tuple of closed terms $\boldsymbol{t}$ of WE-HA$^\omega$ + EPS *such that*

$$\text{WE-HA}^\omega + \mathsf{EPS} \vdash \forall \boldsymbol{y}(A^N)_D(\boldsymbol{ta}, \boldsymbol{y}, \boldsymbol{a}).$$

Moreover, $\boldsymbol{t}$ can be effectively extracted from a proof of the assumption.

5.3 Equivalence to SBR

The special form of bar recursion SBR turns out to be equivalent to EPS, meaning that we can define each from the other one in a primitive recursive way. Actually, SBR is equivalent to the general form of bar recursion BR, and hence so is EPS. For details see [40] or [13].

However, proving that the definition of EPS using SBR is correct requires a principle called bar induction. Since this topic lies outside our purposes, and we only intend to give here intuition on how these principles are equivalent, we avoid this technical question by presenting a variant of EPS, called here $\widetilde{\mathsf{EPS}}$, for which the equivalence to SBR is provable over WE-HA$^\omega$. This variant is actually the one used in [40].

For the most general results about equivalence of forms of bar recursion and the product of selection functions the reader is referred to [13].

Remark 5.9. The symbol @ stands for a binary function on sequences overwriting the first part of the second sequence, so that for $s : \sigma^*$ and $\alpha : \sigma^0$,

$$(s@\alpha)(k) := \begin{cases} s(k) & \text{if } k < |s| \\ \alpha(k) & \text{otherwise.} \end{cases}$$

Consider the following definitions. For all $n : 0$, $s : \sigma^0$, $\omega : \sigma^0 \to 0$, $\varepsilon : 0 \to (\sigma \to \sigma^0) \to \sigma$,

$$\mathsf{SBR}_\sigma n s \omega \varepsilon := [s](n)@\begin{cases} \overline{s,n} & \text{if } \omega(\overline{s,n}) < n \\ \mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{[s](n) * a})\omega\varepsilon & \text{otherwise,} \end{cases}$$

where $a := \varepsilon n\big(\lambda x^\sigma.\mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{[s](n) * x})\omega\varepsilon\big)$. Notice that overwriting the first part with $[s](n)$ is harmless here, since, as seen in Chapter 3, $[s](n)$ is always the initial segment of $\mathsf{SBR}_\sigma n s\omega\varepsilon$. We write it to avoid some technical details below.

On the other hand, for all $n : 0$, $s : \sigma^*$ of length n, $\omega : \sigma^0 \to 0$, $p : \sigma^0 \to \tau$ and $\varepsilon : 0 \to (\sigma \to \tau) \to \sigma$,

$$\widetilde{\mathsf{EPS}}_n s\omega\varepsilon p := s\,@\begin{cases} \mathbf{0} & \text{if } \omega(s * \mathbf{0}) < n \\ \widetilde{\mathsf{EPS}}_{n+1}(s * a)\omega\varepsilon p & \text{otherwise,} \end{cases}$$

where $a := \varepsilon n\big(\lambda x^\sigma.p(\widetilde{\mathsf{EPS}}_{n+1}(s * x)\omega\varepsilon p)\big)$. It is intuitively clear that $\widetilde{\mathsf{EPS}}$ is equivalent to EPS (when the family of selection functions does not depend on the sequence, as in Section 5.2). The idea is that $\widetilde{\mathsf{EPS}}$ computes the whole sequence while EPS forgets the first part, in the sense that $\widetilde{\mathsf{EPS}}_n s\omega\varepsilon p = s * \mathsf{EPS}_n s\omega\varepsilon p_s$. The proof, as stated above, requires bar induction, and can be found in [13].

Theorem 5.10 ([40])**.** *SBR and $\widetilde{\mathsf{EPS}}$ as defined in Remark 5.9 are equivalent over* WE-HA$^\omega$*, in the sense that there is a term t of* WE-HA$^\omega$ *such that the defining equations of SBR are satisfied by $t(\widetilde{\mathsf{EPS}})$ provably in* WE-HA$^\omega$*, and viceversa.*

Proof. Let us define SBR from $\widetilde{\mathsf{EPS}}$. We use $\widetilde{\mathsf{EPS}}$ with $\tau = \sigma^0$. Define $p : \sigma^0 \to \sigma^0$ as the identity function. For all $n : 0$, $s : \sigma^0$, $\omega : \sigma^0 \to 0$, $\varepsilon : 0 \to (\sigma \to \sigma^0) \to \sigma$, define:

$$\mathsf{SBR}_\sigma n s\omega\varepsilon := \widetilde{\mathsf{EPS}}_n\big([s](n)\big)\omega\varepsilon p.$$

We see that this satisfies the definition of SBR_σ of Remark 5.9. If $\omega(\overline{s,n}) < n$,

$$\mathsf{SBR}_\sigma n s\omega\varepsilon = \widetilde{\mathsf{EPS}}_n\big([s](n)\big)\omega\varepsilon p = [s](n)@\mathbf{0} = \overline{s,n}.$$

And if $\omega(\overline{s,n}) \geq n$,

$$\mathsf{SBR}_\sigma n s \omega \varepsilon = \widetilde{\mathsf{EPS}}_n\big([s](n)\big)\omega\varepsilon p = s@\widetilde{\mathsf{EPS}}_{n+1}([s](n) * a)\omega\varepsilon p =$$

$$s@\mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{[s](n) * a})\omega\varepsilon,$$

where:

$$\begin{aligned} a &= \varepsilon n\big(\lambda x^\sigma.p(\widetilde{\mathsf{EPS}}_{n+1}([s](n) * x)\omega\varepsilon p)\big) \\ &= \varepsilon n\big(\lambda x^\sigma.p(\mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{[s](n) * x})\omega\varepsilon)\big) \\ &= \varepsilon n\big(\lambda x^\sigma.\mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{[s](n) * x})\omega\varepsilon\big), \end{aligned}$$

as required.

Conversely, given $n : 0$, $s : \sigma^*$ of length n, $\omega : \sigma^0 \to 0$, $p : \sigma^0 \to \tau$, and $\varepsilon : 0 \to (\sigma \to \tau) \to \sigma$, define for all $k : 0$ and $f : \sigma \to \sigma^0$,

$$\tilde{\varepsilon} k f := \varepsilon k\big(\lambda x^\sigma.p(f(x))\big),$$

and

$$\widetilde{\mathsf{EPS}}_n s\omega\varepsilon p := \mathsf{SBR}_\sigma n\overline{s}\omega\tilde{\varepsilon}.$$

Let us prove that this satisfies the definition of $\widetilde{\mathsf{EPS}}$ above. If $\omega(s * \mathbf{0}) < n$, then

$$\widetilde{\mathsf{EPS}}_n s\omega\varepsilon p = \mathsf{SBR}_\sigma n\overline{s}\omega\tilde{\varepsilon} = s@\mathbf{0},$$

as required. If $\omega(s * \mathbf{0}) \geq n$, then

$$\widetilde{\mathsf{EPS}}_n s\omega\varepsilon p = \mathsf{SBR}_\sigma n\overline{s}\omega\tilde{\varepsilon} = s@\mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{s * a})\omega\tilde{\varepsilon} = s@\widetilde{\mathsf{EPS}}_{n+1}(s * a)\omega\varepsilon p,$$

where:

$$\begin{aligned} a &= \tilde{\varepsilon} n\big(\lambda x^\sigma.\mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{s * x})\omega\tilde{\varepsilon}\big) \\ &= \varepsilon n\big(\lambda x^\sigma.p\big(\mathsf{SBR}_\sigma(\mathbf{S}n)(\overline{s * x})\omega\tilde{\varepsilon}\big)\big) \\ &= \varepsilon n\big(\lambda x^\sigma.p\big(\widetilde{\mathsf{EPS}}_{n+1}(s * x)\omega\varepsilon p\big)\big), \end{aligned}$$

and hence $\widetilde{\mathsf{EPS}}$ defined like this satisfies the definition in Remark 5.9. □

5.4 Advantages of EPS over bar recursion

Recall that the D-interpretation of DNS led to:

$$\begin{aligned} &\exists U\, \forall x, V \neg\neg A_D\big(x, UxV, V(UxV)\big) \to \\ &\qquad \exists W \forall Y, Z \neg\neg A_D\big(Y(WYZ), WYZ(Y(WYZ)), Z(WYZ)\big). \end{aligned}$$

After renaming and removing double negations in front of quantifier-free formulas, we have:

$$\exists \varepsilon\, \forall n, p\; A_D\big(n, \varepsilon_n p, p(\varepsilon_n p)\big) \to \exists \alpha\, \forall \omega, q\; A_D\big(\omega(\alpha_{\omega,q}), \alpha_{\omega,q}(\omega\alpha_{\omega,q}), q\alpha_{\omega,q}\big).$$

That is, DNS^D can be viewed as stating that if we have selection functions that witness A_D at each round and for any outcome function, then there is a functional α that, for every control and outcome function ω, q, gives a global play $\alpha_{\omega,q}$ witnessing A_D. We have seen that α is precisely the explicitly controlled product of the functions ε.

So, basically, the interpretation of DNS asks exactly for what the main theorem on EPS gives.

We have seen that appropriate instances of EPS and SBR are equivalent over WE-HA$^\omega$. All of them are capable of interpreting analysis, and in this sense there is no better option. If the aim is to give a relative consistency proof, all are equally convenient as well.

But when choosing among them for proof mining, it might be the case, and it actually is, that one of them has advantages over the other. The rest of this section is devoted to explaining the two main advantages of EPS over bar recursion regarding proof mining.

The first of them has to do with the fact that the algorithm of term extraction that the proof of soundness of the *ND*-interpretation gives is unpractical, due to several reasons. On the one hand, classical proofs of theorems in analysis are usually written in high-level, and a strict formalization of one of these proofs in our system WE-PA$^\omega$ + QF-AC + CA^0 would involve a huge amount of steps. On the other hand, even if we had such a formalization, the algorithm would yield a huge term, which probably only a computer might obtain.

Therefore, what is usually done in proof mining is a compromise between a high-level proof and a strict formalization. The classical proof is worked out up to a certain, greater than usual, level of detail, and then the interpretation is applied piecewise to some parts of the proof. Finding witnesses and filling the gaps generally requires not only the use of whatever principle we have accepted, EPS or SBR, but also intuition and mathematical thinking, both constructive and classical. Chapter 6 is an extensive example thereof.

If term extraction were just an algorithmic procedure, the choice between EPS and SBR would be completely irrelevant. But, as we have discussed, it is not. The benefit of EPS is that it supplies a clear semantics, and this enables the mathematician to turn his or her effort from a syntactical matter into a semantical one, thus facilitating the task of interpreting theorems and proofs, and presumably leading to a faster growth of the proof mining program.

Yet there is another advantage, maybe more important. Since its birth until now, proof mining has been synonymous with term extraction. As discussed in the introduction, it has allowed for extracting explicit bounds from non-constructive proofs in analysis. However, there is another task that proof mining can accomplish: to understand the constructive content of classical proofs in daily mathematics. The game-theoretic semantics gives the intuition of theorems as statements about games and their proofs as the search for an optimal strategy in the corresponding game. As we will show in Chapter 6, this conceptual reading of the Dialectica interpretation is enlightening.

Chapter 6

A constructive interpretation of Ramsey's Theorem

The Infinite Ramsey Theorem, also referred to here simply as Ramsey's theorem, has been widely studied in logic, and in particular there are several papers [3, 6, 7, 27, 48] studying its constructive content. To this end, the proof by Erdős and Rado [9] is more convenient than the original proof by Ramsey [41], since the latter, although apparently simpler, uses further non-constructive principles.

Recently, Oliva and Powell [38] presented a constructive interpretation of Ramsey's theorem for pairs and two colours, based on the Dialectica interpretation, and a constructive proof using the product of selection functions. Their paper is inspired by [27], which presents a similar interpretation of Ramsey's theorem realized, however, using bar recursion instead of selection functions. In this chapter we expose the results of [38] with a deeper level of detail, and we extend all of them for the case of $r \geq 2$ colours.

