Computable Structure Theory: Beyond the arithmetic

Draft

Antonio Montalbán

Draft of Part 2 - Compiled on September 23, 2022

Saved: Submitted for publication on April 14th, 2022
Compiled: September 23, 2022

Contents

Preface

We all know that in mathematics there are proofs that are more difficult than others, constructions that are more complicated than others, and objects that are harder to describe than others. The objective of *computable mathematics* is to study this complexity, to measure it, and to find out where it comes from. Among the many aspects of mathematical practice, this book concentrates on the complexity of structures. By *structures*, we mean objects like rings, graphs, or linear orderings, which consist of a domain on which we have relations, functions, and constants.

Computable structure theory studies the interplay between complexity and structure. By *complexity*, we mean descriptional or computational complexity, in the sense of how difficult it is to describe or compute a certain object. By *structure*, we refer to algebraic or structural properties of mathematical structures. The setting of computable structure theory is that of countable infinite structures and thus, within the whole hierarchy of complexity levels developed by logicians, the appropriate tools come from computability theory: Turing degrees, the arithmetic hierarchy, the hyperarithmetic hierarchy, etc. These structures are like the ones studied in model theory, and we will use a few basic tools from there too. The intention is not, however, to effectivize model theory, and our motivations are very different than those of model theory. Our motivations come from questions of the following sort: Are there syntactical properties that explain why certain objects (like structures, relations, or isomorphisms) are easier or harder to compute or to describe?

The objective of this book is to describe some of the main ideas and techniques used in the field. Most of these ideas are old, but for many of them, the style of the presentation is not. Over the last few years, the author has developed new frameworks for dealing with these old ideas — for instance, for forcing, r.i.c.e. relations, jumps, Scott ranks, and back-and-forth types. One of the objectives of the book is to present these frameworks in a concise and self-contained form.

The modern state of the field, and also the author's view of the subject, has been influenced greatly by the monograph by Ash and Knight [**AK00**] published in 2000. There is, of course, some intersection between that book and this one. But, even within that intersection, the approach is different.

The intended readers are graduate students and researchers working on mathematical logic. Basic background in computability and logic, as is covered in standard undergraduate courses in logic and computability, is assumed. The objective of this book is to describe some of the main ideas and techniques of the field so that graduate students and researchers can use it for their own research.

This book is part II of a monograph that actually consists of two parts: *within the arithmetic* and *beyond the arithmetic*.

Part I, Within the arithmetic. [**Part 1**] The first book is about the part of the theory that can be developed below a single Turing jump. The first chapters introduce what the author sees as the basic tools to develop the theory: ω-presentations, relations, and $\exists$-atomic structures. It then goes into many topics where there is current active research going on. Many of the topics covered in Part I (like Scott sentences, 1-generics, the method of true stages, categoricity, etc.) are generalized through the transfinite in part II. Here is the list of chapters of part I.

Table of contents of Part 1

Part II, Beyond the arithmetic. This book moves into the realm of the hyperarithmetic and the infinitary languages. To fully analyze the complexity of a structure, staying within the arithmetic hierarchy is not enough. The hyperarithmetic hierarchy goes far enough to capture the complexity levels of relations in almost all structures

— although, we will see there are some structures whose complexity goes just beyond. The first half of Part II develops the basic theory of infinitary logic, Π^1_1 sets, and the hyperarithmetic hierarchy. In the second half, the main chapters are those on forcing and the $0^{(\eta)}$-priority method. The exposition of forcing is only aesthetically new. The presentation of $0^{(\eta)}$-priority method is more than just aesthetically different from previous approaches. It will use the method of η-true stages.

The two books are independent of each other, and there is no need to read part I before reading part II.

Acknowledgements

...

Notation and Conventions from computability theory

The intention of this section is to refresh the basic concepts of computability theory and structures and set up the basic notation we use throughout the book. If the reader has not seen basic computability theory before, this section will be too fast an introduction and we recommend starting with other textbooks like Cutland [**Cut80**], Cooper [**Coo04**], Enderton [**End11**], or Soare [**Soa16**].

The computable functions

A function is *computable* if there is a purely mechanical process to calculate its values. In today's language, we would say that $f\colon \mathbb{N} \to \mathbb{N}$ is computable if there is a computer program that, on input n, outputs $f(n)$. This might appear to be too informal a definition, but the Turing–Church thesis tells us that it does not matter which method of computation you choose: you always get the same class of functions from $\mathbb{N}$ to $\mathbb{N}$. The reader should choose to keep in mind whichever definition of computability feels intuitively most comfortable to him or her, be it Turing machines, μ-recursive functions, lambda calculus, register machines, Pascal, Basic, C++, Java, Haskel, or Python.[*] We will not use any particular definition of computability, and instead, every time we need to define a computable function, we will just describe the algorithm in English and let the reader convince himself or herself that it can be written in the programing language he or she has in mind.

The choice of $\mathbb{N}$ as the domain and image for the computable functions is not as restrictive as it may sound. Every hereditarily finite object[†] can be encoded by just a single natural number. Even if formally we define computable functions as having domain $\mathbb{N}$, we think

[*] For the reader with a computer science background, let us remark that we do not impose any time or space bound on our computations — computations just need to halt and return an answer after finitely many steps using a finite amount of memory.

[†] A hereditarily finite object consists of a finite set or finite tuple of hereditarily finite objects.

of them as using any kind of finitary object as inputs or outputs. This should not be surprising. It is what computers do when they encode everything you see on the screen using finite binary strings, or equivalently, natural numbers written in binary. For instance, we can encode pairs of natural numbers by a single number using the *Cantor pairing function* $\langle x, y\rangle \mapsto ((x+y)(x+y+1)/2+y)$, which is a bijection from $\mathbb{N}^2$ to $\mathbb{N}$ whose inverse is easily computable too. One can then encode triples by using pairs of pairs, and then encode n-tuples, and then tuples of arbitrary size, and then tuples of tuples, etc. In the same way, we can consider standard effective bijections between $\mathbb{N}$ and various other sets like $\mathbb{Z}$, $\mathbb{Q}$, V_ω, $\mathcal{L}_{\omega,\omega}$, etc. Given any finite object a, we use Quine's notation $\ulcorner a \urcorner$ to denote the number coding a. Which method of coding we use is immaterial for us so long as the method is sufficiently effective. We will just assume that these methods exist and hope the reader can figure out how to define them.

Let

$$\Phi_0, \Phi_1, \Phi_2, \Phi_3, \ldots$$

be an enumeration of the computer programs ordered in some effective way, say lexicographically. Given n, we write $\Phi_e(n)$ for the output of the eth program on input n. Each program Φ_e calculates the values of a *partial computable function* $\mathbb{N} \rightharpoonup \mathbb{N}$. Let us remark that, on some inputs, $\Phi_e(n)$ may run forever and never halt with an answer, in which case $\Phi_e(n)$ is undefined. If Φ_e returns an answer for all n, Φ_e is said to be *total* — even if total, these functions are still included within the class of partial computable functions. The *computable functions* are the total functions among the partial computable ones. We write $\Phi_e(n)\downarrow$ to mean that this computation *converges*, that is, that it halts after a finite number of steps; and we write $\Phi_e(n)\uparrow$ to mean that it *diverges*, i.e., it never returns an answer. Computers, as Turing machines, run on a step-by-step basis. We use $\Phi_{e,s}(n)$ to denote the output of $\Phi_e(n)$ after s steps of computation, which can be either not converging yet ($\Phi_{e,s}(n)\uparrow$) or converging to a number ($\Phi_{e,s}(n)\downarrow = m$). Notice that, given e, s, n, we can computably decide whether $\Phi_{e,s}(n)$ converges or not: All we have to do is run $\Phi_e(n)$ for s steps. If f and g are partial functions, we write $f(n) = g(m)$ to mean that either both $f(n)$ and $g(m)$ are undefined, or both are defined and have the same value. We write $f = g$ if $f(n) = g(n)$ for all n. If $f(n) = \Phi_e(n)$ for all n, we say that e is an *index* for f.

In his famous 1936 paper, Turing showed there is a partial computable function $U\colon \mathbb{N}^2 \to \mathbb{N}$ that encodes all other computable functions in the sense that, for every e, n,

$$U(e,n) = \Phi_e(n).$$

This function U is said to be a *universal partial computable function.* It does essentially what computers do nowadays: You give them an index for a program and an input, and they run it for you. We will not use U explicitly throughout the book, but we will constantly use the fact that we can computably list all programs and start running them one at the time, using U implicitly.

We identify subsets of $\mathbb{N}$ with their characteristic functions in $2^{\mathbb{N}}$, and we will move from one viewpoint to the other without even mentioning it. For instance, a set $A \subseteq \mathbb{N}$ is said to be *computable* if its characteristic function is.

An *enumeration* of a set A is nothing more than an onto function $g\colon \mathbb{N} \to A$. A set A is *computably enumerable* (*c.e.*) if it has an enumeration that is computable. The empty set is computably enumerable too. Equivalently, a set is computably enumerable if it is the domain of a partial computable function.[‡] We denote

$$W_e = \{n \in \mathbb{N} : \Phi_e(n)\downarrow\} \quad \text{and} \quad W_{e,s} = \{n \in \mathbb{N} : \Phi_{e,s}(n)\downarrow\}.$$

As a convention, we assume that $W_{e,s}$ is finite, and furthermore, that only on inputs less than s can Φ_e converge in less than s steps. One way to make sense of this is that numbers larger than s should take more than s steps to even be read from the input tape. In general, if a is an object built during a construction and whose value might change along the stages of the construction, we use $a[s]$ to denote its value at stage s. A set is *co-c.e.* if its complement is c.e.

Recall that a set is computable if and only if it and its complement are computably enumerable.

The *recursion theorem* gives us one of the most general ways of using recursion when defining computable functions. It states that for every computable function $f\colon \mathbb{N}^2 \to \mathbb{N}$ there is an index $e \in \mathbb{N}$ such that $f(e,n) = \varphi_e(n)$ for all $n \in \mathbb{N}$.[§] Thus, we can think of $f(e,\cdot) = \varphi_e(\cdot)$ as a function of n which uses its own index, namely e, as a parameter

[‡] If $A = range(g)$, then A is the domain of the partial function that, on input m, outputs the first n with $g(n) = m$ if it exists.

[§] To prove the recursion theorem, for each i, let $g(i)$ be an index for the partial computable function $\varphi_{g(i)}(n) = f(\varphi_i(i), n)$. Let e_0 be an index for the total computable function g, and let $e = g(e_0)$. Then $\varphi_e(n) = \varphi_{g(e_0)} = f(\varphi_{e_0}(e_0), n) = f(g(e_0), n) = f(e,n)$.

during its own computation, and in particular is allowed to call and run itself. An equivalent formulation of this theorem is that, for every computable function $h\colon \mathbb{N} \to \mathbb{N}$, there is an e such that $W_{h(e)} = W_e$.

Sets and strings

The natural numbers are $\mathbb{N} = \{0, 1, 2, ...\}$. For $n \in \mathbb{N}$, we sometimes use n to denote the set $\{0, ..., n-1\}$. For instance, $2^{\mathbb{N}}$ is the set of functions from $\mathbb{N}$ to $\{0, 1\}$, which we will sometimes refer to as *infinite binary sequences* or *infinite binary strings.* For any set X, we use $X^{<\mathbb{N}}$ to denote the set of finite tuples of elements from X, which we call *strings* when $X = 2$ or $X = \mathbb{N}$. For $\sigma \in X^{<\mathbb{N}}$ and $\tau \in X^{\leq\mathbb{N}}$, we use $\sigma^\frown\tau$ to denote the concatenation of these sequences. Similarly, for $x \in X$, $\sigma^\frown x$ is obtained by appending x to σ. We will often omit the $^\frown$ symbol and just write $\sigma\tau$ and σx. We use $\sigma \subseteq \tau$ to denote that σ is an initial segment of τ, that is, that $|\sigma| \leq |\tau|$ and $\sigma(n) = \tau(n)$ for all $n < |\sigma|$. This notation is consistent with the subset notation if we think of a string σ as its graph $\{\langle i, \sigma(i)\rangle : i < |\sigma|\}$. We use $\langle\rangle$ to denote the empty tuple. If Y is a subset of the domain of a function f, we use $f \upharpoonright Y$ for the restriction of f to Y. Given $f \in X^{\leq\mathbb{N}}$ and $n \in \mathbb{N}$, we use $f \upharpoonright n$ to denote the initial segment of f of length n. We use $f \upharpoonright\!\!\upharpoonright n$ for the initial segment of length $n + 1$. For a tuple $\bar{n} = \langle n_0, ..., n_k\rangle \in \mathbb{N}^{<\mathbb{N}}$, we use $f \upharpoonright \bar{n}$ for the tuple $\langle f(n_0), ..., f(n_k)\rangle$. Given a nested sequence of strings $\sigma_0 \subseteq \sigma_1 \subseteq \cdots$, we let $\bigcup_{i\in\mathbb{N}} \sigma_i$ be the possibly infinite string $f \in X^{\leq\mathbb{N}}$ such that $f(n) = m$ if $\sigma_i(n) = m$ for some i.

Given $f, g \in X^{\mathbb{N}}$, we use $f \oplus g$ for the function $(f \oplus g)(2n) = f(n)$ and $(f \oplus g)(2n + 1) = g(n)$. We can extend this to ω*-sums* and define $\bigoplus_{n\in\mathbb{N}} f_n$ to be the function defined by $(\bigoplus_{n\in\mathbb{N}} f_n)(\langle m, k\rangle) = f_m(k)$. Conversely, we define $f^{[n]}$ to be the nth column of f, that is, $f^{[n]}(m) = f(\langle n, m\rangle)$. All these definitions work for sets if we think in terms of their characteristic functions. So, for instance, we can encode countably many sets $\{A_n : n \in \mathbb{N}\}$ with one set $A = \{\langle n, m\rangle : m \in A_n\}$.

For a set $A \subseteq \mathbb{N}$, the complement of A with respect to $\mathbb{N}$ is denoted by A^c.

A *tree* on a set X is a subset T of $X^{<\mathbb{N}}$ that is closed downward, i.e., if $\sigma \in T$ and $\tau \subseteq \sigma$, then $\tau \in T$ too. A *path* through a tree T is a function $f \in X^{\mathbb{N}}$ such that $f \upharpoonright n \in T$ for all $n \in \mathbb{N}$. We use $[T]$ to denote the set of all paths through T. A tree is *well-founded* if it has no paths.

Reducibilities

There are various ways to compare the complexity of sets of natural numbers. Depending on the context or application, some may be more appropriate than others.

Many-one reducibility. Given sets $A, B \subseteq \mathbb{N}$, we say that A is *many-one reducible* (or *m-reducible*) to B, and write $A \leq_m B$, if there is a computable function $f \colon \mathbb{N} \to \mathbb{N}$ such that $n \in A \iff f(n) \in B$ for all $n \in \mathbb{N}$. One should think of this reducibility as saying that all the information in A can be decoded from B. Notice that the classes of computable sets and of c.e. sets are both closed downwards under $\leq_m$-reducibility. A set B is said to be *c.e. complete* if it is c.e. and, for every other c.e. set A, $A \leq_m B$.

Two sets are *m-equivalent* if they are m-reducible to each other, denoted $A \equiv_m B$. This is an equivalence relation, and the equivalence classes are called *m-degrees*

There are, of course, various other ways to formalize the idea of one set encoding the information from another set. Many-one reducibility is somewhat restrictive in various ways: (1) to figure out if $n \in A$, one is allowed to ask only one question of the form "$m \in B$?"; (2) the answer to "$n \in A$?" has to be the same as the answer to "$f(n) \in B$?". Turing reducibility is much more flexible.

One-one reducibility. *1-reducibility* is is like m-reducibility it requires the reduction to be one-to-one. The equivalence induced by it, *1-equivalence*, is one of the strongest notions of equivalence between sets in computability theory — a computability theorist would view sets that are 1-equivalent as being the same. Myhill's theorem states that two sets of natural numbers are 1-equivalent, i.e., each is 1-reducible to the other, if and only if there is a computable bijection of $\mathbb{N}$ that matches one set with the other.

Turing reducibility. Given a function $f \colon \mathbb{N} \to \mathbb{N}$, we say that a partial function $g \colon \mathbb{N} \rightharpoonup \mathbb{N}$ is *partial f-computable* if it can be computed by a program that is allowed to use the function f as a primitive function during its computation; that is, the program can ask questions about the value of $f(n)$ for different n's and use the answers to make decisions while the program is running. The function f is called the *oracle* of this computation. If g and f are total, we write $g \leq_T f$ and say that g is *Turing reducible* to f, that f *computes* g, or that g is *f-computable.* The class of partial f-computable functions can be enumerated the same way as the class of the partial computable functions. Programs that are allowed to query an oracle are called

Turing operators or *computable operators.* We list them as Φ_0, Φ_1,... and we write $\Phi_e^f(n)$ for the output of the eth Turing operator on input n when it uses f as oracle. Notice that Φ_e represents a fixed program that can be used with different oracles. When the oracle is the empty set, we may write Φ_e for $\Phi_e^{\emptyset}$ matching the previous notation.

As we already mentioned, for a fixed input n, if $\Phi_e^f(n)$ converges, it does so after a finite number of steps s. As a convention, let us assume that in just s steps, it is only possible to read the first s entries from the oracle. Thus, if σ is a finite substring of f of length greater than s, we could calculate $\Phi_{e,s}^{\sigma}(n)$ without ever noticing that the oracle is not an infinite string.

> Convention: For $\sigma \in \mathbb{N}^{<\mathbb{N}}$, $\Phi_e^{\sigma}(n)$ is shorthand for $\Phi_{e,|\sigma|}^{\sigma}(n)$, which runs for at most $|\sigma|$ stages.

Notice that given e, σ, n, it is computable to decide if $\Phi_e^{\sigma}(n)\downarrow$.

As the class of partial computable functions, the class of partial X-computable functions contains the basic functions; is closed under composition, recursion, and minimization; and can be listed in such a way that we have a universal partial X-computable function (that satisfies the s-m-n theorem). In practice, with very few exceptions, those are the only properties we use of computable functions. This is why almost everything we can prove about computable functions, we can also prove about X-computable functions. This translation is called *relativization.* All notions whose definition are based on the notion of partial computable function can be relativized by using the notion of partial X-computable function instead. For instance, the notion of c.e. set can be relativized to that of c.e. in X or X-c.e. set: These are the sets which are the images of X-computable functions (or empty), or, equivalently, the domains of partial X-computable functions. We use W_e^X to denote the domain of Φ_e^X.

When two functions are Turing reducible to each other, we say that they are *Turing equivalent*, which we denote by $\equiv_T$. This is an equivalence relation, and the equivalence classes are called *Turing degrees.*

Computable operators can be encoded by computable subsets of $\mathbb{N}^{<\mathbb{N}} \times \mathbb{N} \times \mathbb{N}$. Given $\Phi \subseteq \mathbb{N}^{<\mathbb{N}} \times \mathbb{N} \times \mathbb{N}$, $\sigma \in \mathbb{N}^{<\mathbb{N}}$, n, m, we write $\Phi^{\sigma}(n) = m$ as shorthand for $\langle \sigma, n, m \rangle \in \Phi$. Then, given $f \in \mathbb{N}^{\mathbb{N}}$, we let

$$\Phi^f(n) = m \iff (\exists \sigma \subset f)\ \Phi^{\sigma}(n) = m.$$

We then have that g is computable in f if and only if there is a c.e. subset $\Phi \subseteq \mathbb{N}^{<\mathbb{N}} \times \mathbb{N} \times \mathbb{N}$ such that $\Phi^f(n) = g(n)$ for all $n \in \mathbb{N}$. A standard assumption is that $\langle \sigma, n, m \rangle \in \Phi$ only if $n, m < |\sigma|$.

We can use the same idea to encode c.e. operators by computable subsets of $\mathbb{N}^{<\mathbb{N}} \times \mathbb{N}$. Given $W \subseteq \mathbb{N}^{<\mathbb{N}} \times \mathbb{N}$, $\sigma \in \mathbb{N}^{<\mathbb{N}}$, and $f \in \mathbb{N}^{\mathbb{N}}$, we let

$$W^\sigma = \{n \in \mathbb{N} : \langle \sigma, n\rangle \in W\} \quad \text{and} \quad W^f = \bigcup_{\sigma \subset f} W^\sigma.$$

We then have that X is c.e. in Y if and only if there is a c.e. subset $W \subseteq \mathbb{N}^{<\mathbb{N}} \times \mathbb{N}$ such that $X = W^Y$. A standard assumption is that $\langle \sigma, n\rangle \in W$ only if $n < |\sigma|$.

The Turing jump. Let K be the domain of the universal partial computable function. That is,

$$K = \{\langle e, n\rangle : \Phi_e(n)\downarrow\} = \bigoplus_{e \in \mathbb{N}} W_e.$$

K is called the *halting problem.*[¶] It is not hard to see that K is c.e. complete. Using a standard diagonalization argument, one can show that K is not computable.[‖] It is common to define K as $\{e : \Phi_e(e)\downarrow\}$ instead — the two definitions give 1-equivalent sets. We will use whichever is more convenient in each situation. We will often write $0'$ for K.

We can relativize this definition and, given a set X, define the *Turing jump* of X as

$$X' = \{e \in \mathbb{N} : \Phi_e^X(e)\downarrow\}.$$

Relativizing the properties of K, we get that X' is X-c.e.-complete, that $X \leq_T X'$, and that $X' \not\leq_T X$. The Turing degree of X' is strictly above that of X — this is why it is called a jump. The jump defines an operation on the Turing degrees. Furthermore, for $X, Y \subseteq \mathbb{N}$, $X \leq_T Y \iff X' \leq_m Y'$.

The double iteration of the Turing jump is denoted X'' and the n-th iteration by $X^{(n)}$.

Vocabularies and languages

Let us quickly review the basics about vocabularies and structures. Our vocabularies will always be countable.

A *vocabulary* τ consists of three sets of symbols $\{\mathsf{R}_i : i \in I_R\}$, $\{\mathsf{f}_i : i \in I_F\}$, and $\{\mathsf{c}_i : i \in I_C\}$; and two functions $a_R \colon I_R \to \mathbb{N}$ and $a_F \colon I_F \to \mathbb{N}$. Each of I_R, I_F, and I_C is an initial segment of $\mathbb{N}$. The symbols R_i, f_i, and c_i represent *relations*, *functions,* and *constants*,

[¶] The 'K' is for Kleene.

[‖] If it were computable, so would be the set $A = \{e : \langle e, e\rangle \notin K\}$. But then $A = W_e$ for some e, and we would have that $e \in A \iff \langle e, e\rangle \notin K \iff e \notin W_e \iff e \notin A$.

respectively. For $i \in I_R$, $a_R(i)$ is the arity of R_i, and for $i \in I_F$, $a_F(i)$ is the arity of f_i.

A vocabulary τ is *computable* if the arity functions a_R and a_F are computable. This only matters when τ is infinite; finite vocabularies are trivially computable. Except for a few occasions, the vocabularies we use will always be computable.

Given such a vocabulary τ, a *τ-structure* is a tuple

$$\mathcal{M} = (M; \{R_i^{\mathcal{M}} : i \in I_R\}, \{f_i^{\mathcal{M}} : i \in I_F\}, \{c_i^{\mathcal{M}} : i \in I_C\}),$$

where M is just a set called the *domain* of $\mathcal{M}$, and the rest are interpretations of the symbols in τ. That is, $R_i^{\mathcal{M}} \subset M^{a_R(i)}$, $f_i^{\mathcal{M}} \colon M^{a_F(i)} \to M$, and $c_i^{\mathcal{M}} \in M$. A *structure* is a τ-structure for some τ.

Given two τ-structures $\mathcal{A}$ and $\mathcal{B}$, we write $\mathcal{A} \subseteq \mathcal{B}$ to mean that $\mathcal{A}$ is a substructure of $\mathcal{B}$, that is, that $A \subseteq B$, $f_i^{\mathcal{A}} = f_i^{\mathcal{B}} \restriction A^{a_F(i)}$, $R_j^{\mathcal{A}} = R_j^{\mathcal{B}} \restriction A^{a_R(i)}$ and $c_k^{\mathcal{A}} = c_k^{B}$ for all symbols f_i, R_j and c_k. This notation should not be confused with $A \subseteq B$ which only means that the domain of $\mathcal{A}$ is a subset of the domain of $\mathcal{B}$. If $\mathcal{A}$ is a τ_0-structure and $\mathcal{B}$ a τ_1-structure with $\tau_0 \subseteq \tau_1$,[**] $\mathcal{A} \subseteq \mathcal{B}$ means that $\mathcal{A}$ is a τ_0-substructure of $\mathcal{B} \restriction \tau_0$, where $\mathcal{B} \restriction \tau_0$ is obtained by forgetting the interpretations of the symbols of $\tau_1 \smallsetminus \tau_0$ in $\mathcal{B}$. $\mathcal{B} \restriction \tau_0$ is called the τ_0-*reduct* of $\mathcal{B}$, and $\mathcal{B}$ is said to be an *expansion* of $\mathcal{B} \restriction \tau_0$.

Given a vocabulary τ, we define various languages over it. First, recursively define a *τ-term* to be either a variable x, a constant symbol c_i, or a function symbol applied to other τ-terms, that is, $\mathsf{f}_i(t_1, ..., t_{a_F(i)})$, where each t_j is a τ-term we have already built. The *atomic τ-formulas* are the ones of the form $\mathsf{R}_i(t_1, ..., t_{a_R(i)})$ or $t_1 = t_2$, where each t_i is a τ-term. A *τ-literal* is either a τ-atomic formula or a negation of a τ-atomic formula. A *quantifier-free τ-formula* is built out of literals using conjunctions and disjunctions. If we also use existential quantification, we get the *existential τ-formulas*, or *$\exists$-formulas*. Every τ-existential formula is equivalent to one of the form $\exists x_1 \cdots \exists x_k\ \varphi$, where φ is quantifier-free. A *universal τ-formula*, or *$\forall$-formula*, is one equivalent to $\forall x_1 \cdots \forall x_k\ \varphi$ for some quantifier-free τ-formula φ. An *elementary τ-formula* is built out of quantifier-free formulas using existential and universal quantifiers. We also call these the *finitary first-order formulas.*

Given a τ-structure $\mathcal{A}$, and a tuple $\bar{a} \in A^{<\mathbb{N}}$, we write $(\mathcal{A}, \bar{a})$ for the $\tau \cup \bar{\mathsf{c}}$-structure where $\bar{\mathsf{c}}$ is a new tuple of constant symbols and $\bar{\mathsf{c}}^{\mathcal{A}} = \bar{a}$. Given $R \subseteq \mathbb{N} \times A^{<\mathbb{N}}$, we write $(\mathcal{A}, R)$ for the $\widetilde{\tau}$-structure where $\widetilde{\tau}$ is

[**] By $\tau_0 \subseteq \tau_1$ we mean that every symbol in τ_0 is also in τ_1 and has the same arity.

defined by adding to τ relations symbols $\mathsf{R}_{i,j}$ of arity j for $i, j \in \mathbb{N}$, and $\mathsf{R}_{i,j}^{\mathcal{A}} = \{\bar{a} \in A^j : \langle i, \bar{a}\rangle \in R\}$.

Orderings

Here are some structures we will use quite often in examples. A *partial order* is a structure over the vocabulary $\{\leq\}$ with one binary relation symbol which is transitive ($x \leq y$ & $y \leq z \to x \leq z$), reflexive ($x \leq x$), and anti-symmetric ($x \leq y$ & $y \leq x \to x = y$). A *linear order* is a partial order where every two elements are comparable ($\forall x, y\ (x \leq y \vee y \leq x)$). We will often add and multiply linear orderings. Given linear orderings $\mathcal{A} = (A; \leq_A)$ and $\mathcal{B} = (B; \leq_B)$, we define $\mathcal{A} + \mathcal{B}$ to be the linear ordering with domain $A \sqcup B$, where the elements of A stand below the elements of B. We define $\mathcal{A} \times \mathcal{B}$ to be the linear ordering with domain $A \times B$ where $\langle a_1, b_1\rangle \leq_{A\times B} \langle a_2, b_2\rangle$ if either $b_1 <_B b_2$ or $b_1 = b_2$ and $a_1 \leq_A a_2$ — notice we compare the second coordinate first.[††] We will use ω to denote the linear ordering of the natural numbers and $\mathbb{Z}$ and $\mathbb{Q}$ for the orderings of the integers and the rationals. We denote the finite linear ordering with n elements by $\mathbf{n}$. We use $\mathcal{A}^*$ to denote the reverse ordering $(A; \geq_A)$ of $\mathcal{A} = (A, \leq_A)$. For $a <_{\mathcal{A}} b \in \mathcal{A}$, we use the notation $\mathcal{A} \upharpoonright (a, b)$ or the notation $(a, b)_{\mathcal{A}}$ to denote the open interval $\{x \in A : a <_{\mathcal{A}} x <_A b\}$. We also use $\mathcal{A} \upharpoonright a$ to denote the initial segment of $\mathcal{A}$ below a, which we could also denote as $(-\infty, a)_{\mathcal{A}}$.