Recall that Ramsey's theorem states the existence of a monochromatic infinite subset $H \subseteq \mathbb{N}$ for any colouring with r colours of subsets of size k of the natural numbers. If we write $[X]^k$ for the set of subsets of X of exactly k elements, and $[r]$ for the set $\{0, \dots, r-1\}$[1], we can state the theorem a bit more formally as:

Theorem 6.1 (Ramsey's theorem)**.** *For every $c : [\mathbb{N}]^k \to [r]$, there exists a colour $x \in [r]$ and an infinite subset $H \subseteq \mathbb{N}$ such that for every $A \in [H]^k$, $c(A) = x$.*

However, this version is still too informal for our purposes. When we apply the Dialectica interpretation, in principle we should do it with strict logical syntax and follow the proof of soundness to extract a precise term realizing the theorem. But, in practice, this procedure is unfeasible to carry out by hand. Hence, our syntax is a compromise between formality and readability.

In general terms, the only violations to the syntax of WE-HA$^\omega$ will be the use of the type $[r]$ and $\mathbb{B} := [2]$ (the booleans), the type $\mathbb{B}^*$ of finite sequences over $\mathbb{B}$, and bounded quantifiers, which are understood as usual. All of these are just abbreviations: a term $t : [r]$ can be formally encoded by a term $t : 0$ such that $t < r$; as for a term $s : \mathbb{B}^*$, it can be coded by $s : 0$ by means of a primitive

[1] We do not write simply r, as usual in set theory, to distinguish $[r]$ from the pure type r.

recursive encoding similar to Gödel's well-known encoding for finite sequences of natural numbers (see Remark 5.2).

Throughout this chapter, we relax our notation to improve readability. We write $\mathbb{N}$ instead of 0, we abbreviate the type $\sigma \to \tau$ as τ^σ and the type $\sigma \to \sigma \to \tau$ as $\sigma^2 \to \tau$. The variables i, j, k, n, m are always assumed to be of type $\mathbb{N}$.

In systems in all finite types, as WE-HA$^\omega$, it is not possible to directly express the existence of an infinite subset of the natural numbers satisfying some condition. Instead, we state the existence of a function $f : \mathbb{N} \to \mathbb{N}$ with unbounded image (imposing, for instance, that for all n, $fn \geq n$) such that the image satisfies the required condition. That is, instead of subsets of $\mathbb{N}$ we use (not necessarily injective) enumerations of subsets of $\mathbb{N}$. This will be common practice in what follows.

6.1 Preliminary definitions and theorems

This section is devoted to presenting some usual definitions and well-known theorems that are needed throughout the chapter. Our definitions are stated in order to be well-suited for our proof of Ramsey's theorem: no attempt is made to give the most general definitions and results.

There is an easy particular case of Ramsey's theorem, when $k = 1$. This case is called *infinite pigeon-hole principle*. We state it here for clarity:

Theorem 6.2 (Infinite pigeon-hole principle)**.** *For every colouring $c : \mathbb{N} \to [r]$, there exists a colour $x \in [r]$ and an infinite subset $H \subseteq \mathbb{N}$ such that for every $n \in H$, $c(n) = x$.*

Our proof will make use of trees, for which there are several definitions in mathematics. We present here two of them.

Definition 6.3. A partial order $\prec$ on $\mathbb{N}$ describes a **tree** if it has a unique minimal element and for every $i \in \mathbb{N}$, the set of its predecessors $\mathrm{pd}(i) = \{k \in \mathbb{N} : k \prec i\}$ is well-ordered by $\prec$. A **branch** is a maximal chain of the tree. Moreover, the tree is **finitely branching** if every $i \in \mathbb{N}$ has at most a finite number of immediate successors. It is n**-ary branching** if every $i \in \mathbb{N}$ has at most n different immediate successors.

We also use a description of trees by means of a predicate over finite sequences, that tells us whether a given sequence is an initial segment of a branch of the tree or not.

Definition 6.4. Given a type σ, a predicate T over σ^* (in the language of WE-HA$^\omega$) is a **tree** predicate if T is prefix closed, i.e., for every $s : \sigma^*$, if $T(s)$ holds, then $T([s](m))$ holds for every $m < |s|$. If $\sigma = \mathbb{B}$, then T is a **binary tree**. If for every $s : \sigma^*$, WE-HA$^\omega$ proves $T(s) \vee \neg T(s)$, then T is said to be **decidable**.

For a tree predicate T over σ^*, if $s : \sigma^*$ and we have $T(s)$, then we say that s is a branch of T.

Now we state König's lemma, a well-known result from set theory that is not provable in Peano arithmetic.

Theorem 6.5 (König's lemma). *A finitely branching tree $\prec$ on $\mathbb{N}$ has an infinite branch.*

Since our aim is to give a constructive realizer for Ramsey's theorem for pairs, we will avoid König's lemma, taking a detour and using weak König's lemma instead. In our setting, the most convenient way to state weak König's lemma is as follows:

Theorem 6.6 (Weak König's lemma). *Let T be a tree predicate over $\mathbb{B}^*$. If T is decidable, then T has an infinite branch, i.e., there is a sequence $\alpha : \mathbb{N} \to \mathbb{B}$ such that $T([\alpha](n))$ holds for every $n \in \mathbb{N}$.*

6.2 A formal (classical) proof

First of all we need to formalize the statements that will be used throughout the proof. We present a rough sketch of the classical proof, and from that point we analyse which principles it will require.

The idea behind the proof is as follows. Given a colouring c, we arrange the natural numbers on a tree, described by an order $\prec$ to be defined below. This tree is called *Erdős-Rado tree*, or E-R tree for short. It is finitely branching, or more concretely, each node has at most r different immediate successors. The key property of the E-R tree is that for any $i, j, k \in \mathbb{N}$, if $i \prec j \prec k$, then $c(\{i, j\}) = c(\{i, k\})$. By König's lemma, this tree has an infinite branch. Suppose that $b : \mathbb{N} \to \mathbb{N}$ is an injective $\prec$-increasing enumeration of the infinite branch, that is, for each $i \in \mathbb{N}$, $b(i)$ is the ith node of the branch under the order $\prec$. Now we define a colouring $c' : \mathbb{N} \to [r]$ as $c'(i) = c(b(i), b(i + 1))$, and by the infinite pigeon-hole principle there is an infinite monochromatic subset H of $\mathbb{N}$ with respect to c'. But then the set $H^b := \{b(i) : i \in H\}$ is pairwise monochromatic under c.

For simplicity, the type of a colouring of pairs of natural numbers will be $\mathbb{N}^2 \to [r]$ (or, equivalently, $0 \to 0 \to [r]$), instead of $[\mathbb{N}]^2 \to [r]$. Of course, we need to impose the condition $c(i, j) = c(j, i)$ for every $i, j \in \mathbb{N}$. Note that there is a canonical way of assigning a colouring to every $c : \mathbb{N}^2 \to [r]$, namely:

$$\check{c}(i, j) := \begin{cases} c(i, j) & \text{if } i < j \\ c(j, i) & \text{otherwise.} \end{cases}$$

Therefore, each time we quantify over colourings, we can think, if we wish, that we quantify over $c : \mathbb{N}^2 \to [r]$ and then we use $\check{c}$ everywhere.

We notice that the principles that we use amount to König's lemma and the infinite pigeon-hole principle. In our formalization, we avoid the use of full König's lemma by encoding the E-R tree by a binary decidable tree, and then using weak König's lemma on it.

Our first encoding below of the E-R tree by a binary tree is Σ^0_1, and in order to make it decidable we will need a weak form of choice, more precisely Π^0_1-countable choice. Therefore, the principles that we need are the infinite pigeon-hole principle, IPHP; weak König's lemma, WKL; and countable choice for Π^0_1-formulas, Π^0_1-AC^0.

The infinite pigeon-hole principle is a single axiom stating, as we have seen, that for every colouring of the natural numbers with r colours, there is an infinite monochromatic subset H.

$$\mathsf{IPHP} : \forall c^{\mathbb{N}\to[r]} \exists x^{[r]}, p^{\mathbb{N}\to\mathbb{N}} \forall k \big(pk \geq k \wedge c(pk) = x\big).$$

Here the infinite subset is encoded by a function p that enumerates it, that is, $H = \{pk : k \in \mathbb{N}\}$.

Our instance of choice is an axiom schema that reads:

$$\Pi^0_1\text{-}\mathsf{AC}^0 : \forall n \exists x^\sigma \forall m A_0(n, x, m) \to \exists \alpha^{0\to\sigma} \forall n, m A_0(n, \alpha n, m),$$

where A_0 is quantifier-free.

Weak König's lemma is an axiom schema stating the existence of an infinite branch in every externally given infinite decidable binary tree T:

$$\mathsf{WKL}(T) : \forall n \exists s^{\mathbb{B}^*} \big(|s| = n \wedge T(s)\big) \to \exists \alpha^{\mathbb{N}\to\mathbb{B}} \forall n T([\alpha](n)).$$

Finally, we formalize Ramsey's theorem for pairs and r colours as:

$$\mathsf{RT}^2_r(c) : \exists x^{[r]}, F^{\mathbb{N}\to\mathbb{N}} \forall k \big(Fk \geq k \wedge \forall i, j \leq k (Fi < Fj \to c(Fi, Fj) = x)\big),$$

where $c : \mathbb{N}^2 \to [r]$ is a colouring. Here the infinite set is encoded by F as before. Notice that instead of $Fi < Fj$ we could have required simply $Fi \neq Fj$.

Now we are ready to start the proof. We will use the following convention: we call Propositions or Theorems, depending on their relevance, those statements that will be interpreted in subsequent sections, and we call Lemmas the technical results needed for Propositions and Theorems, which are not directly translated from the classical proof into the interpretation.

The first thing we need is the definition of the E-R tree.

Definition 6.7 (Erdős-Rado tree)**.** Let $c : \mathbb{N}^2 \to [r]$ be a colouring. We define a partial order $\prec$ on $\mathbb{N}$ recursively as follows:

(i) 0 has no predecessors.

(ii) $0 \prec 1$, and moreover 0 is the only predecessor of 1.

(iii) If the predecessors of every $j \in [k]$ are already defined, we proceed to define the predecessors of k. For every $j < k$, define:

$$j \prec k \quad \text{iff} \quad c(i, k) = c(i, j) \text{ for all } i \prec j.$$

Example 6.8. Consider the colouring:

$$c(i, j) := \begin{cases} 0 & \text{if } i \equiv j \bmod 3 \\ 1 & \text{otherwise.} \end{cases}$$

This generates the E-R tree of Figure 6.1. The edge labels are there for clarity: the edge from i to j is labelled as $c(i, j)$. We shall see that $\prec$ defines a transitive relation and the figure should be understood as such.

The next lemma states properties of the E-R tree that will be needed later.

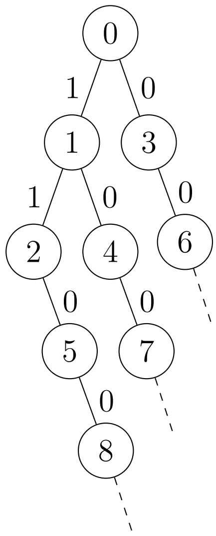

Figure 6.1: Tree obtained in Example 6.8

Lemma 6.9 ([27]). *Given a colouring $c : \mathbb{N}^2 \to [r]$, let $\prec$ be the relation of Definition 6.7. The following hold:*

(i) *The relation $\prec$ is a strict partial order and, moreover, $\prec \subseteq <$, where $<$ is the usual order on $\mathbb{N}$.*

(ii) *For every $k \in \mathbb{N}$, on the set $\mathrm{pd}(k)$ of its predecessors the orders $\prec$ and $<$ coincide. That is, if $i \prec k$ and $j \prec k$, then $i \prec j$ iff $i < j$.*

(iii) *The order $\prec$ defines an r-ary branching tree on $\mathbb{N}$.*

(iv) *The branches of the E-R tree are min-monochromatic: if $i \prec j \prec k$, then $c(i, j) = c(i, k)$.*

Proof. For (i), first note that $\prec \subseteq <$ follows directly from the definition, as well as the irreflexivity of $\prec$. To prove transitivity, we proceed by complete induction on k to prove that if $i \prec j$ and $j \prec k$, then $i \prec k$.