As mentioned above, a *tree* T is a downward closed subset of $X^{<\mathbb{N}}$. As a structure, a tree can be represented in various ways. One is as a partial order $(T; \subseteq)$ using the ordering on strings. Another is as a graph where each node $\sigma \in T$ other than the root is connected to its parent node $\sigma \upharpoonright |\sigma - 1|$, and there is a constant symbol used for the root of the tree. We will refer to these two types of structures as *trees as orders* and *trees as graphs.*

A partial order where every two elements have a least upper bound $(x \vee y)$ and a greatest lower bound $(x \wedge y)$ is called a *lattice.* A lattice with a top element 1, a bottom element 0, where $\vee$ and $\wedge$ distribute over each other, and every element x has a *complement* (that is an element x^c such that $x \vee x^c = 1$ and $x \wedge x^c = 0$) is called a *Boolean algebra.* The vocabulary for Boolean algebras is $\{0, 1, \vee, \wedge, \cdot^c\}$, and the ordering can be defined by $x \leq y \iff y = x \vee y$.

The arithmetic hierarchy

Consider the structure $(\mathbb{N}; 0, 1, +, \times, \leq)$. In this vocabulary, the *bounded formulas* are built out of the quantifier-free formulas using

[††] $\mathcal{A}$ times $\mathcal{B}$ is $\mathcal{A}$ $\mathcal{B}$ times.

bounded quantifiers of the form $\forall x < y$ and $\exists x < y$. A Σ^0_1 formula is one of the form $\exists x\ \varphi$, where φ is bounded. A Π^0_1 formula is one of the form $\forall x\ \varphi$, where φ is bounded. By coding tuples of numbers by a single natural number, one can show that formulas of the form $\exists x_0 \exists x_1 \cdots ... \exists x_k\ \varphi$ are equivalent to Σ^0_1 formulas. Post's theorem asserts that a set $A \subseteq \mathbb{N}$ is c.e. if and only if it can be defined by a Σ^0_1 formula. Thus, a set is computable if and only if it is Δ^0_1, that is, if it can be defined both by a Σ^0_1 formula and by a Π^0_1 formula.

By recursion, we define the Σ^0_{n+1} formulas as those of the form $\exists x\ \varphi$, where φ is Π^0_n; and the Π^0_{n+1} formulas as those of the form $\forall x\ \varphi$, where φ is Σ^0_n. A set is Δ^0_n if it can be defined by both a Σ^0_n formula and a Π^0_n formula. Again, in the definition of Σ^0_{n+1} formulas, using one existential quantifier or many makes no difference. What matters is the number of alternations of quantifiers. Post's theorem asserts that a set $A \subseteq \mathbb{N}$ is c.e. in $0^{(n)}$ if and only if it can be defined by a Σ^0_{n+1} formula. In particular, a set is computable from $0'$ if and only if it is Δ^0_2. The Shoenfield *Limit Lemma* says that a set A is Δ^0_2 if and only if there is a computable function $f\colon \mathbb{N}^2 \to \mathbb{N}$ such that, for each $n \in \mathbb{N}$, if $n \in A$ then $f(n,s) = 1$ for all sufficiently large s, and if $n \notin A$ then $f(n,s) = 0$ for all sufficiently large s. This can be written as $\chi_A(n) = \lim_{s\to\infty} f(n,s)$, where χ_A is the characteristic function of A and the limit is with respect to the discrete topology of $\mathbb{N}$ where a sequence converges if and only if it is eventually constant.

The language of second-order arithmetic is a two-sorted language for the structure $(\mathbb{N}, \mathbb{N}^{\mathbb{N}}; 0, 1, +, \times, \leq)$. The elements of the first sort, called *first-order elements*, are natural numbers. The elements of the second sort, called *second-order elements* or *reals*, are functions $\mathbb{N} \to \mathbb{N}$. The vocabulary consists of the standard vocabulary of arithmetic, 0, 1, $+$, $\times$, $\leq$, which is used on the first-order elements, and an application operation denoted $F(n)$ for a second-order element F and a first-order element n. A formula in this language is said to be *arithmetic* if it has no quantifiers over second-order objects. Among the arithmetic formulas, the hierarchy of Σ^0_n and Π^0_n formulas are defined exactly as above. Post's theorem that Σ^0_1 sets are c.e. also applies in this context: For every Σ^0_1 formula $\psi(F,n)$, where n a number variable and F is a function variable, there is c.e. operator W such that $n \in W^F \iff \psi(F,n)$. We can then build the computable tree $T_n = \{\sigma \in \mathbb{N}^{<\mathbb{N}} : n \notin W^\sigma\}$, and we have that $\psi(F,n)$ holds if and only if F is not a path through T_n. A Π^0_1 *class* is a set of the form $\{F \in \mathbb{N}^{\mathbb{N}} : \psi(F)\}$ for some Π^0_1 formula $\psi(F)$. The observation above shows how every Π^0_1 class is of the form $[T]$ for some computable tree $T \subseteq \mathbb{N}^{<\mathbb{N}}$.

Notation and Conventions from Part I

Knowledge of [**Part 1**] is not necessary to read this book. However, there are some basic notations and concepts developed at the beginning of [**Part 1**] that we will review here. Most of these concepts are carefully developed in [**Part 1**, Chapter I]. Here we review them rather quickly.

Presentations

All the structures we consider are countable. So, unless otherwise stated, "structure" means "countable structure." Furthermore, we usually assume that the domains of our structures are subsets of $\mathbb{N}$. This will allow us to have everything we already know about computable functions on $\mathbb{N}$ at our disposal.

Definition .1. An *ω-presentation* is nothing more than a structure whose domain is $\mathbb{N}$. Given a τ-structure $\mathcal{A}$, when we refer to *an ω-presentation of $\mathcal{A}$* or to a *copy of $\mathcal{A}$*, we mean an ω-presentation $\mathcal{M}$ which is isomorphic to $\mathcal{A}$. An ω-presentation $\mathcal{M}$ is *computable* if all its relations, functions, and constants are uniformly computable; that is, if the set $\tau^{\mathcal{M}}$, defined as

$$\tau^{\mathcal{M}} = \bigoplus_{i \in I_R} R_i^{\mathcal{M}} \oplus \bigoplus_{i \in I_F} F_i^{\mathcal{M}} \oplus \bigoplus_{i \in I_C} \{c_i^{\mathcal{M}}\}, \tag{1}$$

is computable.

Atomic diagrams

Another standard way of defining when an ω-presentation is computable is via its atomic diagram. Let $\{\varphi_i^{\mathrm{at}} : i \in \mathbb{N}\}$ be an effective enumeration of all atomic τ-formulas with free variables from the set $\{x_0, x_1, ...\}$. (An *atomic τ-formula* is one of the form $\mathsf{R}(\mathsf{t}_1, ..., \mathsf{t}_a)$, where R is either "=" or R_j for $j \in I_R$, and each t_i is a term built out of the function, constant, and variable symbols.)

Definition .2. The *atomic diagram* of an ω-presentation $\mathcal{M}$ is the infinite binary string $D(\mathcal{M}) \in 2^{\mathbb{N}}$ defined by

$$D(\mathcal{M})(i) = \begin{cases} 1 & \text{if } \mathcal{M} \models \varphi_i^{\text{at}}[x_j \mapsto j : j \in \mathbb{N}] \\ 0 & \text{otherwise.} \end{cases}$$

It is not hard to see that $D(\mathcal{M})$ and $\tau^{\mathcal{M}}$ are Turing equivalent. We will often treat the ω-presentation $\mathcal{M}$, the real $\tau^{\mathcal{M}}$, and the real $D(\mathcal{M})$ as the same thing. For instance, we define the *Turing degree of the ω-presentation $\mathcal{M}$* to be the Turing degree of $D(\mathcal{M})$. When we say that $\mathcal{M}$ *is computable from a set X*, that *a set X is computable from $\mathcal{M}$*, that $\mathcal{M}$ is Δ_2^0, that $\mathcal{M}$ is *arithmetic*, that $\mathcal{M}$ is *low*, etc., we always mean $D(\mathcal{M})$ instead of $\mathcal{M}$.

Relaxing the domain

In many cases, it will be useful to consider structures whose domain is a subset of $\mathbb{N}$. We call those $(\subseteq\omega)$*-presentations*. If M, the domain of $\mathcal{M}$, is a proper subset of $\mathbb{N}$, we can still define $D(\mathcal{M})$ by letting $D(\mathcal{M})(i) = 0$ if φ_i^{at} mentions a variable x_j with $j \notin M$. In this case, we have

$$D(\mathcal{M}) \equiv_T M \oplus \tau^{\mathcal{M}}.$$

To see that $D(\mathcal{M})$ computes M, notice that, for $j \in \mathbb{N}$, $j \in M$ if and only if $D(\mathcal{M})(\ulcorner x_j = x_j \urcorner) = 1$, where $\ulcorner \varphi \urcorner$ is the index of the atomic formula φ in the enumeration $\{\varphi_i^{\text{at}} : i \in \mathbb{N}\}$.

The following observation will simplify many of our constructions later on.

Observation .3. We can always associate to an infinite $(\subseteq\omega)$-presentation $\mathcal{M}$ an isomorphic ω-presentation $\mathcal{A}$: If $M = \{m_0 < m_1 < m_2 < \cdots\} \subseteq \mathbb{N}$, we can use the bijection $i \mapsto m_i \colon \mathbb{N} \to M$ to get a copy $\mathcal{A}$ of $\mathcal{M}$, now with domain $\mathbb{N}$. Since this bijection is computable in M, it is not hard to see that $D(\mathcal{A}) \leq_T D(\mathcal{M})$, and furthermore that $D(\mathcal{A}) \oplus M \equiv_T D(\mathcal{M})$.

One of the advantages of $(\subseteq\omega)$-presentations is that they allow us to present finite structures.

Relational vocabularies

A vocabulary is *relational* if it has no function or constant symbols and has only relational symbols. Every vocabulary τ can be made into a relational one, $\widetilde{\tau}$, by replacing each n-ary function symbol by an $(n+1)$-ary relation symbol coding the graph of the function and each constant symbol by a 1-ary relation symbol coding it as a singleton.

Depending on the situation, this change in vocabulary might be more or less significant. For instance, the class of quantifier-free definable sets changes, but the class of $\exists$-definable sets does not (see Exercise [**Part 1**, Exercise I.4]). For most computational properties, this change is nonessential; for instance, if $\mathcal{M}$ is an ω-presentation of a τ-structure, and $\widetilde{\mathcal{M}}$ is the associated ω-presentation of $\mathcal{M}$ as a $\widetilde{\tau}$-structure, then $D(\mathcal{M}) \equiv_T D(\widetilde{\mathcal{M}})$ (as it follows from [**Part 1**, Exercise I.4]). Because of this, and for the sake of simplicity, we will often restrict ourselves to relational vocabularies.

Diagrams of tuples

When dealing with an infinite vocabulary, we sometimes need to approximate it using finite sub-vocabularies. We assume that all computable vocabularies τ come with an associated effective approximation $\tau_0 \subseteq \tau_1 \subseteq \cdots \subseteq \tau$, where each τ_s is finite, and $\tau = \bigcup_s \tau_s$. In general and unless otherwise stated, we let τ_s consist of the first s relation, constant and function symbols in τ, but in some particular cases, we might prefer other approximations. For instance, if τ is already finite, we usually prefer to let $\tau_s = \tau$ for all s.

As a convention, when we enumerate the atomic formulas as $\{\varphi_i^{\text{at}} : i \in \mathbb{N}\}$ we do it an way that for each s, the τ_s-atomic formulas on the variables $\{x_0, ..., x_{s-1}\}$ are listed before the rest; that is, they are $\varphi_0^{\text{at}}, ..., \varphi_{\ell_s - 1}^{\text{at}}$ for some $\ell_s \in \mathbb{N}$.

As a useful technical device, we define the atomic diagram of a finite tuple as the finite binary sequence coding the set of atomic formulas true of the tuple restricted to the smaller vocabulary. We assume that τ is relational.

Definition .4. Let $\mathcal{M}$ be a τ-structure and let $\bar{a} = \langle a_0, ..., a_{s-1}\rangle \in M^s$. We define the *atomic diagram of $\bar{a}$ in $\mathcal{M}$, denoted $D_{\mathcal{M}}(\bar{a})$,* as the string in 2^{ℓ_s} such that

$$D_{\mathcal{M}}(\bar{a})(i) = \begin{cases} 1 & \text{if } \mathcal{M} \models \varphi_i^{\text{at}}[x_j \mapsto a_j, j < s], \\ 0 & \text{otherwise.} \end{cases}$$

Observation .5. For every $\sigma \in 2^{<\mathbb{N}}$ and every s with $\ell_s \geq |\sigma|$, there is a quantifier-free τ-formula $\varphi_\sigma^{\text{at}}(x_0, ..., x_{s-1})$ such that

$$\mathcal{A} \models \varphi_\sigma^{\text{at}}(\bar{a}) \iff \sigma \subseteq D_{\mathcal{A}}(\bar{a})$$

for every τ-structure $\mathcal{A}$ and tuple $\bar{a} \in A^s$, namely

$$\varphi^{\mathrm{at}}_\sigma(\bar{x}) \quad \equiv \quad \left(\bigwedge_{i<|\sigma|, \sigma(i)=1} \varphi^{\mathrm{at}}_i(\bar{x}) \right) \wedge \left(\bigwedge_{i<|\sigma|, \sigma(i)=0} \neg\varphi^{\mathrm{at}}_i(\bar{x}) \right).$$

Congruence structures

It will often be useful to consider structures where equality is interpreted by an equivalence relation. A *congruence τ-structure* is a structure $\mathcal{M} = (M; =^{\mathcal{M}}, \{R_i^{\mathcal{M}} : i \in I_R\}, \{f_i^{\mathcal{M}} : i \in I_F\}, \{c_i^{\mathcal{M}} : i \in I_C\})$, where $=^{\mathcal{M}}$ is an equivalence relation on M, and the interpretations of all the τ-symbols are invariant under $=^{\mathcal{M}}$ (that is, if $\bar{a} =^{\mathcal{M}} \bar{b}$, then $\bar{a} \in R_i^{\mathcal{M}} \iff \bar{b} \in R_i^{\mathcal{M}}$ and $f_j^M(\bar{a}) =^{\mathcal{M}} f_j(\bar{b})$ for all relations symbols R_i and function symbols f_j). If $M = \mathbb{N}$, we say that $\mathcal{M}$ is a *congruence ω-presentation*. We can then define $D(\mathcal{M})$ exactly as in Definition .2, using $=^{\mathcal{M}}$ to interpret equality.

Given a congruence τ-structure, one can always take the quotient $\mathcal{M}/{=^{\mathcal{M}}}$ and get a τ-structure where equality is the standard $\mathbb{N}$-equality. To highlight the difference, we will sometimes use the term *injective ω-presentations* when equality is $\mathbb{N}$-equality.