If $k = 0$, it has no predecessors and so the implication follows. Now assume $k > 0$ and that transitivity holds for all $k' < k$. Suppose we have $i \prec j$ and $j \prec k$. By definition, $i \prec k$ iff for all $i' \prec i$, $c(i', i) = c(i', k)$. Fix $i' \prec i$ to see that $c(i', i) = c(i', k)$. Since $i \prec j$, $c(i', i) = c(i', j)$. Also, $j < k$, so by IH we have that $i' \prec j$. Since $j \prec k$, we have that $c(i', k) = c(i', j)$, and this ends the proof.

For (ii), we proceed by induction on k. If $k = 0$ the claim is obvious. Assume that the claim is proved for every $k' < k$. Let $i, j \prec k$. Since $\prec \subseteq <$, we only need to prove that $i < j$ implies $i \prec j$. So assume $i < j$. Of course $0 \prec i$, $0 \prec j$. Let l be the $<$-greatest such that $l \prec i$, $l \prec j$. Among the immediate successors of l, there is at least one that is $\prec k$, since $l \prec i \prec k$. Among those, take m the $<$-minimal. Note that, since $l \prec m \prec k$, $l \prec i \prec k$ and $l \prec j \prec k$, we have that

$$c(l, k) = c(l, m) = c(l, i) = c(l, j). \tag{6.1}$$

We have that $m \neq j$. Indeed, since $l \prec i, j$ and is maximal with this property, $m \not\prec i$ or $m \not\prec j$. But since m is minimal being an immediate succesor of l such that $m \prec k$, then $m \leq i < j$, and so $m < j$. Now, we show that $m \prec j$. Let $i' \prec m$. By the IH applied to m, either $i' \prec l$ or $i' = l$ (the case $l \prec i'$ is impossible because m is an immediate successor of l). If $i' \prec l \prec m$, then, since $i' \prec l \prec j$,

$$c(i', m) = c(i', l) = c(i', j),$$

as required, and if $i' = l$, by (6.1) we have $c(l, m) = c(l, j)$. Therefore, $m \prec j$.

But now, if we assume $i \neq m$, the same argument works for showing $m \prec i$, and this is impossible by the maximality of l and the fact that $m \prec j$. Therefore, $i = m$ and so $i \prec j$. This ends the proof of (ii).

Now, to obtain (iii), observe that, since the set of predecessors of a node is ordered by $<$, it is well-ordered. Hence, $\prec$ is a tree order. Moreover, it is r-ary branching because if i, j are different immediate successors of k, then from the definition of $\prec$ follows that $c(i, k) \neq c(j, k)$.

Finally, (iv) follows from the definition of $j \prec k$. □

Now that we have the definition and key properties of the E-R tree, the next step is to define a binary tree predicate that encodes its branches. The following definition presents a Σ^0_1-predicate T which will be turned into a decidable predicate by means of our limited principle of choice, Π^0_1-AC^0.

Definition 6.10. Given a colouring $c : \mathbb{N}^2 \to [r]$ and the corresponding E-R tree described by $\prec$, define for $s : \mathbb{B}^*$ and $k : \mathbb{N}$:

$$T'(s, k) := \exists k' \in [|s|, k]\, \forall i < |s| (s_i = 0 \leftrightarrow i \prec k'),$$

and:

$$T(s) := \exists k T'(s, k).$$

Note that $T'(s, k)$ can be written as a quantifier-free formula, since its quantifiers are bounded and $\prec$ is primitive recursive. Hence, $T(s)$ is Σ^0_1. The bounded existential quantifier is there in order to make $T(s)$ monotone on k, as the following lemma states. This will be useful later.

Lemma 6.11. *For any $c : \mathbb{N}^2 \to [r]$, let T' and T be the predicates from Definition 6.10. The following properties hold:*

(i) *T is an infinite tree.*

(ii) *Every branch of T is the characteristic function of an initial segment of a branch of the E-R tree. More precisely, if we have $T(s)$, then we have that $\{i : s_i = 0\}$ is the initial segment of a branch of the E-R tree.*

(iii) *The following monotonicity conditions hold:*

  (M1) *$T'(s * t, k) \to T'(s, k)$.*

  (M2) *$T'(s, k) \to T'(s, k + m)$.*

Proof. For (i), note that it is clear by definition that T is prefix closed. Also, for every n there is $s : \mathbb{B}^n$ such that $T(s)$, taking simply s such that $s_i = 0$ iff $i \prec n$.

For (ii), assume $T(s)$ holds. Then $T'(s, k)$ holds for some k, and so there is $k' \in [|s|, k]$ such that for all $i < |s|$, $s_i = 0$ iff $i \prec k'$. Hence, $\{i : s_i = 0\}$ is an initial segment of $\mathrm{pd}(k')$.

Finally, (iii) is obvious. □

The following lemma uses Π^0_1-AC^0 in order to prove the existence of a function β which will allow for turning T into a decidable tree.

Proposition 6.12. *For any* $c : \mathbb{N}^2 \to [r]$*, there exists a function* $\beta : \mathbb{N} \to \mathbb{N}$ *such that:*

$$\forall n \forall s^{\mathbb{B}^*} \big(|s| = n \wedge \exists k T'(s,k) \to T'(s, \beta n)\big). \tag{def-β}$$

Proof. Clearly, we have:

$$\forall n \forall s \big[|s| = n \to \big(\exists k T'(s,k) \to \exists m T'(s,m)\big)\big].$$

Classically we can derive:

$$\forall n \forall s \big[|s| = n \to \exists m \big(\exists k T'(s,k) \to T'(s,m)\big)\big].$$

Since there are finitely many $s : \mathbb{B}^n$, using condition (M2) we obtain:

$$\forall n \exists m \forall s \big[|s| = n \to \big(\exists k T'(s,k) \to T'(s,m)\big)\big].$$

Finally, since, as we have discussed at the beginning of this chapter, $s : \mathbb{B}^*$ is primitive recursively encoded by $s : \mathbb{N}$, we can apply Π^0_1-AC^0 and obtain:

$$\exists \beta^{\mathbb{N}^\mathbb{N}} \forall n, s \big(|s| = n \wedge \exists k T'(s,k) \to T'(s, \beta n)\big).$$

$\square$

Assuming we are given this function β, our tree T becomes decidable. The only undecidable element in T was the unbounded existential quantifier on k. The idea is that β gives us, for each n, a $k = \beta n$ uniform for every $s : \mathbb{B}^n$.

Corollary 6.13. *Given a colouring* $c : \mathbb{N}^2 \to [r]$ *and a function* $\beta : \mathbb{N} \to \mathbb{N}$ *satisfying* (def-β)*, for each* $n : \mathbb{N}$ *and each* $s : \mathbb{B}^n$*,* $T(s)$ *is equivalent to:*

$$T^\beta(s) := T'(s, \beta n).$$

Now we are ready to apply WKL.

Proposition 6.14. *Given a colouring* $c : \mathbb{N}^2 \to [r]$ *and a function* $\beta : \mathbb{N} \to \mathbb{N}$ *satisfying* (def-β)*, let* T^β *be as in Corollary 6.13. There exists an infinite sequence* $\alpha : \mathbb{N} \to \mathbb{B}$ *such that:*

$$\forall n T^\beta([\alpha](n)),$$

i.e.,

$$\forall n \exists k \in [n, \beta n]\, \forall i < n \big(\alpha(i) = 0 \leftrightarrow i \prec k\big). \tag{def-α}$$

Proof. A direct application of $\mathsf{WKL}(T^\beta)$. $\square$

We need to see now that this infinite branch of T encodes an infinite branch of the E-R tree, that is, it has infinitely many zeros.

Proposition 6.15. *Given a colouring $c : \mathbb{N}^2 \to [r]$ and a function $\beta : \mathbb{N} \to \mathbb{N}$ satisfying* (def-β)*, let α be an infinite sequence as obtained in Proposition 6.14. Then, α is the characteristic function of an infinite set, i.e., it has infinitely many zeros. Furthermore, we can construct a function $a : \mathbb{N} \to \mathbb{N}$ such that, for all n, $a(n)$ is the first $k \geq n$ with $\alpha(k) = 0$.*

Proof. Given β, we first define an auxiliary function $\gamma : \mathbb{N} \to \mathbb{N}$ as $\gamma m := \beta m + 1$. For every m, define:

$$a(m) := \begin{cases} 0 & \text{if } m = 0 \\ k & \text{for the least } k \in [m, \beta(\gamma^{r-1}(m))] \text{ with } \alpha(k) = 0. \end{cases}$$

If we show that a is well-defined, the lemma will follow, since the image of a is unbounded.

We have that $a(0) = 0$, and by definition of $\prec$, $\alpha(0) = 0$. Let us now assume $m > 0$, and let $i < m$ be the greatest such that $\alpha(i) = 0$. We will find $k_0, k_1, \ldots, k_r \in \mathbb{N}$ pairwise different such that $i \prec k_j$ for all $j = 0, 1, \ldots, r$.

By (def-α) applied with $n := m$, there is $k_0 \in [m, \beta m]$ such that $i \prec k_0$. Applying (def-α) now with $n := \beta m + 1 = \gamma m$, we obtain $k_1 \in [\gamma m, \beta(\gamma m)]$ such that $i \prec k_1$. If we apply this procedure $r + 1$ times we get:

$$\begin{aligned} k_0 &\in [m, \beta m], \\ k_1 &\in [\gamma m, \beta(\gamma m)], \\ &\vdots \\ k_{r-1} &\in [\gamma^{r-1} m, \beta(\gamma^{r-1} m)], \\ k_r &\in [\gamma^r m, \beta(\gamma^r m)], \end{aligned}$$

such that $i \prec k_j$ for all $j = 0, \ldots, r$. We observe that, by the definition of γ, the above intervals are all disjoint, and so all k_j are different. Now since the E-R tree is r-ary branching, some of these have to be comparable, i.e., there are j_1, j_2 such that $j_1 < j_2$ and $k_{j_1} \prec k_{j_2}$. But now recall that, by (def-α), $k_{j_2} \in [\gamma^{j_2} m, \beta(\gamma^{j_2} m)]$ satisfies:

$$\forall i' < \gamma^{j_2} m \big(\alpha(i') = 0 \leftrightarrow i' \prec k_{j_2}\big).$$

Therefore, $\alpha(k_{j_1}) = 0$, and so, we have, as we required, some $k \in [m, \beta(\gamma^{r-1}(m))]$ such that $\alpha(k) = 0$. Thus a is well-defined. ☐

Now that we know that a encodes an infinite branch of the E-R tree, we have the following:

Corollary 6.16. *Given $c : \mathbb{N}^2 \to [r]$ and $\beta : \mathbb{N} \to \mathbb{N}$ satisfying* (def-β)*, let $a : \mathbb{N} \to \mathbb{N}$ be as constructed in Proposition 6.15. The set $\{an : n \in \mathbb{N}\}$ with the order $<$ is an infinite min-monochromatic set under the colouring c, i.e., for any $i, j, k : \mathbb{N}$, if $ai < aj < ak$, then $c(ai, aj) = c(ai, ak)$.*

Proof. By Proposition 6.15 the set is infinite. Moreover, if $ai < aj < ak$, applying (def-α) with $n := ak + 1$, we obtain that ai, aj and ak are $\prec$-predecessors of the same node, and therefore by Lemma 6.9, we obtain $ai \prec aj \prec ak$. Now this implies $c(ai, aj) = c(ai, ak)$. ☐

In fact, the property of a that we will use is:

$$\exists a^{\mathbb{N}^\mathbb{N}} \forall n \big(an \geq n \wedge \forall i, j, k < n \big(ai < aj \wedge ai < ak \to c(ai, aj) = c(ai, ak)\big)\big). \quad \text{(def-}a\text{)}$$

We can give now a proof of Ramsey's theorem, which will involve an application of IPHP.

Theorem 6.17 (Ramsey's theorem)**.** *For every colouring* $c : \mathbb{N}^2 \to [r]$*,*

$$\exists x^{[r]}, F^{\mathbb{N}\to\mathbb{N}} \forall k \big(Fk \geq k \wedge \forall i, j \leq k \big(Fi < Fj \to c(Fi, Fj) = x\big)\big).$$

Proof. We consider the function a defined as above, and so satisfying (def-a). We define a colouring $c^a : \mathbb{N} \to [r]$ as:

$$c^a(i) := c\big(ai, a(ai + 1)\big).$$

We notice that $c^a(i)$ is the colour assigned to a pair of two consecutive elements of the infinite E-R branch under c. By IPHP there is an $x : [r]$ and a $p : \mathbb{N} \to \mathbb{N}$ such that for all k:

$$pk \geq k \wedge c^a(pk) = x.$$

We prove that $F := a \circ p$ works. We have that $F(k) = a(pk) \geq a(k) \geq k$. Moreover, if $Fi < Fj$, we have to see that $c(Fi, Fj) = x$. Since $a(pi) < a(pj)$, we have that $a(a(pi) + 1) \leq a(pj)$, and so, as seen above,

$$c\big(a(pi), a(a(pi) + 1)\big) = c\big(a(pi), a(pj)\big).$$

But then,

$$x = c^a(pi) = c\big(a(pi), a(a(pi) + 1)\big) = c\big(a(pi), a(pj)\big) = c(Fi, Fj).$$

□

6.3 The Dialectica interpretation in action

In order to extract computational information from the classical proof above, a possible path would be to apply the negative translation and then the Dialectica interpretation to our theorem, and then proceed as in the proof of soundness to obtain a term realizing the interpretation. But it turns out that this is totally unpractical, because of two reasons: first, the classical proof above is by no means a formal proof of our system of analysis, as writing down all the steps and details would probably take more pages than this thesis; second, the Dialectica interpretation of double negated formulas becomes unreadable for human beings, and one of our aims is to understand the semantics of the interpreted theorem.

So, instead, we present here convenient transformations of the main lemmas and principles used in the proof. Basically, we apply the negative translation, then the Dialectica interpretation, and finally we transform the result into another formula that is more readable.

The outline of this section is as follows: first we explain in full detail the transformation announced above in the simple but paradigmatic case of IPHP. Then we give (this relaxed version of) the interpretation of the other principles and theorems needed.

Recall that the IPHP states that for every colouring $c : \mathbb{N} \to [r]$, we have:

$$\exists x^{[r]}, p^{\mathbb{N}\to\mathbb{N}} \forall k \big(pk \geq k \wedge c(pk) = x\big). \tag{6.2}$$

First, the negative translation of this is:

$$\neg\neg\exists x^{[r]}, p^{\mathbb{N}\to\mathbb{N}} \forall k \neg\neg \big(pk \geq k \wedge c(pk) = x\big).$$

The Dialectica interpretation of this formula is (supressing double negations in front of quantifier-free formulas):

$$\begin{gathered}\exists X^{([r]\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N})\to[r]}, P^{([r]\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N})\to\mathbb{N}^{\mathbb{N}}} \forall K^{[r]\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N}} \\ \big((PK)(K(XK)(PK)) \geq K(XK)(PK) \\ \wedge\, c((PK)(K(XK)(PK))) = XK\big).\end{gathered}$$

We observe that, after applying D, occurrences of x in the matrix of (6.2) have been replaced by occurrences of XK; of p, by PK; and of k, by $K(XK)(PK)$.

So now, in order to make the dependencies of X and P on K implicit, we swap the universal and existential quantifier. That is, instead of saying that there are functionals X and P such that for any value of K, XK and PK satisfy something, we say that for every value of K there are values x and p satisfying something. In this step we obtain a weaker formula, since in general we cannot go back to the previous form without some form of choice. Anyway, it is a matter of notational convenience, and our proof will give explicit constructions. Here we also change the name of K to ε, to be consistent with our notation on selection functions. In doing this, we obtain:

$$\forall \varepsilon^{[r]\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N}} \exists x^{[r]}, p^{\mathbb{N}^{\mathbb{N}}} \big(p(\varepsilon x p) \geq \varepsilon x p \wedge c(p(\varepsilon x p)) = x\big).$$

But we have lost important information, because now we are saying that x and p satisfy the conditions just for one input. To fix this, we come back to the original IPHP and we consider, for each colouring $c : \mathbb{N} \to [r]$, the following equivalent statement:

$$\exists x^{[r]}, p^{\mathbb{N}\to\mathbb{N}} \forall k \forall i \leq k \big(pi \geq i \wedge c(pi) = x\big).$$

Observe that the bounded quantifier can be encoded as a primitive recursive function and, therefore, we can consider the formula $\forall i \leq k(pi \geq i \wedge c(pi) = x)$ as quantifier-free. Therefore, after applying Dialectica we obtain:

$$\begin{gathered}\exists X^{([r]\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N})\to[r]}, P^{([r]\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N})\to\mathbb{N}^{\mathbb{N}}} \forall K^{[r]\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N}} \\ \forall i \leq K(XK)(PK)\big[(PK)i \geq i \wedge c\big((PK)i\big) = XK\big].\end{gathered}$$

Now, we can take the K out as before, and we obtain:

$$\overline{\mathsf{IPHP}(c)} : \forall \varepsilon^{[r]\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N}} \exists x^{[r]}, p^{\mathbb{N}^{\mathbb{N}}} \forall i \le \varepsilon_x p \big(pi \ge i \wedge c(pi) = x\big).$$

This is our interpretation of the infinite pigeon-hole principle. The intuition is that ε tries to give a counterexample to $\mathsf{IPHP}(c)$, in the sense that a counterexample to $\mathsf{IPHP}(c)$ would be a functional $\varepsilon : [r] \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ such that for every colour $x : [r]$ and function $p : \mathbb{N}^{\mathbb{N}}$, the value εxp tells us 'x and p fail in some $i \le \varepsilon xp$', i.e., for some $i \le \varepsilon xp$, $pi < i$ or $c(pi) \neq x$. But $\overline{\mathsf{IPHP}(c)}$ states that for every possible counterexample, there are a colour and a set that prove it wrong.

We proceed as in [38] and summarize the results of applying the same kind of reasoning to other formulas that we be used. From the original

$$\mathsf{RT}^2_r(c) : \exists x^{[r]}, F^{\mathbb{N}\to\mathbb{N}} \forall k \big(Fk \ge k \wedge \forall i, j \le k (Fi < Fj \to c(Fi, Fj) = x)\big),$$

we get our interpreted version

$$\overline{\mathsf{RT}^2_r(c)} : \ \forall \eta^{[r]\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N}} \exists x^{[r]}, F^{\mathbb{N}^{\mathbb{N}}} \forall k \le \eta_x F \\ \Big(Fk \ge k \wedge \forall i, j \le k \big(Fi < Fj \to c(Fi, Fj) = x\big)\Big).$$

Also, (def-β), which is the key property of β and reads:

$$\exists \beta^{\mathbb{N}^{\mathbb{N}}} \forall n \forall s^{\mathbb{B}^*} \big(|s| = n \wedge \exists k T'(s, k) \to T'(s, \beta n)\big),$$

becomes:

$$\forall \tilde{\omega}^{\mathbb{N}^{\mathbb{N}}\to\mathbb{N}}, \tilde{q}^{\mathbb{N}^{\mathbb{N}}\to\mathbb{N}} \exists \beta^{\mathbb{N}^{\mathbb{N}}} \forall n \le \tilde{\omega}\beta \ \forall s^{\mathbb{B}^n} \big(\exists k \le \tilde{q}\beta \, T'(s, k) \to T'(s, \beta n)\big). \tag{$\overline{\text{def-}\beta}$}$$

Proposition 6.14 is interpreted as:

$$\forall \omega^{\mathbb{B}^{\mathbb{N}}\to\mathbb{N}^{\mathbb{N}}\to\mathbb{N}} \exists \alpha^{\mathbb{B}^{\mathbb{N}}}, \beta^{\mathbb{N}^{\mathbb{N}}} \forall n \le \omega\alpha\beta \ \ T^{\beta}([\alpha](n)). \tag{$\overline{\text{def-}\alpha}$}$$

And finally, (def-a) becomes:

$$\forall \psi^{\mathbb{N}^{\mathbb{N}}\to\mathbb{N}} \exists a^{\mathbb{N}^{\mathbb{N}}} \forall n \le \psi a \\ \Big(an \ge n \wedge \forall i, j, k < n \big(ai < aj \wedge ai < ak \to c(ai, aj) = c(ai, ak)\big)\Big). \tag{$\overline{\text{def-}a}$}$$

6.4 A constructive proof

Now that all our main results of the previous section are interpreted, the key observation is that all interpreted statements ask for finite approximations of the infinite sets whose existence is classically proved, since the inner quantifiers are bounded and so the properties need to be satisfied over a finite number of inputs.

This section is devoted to extracting from the classical proof of Section 6.2 a program that computes arbitrarily good approximations to a pairwise monochromatic set. The proofs of the theorems will reveal how much of each infinite set or function (such as α or β) is necessary. Hence, the theorems presented in this section are versions of those of Section 6.2 that include bounds on the hypotheses.

6.4.1 Interpreting our application of Π^0_1-AC^0

This subsection is devoted to witnessing ($\overline{\text{def-}\beta}$). The following lemma defines the selection functions that will achieve this.

Lemma 6.18. *Let $c : \mathbb{N}^2 \to [r]$ be a colouring. Define $\delta := \langle \delta_n \rangle_{n \in \mathbb{N}}$, where for each $n \in \mathbb{N}$, $\delta_n : J_{\mathbb{N}}\mathbb{N}$ is defined as*

$$\delta_n p := p^i(0)$$

for every $p : \mathbb{N}^{\mathbb{N}}$, where i is the least $\leq 2^n$ such that for all $s : \mathbb{B}^n$,

$$T'(s, p^{i+1}(0)) \to T'(s, p^i(0)).$$

Then, for every $n : \mathbb{N}$ and $p : \mathbb{N}^{\mathbb{N}}$, we have:

$$\forall s^{\mathbb{B}^n} \big(T'(s, p(\delta_n p)) \to T'(s, \delta_n p) \big). \tag{6.3}$$

Proof. We note that (6.3) is obvious once we have established that δ_n is well-defined, that is, that such an i exists. Let us proceed by contradiction. Thus, we assume that for all $i \leq 2^n$ there exists $s : \mathbb{B}^n$ such that $T'(s, p^{i+1}(0))$ and $\neg T'(s, p^i(0))$. By (M2), this implies that for all $i \leq 2^n$, $p^i(0) \leq p(p^i(0))$.

Since there are $2^n + 1$ possible values for i and only 2^n for s, there are by the assumption $i < j$ such that for the same $s : \mathbb{B}^n$,

$$\begin{aligned} &T'(s, p^{i+1}(0)), \quad \neg T'(s, p^i(0)); \\ &T'(s, p^{j+1}(0)), \quad \neg T'(s, p^j(0)). \end{aligned}$$

But since $i + 1 \leq j$, and so $p^{i+1}(0) \leq p^j(0)$, $T'(s, p^{i+1}(0))$ and $\neg T'(s, p^j(0))$ yield a contradiction. □

Now we can prove ($\overline{\text{def-}\beta}$) using EPS and the lemma above.