LEMMA .6. *Given a congruence ω-presentation $\mathcal{M}$ with infinitely many equivalence classes, the quotient $\mathcal{M}/{=^{\mathcal{M}}}$ has an injective ω-presentation $\mathcal{A}$ computable from $D(\mathcal{M})$. Furthermore, the natural projection $\mathcal{M} \to \mathcal{A}$ is also computable from $D(\mathcal{M})$.*

See [**Part 1**, Lemma I.11] for a proof.

It follows that, from a computational viewpoint, there is no real difference in considering congruence structures or injective structures.

Enumerations

Assume τ is a relational vocabulary. An *enumeration of a structure* $\mathcal{M}$ is just an onto map $g\colon \mathbb{N} \to M$. To each such enumeration we can associate a congruence ω-presentation $g^{-1}(\mathcal{M})$ by taking the *pull-back* of $\mathcal{M}$ through g:

$$g^{-1}(\mathcal{M}) = (\mathbb{N}; \sim, \{R_i^{g^{-1}(\mathcal{M})} : i \in I_R\}),$$

where $a \sim b \iff g(a) = g(b)$ and $R_i^{g^{-1}(\mathcal{M})} = g^{-1}(R_i^{\mathcal{M}}) \subseteq \mathbb{N}^{a(i)}$. The assumption that τ is relational was used here so that the pull-backs of functions and constants are not multi-valued. Let us remark that if g is injective, then $\sim$ becomes $=_{\mathbb{N}}$, and hence $g^{-1}(\mathcal{M})$ is an injective ω-presentation. In this case, the assumption that τ is relational is not important, as we can always pull-back functions and constants through bijections.

It is not hard to see that

$$D(g^{-1}(\mathcal{M})) \leq_T g \oplus D(\mathcal{M}).$$

Furthermore, $D(g^{-1}(\mathcal{M})) \leq_T g \oplus \tau^{\mathcal{M}}$, where $\tau^{\mathcal{M}}$ is as in Definition .1.

Throughout the book, there will be many constructions where we need to build a copy of a given structure with certain properties. In most cases, we will do it by building an enumeration of the structure and then taking the pull-back. The following observation will allow us to approximate the atomic diagram of the pull-back, and we will use it countless times.

OBSERVATION .7. Let g be an enumeration of $\mathcal{M}$. Notice that for every tuple $\bar{a} \in M^{<\mathbb{N}}$,

$$D_{g^{-1}(\mathcal{M})}(\bar{a}) = D_{\mathcal{M}}(g(\bar{a})).$$

Suppose that g was defined as $\bigcup_{s\in\mathbb{N}} \sigma_s$ where $\sigma_0 \subseteq \sigma_1 \subseteq \sigma_2 \subset \cdots$ are tuples of elements of M. Then $D_{g^{-1}(\mathcal{M})}(\langle 0, ..., |\sigma_k| - 1\rangle) = D_{\mathcal{M}}(\sigma_k)$, and the diagram of the pull-back can be calculated in terms of the diagrams of tuples in $\mathcal{M}$ as follows:

$$D(g^{-1}(\mathcal{M})) = \bigcup_{k\in\mathbb{N}} D_{\mathcal{M}}(\sigma_k).$$

CHAPTER I

Ordinals

The ordinal numbers were introduced by Cantor in 1883 with the intention of extending the iteration of his derivative process beyond just the finite steps. They turned out to have a beautiful structure that we describe in this chapter. Ordinal numbers extend the natural numbers into the transfinite and allow us to define complexity classes beyond the arithmetic. A set is said to be *arithm**e**tic* if it can be defined within ar**i**thmetic, that is, within the structure $(\mathbb{N}; 0, 1, +, \times, \leq)$. The first step to go beyond *the* arithm**e**tic is to extend ar**i**thmetic.

The first couple sections describe the elementary properties of ordinals and well-founded partial orderings. Even if this is basic background for most readers, it is so important for the rest of the textbook that we had to include it. We recommend the reader to skim through the statements as there might be some interesting lemma here or there. We then turn into complexity issues in Section I.3 and define computable ordinals in Section I.4.

I.1. Well-orderings

We start with a very quick introduction to ordinals and their properties. The first half of this section can be found in most basic logic textbooks. The second half, which is about ordinal exponentiation, not as much.

Definition I.1. We say that a linear ordering is *well-ordered* if it has no infinite descending sequences.

Equivalently, a linear ordering is well-ordered if every subset has a least element: If a subset has no least element, one can easily define an infinite descending sequence inside the set, and if we are given an infinite descending sequence, its elements form a set which has no least element.

In this book, we use the word *ordinal* to refer to the isomorphism type of a well-ordering.[*]

[*] By *isomorphism type* we mean an equivalence class under the equivalence relation given by isomorphism. In the case of linear orderings, isomorphism types are often called *order types*.

All finite orderings are well-ordered. We use the number $\mathbf{n}$ to represent the linear ordering of size n. The first infinite ordinal is ω, which corresponds to the order on the natural numbers $(\mathbb{N}; \leq)$. Next come

$$\omega + 1,\ \omega + 2, ..., \omega + \omega,\ \omega 2 + 1, ..., \omega 3, ..., \omega \cdot \omega, ..., \omega^3, ..., \omega^\omega, ..., \omega^{\omega^\omega}, ...$$

Exercise I.2. Consider $\mathbb{N}[x]$, the set of polynomials with coefficients in $\mathbb{N}$. Order $\mathbb{N}[x]$ as follows: $p \leq q$ if $\lim_{x\to\infty} q(x) - p(x) \geq 0$. Prove that $(\mathbb{N}[x]; \leq)$ is a well-ordering.

Let $\mathbb{LO}$ denote the class of $(\subseteq\omega)$-presentations of linear orderings.[†] Let $\mathbb{WO}$ denote the class of $(\subseteq\omega)$-presentations of well-orderings. One way to represent the set of countable ordinals is as the quotient $\mathbb{WO}/\cong$. We often abuse notation and refer to an ordinal when we actually mean a particular $(\subseteq\omega)$-presentation of that ordinal instead of an equivalence class of $(\subseteq\omega)$-presentations.

We start by proving the three main properties of well-orderings: transfinite induction, transfinite recursion, and comparability. We need the following notation: Given a partial ordering $\mathcal{P} = (P; \leq_P)$ and $a \in P$, we use $\mathcal{P}_{<a}$ to denote the sub-ordering of $\mathcal{P}$ with domain $P_{<a} = \{x \in P : x <_P a\}$.

Theorem I.3 (Transfinite induction). *Let $\mathcal{W} = (W; \leq_W)$ be a well-ordering and I a subset of W that satisfies that, for every $a \in W$, if $W_{<a} \subseteq I$, then $a \in I$. Then $I = W$.*

Proof. If $I \neq W$, the set $W \smallsetminus I$ has a minimal element. Call it a. It satisfies that $W_{<a} \subseteq I$ while $a \notin I$, contradicting the hypothesis. □

Theorem I.4 (Transfinite recursion). *Let $\mathcal{W} = (W; \leq_W)$ be a well-ordering, X be any set, and Ψ be an operator that, given $a \in W$ and a function $W_{<a} \to X$, outputs an element of X. Then there is a unique total function $g\colon W \to X$ such that*

$$g(a) = \Psi(a, g \restriction W_{<a}) \qquad \text{for every } a \in W.$$

Proof. Let $\mathcal{C}$ be the class of all functions g whose domain is a downward-closed subset of W and which satisfy

$$g(a) = \Psi(a, g \restriction W_{<a}) \quad \text{for every } a \in \text{dom}(g). \tag{2}$$

First, we claim that if $f, g \in \mathcal{C}$, then f and g coincide on their common domain: If not, let $a \in \text{dom}(f) \cap \text{dom}(g)$ be a minimal element such that $f(a) \neq g(a)$. By the minimality of a, $f \restriction W_{<a} = g \restriction W_{<a}$, and

[†] Recall that an $(\subseteq\omega)$-presentations is a structure whose domain is a subset of ω. We use $(\subseteq\omega)$-presentations instead of plain old ω-presentations because we want to allow for finite linear orderings.

hence $f(a) = \Psi(a, f \upharpoonright W_{<a}) = \Psi(a, g \upharpoonright W_{<a}) = g(a)$, contradicting our choice of a.

Now, since all the functions in $\mathcal{C}$ are compatible, their union $g = \bigcup \mathcal{C}$ is also a function, given by $g(a) = b$ if there is some $f \in \mathcal{C}$ with $f(a) = b$. It is easy to see that g is itself a member of $\mathcal{C}$.

Last, we claim that the domain of g is the whole of W. If not, let a be a minimal element outside the domain of g. Define a new function $f\colon \operatorname{dom}(g) \cup \{a\} \to X$ by copying g on $\operatorname{dom}(g)$, and letting $f(a) = \Psi(a, g)$. This new function clearly belongs to $\mathcal{C}$ but it has larger domain than g, contradicting the maximality of g in $\mathcal{C}$. □

Observation I.5. There is no one-to-one order-preserving function from an ordinal to a proper initial segment of itself: To see this, suppose towards a contradiction that f is a one-to-one order-preserving function from an ordinal α to $\alpha_{<a}$ for some $a \in \alpha$. We claim that then, the sequence $a, f(a), f(f(a)), \ldots$ would be an infinite descending sequence in α, which would contradict the well-orderness of α. To see this, we first note that $a > f(a)$ just because $f(a) \in \alpha_{<a}$. Using that f preserves order, we then get that $f(a) > f(f(a))$ and then by induction that $f^n(a) > f^{n+1}(a)$.

Theorem I.6. *Given two well-orderings α and β, we have one of the following three exclusive possibilities:*

- *α and β are isomorphic.*
- *α is isomorphic to $\beta_{<b}$ for some $b \in \beta$.*
- *β is isomorphic to $\alpha_{<a}$ for some $a \in \alpha$.*

Proof. To see that the possibilities are mutually exclusive, notice that if two of them were true, we could compose the isomorphisms and get either that α is isomorphic to a proper initial segment of itself, or that β is isomorphic to a proper initial segment of itself. Either way we find a contradiction with the previous observation.

To prove that one of these isomorphisms exists, we start by defining a partial function $g\colon \alpha \rightharpoonup \beta$ as follows: Given $a \in \alpha$, let $g(a)$ be the $b \in \beta$ such that $\alpha_{<a} \cong \beta_{<b}$ if it exists, and let $g(a)$ be undefined if it does not. Note that there can be at most one such b, as otherwise we would get $\beta_{<b_0} \cong \beta_{<b_1}$ for $b_0 \neq b_1$, contradicting the observation above. Also, note that g is injective and order preserving, as if we had $a_0 < a_1$ with $g(a_0) \geq g(a_1)$, we could again compose the isomorphisms and contradict the observation above. A key observation is that the domain of g is an initial segment of α, as if $c < a$ and $a \in \operatorname{dom}(g)$, then if f is the isomorphism $\alpha_{<a} \cong \beta_{<g(a)}$, we get that $\alpha_{<c} \cong \beta_{<f(c)}$, and hence $g(c)$ is defined and equals $f(c)$. A symmetric argument shows

that the range of g is also an initial segment of β. We now claim that either the domain of g is the whole of α, the range of g is the whole of β, or both. Otherwise, let a be the least element in α not in the domain of g and let b be the least element in β not in the range of g. Then g is an isomorphism from $\alpha_{<a}$ to $\beta_{<b}$, and we should have $g(a) = b$, contradicting our choice of a and b.

There are now three cases: If $\mathrm{dom}(g) = \alpha$ and $\mathrm{ran}(g) = \beta$, then g is an isomorphism from α to β; If $\mathrm{dom}(g) = \alpha$ but $\mathrm{ran}(g) \subsetneq \beta$ and b is the least element of $\beta \smallsetminus \mathrm{ran}(g)$, then g is an isomorphism from α to $\beta_{<b}$; If $\mathrm{dom}(g) \subsetneq \alpha$, $\mathrm{ran}(g) = \beta$, and a is the least element of $\alpha \smallsetminus \mathrm{dom}(g)$, then g is an isomorphism from $\alpha_{<a}$ to β. □

Corollary I.7. *If there is an order-preserving embedding from α to β, then there is an embedding from α to β whose image is an initial segment of β.*

Proof. If there is an order preserving embedding from α to β, then the third case of the theorem cannot be the case, as we would end up with an embedding from α to $\alpha_{<a}$ for some $a \in \alpha$, which we know cannot happen. □

Given linear orderings $\mathcal{A}$ and $\mathcal{B}$, we use $\mathcal{A} \preccurlyeq \mathcal{B}$ to denote that there exists an embedding from $\mathcal{A}$ to $\mathcal{B}$. We have proved that the embeddability relation on ordinals is linear. Define ω_1 as the quotient of $\mathbb{WO}$, the class of ($\subseteq\omega$)-presentations of well-orderings, over the isomorphism relation ordered by embeddability. That is,

$$\omega_1 = (\mathbb{WO}/{\cong}; \preccurlyeq).$$

If $\alpha \in \omega_1$, it follows from the theorem above that ${\omega_1}_{<\alpha} \cong \alpha$. Thus, all countable well-orderings are proper initial segments of ω_1, and all proper initial segments of ω_1 are countable well-orderings. A descending sequence in ω_1 would be a descending sequence in some $\alpha \in \omega_1$. Thus, ω_1 is itself well-ordered. Since no well-ordering is isomorphic to a proper initial segment of itself, it follows that ω_1 is not a countable well-ordering: It is the first uncountable ordinal.

For every ordinal α we can form a new ordinal by adding an element on top. We call this new ordinal the *successor* of α, and we denote it by $\alpha + 1$. Non-zero ordinals that are not successors of another ordinals are said to be *limit ordinals.*

The operations of addition and multiplication on ω_1 are just the addition and multiplication of linear orderings defined in page xix. One can prove that if $\mathcal{A}$ and $\mathcal{B}$ are well-orders, then so are $\mathcal{A} + \mathcal{B}$ and

$\mathcal{A} \times \mathcal{B}$. We know that those operations coincide with addition and multiplication on natural numbers when $\mathcal{A}$ and $\mathcal{B}$ are finite.

These operations are not commutative: $1 + \omega \cong \omega \ncong \omega + 1$, and $2 \times \omega \cong \omega \ncong \omega + \omega \cong \omega \times 2$. They are associative, they have identities — 0 and 1 respectively — and left multiplication distributes over addition. Right multiplication does not distribute over addition: $(1+1) \times \omega \cong \omega$ while $1 \times \omega + 1 \times \omega \cong \omega + \omega$. Addition and multiplication are order preserving: If $\alpha_0 \leq \alpha_1$ and $\beta_0 \leq \beta_1$, then $\alpha_0 + \beta_0 \leq \alpha_1 + \beta_1$ and $\alpha_0 \times \beta_0 \leq \alpha_1 \times \beta_1$. They are strict-order preserving on the right: If $\beta_0 < \beta_1$, then $\alpha + \beta_0 < \alpha + \beta_1$ and $\alpha \times \beta_0 < \alpha \times \beta_1$.

We will often write $\alpha \cdot \beta$, and sometimes even $\alpha\beta$, for $\alpha \times \beta$.

On ordinals we have *right subtraction*: Given ordinals $\alpha < \beta$, there is a unique γ satisfying $\alpha + \gamma = \beta$. To see this, let $b \in \beta$ be such that $\alpha \cong \beta_{<b}$ and let $\gamma \cong \beta_{\geq b}$. Uniqueness follows from the fact that addition preserves strict-order on the right. We also have *left division* with remainder: Given ordinals ν and $\delta > 0$, there exist unique ordinals $\pi \leq \nu$ and $\rho < \delta$ such that $\nu = \delta \times \pi + \rho$. To see this, note that either $\delta \times \nu \cong \nu$ or $\delta \times \nu \succ \nu$. In the former case, let $\pi = \nu$ and $\rho = 0$. In the latter case, let $(d, n) \in \delta \times \nu$ be such that $(\delta \times \nu)_{<(d,n)} \cong \nu$, and then let $\pi = \nu_{<n}$ and $\rho = \delta_{<d}$. Uniqueness again follows from the fact that addition and multiplication preserve strict-order on the right.

We can also consider the addition of infinitely many linear orderings: Given a list of linear orderings $\mathcal{A}_i$ for $i \in \mathcal{L}$, where $\mathcal{L}$ is also linearly ordered, we define $\sum_{i \in \mathcal{L}} \mathcal{A}_i$ to be the concatenation of the $\mathcal{A}_i$'s according to $\mathcal{L}$. That is, as domain use the disjoint union of the $\mathcal{A}_i$'s, and let $a \leq b$ for $a \in \mathcal{A}_i$ and $b \in \mathcal{A}_j$ if either $i <_{\mathcal{L}} j$, or $i = j$ and $a \leq_{\mathcal{A}_i} b$. One can prove that if $\mathcal{L}$ and all the $\mathcal{A}_i$'s are well-ordered, so is $\sum_{i \in \mathcal{L}} \mathcal{A}_i$.

Another important operation is the *supremum*. Given a countable set $\{\mathcal{A}_i : i \in \mathbb{N}\}$ of countable well-orderings, we let $\sup_i \mathcal{A}_i$ be the least upper bound of the $\mathcal{A}_i$'s. To see this exists, notice that we already know that there is an upper bound, namely $\sum_{i \in \mathbb{N}} \mathcal{A}_i$, and since ω_1 is well-ordered, there must be a least upper bound.

I.1.1. Exponentiation. We will use ordinal exponentiation extensively throughout this book. It can be defined either by transfinite recursion or by a direct construction on linear orderings. We give both definitions.

An order-preserving function $f \colon \omega_1 \to \omega_1$ is said to be *continuous* if, for every limit ordinal λ,

$$f(\lambda) = \sup_{\beta < \lambda} f(\beta).$$

The reader can verify that addition and multiplication are both continuous on their second input. That is, if we fix an ordinal α, then for every limit ordinal λ,

- $\alpha + \lambda = \sup_{\beta<\lambda} \alpha + \beta$.
- $\alpha \times \lambda = \sup_{\beta<\lambda} \alpha \times \beta$.

One could use these properties to define addition and multiplication using recursion instead of a direct construction as above. These formulas would be used for the limit case, and, at the successor cases, we would use the following formulas:

- $\alpha + (\beta + 1) = (\alpha + \beta) + 1$
- $\alpha \times (\beta + 1) = (\alpha \times \beta) + \alpha$.

In a similar fashion, one can define exponentiation by recursion:

- $\alpha^0 = 1$,
- $\alpha^{\beta+1} = \alpha^\beta \times \alpha$, and
- $\alpha^\lambda = \sup_{\gamma<\lambda} \alpha^\gamma$ for λ limit.

Alternatively, we could write these three equations into one that works for all α and β:

$$\alpha^\beta = \sup\{\alpha^\gamma \times \alpha : \gamma < \beta\}.$$

It is not hard to see that exponentiation is order preserving on both inputs and is continuous on its second input.

Recall that the base-b expansion of a natural number m is a sequence of numbers $n_0, ..., n_k$ between 0 and $b-1$ such that $m = b^k \cdot n_0 + \cdots + b \cdot n_1 + n_0$. The same is true for ordinals:

Lemma I.8. *Fix an ordinal β. For every ordinal μ, there are ordinals $\alpha_0 > \alpha_1 > \cdots > \alpha_k$ and $\nu_0, ..., \nu_k < \beta$ such that*

$$\mu = \beta^{\alpha_0} \cdot \nu_0 + \beta^{\alpha_1} \cdot \nu_1 + \cdots + \beta^{\alpha_k} \cdot \nu_k.$$

Furthermore, $k, \alpha_0, ..., \alpha_k, \nu_0, ..., \nu_k$ are uniquely determined from β and μ.

Proof. We use transfinite induction on μ and assume such a unique decomposition exists for all $\rho < \mu$. If μ had such a decomposition, the first thing to observe is that $\beta^{\alpha_1} \cdot \nu_1 + \cdots + \beta^{\alpha_k} \cdot \nu_k < \beta^{\alpha_0}$, which can be easily proved by induction on k. We must then have

$$\beta^{\alpha_0} \leq \beta^{\alpha_0} \cdot \nu_0 \leq \quad \mu \quad < \beta^{\alpha_0} \cdot (\nu_0 + 1) < \beta^{\alpha_0+1}.$$

From this, we first observe that α_0 must be the supremum of all the α's with $\beta^\alpha \leq \mu$. Second, that there is then a unique possible value for ν_0: Using left-division with reminder, we can find ν_0 and $\rho < \beta^{\alpha_0}$ such that

$$\mu = \beta^{\alpha_0} \cdot \nu_0 + \rho.$$

Since $\beta^{\alpha_0} \times \beta = \beta^{\alpha+1} > \mu$, we must have $\nu_0 < \beta$. Since $\rho < \beta^{\alpha_0} \leq \mu$, by the induction hypothesis, we can write ρ uniquely as

$$\rho = \beta^{\alpha_1} \cdot \nu_1 + \cdots + \beta^{\alpha_k} \cdot \nu_k.$$

Putting these last two equations together, we get the decomposition of μ we were looking for. Note that $\alpha_1 < \alpha_0$, as $\beta^{\alpha_1} \leq \rho < \beta^{\alpha_0}$. □

The preferred base when dealing with ordinals is, of course, ω. In the case when $\beta = \omega$, this decomposition of μ is called the *Cantor normal form* of μ.

One can use the base-β decomposition of the elements of β^α to give an order-theoretic and more constructive definition of exponentiation. Given linear orderings $\mathcal{A}$ and $\mathcal{B}$, where $\mathcal{B}$ has an element designated as $0_\mathcal{B}$, we define a new linear ordering $\mathcal{B}^\mathcal{A}$ as follows: We let the domain of $\mathcal{B}^\mathcal{A}$ be the set of all functions from $\mathcal{A}$ to $\mathcal{B}$ of finite support, i.e. equal to $0_\mathcal{B}$ in all but finitely many inputs. We define an ordering on $\mathcal{B}^\mathcal{A}$ as follows: Given two different functions, $f, g : \mathcal{A} \to \mathcal{B}$ with finite support, we let $f <_{\mathcal{B}^\mathcal{A}} g$ if and only if, for the $\mathcal{A}$-greatest $a \in \mathcal{A}$ with $f(a) \neq g(a)$, we have $f(a) <_\mathcal{B} g(a)$.

When $\mathcal{A}$ and $\mathcal{B}$ are presentations of ordinals α and β with $0_\mathcal{B}$ being the least element of $\mathcal{B}$, one can prove that $\mathcal{B}^\mathcal{A}$ has the same order type as the ordinal β^α we defined before. In this isomorphism, a function $f\colon \mathcal{A} \to \mathcal{B}$ with finite support corresponds to the element of β^α given by

$$\sum_{\substack{a \in \alpha^* \\ f(a) \neq 0_\mathcal{B}}} \beta^a \cdot f(a),$$

where α^* is the inverse order of α. Since almost all of the values of $f(a)$ are zero, the summation above is a finite sum. We sum over the inverse order of α because we put the terms corresponding to higher exponents to the left and lower exponents to the right. That is, if $\{a \in A : f(a) \neq 0_\mathcal{B}\} = \{a_0 > a_1 > \cdots > a_k\}$, then

$$\sum_{a \in \mathcal{A}^*} \beta^a \cdot f(a) = \beta^{a_0} \cdot f(a_0) + \beta^{a_1} \cdot f(a_1) + \cdots + \beta^{a_k} \cdot f(a_k).$$

Exponentiation on linear orderings satisfies the usual properties of exponentiation of real numbers:

$$\mathcal{B}^{\mathcal{C}+\mathcal{D}} \cong \mathcal{B}^\mathcal{C} \times \mathcal{B}^\mathcal{D} \quad \text{and} \quad \mathcal{B}^{\mathcal{C}\times\mathcal{D}} \cong (\mathcal{B}^\mathcal{C})^\mathcal{D}.$$

We leave the verification of these properties to the reader.

Observation I.9. If $\mathcal{A}$ is a computable linear ordering, using these functions of finite support we get a computable ω-presentation of $\omega^\mathcal{A}$.

Furthermore, the operation of addition is computable in $\omega^{\mathcal{A}}$, independently of whether addition was or was not computable in $\mathcal{A}$. If $\mathcal{A}$ has a least element $0_{\mathcal{A}}$, then $\omega^{0_{\mathcal{A}}}$ is the second least element of $\omega^{\mathcal{A}}$, which we call $1_{\omega^{\mathcal{A}}}$. We also get a computable successor operator and a computable way of deciding if a member of $\omega^{\mathcal{A}}$ is limit or successor.

Observation I.10. Let us consider the particular case when $\mathcal{A}$ has no least element, just for a minute. In this case, one can show that $\mathcal{B}^{\mathcal{A}}$ is dense and has no endpoints, and thus is isomorphic to the rationals (Exercise I.13). In general, every linear ordering $\mathcal{A}$ can be decomposed as $\mathcal{A}_{WO} + \mathcal{A}_{IO}$ where $\mathcal{A}_{WO}$ is well-ordered and $\mathcal{A}_{IO}$ has no least element. We then get that $\mathcal{B}^{\mathcal{A}} \cong \mathcal{B}^{\mathcal{A}_{WF}} \times \mathbb{Q}$.

Exercise I.11. Prove that the well-ordering from Exercise I.2 is isomorphic to ω^{ω}.

Exercise I.12. Prove that if $f\colon \omega_1 \to \omega_1$ is order preserving and continuous, it has uncountably many fixed points.

Exercise I.13. Prove that if $\mathcal{A}$ has no least element, $\mathcal{B}^{\mathcal{A}}$ is dense and has no endpoints.

Exercise I.14. A linear ordering that will appear often in examples is $\mathbb{Z}^{\alpha}$ for ordinal α.

(a) Prove that any two elements of $\mathbb{Z}^{\alpha}$ are automorphic.

(b) Prove that if a linear ordering $\mathcal{L}$ satisfies that any two elements are automorphic, then it must be isomorphic to $\mathbb{Z}^{\mathcal{A}}$ for some linear ordering $\mathcal{A}$. See hint in footnote.[‡]

I.2. Well-foundedness

We now move to well-founded partial orderings, which we will also use extensively throughout the book. Again, the first half of this section can be found in most basic logic textbooks, though the second half not as much.

Definition I.15. We say that a partial ordering is *well-founded* if it has no infinite descending sequences. Otherwise, we say it is *ill-founded.* A tree $T \subseteq \mathbb{N}^{<\mathbb{N}}$ is *well-founded* if it has no infinite paths, or equivalently, if $(T; \supseteq)$ is a well-founded partial ordering. (Notice the order in $(T; \supseteq)$ is reverse inclusion, with the root sitting on top.)

[‡] For each element ℓ of $\mathcal{L}$, consider the supremum of the ordinals α such that ℓ belongs to a segment isomorphic to $\mathbb{Z}^{\alpha}$. Then, consider the quotient of $\mathcal{L}$ over these segments.

It is not hard to see that a partial ordering is well-founded if and only if every subset has a minimal element, that is, an element with no other element from the subset below it.

Well-founded partial orderings do not behave as neatly as ordinals. However, some useful properties still hold. The induction and recursion principles can be proved for well-founded partial orderings using exactly the same proofs we used for transfinite induction and transfinite recursion on page 2.

Theorem I.16 (Well-founded induction). *Let $\mathcal{P} = (P; \leq_P)$ be a well-founded partial ordering and I a subset of P that satisfies that, for every $a \in P$, if $P_{<a} \subseteq I$, then $a \in I$. Then $I = P$.*

Theorem I.17 (Well-founded recursion). *Let $\mathcal{P} = (P; \leq_P)$ be a well-founded partial ordering, X any set, and Ψ an operator that, given $a \in P$ and a function $P_{<a} \to X$, outputs an element of X. Then there is a unique total function $g \colon P \to X$ such that*

$$g(a) = \Psi(a, g \restriction P_{<a}) \qquad \textit{for every } a \in P.$$

We will assign to each well-founded partial ordering a rank, which is an ordinal that in some sense measures its well-foundedness. We start by assigning a rank to each element of a partial ordering as follows: All the minimal elements in a partial ordering get rank 0. Among the remaining elements, the minimal ones get rank 1. Among the remaining elements, the minimal ones get rank 2, and so on and so forth, continuing throughout the ordinals. An element that is *never*[§] reached through this process gets rank ∞. Here is a more formal definition.

Definition I.18. For technical convenience, we let ∞ be a symbol for an element that we think of as larger than all ordinals. Also for technical convenience, we let ∞ satisfy $\infty + 1 = \infty$ and $\infty < \infty$. The *well-founded part* $\mathrm{WF}(\mathcal{P})$ of a partial ordering $\mathcal{P}$ is the set of $p \in \mathcal{P}$ for which $\mathcal{P}_{<p}$ is well-founded.

We define the *rank function* $\mathrm{rk}_{\mathcal{P}} \colon P \to \omega_1 \cup \{\infty\}$ as follows: All elements in the ill-founded part of $\mathcal{P}$, namely $P \smallsetminus \mathrm{WF}(\mathcal{P})$, are assigned rank ∞. On $\mathrm{WF}(\mathcal{P})$, the *rank function* is defined by well-founded recursion:

$$\mathrm{rk}_{\mathcal{P}}(p) = \sup\{\mathrm{rk}_{\mathcal{P}}(q) + 1 : q \in P, q <_P p\}.$$

[§] In this context, the informal word ‘never’ means not even after α many steps for any ordinal α.