Proposition 6.19 (Interpreted Proposition 6.12)**.** *Let $c : \mathbb{N}^2 \to [r]$ be a colouring. Given $\tilde{\omega}, \tilde{q} : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$, define δ as in Lemma 6.18 and:*

$$\beta := \mathsf{EPS}_0 \langle\rangle \tilde{\omega} \delta \tilde{q}.$$

Then β satisfies ($\overline{\text{def-}\beta}$) applied to $\tilde{\omega}, \tilde{q}$, that is:

$$\forall n \leq \tilde{\omega}\beta \; \forall s^{\mathbb{B}^n} \big(\exists k \leq \tilde{q}\beta \, T'(s, k) \to T'(s, \beta n) \big).$$

Proof. By Theorem 5.3, if $n \leq \tilde{\omega}\beta$, there is $p : \mathbb{N}^{\mathbb{N}}$ such that $\beta n = \delta_n(p)$ and $\tilde{q}\beta = p(\delta_n p)$. By (6.3), we have:

$$\forall n \leq \tilde{\omega}\beta \forall s^{\mathbb{B}^n} \big(T'(s, \tilde{q}\beta) \to T'(s, \beta n) \big).$$

Using (M2), we obtain:

$$\forall n \leq \tilde{\omega}\beta \; \forall s^{\mathbb{B}^n} \big(\exists k \leq \tilde{q}\beta \, T'(s, k) \to T'(s, \beta n) \big),$$

as required. □

6.4.2 Interpreting our application of WKL

The aim of this subsection is to prove ($\overline{\text{def-}\alpha}$):

$$\forall \omega^{\mathbb{B}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}} \exists \alpha^{\mathbb{B}^{\mathbb{N}}}, \beta^{\mathbb{N}^{\mathbb{N}}} \forall n \leq \omega\alpha\beta \;\; T^{\beta}([\alpha](n)).$$

To this end, we will use a sufficient approximation of β.

Definition 6.20. Given a predicate P on $\mathbb{B}^*$, for each $n : \mathbb{N}$ we define a predicate:

$$\mathsf{Depth}_n(P) :\equiv \exists s^{\mathbb{B}^n} P(s).$$

In order to satisfy ($\overline{\text{def-}\alpha}$), given $\omega : \mathbb{B}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$, we will first build $\alpha : \mathbb{B}^{\mathbb{N}}$ and $\beta : \mathbb{N}^{\mathbb{N}}$ such that:

$$\forall n < \omega\alpha\beta \big(\mathsf{Depth}_{\omega\alpha\beta - n}(T^{\beta}_{[\alpha](n)}) \to \mathsf{Depth}_{\omega\alpha\beta - n - 1}(T^{\beta}_{[\alpha](n+1)})\big). \tag{6.4}$$

Also, Lemma 6.21 below will establish $\mathsf{Depth}_{\omega\alpha\beta}(T^{\beta})$, and so applying (6.4) $\omega\alpha\beta$ times, we will obtain $\mathsf{Depth}_0(T^{\beta}([\alpha](\omega\alpha\beta)))$, that is, $T^{\beta}([\alpha](\omega\alpha\beta))$, and so, we will have ($\overline{\text{def-}\alpha}$).

The key fact is that the construction of an approximation of α does not require the whole β to be constructed, since it will need just a finite number of calls to β. Hence, given ω, there are concrete $\tilde{\omega}, \tilde{q}$ such that the β obtained applying ($\overline{\text{def-}\beta}$) to them is sufficient for approximating α up to $\omega\alpha\beta$. We will give explicit expressions for $\tilde{\omega}, \tilde{q}$ by inspecting the following proofs: more precisely, we will pay attention to how much of β is used and, as in [38], we will highlight every use of it with a $\boxed{\text{box}}$. Lemmas 6.21 and 6.23 and Proposition 6.26 assume the existence of an (ineffective) β satisfying (def-β). Each one of these Lemmas and Proposition is followed respectively by Corollary 6.22, 6.24 and 6.27, which are counterparts that assume just an approximation of β: these corollaries define explicit $\tilde{\omega}$ and $\tilde{q}$ using the bounds found in the proof of their corresponding Lemma or Proposition, and then give a bounded version thereof for the approximation of β satisfying:

$$\forall n \leq \tilde{\omega}\beta \; \forall s^{\mathbb{B}^n} \big(\exists k \leq \tilde{q}\beta \, T'(s, k) \to T'(s, \beta n)\big). \tag{6.5}$$

Lemma 6.21. *Let $c : \mathbb{N}^2 \to [r]$ be a colouring, and assume that β is a function satisfying* (def-β). *The tree T^{β} has branches of arbitrary length, i.e., for all $m : \mathbb{N}$ there exists $s : \mathbb{B}^m$ such that $T^{\beta}(s)$.*

Proof. Recall that:

$$T^{\beta}(s) = T'(s, \beta|s|) = \exists k' \in [|s|, \beta|s|] \; \forall i < |s| (s_i = 0 \leftrightarrow i \prec k').$$

Given m, define $s : \mathbb{B}^m$ as for all $i < m$, $s_i = 0$ iff $i \prec m$. This implies $T'(s, m)$. Therefore, by (def-β) with $\boxed{n = m \text{ and } k = m}$, we have $T'(s, \beta m)$. □

Corollary 6.22. *Let $c : \mathbb{N}^2 \to [r]$ be a colouring. Given $m : \mathbb{N}$, define for all $\beta : \mathbb{N}^{\mathbb{N}}$,*

$$\begin{aligned} \tilde{\omega}\beta &:= m, \\ \tilde{q}\beta &:= m. \end{aligned} \tag{6.6}$$

Let β be as in Proposition 6.19, hence satisfying (6.5). *For any $j \leq m$, the tree T^{β} has branches of length j, i.e., for all there exists $s : \mathbb{B}^j$ such that $T^{\beta}(s)$.*

Proof. The same proof of Lemma 6.21, replacing the use of (def-β) with (6.5). $\square$

Recall that by the notation T_s^β we mean that, given $t : \mathbb{B}^*$, $T_s^\beta(t) = T^\beta(s * t)$.

Lemma 6.23. *Let $c : \mathbb{N}^2 \to [r]$ be a colouring and suppose that $\beta : \mathbb{N} \to \mathbb{N}$ satisfies* (def-β)*. For each $s : \mathbb{B}^*$, define the selection function $\varepsilon_s : J_{\mathbb{N}}\mathbb{B}$ by, for each $p : \mathbb{B} \to \mathbb{N}$,*

$$\varepsilon_s p := \begin{cases} 0 & \textit{if } \mathsf{Depth}_{p(0)+1}(T_s^\beta) \to \mathsf{Depth}_{p(0)}(T_{s*0}^\beta) \\ 1 & \textit{otherwise.} \end{cases}$$

Then,

$$\forall p^{\mathbb{B}\to\mathbb{N}} \forall s^{\mathbb{B}^*} \big(\mathsf{Depth}_{p(\varepsilon_s p)+1}(T_s^\beta) \to \mathsf{Depth}_{p(\varepsilon_s p)}(T_{s*\varepsilon_s p}^\beta)\big).$$

Proof. Fix s and p and assume $\mathsf{Depth}_{p(\varepsilon_s p)+1}(T_s^\beta)$. Let $m = |s|$. If

$$\mathsf{Depth}_{p(0)+1}(T_s^\beta) \to \mathsf{Depth}_{p(0)}(T_{s*0}^\beta)$$

holds, then $\varepsilon_s p = 0$ and we are done.

If not, then we have

$$\mathsf{Depth}_{p(0)+1}(T_s^\beta) \wedge \neg\mathsf{Depth}_{p(0)}(T_{s*0}^\beta)$$

and $\varepsilon_s p = 1$. But then, the assumption $\mathsf{Depth}_{p(\varepsilon_s p)+1}(T_s^\beta)$ says $\mathsf{Depth}_{p(1)+1}(T_s^\beta)$. Our current list of assumptions is:

(i) $\mathsf{Depth}_{p(1)+1}(T_s^\beta)$.

(ii) $\mathsf{Depth}_{p(0)+1}(T_s^\beta)$.

(iii) $\neg\mathsf{Depth}_{p(0)}(T_{s*0}^\beta)$.

We consider two cases:

Case 1. $p(0) \geq p(1)$.

By (ii) and (iii) we have $\mathsf{Depth}_{p(0)}(T_{s*1}^\beta)$. Therefore, there is $t' : \mathbb{B}^{p(0)}$ such that $T'(s * 1 * t', \beta(m + 1 + p(0)))$. By (M1), since $p(0) \geq p(1)$, we obtain $t : \mathbb{B}^{p(1)}$ such that $T'(s * 1 * t, \beta(m + 1 + p(0)))$. Applying (def-$\beta$) with $\boxed{n = m + 1 + p(1) \text{ and } k = \beta(m + 1 + p(0))}$, we obtain

$$\exists t^{\mathbb{B}^{p(1)}} T'(s * 1 * t, \beta(m + 1 + p(1))),$$

which is equivalent to $\mathsf{Depth}_{p(1)}(T_{s*1}^\beta)$.

Case 2. $p(0) < p(1)$.

Applying (def-β) with $\boxed{n = m + 1 + p(0) \text{ and } k = \beta(m + 1 + p(1))}$, together with (iii) we obtain:

$$\forall t^{\mathbb{B}^{p(0)}} \neg T'(s * 0 * t, \beta(m + 1 + p(1))).$$

Then by (M1) again,

$$\forall t^{\mathbb{B}^{p(1)}} \neg T'(s * 0 * t, \beta(m + 1 + p(1))).$$

which is equivalent to $\neg\mathsf{Depth}_{p(1)}(T^{\beta}_{s*0})$. By (i) we get $\mathsf{Depth}_{p(1)}(T^{\beta}_{s*1})$.

□

Now we give a version of this Lemma that does not require the whole β to be constructed, using the bounds on n and k from (def-β) that we have indicated with a box.

Corollary 6.24. *Let $c : \mathbb{N}^2 \to [r]$ be a colouring. Given $m : \mathbb{N}$ and $p : \mathbb{B} \to \mathbb{N}$, define for any $\beta : \mathbb{N} \to \mathbb{N}$:*

$$\begin{aligned} \tilde{\omega}\beta &:= m + 1 + \max\{p(0), p(1)\}, \\ \tilde{q}\beta &:= \max\nolimits_{i \le \tilde{\omega}\beta} \beta i. \end{aligned} \tag{6.7}$$

Fix β as the one obtained from Proposition 6.19, i.e., β satisfies (6.5) *with $\tilde{\omega}, \tilde{q}$ as defined above. Define ε_s as in Lemma 6.23. Then, for all $k \le m$,*

$$\forall s^{\mathbb{B}^k} \big(\mathsf{Depth}_{p(\varepsilon_s p)+1}(T^{\beta}_{s}) \to \mathsf{Depth}_{p(\varepsilon_s p)}(T^{\beta}_{s*\varepsilon_s p})\big).$$

Proof. The proof of Lemma 6.23 works, again replacing (def-β) with (6.5). □

We now use these facts to construct α. First we need the following definition:

Definition 6.25. Let $c : \mathbb{N}^2 \to [r]$ be a colouring. Given $\omega : \mathbb{B}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ and $\beta : \mathbb{N}^{\mathbb{N}}$, define the function $q : \mathbb{B}^{\mathbb{N}} \to \mathbb{N}$ as, for each $\alpha : \mathbb{B}^{\mathbb{N}}$,

$$q\alpha := \omega\alpha\beta - n - 1,$$

where $n < \omega\alpha\beta$ is the least refuting (6.4), and $\omega\alpha\beta - 1$ if no such n exists.

The following proposition states the existence of our α assuming the (ineffective) existence of a function β satisfying (def-β). Throughout its proof, we highlight the uses of the lemmas above.

Proposition 6.26. *Let $c : \mathbb{N}^2 \to [r]$ be a colouring and suppose that $\beta : \mathbb{N} \to \mathbb{N}$ satisfies* (def-β)*. Given $\omega : \mathbb{B}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$, consider $\omega_\beta := \lambda\alpha.\omega\alpha\beta$. Let q be as in Definition 6.25 and ε as in Lemma 6.23. The sequence:*

$$\alpha := \mathsf{EPS}_0\langle\rangle\omega_\beta\varepsilon q$$

satisfies $T^{\beta}([\alpha](\omega_\beta\alpha))$.