We then define $\mathrm{rk}(\mathcal{P}) = \sup\{\mathrm{rk}_{\mathcal{P}}(q) + 1 : q \in P\}$. When we are computing ranks of trees, it is customary to let

$$\mathrm{rk}(T) = \mathrm{rk}_T(\langle\rangle).$$

Note that the rank of T as a partial ordering and the rank of T as a tree are off by one.

Lemma I.19. *The* rank function *on a countable partial ordering $\mathcal{P}$ is the least $<$-preserving function $f\colon P \to \omega_1 \cup \{\infty\}$.*[¶]

Proof. First observe that rk is indeed $<$-preserving, which is immediate from the definition.

Suppose $f\colon P \to \omega_1 \cup \{\infty\}$ is $<$-preserving. If $p \in \mathcal{P} \smallsetminus \mathrm{WF}(\mathcal{P})$, then $f(p)$ must be ∞, as if $p >_{\mathcal{P}} p_1 >_{\mathcal{P}} p_2 >_{\mathcal{P}} \cdots$, is an infinite descending sequence, then so is $f(p) > f(p_1) > f(p_2) > \cdots$ which could only happen if $f(p) = f(p_1) = \cdots = \infty$. We now use well-founded induction to show that $\mathrm{rk}_{\mathcal{P}}(p) \leq f(p)$ for all $p \in \mathrm{WF}(\mathcal{P})$:

$$\begin{aligned} \mathrm{rk}_{\mathcal{P}}(p) &= \sup\{\mathrm{rk}_{\mathcal{P}}(q) + 1 : q \in P, q <_P p\} \\ &\leq \sup\{f(q) + 1 : q \in P, q <_P p\} \\ &\leq f(p). \end{aligned}$$

The second line follows from the induction hypothesis and the third line from the fact that f is $<$-preserving. □

Corollary I.20. *Let $\mathcal{P}$ and $\mathcal{Q}$ be partial orderings. If there exists a $<$-preserving map $f\colon \mathcal{P} \to \mathcal{Q}$, then* $\mathrm{rk}(\mathcal{P}) \leq \mathrm{rk}(\mathcal{Q})$.

Proof. The composition $\mathrm{rk}_{\mathcal{Q}} \circ f\colon \mathcal{P} \to \omega_1 \cup \{\infty\}$ is $<$-preserving. From the previous lemma, we get that, for all $p \in \mathcal{P}$, $\mathrm{rk}_{\mathcal{P}}(p) \leq \mathrm{rk}_{\mathcal{Q}}(f(p))$. It follows that

$$\begin{aligned} \mathrm{rk}(\mathcal{P}) = \sup\{\mathrm{rk}_{\mathcal{P}}(p) + 1 : p \in P\} \\ \leq \sup\{\mathrm{rk}_{\mathcal{Q}}(f(p)) + 1 : p \in P\} \\ \leq \sup\{\mathrm{rk}_{\mathcal{P}}(q) + 1 : q \in Q\} = \mathrm{rk}(\mathcal{Q}). \quad \square \end{aligned}$$

In the case of trees we also get the converse.

Lemma I.21. *Let $T, S \subseteq \mathbb{N}^{<\mathbb{N}}$ be trees. Then* $\mathrm{rk}(T) \leq \mathrm{rk}(S)$ *if and only if there exists a $\subsetneq$-preserving map $f\colon T \to S$.*

Proof. The $(\Leftarrow)$ direction follows from the previous lemma. Suppose now that $\mathrm{rk}(T) \leq \mathrm{rk}(S)$, and hence $\mathrm{rk}_T(\langle\rangle) \leq \mathrm{rk}_S(\langle\rangle)$. We build a $\subsetneq$-preserving map $f\colon T \to S$ defining $f(\tau)$ by recursion on the length

[¶] A map $f\colon \mathcal{P} \to \mathcal{Q}$ is *$<$-preserving* if whenever $x <_{\mathcal{P}} y$, $f(x) <_{\mathcal{Q}} f(y)$. Such maps need not be one-to-one.

$|\tau|$ of the string τ. At each step, we make sure that $\mathrm{rk}_T(\tau) \leq \mathrm{rk}_S(f(\tau))$. Start by letting $f(\langle\rangle) = \langle\rangle$. Suppose we have already defined $f(\tau)$ and we want to define $f(\sigma)$ for a child σ of τ. Since $\mathrm{rk}_T(\sigma) < \mathrm{rk}_T(\tau) \leq \mathrm{rk}_S(f(\tau))$ and $\mathrm{rk}_S(f(\tau)) = \sup\{\mathrm{rk}_S(\gamma) + 1 : \gamma \in S, \gamma \supsetneq f(\tau)\}$, there must exist a child γ of $f(\tau)$ with $\mathrm{rk}_S(\gamma) \geq \mathrm{rk}_T(\sigma)$. Define $f(\sigma)$ to be one of those γ's. □

I.3. Well-foundedness versus well-orderness

Let us look at complexity. In this section, we show that deciding whether a linear ordering is well-ordered is as hard as deciding whether a partial ordering is well-founded, or deciding whether a tree is well-founded. The ideas in the proofs, which require building one type of object from another, will be useful throughout the book.

Definition I.22. Given classes of reals $\mathcal{A}_0 \subseteq \mathcal{B}_0 \subseteq \mathbb{N}^\mathbb{N}$ and $\mathcal{A}_1 \subseteq \mathcal{B}_1 \subseteq \mathbb{N}^\mathbb{N}$, we say that $\mathcal{A}_0$ *effectively Wadge-reduces* to $\mathcal{A}_1$ within $\mathcal{B}_0$ and $\mathcal{B}_1$ if there is a computable operator $\Phi\colon \mathcal{B}_0 \to \mathcal{B}_1$ such that

$$X \in \mathcal{A}_0 \iff \Phi(X) \in \mathcal{A}_1$$

for all $X \in \mathcal{B}_0$. Two classes are *effectively Wadge-equivalent* if they reduce to each other.

Theorem I.23. *The following classes are effectively Wadge-equivalent:*

(1) *The class of well-orderings within the class of linear orderings.*
(2) *The class of well-founded partial orderings within the class of partial orderings.*
(3) *The class of well-founded trees within the class of trees (viewed as subtrees of $\mathbb{N}^{<\mathbb{N}}$).*

The proof of this theorem requires various lemmas and definitions. We will finish it on page 13. Let us start with the reduction from trees to linear orderings.

Definition I.24. The *Kleene–Brouwer* ordering $\leq_{\text{KB}}$ is an ordering on $\mathbb{N}^{<\mathbb{N}}$ which coincides with the lexicographic ordering on incomparable strings but reverses inclusion on comparable strings: That is, for $\sigma, \tau \in \mathbb{N}^{<\mathbb{N}}$, $\sigma \leq_{\text{KB}} \tau$ if either $\sigma \supseteq \tau$, or $\sigma(i) < \tau(i)$ for the least i with $\sigma(i) \neq \tau(i)$.

Note that $\leq_{\text{KB}}$ linearly orders $\mathbb{N}^{<\mathbb{N}}$.

Exercise I.25. Show that $(\mathbb{N}^{<\mathbb{N}}; \leq_{\text{KB}})$ has the same order type as $\mathbb{Q} \cap (0, 1]$.

When we refer to the Kleene–Brouwer ordering of a tree $T \subseteq \mathbb{N}^{<\mathbb{N}}$, we mean the linear ordering $\mathrm{KB}(T) = (T; \leq_{\mathrm{KB}})$. Notice that

$$(T; \leq_{\mathrm{KB}}) = (T_0; \leq_{\mathrm{KB}}) + (T_1; \leq_{\mathrm{KB}}) + (T_2; \leq_{\mathrm{KB}}) + \cdots + \{\langle\rangle\},$$

where $T_n = \{\sigma \in \mathbb{N}^{<\mathbb{N}} : n^\frown\sigma \in T\}$.

The following theorem gives us the reduction from trees to linear orderings we need:

Theorem I.26. *A tree $T \subseteq \mathbb{N}^{<\mathbb{N}}$ is well-founded if and only if $(T; \leq_{\mathrm{KB}})$ is well-ordered.*

Proof. If T is not well-founded, then a path through T is also a descending sequence on $(T; \leq_{\mathrm{KB}})$.

Suppose now that $(T; \leq_{\mathrm{KB}})$ is not well-ordered and that $\sigma_0 \geq_{\mathrm{KB}} \sigma_1 \geq_{\mathrm{KB}} \sigma_2 \geq_{\mathrm{KB}} \cdots$ is an infinite $\leq_{\mathrm{KB}}$-descending sequence in T; We claim that $f \in \mathbb{N}^{\leq\mathbb{N}}$, defined by $f(n) = \lim_{i\to\infty} \sigma_i(n)$, is actually defined for all $n \in \mathbb{N}$ and is a path through T. The proof that this limit exists is by induction on n. Suppose that $\lim_{i\to\infty} \sigma_i(m)$ exists for all $m < n$, and hence that $f \upharpoonright n$ is defined and belongs to T. Let s be a stage at which all these values have reached their limits. That is, s is such that, $\sigma_t \upharpoonright n = \sigma_s \upharpoonright n$ for all $t > s$. Note that then $f \upharpoonright n = \sigma_s \upharpoonright n \in T$. Since $\sigma_s \geq_{\mathrm{KB}} \sigma_{s+1} \geq_{\mathrm{KB}} \cdots$, we must have $\sigma_s(n) \geq \sigma_{s+1}(n) \geq \cdots$. This non-increasing sequence of natural numbers must eventually stabilize and reach a limit. It follows that $f(n)$ is defined and that $f \upharpoonright n+1 \in T$. Since for every n, $f \upharpoonright n \in T$, f is a path through T. □

Exercise I.27. (a) Prove that for every well-founded tree T

$$\mathrm{rk}(T) + 1 \leq \mathrm{KB}(T) \leq \omega^{\mathrm{rk}(T)} + 1.$$

(b) Prove that, for every ordinal $\alpha > 0$, there is a tree S with $\mathrm{rk}(S) = \alpha$ and $\mathrm{KB}(S) \cong \omega^\alpha + 1$. See hint in footnote.[‖]

To reduce well-founded partial orderings to well-founded trees, we consider the tree of descending sequences: Given an ω-presentation of a partial ordering $\mathcal{P} = (P; \leq)$, let

$$T_{\mathcal{P}} = \{\sigma \in P^{<\mathbb{N}} : \sigma(0) >_P \sigma(1) >_P \cdots >_P \sigma(|\sigma| - 1)\}.$$

It is easy to see that $T_{\mathcal{P}}$ is a tree and that it has an infinite path if and only if $\mathcal{P}$ has an infinite descending sequence.

Observation I.28. The rank of the tree of descending sequences of a partial ordering $\mathcal{P}$ is the same as the rank of $\mathcal{P}$. The proof is, of course, by well-founded induction. One needs to show that, for each

[‖] Repeat each branch infinitely often.

$p \in \mathcal{P}$, if $\sigma \in T_{\mathcal{P}}$ is a string whose last element is p, then $\mathrm{rk}_T(\sigma) = \mathrm{rk}_{\mathcal{P}}(p)$. The reason is that

$$\mathrm{rk}_T(\sigma) = \sup_{q <_{\mathcal{P}} p} (\mathrm{rk}_T(\sigma^\frown q) + 1) = \sup_{q <_{\mathcal{P}} p} (\mathrm{rk}_{\mathcal{P}}(q) + 1) = \mathrm{rk}_{\mathcal{P}}(p).$$

In particular, the rank of the tree of descending sequences of an ordinal α is α.

Proof of Theorem I.23. The class of well-founded trees effectively Wadge-reduces to the class of well-orderings via the Kleene-Brouwer ordering as in Theorem I.26. The class of well-orderings effectively Wadge-reduces to the class of well-founded partial orderings via the inclusion map. The class of well-founded partial orderings effectively Wadge-reduces to the class of well-founded trees via the tree of descending sequences as in the paragraph above. All these reductions stay within the classes of trees, linear orderings, and partial orderings, respectively. □

Theorem I.23 holds the same way if, instead of considering $(\subseteq\omega)$-presentations, we consider indices for computable $(\subseteq\omega)$-presentations. That is, the sets

- $\{e \in \mathbb{N} : \Phi_e$ is the atomic diagram of a well-ordered linear ordering$\}$,
- $\{e \in \mathbb{N} : \Phi_e$ is the atomic diagram of a well-founded partial ordering$\}$, and
- $\{e \in \mathbb{N} : \Phi_e$ is the characteristic function of a well-founded subtree of $\mathbb{N}^{<\mathbb{N}}\}$,

are m-equivalent. This is an important m-degree, which we will call *Kleene's $\mathcal{O}$*. Before defining Kleene's $\mathcal{O}$ formally as a set, the following lemma specifies an indexing of linear orderings that is slightly nicer than the usual one. This is just a technicality that will simplify our notation later. The objective of this technicality is not to have to worry about whether a number is an index for a linear ordering or not later. Essentially, we will let $\mathcal{L}_e$ be the linear ordering computed by the Turing functional Φ_e. For the numbers e for which Φ_e is not the diagram of a linear ordering, we still want $\mathcal{L}_e$ to be a linear ordering, as this will simplify our constructions and definitions. For this, we need to modify the definition of $\mathcal{L}_e$ just a tiny bit.

Lemma I.29. *There is a computable sequence $\{\mathcal{L}_e : e \in \mathbb{N}\}$ of computable $(\subseteq\omega)$-presentations of linear orderings such that, if Φ_e happens to be the diagram of a $(\subseteq\omega)$-presentation of a linear ordering, then $\mathcal{L}_e$ is computably isomorphic to that linear ordering.*

Proof. For each e, we first build a finite approximation $\mathcal{A}_{e,0} \subseteq \mathcal{A}_{e,1} \subseteq \cdots$ to the linear ordering with diagram Φ_e. Let $\mathcal{A}_{e,s}$ be the largest linear ordering whose domain is an initial segment of $\mathbb{N}$ for which $D(\mathcal{A}_{e,s})$, as a finite binary string, is contained in $\Phi_{e,s}$, the step s approximation to Φ_e. (I.e., for all $i < |D(\mathcal{A}_{e,s})|$, $\Phi_{e,s}(i) \downarrow = D(\mathcal{A}_{e,s})(i)$.) The limit of the sequence $\mathcal{A}_{e,0} \subseteq \mathcal{A}_{e,1} \subseteq \cdots$ is a linear ordering with diagram Φ_e. Notice that even if Φ_e is not the diagram of a linear ordering, this limit is still a linear ordering. The only obstacle to building an ω-presentation of $\bigcup_s \mathcal{A}_{e,s}$ is that the sequence may stabilize and we might never know it. We thus define $\mathcal{L}_e$ as a $(\subseteq\omega)$-presentation of this limit by letting the domain of $\mathcal{L}_e$ be $\bigcup_{s\in\mathbb{N}}(\{s\} \times (A_{e,s} \smallsetminus A_{e,s-1}))$. This is a computable set computably isomorphic to $\bigcup_s \mathcal{A}_{e,s}$. □

Definition I.30. We define $\mathcal{O}_{wo}$ as the index set of the computable well-orderings according to the indexing of the previous lemma. That is,

$$\mathcal{O}_{\text{wo}} = \{e \in \mathbb{N} : \ \mathcal{L}_e \text{ is well-ordered}\}.$$

The same way, we define $\mathcal{O}_{\text{wf}}$ to be the set of indices for computable well-founded posets. As we mentioned above, these two sets are m-equivalent. These sets are both m-equivalent to the well-known *Kleene's* $\mathcal{O}$, which is a very important object in the study of the hyperarithmetic hierarchy. In this book we will use $\mathcal{O}_{\text{wo}}$ instead of Kleene's old definition of $\mathcal{O}$ as we find $\mathcal{O}_{\text{wo}}$ more natural, more direct, and closer to intuition. Kleene's original definition was quite different in format but similar in spirit. Kleene created his own way of indexing the computable well-orders and then defined $\mathcal{O}$ to be this set of indices. His definition has a computable successor and limit relations, though as we will see soon enough, this does not make a big difference.

Exercise I.31. Show that $\mathcal{O}_{\text{wo}}$ is m-equivalent to the set of indices e for which Φ_e is total and is the diagram of a $(\subseteq\omega)$-presentation of a well-ordering.

Exercise I.32. Show that $\mathcal{O}_{\text{wo}}$ is m-equivalent to the set of indices e for which there is no infinite sequence $\langle a_n : n \in \mathbb{N}\rangle$ such that $\Phi_e(a_{n+1}) = a_n$ for all $n \in \mathbb{N}$.

Let us observe that the use of $(\subseteq\omega)$-presentations instead of the nicer ω-presentations is just to allow for finite linear orderings. This choice is of course not essential, and other choices would have been equally good, as for instance using congruence ω-presentations. The reader should not put much emphasis on this, as it distracts from the main underlying ideas.

I.4. Computable Well-orderings

A *computable ordinal* is an ordinal that has a computable ($\subseteq\omega$)-presentation. We will often refer to *a computable ordinal* α, and mean a computable ($\subseteq\omega$)-presentation $(A; \leq_\alpha)$ of a well-ordering of order type α. We define

$$\omega_1^{CK}$$

to be the least ordinal without a computable ($\subseteq\omega$)-presentation. The 'CK' stands for 'Church Kleene.' ω_1^{CK} is the effective analog of ω_1 in the sense that it is the first ordinal for which there is no effective bijection between it and ω. Notice that the set of ordinals with computable ($\subseteq\omega$)-presentations is closed downwards, as we can always truncate an ($\subseteq\omega$)-presentation of a well-ordering. Not all countable ordinals have computable ($\subseteq\omega$)-presentations, as there are only countably many computable ordinals and uncountably many countable ordinals. Thus, ω_1^{CK} is a countable ordinal, all ordinals below it are computable, and no ordinal above it is.

Let us remark that $\mathcal{O}_{\text{wo}}$ can compute an ω-presentation of ω_1^{CK}:[**]

$$\mathcal{L} = \sum_{e \in \mathcal{O}_{\text{wo}}} \mathcal{L}_e.$$

Since every ordinal below ω_1^{CK} is isomorphic to some $\mathcal{L}_e$, we get that $\mathcal{L} \geq \omega_1^{CK}$. Every initial segment of $\mathcal{L}$ is contained in a finite sum of $\mathcal{L}_e$'s with $e \in \mathcal{O}_{\text{wo}}$, and hence is computable and below ω_1^{CK}. It follows that $\mathcal{L} \cong \omega_1^{CK}$.

I.4.1. Effective transfinite recursion. We showed in Theorem I.4 how to define functions using transfinite recursion where one is allowed to use the values of the function at lower ordinals to define the new value. If the way of computing this new value from the previous ones is computable, even if we are dealing with an infinite ordinal, the function we get is also computable.

Let α be a computable well-ordering. Given $a \in \alpha$ and $e \in \mathbb{N}$, let $e\restriction_{\alpha_{<a}}$ be an index for the computable function obtained by restricting the domain of Φ_e to $\alpha_{<a}$, that is,

$$\Phi_{e\restriction_{\alpha_{<a}}}(y) = \begin{cases} \Phi_e(y) & \text{if } y \in \alpha \text{ and } y <_\alpha a \\ \uparrow & \text{if } y \notin \alpha \text{ or } y \geq_\alpha a. \end{cases}$$

[**] This is an ($\subseteq\omega$)-presentation, but, since it is infinite, one can easily make it into an ω-presentation as in [**Part 1**, Observation I.3].

Theorem I.33. *Let Ψ be a partial computable operator such that, for every $a \in \alpha$ and $i \in \mathbb{N}$, if dom$(\Phi_i) = \alpha_{<a}$, then $\Psi(a, i)$ is defined. Then, there is an index e for a partial computable function Φ_e with domain α such that, for all $a \in \alpha$,*

$$\Phi_e(a) = \Psi(a, e{\restriction}_{\alpha_{<a}}).$$

Proof. By the recursion theorem, there is an index e for a partial computable function Φ_e such that, for all $a \in \alpha$, $\Phi_e(a) = \Psi(a, e{\restriction}_{\alpha_{<a}})$, and, for all $a \notin \alpha$, $\Phi_e(a)$ is undefined.[††] We claim that Φ_e is defined on every $a \in \alpha$. If not, let $b \in \alpha$ be the least element for which $\Phi_e(b)$ is undefined. Then, Φ_e is defined everywhere on $\alpha_{<b}$, and hence $\Psi(b, e{\restriction}_{\alpha_{<b}})$ converges. But then $\Phi_e(b)$ would have to be defined too. □

[††] Apply the recursion theorem as in page xiii to the function f where $f(e, n) = \Psi(n, e{\restriction}_{\alpha_{<n}})$ for $n \in \alpha$ and $f(e, n) \uparrow$ for $n \notin \alpha$.

CHAPTER II

Infinitary Logic

In this chapter, we introduce the infinitary language $\mathcal{L}_{\omega_1,\omega}$, where one is allowed to take conjunctions or disjunctions of infinite sets of formulas. Chris Ash was the first to notice that the *computable* infinitary language, which we will see in Chapter III, provides the appropriate syntax to describe computational properties of structures — finitary first-order logic does not do the job. In this chapter, we introduce the general theory of infinitary languages. We concentrate on the part of the theory that deals with countable structures. For a more extensive development of infinitary logic, we recommend Marker's recent book [**Mar16**].

There is no computability theory in this chapter.

II.1. Definitions

Given a vocabulary τ, the infinitary language $\mathcal{L}_{\omega_1,\omega}$ over τ is built the same way as the finitary language, except that one is allowed to use infinitary conjunctions and infinitary disjunctions, so long as the number of free variables remains finite, and the number of conjuncts or disjuncts is countable:

Definition II.1. Fix a vocabulary τ. $\mathcal{L}_{\omega_1,\omega}$ is the smallest class such that:

(1) All finitary quantifier-free τ-formulas are in $\mathcal{L}_{\omega_1,\omega}$.
(2) If φ is in $\mathcal{L}_{\omega_1,\omega}$, then so are $\forall x\varphi$ and $\exists x\varphi$.
(3) If $\bar{x}$ is a finite tuple of variables and $S \subseteq \mathcal{L}_{\omega_1,\omega}$ is a countable set of formulas whose free variables are contained in $\bar{x}$, then both the infinitary disjunction of the formulas in S, denoted $\bigvee\!\!\bigvee_{\varphi\in S}\varphi$, and the infinitary conjunction of the formulas in S, denoted $\bigwedge\!\!\bigwedge_{\varphi\in S}\varphi$, are in $\mathcal{L}_{\omega_1,\omega}$.

Notice that formally, according to in this definition, negations only occur at the level of the finitary quantifier-free formulas. In general, if we want to take the negation of an $\mathcal{L}_{\omega_1,\omega}$ formula, we have to use the De Morgan laws recursively and bring the negations down to the level of the atomic formulas. For instance, $\neg\bigvee\!\!\bigvee_{\varphi\in S}\varphi$ is defined recursively to be $\bigwedge\!\!\!\bigwedge_{\varphi\in S}\neg\varphi$. This restriction is not essential, and the only reason for this convention is that it will simplify the definition of the complexity hierarchy later on.

In Section III.1, we will see how to represent $\mathcal{L}_{\omega_1,\omega}$ formulas as concrete countable objects, but for now the definition above is good enough. Given an $\mathcal{L}_{\omega_1,\omega}$ formula $\varphi(\bar{x})$, a structure $\mathcal{A}$, and a tuple $\bar{a}\in A^{|\bar{x}|}$, we should also define what it means for $\varphi(\bar{x})$ to be *satisfied*, to *hold*, or to be *true of* a $\bar{a}$ in $\mathcal{A}$. We denote this by $\mathcal{A}\models\varphi(\bar{a})$. These definitions are straightforward, and the only reason we will pay more attention to them in Section III.1 is to study their complexity.

The 'ω_1' and the 'ω' in the notation $\mathcal{L}_{\omega_1,\omega}$ come from the following more general setting. Given cardinals κ and λ, $\mathcal{L}_{\kappa,\lambda}$ is the language where one can take conjunctions and disjunctions of any size less than κ, the number of free variables can be of any cardinality less than λ, and one can have strings of $\forall$'s or strings of $\exists$'s of any length less than λ. Then, for instance, $\mathcal{L}_{\omega,\omega}$ denotes the standard *finitary* language where all the disjunctions and conjunctions are finite. In $\mathcal{L}_{\infty,\omega}$, one allows conjunctions and disjunctions of any size, but formulas can only have finitely many free variables. We will only deal with $\mathcal{L}_{\omega_1,\omega}$ in this book, and when we refer to *infinitary* formulas, we will mean $\mathcal{L}_{\omega_1,\omega}$. Some of the concepts we introduce can be generalized to uncountable structures if one uses $\mathcal{L}_{\infty,\omega}$. In contrast, languages $\mathcal{L}_{\kappa,\lambda}$ for $\lambda>\omega$ behave quite differently and do not have any connection to the material of this book.

II.1.1. Examples. Consider the vocabulary $\tau=\{e,*\}$ of groups. A classical example of a class of structures that is not axiomatizable in finitary first-order logic is *torsion groups*. These are groups on which every element becomes the identity if you multiply it with itself enough times. That torsion groups are not elementary axiomatizable can be shown by a simple application of compactness. They are, however, axiomatizable in $\mathcal{L}_{\omega_1,\omega}$. The following infinitary sentence φ says that a group is a torsion group:

$$\forall x \bigvee\!\!\!\bigvee_{n\in\mathbb{N}} \underbrace{x*x*x*\cdots*x}_{n\text{ times}}=e.$$

That is, a group $\mathcal{G}$ is a *torsion* group if and only if $\mathcal{G}\models\varphi$.

Consider now the vocabulary $\tau=\{E\}$ of graphs. Another class that is not axiomatizable by finitary first-order logic is *connected graphs*. The following infinitary sentence says that a graph is connected:

$$\forall x,y \bigvee\!\!\!\bigvee_{n\in\mathbb{N}} \exists z_1,...,z_n \ \big(xEz_1\wedge z_1Ez_2\wedge z_2Ez_3\wedge\cdots\wedge z_nEy\big).$$

Consider the vocabulary $\tau = \{<\}$ of orderings. Given two points x and y in a linear ordering, the property of x and y being *finitely apart* cannot be expressed in finitary first-order logic. The following formula $\mathrm{Fin}(x,y)$ says that there are only finitely many elements between x and y:

$$\bigvee\!\!\!\!\bigvee_{n\in\mathbb{N}} \exists z_1, ..., z_n\ \forall w\ \big(x < w < y \Rightarrow \bigvee_{i\leq n} w = z_i\big).$$

Notice that the second disjunction is finite, and that is why we use the notation $\bigvee$ instead of $\bigvee\!\!\!\!\bigvee$.

Suppose now that we want to describe the linear ordering of the integers $(\mathbb{Z}; <)$. In addition to the axioms of linear orderings, we need to say the following: The structure has no first element, has no last element, and every two elements are finitely apart. We can thus write a single infinitary sentence that is true only of the structure $(\mathbb{Z}; <)$.

Exercise II.2. Write down the sentence describing the linear ordering $\mathbb{Z}^2$.

As for limitations of $\mathcal{L}_{\omega_1,\omega}$, we will prove in Corollary II.41 that the class of well-orders cannot be described with an infinitary sentence.

II.1.2. Quantifier complexity. We want to measure the complexity of formulas in a way that matches the computational complexity of the relations they define. For formulas of arithmetic, the way to do this is by counting alternations of quantifiers. For infinitary formulas, when counting alternations, we count infinitary disjunctions as existential quantifiers and we count infinitary conjunctions as universal quantifiers. Thus, for instance, a $\Sigma^{\mathtt{in}}_4$ formula is one of the form:

$$\underbrace{\underbrace{\bigvee\!\!\!\!\bigvee_{i_1\in\mathbb{N}}\exists\bar{y}_1}\ \underbrace{\bigwedge\!\!\!\!\bigwedge_{i_2\in\mathbb{N}}\forall\bar{y}_2}\ \underbrace{\bigvee\!\!\!\!\bigvee_{i_3\in\mathbb{N}}\exists\bar{y}_3}\ \underbrace{\bigwedge\!\!\!\!\bigwedge_{i_4\in\mathbb{N}}\forall\bar{y}_4}}_{\text{4 alternations}}\ \underbrace{\Big(\psi_{i_1,i_2,i_3,i_4}(\bar{x},\bar{y}_1,\bar{y}_2,\bar{y}_3,\bar{y}_4)\Big)}_{\text{finitary, quantifier free}}.$$

There are infinitary formulas that are not $\Sigma^{\mathtt{in}}_n$ for any n, as, for instance, an infinitary disjunction of formulas φ_n where φ_n is $\Sigma^{\mathtt{in}}_n$. Such a formula would be $\Sigma^{\mathtt{in}}_\omega$. We need to continue through the ordinals.