Proof. By Theorem 5.3, we know that α and $p_{[\alpha](n)}$ as defined in the theorem, for $n \le \omega_\beta\alpha$, satisfy:

$$\begin{aligned} \alpha n &= \varepsilon_{[\alpha](n)} p_{[\alpha](n)}, \\ q\alpha &= p_{[\alpha](n)}(\varepsilon_{[\alpha](n)} p_{[\alpha](n)}). \end{aligned}$$

By Lemma 6.23,

$$\forall p^{\mathbb{B}\to\mathbb{N}} \forall s^{\mathbb{B}^*} \big(\mathsf{Depth}_{p(\varepsilon_s p)+1}(T^\beta_s) \to \mathsf{Depth}_{p(\varepsilon_s p)}(T^\beta_{s*\varepsilon_s p})\big).$$

Taking $\boxed{s = [\alpha](\omega_\beta\alpha - q\alpha - 1) \text{ and } p = p_s}$, we obtain:

$$\mathsf{Depth}_{q\alpha+1}(T^\beta_{[\alpha](\omega_\beta\alpha - q\alpha - 1)}) \to \mathsf{Depth}_{q\alpha}(T^\beta_{[\alpha](\omega_\beta\alpha - q\alpha)}). \tag{6.8}$$

Assume towards a contradiction that (6.4) does not hold. Then, $q\alpha = \omega_\beta\alpha - n - 1$ for some $n < \omega_\beta\alpha$ such that:

$$\neg\big(\mathsf{Depth}_{\omega_\beta\alpha - n}(T^\beta_{[\alpha](n)}) \to \mathsf{Depth}_{\omega_\beta\alpha - n - 1}(T^\beta_{[\alpha](n+1)})\big),$$

but also (6.8) is equivalent to:

$$\mathsf{Depth}_{\omega_\beta\alpha - n}(T^\beta_{[\alpha](n)}) \to \mathsf{Depth}_{\omega_\beta\alpha - n - 1}(T^\beta_{[\alpha](n+1)}),$$

a contradiction. Therefore, there is no n refuting (6.4), and so it holds.

Since by Lemma 6.21 we have $\mathsf{Depth}_{\omega_\beta\alpha}(T^\beta)$ (that is, we take $\boxed{m = \omega_\beta\alpha}$), applying (6.4) from $n = 0$ to $n = \omega_\beta\alpha - 1$, we conclude $\mathsf{Depth}_0(T^\beta_{[\alpha](\omega_\beta\alpha)})$, i.e., $T^\beta([\alpha](\omega_\beta\alpha))$. □

Corollary 6.27. *Let $c : \mathbb{N}^2 \to [r]$ be a colouring. Given $\omega : \mathbb{B}^\mathbb{N} \to \mathbb{N}^\mathbb{N} \to \mathbb{N}$ and $\beta : \mathbb{N} \to \mathbb{N}$, consider $\omega_\beta := \lambda\alpha.\omega\alpha\beta$. Let q be as in Definition 6.25 and ε as in Lemma 6.23. Define:*

$$\alpha := \mathsf{EPS}_0\langle\rangle\omega_\beta\varepsilon q.$$

If β satisfies (6.5) for:

$$\begin{aligned} \tilde{\omega}\beta &:= \max\{\omega_\beta\alpha, |\omega_\beta\alpha - q\alpha - 1| + \max\{p_s(0), p_s(1)\} + 1\}, \\ \tilde{q}\beta &:= \max\{\omega_\beta\alpha, \max_{i \le \tilde{\omega}\beta} \beta i\}, \end{aligned} \tag{6.9}$$

then we have $T^\beta([\alpha](\omega_\beta\alpha))$.

Proof. The proof of Proposition 6.26 works, replacing the use of Lemma 6.21 with Corollary 6.22 and of Lemma 6.23 with Corollary 6.24. □

Now we are ready to arrange all contents of this subsection into a proof of ($\overline{\text{def-}\alpha}$).

Theorem 6.28 (Interpreted Proposition 6.14)**.** *Let $c : \mathbb{N}^2 \to [r]$ be a colouring. We have ($\overline{\text{def-}\alpha}$), i.e., for every $\omega : \mathbb{B}^\mathbb{N} \to \mathbb{N}^\mathbb{N} \to \mathbb{N}$, there exist $\alpha : \mathbb{B}^\mathbb{N}$ and $\beta : \mathbb{N}^\mathbb{N}$ such that $\forall n \le \omega\alpha\beta\, T^\beta([\alpha](n))$.*

Proof. Fix ω. For any $\beta : \mathbb{N}^\mathbb{N}$, let $\beta \mapsto \alpha^\beta$ denote the construction of α from Corollary 6.27. Consider $\tilde{\omega}$ and $\tilde{q}$ as in (6.9). There is no circularity here, since $\tilde{\omega}\beta, \tilde{q}\beta$ depend on β and on the $\alpha = \alpha^\beta$ constructed from β, not on the particular α satisfying ($\overline{\text{def-}\alpha}$) that we are aiming to build now. Once and for all, fix β as the one obtained from Proposition 6.19 using $\tilde{\omega}, \tilde{q}$. And now, fix $\alpha = \alpha^\beta$. These α and β satisfy the theorem. By Corollary 6.27, we have $T^\beta([\alpha](\omega\alpha\beta))$. If $n \le \omega\alpha\beta$, then by (M1) we obtain $T'([\alpha](n), \beta(\omega\alpha\beta))$. Finally, by (6.5) with $k = \beta(\omega\alpha\beta)$, we obtain $T'([\alpha](n), \beta n) = T^\beta([\alpha](n))$, and so we are done. □

The last step of this part of the proof is to prove ($\overline{\text{def-}a}$):

Proposition 6.29 (Interpreted Proposition 6.15)**.** *Let $c : \mathbb{N}^2 \to [r]$ be a colouring. For all $\psi : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ there exists a function $a : \mathbb{N}^{\mathbb{N}}$ such that for all $n \leq \psi a$,*

$$an \geq n \wedge \forall i, j, k < n\big(ai < aj \wedge ai < ak \to c(ai, aj) = c(ai, ak)\big).$$

Proof. For each $\alpha : \mathbb{B}^{\mathbb{N}}$ and $\beta : \mathbb{N}^{\mathbb{N}}$, define $a^{\alpha,\beta}$ as in Proposition 6.15:

$$a^{\alpha,\beta}(m) := \begin{cases} 0 & \text{if } m = 0 \\ k & \text{for the least } k \in [m, \beta(\gamma^{r-1}(m))] \text{ with } \alpha(k) = 0 \\ 0 & \text{if there is no such } k. \end{cases}$$

We take care of this last case because now α and β are arbitrary, and so a k as in the second case may not exist.

Now, define $\omega : \mathbb{B}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ as, for all $\alpha : \mathbb{B}^{\mathbb{N}}$ and $\beta : \mathbb{N}^{\mathbb{N}}$,

$$\omega\alpha\beta := \max_{i \leq \psi(a^{\alpha,\beta})} \big(\max\{\gamma^j i \mid 0 \leq j \leq r\}\big),$$

and define α and β as in Theorem 6.28. Then $a = a^{\alpha,\beta}$ satisfies our proposition, the proof being the same as in Proposition 6.15, since that proof uses the fact $T^{\beta}([\alpha](n))$ only for $n < \omega\alpha\beta$ as defined above. $\square$

6.4.3 Interpreting our application of IPHP

Theorem 6.30 ($\overline{\mathsf{IPHP}(c)}$)**.** *Let $c : \mathbb{N} \to [r]$ be a colouring. We have:*

$$\forall \varepsilon^{[r] \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}} \exists x^{[r]}, p^{\mathbb{N}^{\mathbb{N}}} \forall i \leq \varepsilon_x p\big(pi \geq i \wedge c(pi) = x\big).$$

Proof. Given $\varepsilon : [r] \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$, for each $x : [r]$ and $p : \mathbb{N}^{\mathbb{N}}$ define $\tilde{\varepsilon}_x p$ as the least $i \leq \varepsilon_x p$ refuting

$$pi \geq i \wedge c(pi) = x,$$

if such an i exists, and as $\varepsilon_x p$, if not.

Let

$$\langle k_0, \ldots, k_{r-1} \rangle := \left(\bigotimes_{x=0}^{r-1} \tilde{\varepsilon}_x \right) (\max),$$

and let $N = \max\{k_0, \ldots, k_{r-1}\}$. By Theorem 5.3 (with constant control function $\lambda s^{\mathbb{N}^{\mathbb{N}}}.(r-1)$), there are $p_0, \ldots, p_{r-1}$ such that for every $x : [r]$,

$$k_x = \tilde{\varepsilon}_x p_x, \quad N = p_x(k_x).$$

Let $x := c(N)$ and $p := p_x$. If there were an $i \leq \varepsilon_x p$ satisfying

$$\neg\big(pi \geq i \wedge c(pi) = x\big),$$

then we would have $p(\tilde{\varepsilon}_x p) < \tilde{\varepsilon}_x p$ or $c(p(\tilde{\varepsilon}_x p)) \neq x$. But this is not the case, since

$$p(\tilde{\varepsilon}_x p) = p_x k_x = N \geq k_x$$

and

$$c(p(\tilde{\varepsilon}_x p)) = c(N) = x.$$

Therefore,

$$\forall i \le \varepsilon_x p\big(pi \ge i \wedge c(pi) = x\big),$$

and we are done. □

Finally, we present the proof of the interpreted Ramsey theorem.

Theorem 6.31 (Interpreted Ramsey theorem)**.** *Let $c : \mathbb{N}^2 \to [r]$ be a colouring. For any $\eta : [r] \to J_{\mathbb{N}}\mathbb{N}$, there exist $F : \mathbb{N}^{\mathbb{N}}$ and $x : [r]$ such that:*

$$\forall k \le \eta_x F\Big(Fk \ge k \wedge \forall i, j \le k\big(Fi < Fj \to c(Fi, Fj) = x\big)\Big).$$

Proof. Let c and η be fixed. For any $a : \mathbb{N}^{\mathbb{N}}$, define a colouring $c^a : \mathbb{N} \to [r]$ as $c^a(i) := c(ai, a(ai+1))$. Define $\varepsilon^a : [r] \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ as $\varepsilon^a_x p := \eta_x(a \circ p)$. By $\overline{\mathsf{IPHP}}(c)$, there are $x^a : [r]$ and $p^a : \mathbb{N}^{\mathbb{N}}$ such that:

$$\forall i \le \varepsilon^a_{x^a} p^a\big(p^a i \ge i \wedge c(p^a i) = x^a\big). \tag{6.10}$$

Now define $\psi : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ as

$$\psi a := \max\nolimits_{i \le \varepsilon^a_{x^a} p^a} p^a i.$$

By Proposition 6.29, there exists $a : \mathbb{N}^{\mathbb{N}}$ such that for all $n \le \psi a$ we have

$$an \ge n \wedge \forall i, j, k < n\big(ai < aj \wedge ai < ak \to c(ai, aj) = c(ai, ak)\big). \tag{6.11}$$

Now take $F := a \circ p^a$ and $x := x^a$. Let $k \le \eta_x F = \varepsilon^a_x p^a$. Then $p^a k \le \psi a$ and we have:

$$F(k) = a(p^a k) \overset{(6.11)}{\ge} p^a k \overset{(6.10)}{\ge} k.$$

Moreover, if $i, j \le k$ and $Fi < Fj$, we have to see that $c(Fi, Fj) = x$. Since $a(p^a i) < a(p^a j)$, we have that $a(a(p^a i) + 1) \le a(p^a j)$, and so, by (6.11),

$$c\big(a(p^a i), a(a(p^a i) + 1)\big) = c\big(a(p^a i), a(p^a j)\big).$$

But then,

$$x = c^a(p^a i) = c\big(a(p^a i), a(a(p^a i) + 1)\big) = c\big(a(p^a i), a(p^a j)\big) = c(Fi, Fj).$$

□

6.5 Summary of the construction

The input is a colouring $c : \mathbb{N}^2 \to [r]$ and a functional $\eta : [r] \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$. Throughout this section, we use superscripts to denote parametrization: for instance, the construction of $x^a : [r]$ builds a colour from any $a : \mathbb{N}^{\mathbb{N}}$. We do not write each time the types of the superscripts, as they are always as follows:

$$\begin{aligned} a &: \mathbb{N}^{\mathbb{N}} \\ \alpha &: \mathbb{B}^{\mathbb{N}} \\ \beta &: \mathbb{N}^{\mathbb{N}} \\ \omega &: \mathbb{B}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N} \\ \tilde{\omega} &: \mathbb{N}^{\mathbb{N}} \to \mathbb{N} \\ \tilde{q} &: \mathbb{N}^{\mathbb{N}} \to \mathbb{N} \end{aligned}$$

Also, recall the definitions of the following predicates:

$$\begin{aligned} T'(s,k) &:= \exists k' \in [|s|, k]\, \forall i < |s| (s_i = 0 \leftrightarrow i \prec k') \\ T^{\beta}(s) &:= T'(s, \beta n) \\ \mathsf{Depth}_n(P) &:= \exists s^{\mathbb{B}^n} P(s) \end{aligned}$$

(A) Construction of $x^a : [r]$ and $F^a : \mathbb{N}^{\mathbb{N}}$ (Theorems 6.30, 6.31).

Define $c^a(i) := c\big(ai, a(ai+1)\big)$ and $\varepsilon^a_x p := \eta_x(a \circ p)$. For each $x : [r]$ and $p : \mathbb{N}^{\mathbb{N}}$ define $\tilde{\varepsilon}^a_x p$ as the least $i \le \varepsilon^a_x p$ refuting

$$pi \ge i \wedge c(pi) = x,$$

if such an i exists, and as $\varepsilon^a_x p$, if not. Let

$$\langle k_0, \ldots, k_{r-1} \rangle := \left(\bigotimes_{x=0}^{r-1} \tilde{\varepsilon}^a_x \right) (\max),$$

$N^a := \max\{k_0, \ldots, k_{r-1}\}$, and $x^a := c^a(N)$.

Also, we give the explicit construction of the $p_0, \ldots, p_{r-1}$ given by Theorem 5.3. For any $k : \mathbb{N}$,

$$p^a_{r-1}(k) := \max\{k_0, \ldots, k_{r-2}, k\}$$

and for $i < r-1$,

$$p^a_i(k) := \max\{k_0, \ldots, k_{i-1}, k, k^k_{i+1}, \ldots, k^k_{r-1}\},$$

where $\langle k^k_{i+1}, \ldots, k^k_{r-1} \rangle$ is defined as:

$$\left(\bigotimes_{j=i+1}^{r-1} \tilde{\varepsilon}^a_j \right) \big(\lambda k'_{i+1}, \ldots, k'_{r-1}. \max\{k_0, \ldots, k_{i-1}, k, k'_{i+1}, \ldots, k'_{r-1}\} \big).$$

Define $F^a := a \circ p^a_{x^a}$.

(B) Construction of $\beta^{\tilde{\omega},\tilde{q}} : \mathbb{N}^{\mathbb{N}}$ (Subsection 6.4.1).

Let $\delta : \mathbb{N} \to J_{\mathbb{N}}\mathbb{N}$ be defined as $\delta_n p := p^i(0)$ where $i \leq 2^n$ is the least such that for all $s : \mathbb{B}^n$, $T'(s, p^{i+1}(0)) \to T'(s, p^i(0))$. Define:

$$\beta^{\tilde{\omega},\tilde{q}} := \mathsf{EPS}_0\langle\rangle\tilde{\omega}\delta\tilde{q}.$$

(C) Construction of $\alpha^{\beta,\omega} : \mathbb{B}^{\mathbb{N}}$ (Results 6.21–6.26).

Define $\omega_\beta = \lambda\alpha.\omega\alpha\beta$ and $q^{\beta,\omega} : \mathbb{B}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ as:

$$q^{\beta,\omega}\alpha := \omega\alpha\beta - n - 1$$

where $n < \omega\alpha\beta$ is the least refuting

$$\forall n < \omega\alpha\beta\big(\mathsf{Depth}_{\omega\alpha\beta-n}(T^{\beta}_{[\alpha](n)}) \to \mathsf{Depth}_{\omega\alpha\beta-n-1}(T^{\beta}_{[\alpha](n+1)})\big),$$

and $\omega\alpha\beta - 1$ if no such n exists.

For each $s : \mathbb{B}^*$, define $\varepsilon^{\beta}_s : J_{\mathbb{N}}\mathbb{B}$ by, for each $p : \mathbb{B} \to \mathbb{N}$,

$$\varepsilon^{\beta}_s p := \begin{cases} 0 & \text{if } \mathsf{Depth}_{p(0)+1}(T^{\beta}_s) \to \mathsf{Depth}_{p(0)}(T^{\beta}_{s*0}) \\ 1 & \text{otherwise.} \end{cases}$$

Define:

$$\alpha^{\beta,\omega} := \mathsf{EPS}_0\langle\rangle\omega_\beta\varepsilon^{\beta}q^{\beta,\omega}.$$

(D) Construction of $\beta^{\omega} : \mathbb{N}^{\mathbb{N}}$ **using (B) and (C)** (Corollary 6.27).

Define, for each $\beta : \mathbb{N}^{\mathbb{N}}$,

$$\begin{aligned} \tilde{\omega}^{\omega}\beta &:= \max\{\omega_\beta\alpha^{\beta,\omega}, |\omega_\beta\alpha^{\beta,\omega} - q^{\beta,\omega}\alpha^{\beta,\omega} - 1| + \max\{p_s(0), p_s(1)\} + 1\}, \\ \tilde{q}^{\omega}\beta &:= \max\{\omega_\beta\alpha^{\beta,\omega}, \max\nolimits_{i \leq \tilde{\omega}\beta} \beta i\}, \end{aligned}$$

where:

$$\begin{aligned} s &:= [\alpha^{\beta,\omega}](\omega\alpha^{\beta,\omega}\beta - q^{\beta,\omega}\alpha^{\beta,\omega} - 1), \\ p_s(x) &:= \overline{\mathsf{EPS}_{|s|+1}(s * x)\omega_\beta\varepsilon^{\beta}}(q^{\beta,\omega}_{s*x}). \end{aligned}$$

Define $\beta^{\omega} := \beta^{\tilde{\omega}^{\omega},\tilde{q}^{\omega}}$.

(E) Construction of $\alpha^{\omega} : \mathbb{N}^{\mathbb{N}}$ **using (C) and (D)** (Theorem 6.28).

Define $\alpha^{\omega} := \alpha^{\beta^{\omega},\omega}$.

(F) Construction of $a^{\alpha,\beta} : \mathbb{N}^{\mathbb{N}}$ (Proposition 6.29).

Define $\gamma m := \beta m + 1$ and:

$$a^{\alpha,\beta}(m) := \begin{cases} 0 & \text{if } m = 0 \\ k & \text{for the least } k \in [m, \beta(\gamma^{r-1}(m))] \text{ with } \alpha(k) = 0 \\ 0 & \text{if there is no such } k. \end{cases}$$

(G) Construction of $\omega : \mathbb{B}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ using (A) and (F) (Prop. 6.29).

Define $\psi : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ as, for each $a : \mathbb{N}^{\mathbb{N}}$,

$$\psi a := \max_{i \leq \eta_{x^a}(a \circ p^a)} p^a i.$$

Define ω as, for each $\alpha : \mathbb{B}^{\mathbb{N}}$ and $\beta : \mathbb{N}^{\mathbb{N}}$,

$$\omega\alpha\beta := \max_{i \leq \psi a^{\alpha,\beta}}(\max\{\gamma^j i \mid 0 \leq j \leq r\}).$$

(H) Construction of $\alpha : \mathbb{B}^{\mathbb{N}}$ and $\beta : \mathbb{N}^{\mathbb{N}}$ using (D), (E) and (G) (Th. 6.28).

Define $\beta := \beta^\omega$ and $\alpha := \alpha^\omega$.

(I) Construction of $a : \mathbb{N}^{\mathbb{N}}$ using (H) (Proposition 6.29).

Define $a := a^{\alpha,\beta}$.

(J) Construction of x and F using (A) and (I) (Theorem 6.31).

Define $x := x^a$ and $F := F^a$.

6.6 A game-theoretic reading of the proof

Each use of EPS in the proof above gives an optimal strategy in some sequential game. In this section we explain what games arise from the interpretations and how an optimal strategy solves them.

6.6.1 The game behind Π^0_1-AC^0

The aim is to witness $(\overline{\text{def-}\beta})$:

$$\forall \tilde{\omega}^{\mathbb{N}^{\mathbb{N}} \to \mathbb{N}}, \tilde{q}^{\mathbb{N}^{\mathbb{N}} \to \mathbb{N}} \exists \beta^{\mathbb{N}^{\mathbb{N}}} \forall n \leq \tilde{\omega}\beta \, \forall s^{\mathbb{B}^n} \big(\exists k \leq \tilde{q}\beta \, T'(s,k) \to T'(s,\beta n)\big).$$

Let us think of $\tilde{\omega}$ as a control function and of $\tilde{q}$ as an outcome function. Then, what we want is a sequence of moves β such that, for n up to $\tilde{\omega}\beta$, all sequences s of length n which are witnessed by $\tilde{q}\beta$, i.e., $T'(s, \tilde{q}\beta)$, are also witnessed by the move at round n, i.e., $T'(s, \beta n)$.

To this end, given $\tilde{\omega}, \tilde{q}$, our game is given by $(\tilde{q}, \delta, \tilde{\omega})$, where δ is defined as in Lemma 6.18. The strategy of δ is, for each n and $p : \mathbb{N}^{\mathbb{N}}$, to pick a $k = \delta_n p$ such that:

$$\forall s^{\mathbb{B}^n} \big(T'(s, p(k)) \to T'(s,k)\big),$$

that is, for each n, all branches of length n that have as a witness the outcome $p(k)$ are already witnessed by the move k. It may be seen as a no-new-branches strategy.

We have seen that β as defined in Proposition 6.19 is an optimal play in the game and satisfies our requirements.

6.6.2 The game behind WKL

The aim is, given $\omega : \mathbb{B}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ and a sufficient approximation of β (which is computed via the above game), to find $\alpha : \mathbb{B}^{\mathbb{N}}$ satisfying:

$$\forall n \leq \omega_\beta \alpha \, T^\beta([\alpha](n)). \tag{6.12}$$

where recall that $\omega_\beta := \lambda \alpha . \omega \alpha \beta$.

Let us think of ω as a control function. What we want is a sequence α that encodes a branch of the tree up to $\omega_\beta \alpha$.

To this end, our selection funtions will have to pick 0 or 1 at each round, depending on which of them can be extended to a sufficiently long branch. Our game is given by (q, ε, ω), where q is as in Definition 6.25 and ε is as in Lemma 6.23. We notice that the set of possible moves at each round is here $\mathbb{B}$.

Recall that q is defined as, for each $\alpha : \mathbb{B}^{\mathbb{N}}$,

$$q\alpha := \omega_\beta \alpha - n - 1,$$

where $n < \omega_\beta \alpha$ is the least refuting

$$\forall n < \omega\alpha\beta \big(\mathsf{Depth}_{\omega\alpha\beta - n}(T^\beta_{[\alpha](n)}) \to \mathsf{Depth}_{\omega\alpha\beta - n - 1}(T^\beta_{[\alpha](n+1)})\big)$$

if it exists, and $\omega\alpha\beta - 1$ if not. So, basically, $q\alpha$ says how much of α is missing for satisfying (6.12), in the sense that, if α satisfies (6.12), then $q\alpha = 0$, and if not, then $\omega_\beta\alpha - q\alpha - 1$ is the first index at which α fails, i.e., we have:

$$\mathsf{Depth}_{q\alpha+1}(T^\beta_{[\alpha](\omega_\beta\alpha - q\alpha - 1)}) \wedge \neg \mathsf{Depth}_{q\alpha}(T^\beta_{[\alpha](\omega_\beta\alpha - q\alpha)}).$$

The strategy that ε implements is the following: if we have already constructed the initial segment of a branch, say s, then $\varepsilon_s p$ tries to satisfy $\mathsf{Depth}_{p(\varepsilon_s p)}(T^\beta_{s * \varepsilon_s p})$, i.e., it picks a boolean b such that if we know that s can be extended to a branch of length $|s| + 1 + p(b)$, then $s * b$ still can.

Since T^β is infinite, the empty sequence $\langle\rangle$ extends to a branch of length $\omega_\beta\alpha$, and that is exactly what α, defined as an optimal play of the game, accomplishes.

6.6.3 The game behind IPHP

The aim is to witness:

$$\forall \varepsilon^{[r] \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}} \exists x^{[r]}, p^{\mathbb{N}^{\mathbb{N}}} \forall i \leq \varepsilon_x p \big(pi \geq i \wedge c(pi) = x\big).$$

Recall the intuition: ε tries to give a counterexample to IPHP, in the sense that a counterexample to IPHP would be a functional $\varepsilon : [r] \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ such that for every colour $x : [r]$ and function $p : \mathbb{N}^{\mathbb{N}}$, the value εxp tells us 'x and p fail in some $i \leq \varepsilon xp$', i.e., for some $i \leq \varepsilon xp$, $pi < i$ or $c(pi) \neq x$. But $\overline{\mathsf{IPHP}(c)}$ states that for every possible counterexample, there is a colour and a set that prove it wrong.

So given $\varepsilon : [r] \to \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$, for each $x : [r]$ and $p : \mathbb{N}^{\mathbb{N}}$ define $\tilde{\varepsilon}_x p$ as the least $i \leq \varepsilon_x p$ refuting

$$pi \geq i \wedge c(pi) = x,$$

if such an i exists, and as $\varepsilon_x p$, if not. This $\tilde{\varepsilon}$ tries to find the first place where x and p fail.

Our game is an r-round game, which is the same as an unbounded game whose control is constant r. So consider the game $(\max, \tilde{\varepsilon}, r)$. Then we compute an optimal play $\langle k_0, \ldots, k_{r-1} \rangle$, whose outcome is $N = \max\{k_0, \ldots, k_{n-1}\}$. For each colour (round) x, k_x is, if it exists, the first place where x and p_x (as defined in the proof of Theorem 6.30) fail. But then we have seen that for $x := c(N)$ and $p := p_x$, we must have:

$$p(\tilde{\varepsilon}_x p) \geq \tilde{\varepsilon}_x p \wedge c(p(\tilde{\varepsilon}_x p)) = x,$$

so $\tilde{\varepsilon}$ fails to find a counterexample to x and p, and hence we necessarily have:

$$\forall i \leq \varepsilon_x p \big(pi \geq i \wedge c(pi) = x \big).$$

Future work

Proof mining is a relatively young field, and as such there is an eager search for new applications. It would be interesting to find more theorems of classical analysis for term extraction, but also to revisit some performed term extractions via bar recursion in order to understand the meaning of the constructive proof in game-theoretic terms.

In [11], selection functions are studied within the context of category theory. A development of the study of the interaction of category theory with selection functions could lead to obtaining sufficient conditions for the existence of selection functions associated to families of quantifiers. It could also be a starting point for finding more applications of the product of selection functions in the realm of general mathematics and computer science.

Moreover, there are variants of EPS that are capable of witnessing variants of the Dialectica interpretation, for instance, the monotone variant. There are some open questions [13] regarding the equivalence between several forms of the product of selection functions and bar recursion.

Regarding Ramsey's theorem, our presentation treats only the case of colourings of pairs. Since the classical proof of Ramsey's theorem for sets of k numbers, with $k \geq 2$, uses the instance of Ramsey's theorem for sets of $k-1$ numbers, the proof mining for the general case, or even for the case $k = 3$, supposes a challenge. We would be interested in studying these cases, and if possible, obtaining a general algorithm with k as an input.

It would also be interesting to program the extracted algorithm for the interpretation of Ramsey's theorem in a functional programming language, such as OCaml, Haskell, or Agda.

References

[1] Jeremy Avigad and Solomon Feferman. Gödel's functional ("dialectica") interpretation. *Handbook of proof theory*, 137:337–405, 1998.

[2] Hans Bekič. *Programming languages and their definition: selected papers*, volume 177. Springer Science & Business Media, 1984.

[3] Gianluigi Bellin. Ramsey interpreted: a parametric version of Ramsey's theorem. *Logic and computation (Pittsburgh, PA, 1987)*, 106:17–37, 1990.

[4] Ulrich Berger, Wilfried Buchholz, and Helmut Schwichtenberg. Refined program extraction from classical proofs. *Annals of Pure and Applied Logic*, 114(1-3):3–25, 2002.

[5] Ulrich Berger and Helmut Schwichtenberg. Program extraction from classical proofs. In *Logic and Computational Complexity*, pages 77–97. Springer, 1995.

[6] Thierry Coquand. An analysis of Ramsey's theorem. *Information and Computation*, 110(2):297–304, 1994.

[7] Thierry Coquand. A direct proof of Ramsey's theorem. *Author's website, revised in*, 1994, 2011.

[8] Charles Delzell. Kreisel's unwinding of Artin's proof. In P. Odifreddi, editor, *Kreiseliana: about and around Georg Kreisel*. 1999.

[9] Paul Erdös, Attila Máté, Andras Hajnal, and Richard Rado. *Combinatorial set theory: partition relations for cardinals*, volume 106. Elsevier, 2011.

[10] Martín Escardó and Paulo Oliva. Computational interpretations of analysis via products of selection functions. In *Conference on Computability in Europe*, pages 141–150. Springer, 2010.

[11] Martín Escardó and Paulo Oliva. Selection functions, bar recursion and backward induction. *Mathematical structures in computer science*, 20(2):127–168, 2010.

[12] Martín Escardó and Paulo Oliva. Sequential games and optimal strategies. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 467, pages 1519–1545. The Royal Society, 2011.

[13] Martín Escardó and Paulo Oliva. Bar recursion and products of selection functions. *The Journal of Symbolic Logic*, 80(1):1–28, 2015.

[14] Martín Escardó, Paulo Oliva, and Thomas Powell. System T and the product of selection functions. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 12. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2011.

[15] Jaime Gaspar. *Proof interpretations: theoretical and practical aspects*. PhD thesis, Technical University of Darmstadt, 2011.

[16] Gerhard Gentzen. Die Widerspruchsfreiheit der reinen Zahlentheorie. *Mathematische Annalen*, 112(1):493–565, 1936.

[17] Gerhard Gentzen. Über das Verhältnis zwischen intuitionistischer und klassischer Arithmetik. *Archiv für mathematische Logik und Grundlagenforschung*, 16(3-4):119–132, 1974.

[18] Valery Glivenko. Sur quelques points de la logique de M. Brouwer. *Bulletins de la classe des sciences*, 15(5):183–188, 1929.

[19] Kurt Gödel. Zur intuitionistischen Arithmetik und Zahlentheorie. *Ergebnisse eines mathematischen Kolloquiums*, 4(1933):34–38, 1933.

[20] Kurt Gödel. Über eine Bisher noch nicht Benützte Erweiterung des Finiten Standpunktes. *Dialectica*, 12(3-4):280–287, 1958.

[21] Kurt Gödel. In what sense is intuitionistic logic constructive? (1941). *Unpublished Essays and Lectures*, pages 189–201, 1995.

[22] Kurt Gödel. Vortrag bei Zilsel (1938). *Kurt Gödel Collected Works*, 3:86–113, 1995.

[23] Ulrich Kohlenbach. Effective bounds from ineffective proofs in analysis: an application of functional interpretation and majorization. *The Journal of Symbolic Logic*, 57(4):1239–1273, 1992.

[24] Ulrich Kohlenbach. Effective moduli from ineffective uniqueness proofs. An unwinding of de La Vallée Poussin's proof for Chebycheff approximation. *Annals of Pure and Applied Logic*, 64(1):27–94, 1993.

[25] Ulrich Kohlenbach. New effective moduli of uniqueness and uniform a priori estimates for constants of strong unicity by logical analysis of known proofs in best approximation theory. *Numerical Functional Analysis and Optimization*, 14(5-6):581–606, 1993.

[26] Ulrich Kohlenbach. *Applied proof theory: proof interpretations and their use in mathematics*. Springer Science & Business Media, 2008.

[27] Ulrich Kohlenbach and Alexander Kreuzer. Ramsey's theorem for pairs and provably recursive functions. *Notre Dame Journal of Formal Logic*, 50(4):427–444, 2009.

[28] Andréi Kolmogorov. On the principle of excluded middle. Translation of Russian original in [47]. 1925.

[29] Georg Kreisel. On the interpretation of non-finitist proofs – Part I. *The Journal of Symbolic Logic*, 16(4):241–267, 1951.

[30] Georg Kreisel. On the interpretation of non-finitist proofs – Part II. *The Journal of Symbolic Logic*, 17(1):43–58, 1952.

[31] Georg Kreisel. Hilbert's programme. *Dialectica*, 12(3-4):346–372, 1958.

[32] Georg Kreisel. Mathematical significance of consistency proofs. *The Journal of Symbolic Logic*, 23(2):155–182, 1958.

[33] Georg Kreisel. Interpretation of analysis by means of constructive functionals of finite types. In *Constructivity in Mathematics*, pages 101–128. North-Holland, Amsterdam, 1959.

[34] Georg Kreisel and Anne S. Troelstra. Formal systems for some branches of intuitionistic analysis. *Annals of mathematical logic*, 1(3):229–387, 1970.

[35] Sigekatu Kuroda. Intuitionistische Untersuchungen der formalistischen Logik. *Nagoya Mathematical Journal*, 2:35–47, 1951.

[36] Horst Luckhardt. Bounds extracted by Kreisel from ineffective proofs. In P. Odifreddi, editor, *Kreiseliana: about and around Georg Kreisel*. 1999.

[37] Paulo Oliva. *Proof Mining in Subsystems of Analysis*. PhD thesis, 2003.

[38] Paulo Oliva and Thomas Powell. A constructive interpretation of Ramsey's theorem via the product of selection functions. *Mathematical structures in computer science*, 25(8):1755–1778, 2015.

[39] Paulo Oliva and Thomas Powell. A game-theoretic computational interpretation of proofs in classical analysis. In R. Kahle and M. Rathjen, editors, *Gentzen's Centenary*, pages 501–531. Springer, 2015.

[40] Thomas Powell. *On Bar Recursive Interpretations of Analysis*. PhD thesis, Queen Mary University of London, 2013.

[41] Frank P. Ramsey. On a problem of formal logic. In *Classic Papers in Combinatorics*, pages 1–24. Springer, 2009.

[42] Morten Heine Sørensen and Pawel Urzyczyn. *Lectures on the Curry-Howard isomorphism*, volume 149. Elsevier, 2006.

[43] Clifford Spector. Provably recursive functionals of analysis: a consistency proof of analysis by an extension of principles formulated in current intuitionistic mathematics. *Recursive function theory*, pages 1–27, 1962.

[44] Anne S. Troelstra. *Metamathematical investigation of intuitionistic arithmetic and analysis*, volume 344. Springer Science & Business Media, 1973.

[45] Anne S. Troelstra. Introductory note to 1958 and 1972. *Kurt Gödel Collected Works*, 2:217–241, 1990.

[46] Dirk van Dalen. *Logic and structure*. Springer, 2004.

[47] Jean van Heijenoort. *From Frege to Gödel: a source book in mathematical logic, 1879-1931*, volume 9. Harvard University Press, 1967.

[48] Wim Veldman and Marc Bezem. Ramsey's theorem and the pigeonhole principle in intuitionistic mathematics. *Journal of the London Mathematical Society*, 2(2):193–211, 1993.