Definition II.3. Let α be an ordinal. A formula is $\Sigma^{\mathtt{in}}_\alpha$ if it is of the form $\bigvee\!\!\!\!\bigvee_{i\in\mathbb{N}} \exists\bar{x}_i\ \varphi_i(\bar{x}_i,\bar{y})$, where the formulas φ_i are $\Pi^{\mathtt{in}}_\beta$ for some $\beta < \alpha$. Analogously, a formula is $\Pi^{\mathtt{in}}_\alpha$ if it is of the form $\bigwedge\!\!\!\!\bigwedge_i \forall\bar{x}_i\ \varphi_i(\bar{x}_i,\bar{y})$, where the formulas φ_i are $\Sigma^{\mathtt{in}}_\beta$ for some $\beta < \alpha$. Both $\Sigma^{\mathtt{in}}_0$ and $\Pi^{\mathtt{in}}_0$ are used to denote the finitary quantifier-free formulas.

In the examples above, the formulas for torsion of groups and connectedness of graphs are $\Pi^{\mathtt{in}}_2$, and the formula for finitely-apart on linear orderings is $\Sigma^{\mathtt{in}}_2$. Here are examples of formulas of higher complexity.

II.1.3. Well-founded ranks. Using transfinite recursion, we define, for each countable ordinal α, a sentence ψ_α that is true of an element a in a partial ordering $\mathcal{P}$ if and only if $\mathrm{rk}_{\mathcal{P}}(a) \leq \alpha$. First, let $\psi_0(x) \equiv \nexists y\ (y < x)$. Then, assuming we have already defined ψ_γ for $\gamma < \alpha$, let $\psi_\alpha(x)$ be the formula

$$\forall y < x \bigvee\!\!\!\bigvee_{\gamma<\alpha} \psi_\gamma(y).$$

One can show by transfinite induction that ψ_α is a $\Pi^{\mathtt{in}}_{2\cdot\alpha+1}$ sentence. The following lemma shows that we can do better.

Lemma II.4. *For each ordinal α, there is a $\Sigma^{\mathtt{in}}_{2\cdot\alpha}$ formula $\varphi_{\omega\alpha}$ such that, for any partial ordering $\mathcal{P}$ and $a \in \mathcal{P}$,*

$$\mathcal{P} \models \varphi_{\omega\alpha}(a) \iff \mathrm{rk}_{\mathcal{P}}(a) < \omega \cdot \alpha.$$

Proof. Recursively, for each ordinal β, we define a $\Sigma^{\mathtt{in}}_{2\beta}$ formula $\varphi_{\omega\beta}(x)$ that says x has rank below $\omega \cdot \beta$. If β is a limit ordinal, then $\varphi_{\omega\beta}(x)$ is the formula $\bigvee\!\!\!\bigvee_{\gamma<\beta} \varphi_{\omega\gamma}(x)$, which is $\Sigma^{\mathtt{in}}_{\beta}$. (Recall that for β limit, $\beta = 2\beta$.) For the successor case, we need to take an intermediate step. Let $\varphi_{\omega\gamma+n}(x)$ be the $\Pi^{\mathtt{in}}_{2\gamma+1}$ formula which states that x has rank below $\omega \cdot \gamma + n$ for finite $n \geq 1$, namely

$$\forall y_1, ..., y_n\big((y_1 < y_2 < \cdots < y_n < x) \Rightarrow \varphi_{\omega\gamma}(y_1)\big).$$

Finally, if $\beta = \gamma + 1$, let $\varphi_{\omega\beta}(x)$ be the formula $\bigvee\!\!\!\bigvee_{n\in\omega} \varphi_{\omega\gamma+n}(x)$, which is $\Sigma^{\mathtt{in}}_{2\cdot\gamma+2}$. □

In the case of linear orderings, there is an even more efficient formula to calculate ranks.

Lemma II.5. *For each ordinal $\alpha \geq 1$, there is a $\Sigma^{\mathtt{in}}_{2\cdot\alpha}$ sentence which is true of a linear ordering if and only if the linear ordering is well-ordered and has order type less than ω^α.*

Proof. By transfinite recursion, we write a formula $\varphi_{\omega^\beta}(x, y)$ that holds of $a, b \in L$ if and only if the interval $(a, b)_{\mathcal{L}}$ is well-ordered and has order type less than ω^β. If $\beta = 1$, then φ_{ω^1} says that the interval is finite, which we already saw in Section II.1.1 can be said by a $\Sigma^{\mathtt{in}}_2$ formula we called 'Fin(x, y).' If β is a limit ordinal, then $\varphi_{\omega^\beta}(x, y)$ is the formula $\bigvee\!\!\!\bigvee_{\gamma<\beta} \varphi_{\omega^\gamma}(x, y)$. To see that this formula is $\Sigma^{\mathtt{in}}_{2\cdot\beta}$ use that, by inductive hypotheses, the formulas $\varphi_{\omega^\gamma}(x, y)$ are $\Sigma^{\mathtt{in}}_{2\cdot\gamma}$ when $\gamma < \beta$. For the successor case we need an intermediate step. We recursively define a formula $\varphi_{\omega^\gamma\cdot n}$ that says that the interval between x and z has order type below $\omega^\gamma \cdot n$. Let $\varphi_{\omega^\gamma\cdot n}(x,z)$ be the formula that says that if we split the interval x and z into n intervals, one of them must be shorter than ω^γ:

$$\forall y_0, ..., y_n \left(x = y_0 < y_1 < \cdots < y_n = z \;\Rightarrow \bigvee_{i<n} \varphi_{\omega^\gamma}(y_i, y_{i+1}) \right)$$

Note that this formula is $\Pi^{\mathtt{in}}_{2\cdot\gamma+1}$. Finally, for $\beta = \gamma + 1$, let $\varphi_{\omega^\beta}(x,y)$ be the formula $\mathop{\bigvee\mkern-15mu\bigvee}_{n\in\mathbb{N}} \varphi_{\omega^\gamma\cdot n}$, which is a $\Sigma^{\mathtt{in}}_{2\gamma+2}$ formula. □

II.2. Scott sentences

A *Scott sentence* for a structure $\mathcal{A}$ is a sentence φ that identifies $\mathcal{A}$ up to isomorphism among countable structures in the sense that φ is true of a countable structure $\mathcal{B}$ if and only if $\mathcal{B}$ is isomorphic to $\mathcal{A}$.

The goal of this section is to show that every countable structure has a Scott sentence. The following lemma is a first approximation. Before proving the lemma, let us review the definition of a back-and-forth set.

Definition II.6. Given structures $\mathcal{A}$ and $\mathcal{B}$, we say that a set $I \subseteq A^{<\mathbb{N}} \times\ B^{<\mathbb{N}}$ has the *back-and-forth property* if, for every $\langle \bar{a}, \bar{b}\rangle \in I$,

- $D_{\mathcal{A}}(\bar{a}) = D_{\mathcal{B}}(\bar{b})$ (i.e., $|\bar{a}| = |\bar{b}|$, and $\bar{a}$ and $\bar{b}$ satisfy the same $\tau_{|\bar{a}|}$-atomic formulas);
- for every $c \in A$, there exists $d \in B$ such that $\langle \bar{a}c, \bar{b}d\rangle \in I$; and[*]
- for every $d \in B$, there exists $c \in A$ such that $\langle \bar{a}c, \bar{b}d\rangle \in I$.

We showed in [**Part 1**, Lemma III.15] that if I is a back-and-forth set, and $\langle \bar{a}, \bar{b}\rangle \in I$, then there is an isomorphism from $\mathcal{A}$ to $\mathcal{B}$ mapping $\bar{a}$ to $\bar{b}$.

Lemma II.7. *If two countable structures satisfy the same $\mathcal{L}_{\omega_1,\omega}$ sentences, they are isomorphic.*

Proof. Let $\mathcal{A}$ and $\mathcal{B}$ be structures which satisfy the same $\mathcal{L}_{\omega_1,\omega}$ sentences. Define $I \subset A^{<\mathbb{N}} \times B^{<\mathbb{N}}$ to be the set of pairs of tuples $\langle \bar{a}, \bar{b}\rangle$ such that $(\mathcal{A}, \bar{a})$ and $(\mathcal{B}, \bar{b})$ satisfy the same $\mathcal{L}_{\omega_1,\omega}$ sentences. We claim that I has the back-and-forth property. From the hypothesis of the theorem we get that $\langle\langle\rangle, \langle\rangle\rangle \in I$. Therefore, the claim would imply that $\mathcal{A}$ and $\mathcal{B}$ are isomorphic. The first item in the definition of the back-and-forth property is trivial, and the third is analogous to the second, so we only prove the second item. Suppose $\langle \bar{a}, \bar{b}\rangle \in I$, and suppose toward a contradiction that there is a $c \in A$ such that, for every

[*] Recall that we are using the notation $\bar{a}c$ for the concatenation $\bar{a}^\frown c$.

$d \in B$, $\langle \bar{a}c, \bar{b}d \rangle \notin I$. We then have that for each $d \in B$ there is an $\mathcal{L}_{\omega_1,\omega}$ formula $\psi_d(\bar{x}, z)$ such that $\mathcal{A} \models \psi_d(\bar{a}, c)$ but $\mathcal{B} \models \neg\psi_d(\bar{b}, d)$. Therefore, on one side $\mathcal{A} \models \exists z \bigwedge\!\!\!\bigwedge_{d \in B} \psi_d(\bar{a}, z)$ as witnessed by c, while on the other side $\mathcal{B} \models \forall z \bigvee\!\!\!\bigvee_{d \in B} \neg\psi_d(\bar{b}, z)$. We have thus found a formula true about $(\mathcal{A}, \bar{a})$ that is not true about $(\mathcal{B}, \bar{b})$, contradicting that $\langle \bar{a}, \bar{b} \rangle \in I$. □

In particular, we get that two tuples $\bar{a}$ and $\bar{b}$ from the same structure $\mathcal{A}$ are automorphic if they satisfy the same $\mathcal{L}_{\omega_1,\omega}$ formulas, that is, if they have the same $\mathcal{L}_{\omega_1,\omega}$-type: Just consider the structures $(\mathcal{A}, \bar{a})$ and $(\mathcal{A}, \bar{b})$.

Recall that the automorphism orbit of a tuple $\bar{a} \in A^{<\mathbb{N}}$ is the set of all the $\bar{b} \in A^{|\bar{a}|}$ for which there is an automorphism of $\mathcal{A}$ mapping $\bar{a}$ to $\bar{b}$.

Lemma II.8. *The automorphism orbit of every tuple in a countable structure is definable by an $\mathcal{L}_{\omega_1,\omega}$-formula.*

Proof. Fix a tuple $\bar{a}$ from a structure $\mathcal{A}$. By the previous lemma, for each tuple $\bar{b}$ not automorphic to $\bar{a}$, there is a formula $\theta_{\bar{a},\bar{b}}(\bar{x})$ true of $\bar{a}$ and false of $\bar{b}$ in $\mathcal{A}$. We then have that the formula $\varphi_{\bar{a}}(\bar{x})$ defined as

$$\bigwedge\!\!\!\bigwedge_{\bar{b} \in A^{|\bar{a}|}, (\mathcal{A},\bar{a}) \not\cong (\mathcal{A},\bar{b})} \theta_{\bar{a},\bar{b}}(\bar{x})$$

is true of $\bar{a}$, but not of any tuple not automorphic to $\bar{a}$. Since satisfaction of $\mathcal{L}_{\omega_1,\omega}$ formulas is preserved under automorphisms, the formula above is true exactly on the tuples that are automorphic to $\bar{a}$. □

We have already seen in [**Part 1**, Lemma III.33] how to build a Scott sentence if we are given definitions of all automorphism orbits. The idea was to write down a sentence that is true of a structure $\mathcal{B}$ if and only if the set

$$I_{\mathcal{B}} = \{\langle \bar{a}, \bar{b} \rangle \in \mathcal{A}^{<\mathbb{N}} \times \mathcal{B}^{<\mathbb{N}} : \mathcal{B} \models \varphi_{\bar{a}}(\bar{b})\}$$

has the back-and-forth property, where $\varphi_{\bar{a}}(\bar{x})$ is the formula that defines the automorphism orbit of $\bar{a}$. To include the pair of empty tuples $\langle\langle\rangle, \langle\rangle\rangle$ into I, we let $\varphi_{\langle\rangle}()$ be a sentence that is always true. The sentence is:

$$\bigwedge\!\!\!\bigwedge_{\bar{a} \in A^{<\mathbb{N}}} \forall x_1, ..., x_{|\bar{a}|} \Big(\varphi_{\bar{a}}(\bar{x}) \Rightarrow$$

$$D(\bar{x}) = D_{\mathcal{A}}(\bar{a}) \wedge \left(\bigwedge\!\!\!\bigwedge_{c \in A} \exists y \varphi_{\bar{a}c}(\bar{x}y) \right) \wedge \left(\forall y \bigvee\!\!\!\bigvee_{c \in A} \varphi_{\bar{a}c}(\bar{x}y) \right) \Big),$$

where $D_{\mathcal{A}}(\bar{a})$ is the finite atomic diagram of the tuple $\bar{a}$ in $\mathcal{A}$ as defined on page xxiii and, if we let $\sigma = D_{\mathcal{A}}(\bar{a}) \in 2^{<\mathbb{N}}$, then "$D(\bar{x}) = \sigma$" is the quantifier free formula stating that the atomic diagram of $\bar{x}$ is σ as in Observation .5 on xxiii. We get the following corollary.

THEOREM II.9 (Scott [**Sco65**]). *Every countable structure has a Scott sentence in $\mathcal{L}_{\omega_1,\omega}$.*

COROLLARY II.10. *A relation on a structure $\mathcal{A}$ is $\mathcal{L}_{\omega_1,\omega}$ definable if and only if it is closed under automorphisms.*

PROOF. Clearly a definable relation must be closed under automorphisms.

For the converse, let R be a relation in A^k that is closed under automorphism. Given a tuple $\bar{a} \in A^k$, let $\varphi_{\bar{a}}(\bar{x})$ be a formula that defines the automorphism orbit of $\bar{a}$. Then, $\bigvee\!\!\!\bigvee_{\bar{a}\in R} \varphi_{\bar{a}}(\bar{x})$ defines R. □

OBSERVATION II.11. If every automorphism orbit in $\mathcal{A}$ is definable by a $\Sigma^{\mathtt{in}}_{\alpha}$-formula without parameters, then $\mathcal{A}$ has a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence. To see this, just count the quantifiers in the Scott sentence given above.

EXERCISE II.12. Karp [**Kar65**]. Just for this exercise, consider structures of arbitrary cardinality. For structures $\mathcal{A}$ and $\mathcal{B}$, show that they satisfy the same $\mathcal{L}_{\infty,\omega}$ sentences if and only if there is a set $I \subseteq A^{<\mathbb{N}} \times B^{<\mathbb{N}}$ that has the back-and-forth property and contains the pair of empty tuples. See hint in footnote.[†]

II.3. Scott Rank

We dedicated [**Part 1**, Chapter III] to study $\exists$-atomic structures, and showed that from various viewpoints they are the simplest structures around. We will see in the next few sections how every structure can be made $\exists$-atomic if one adds enough relations to the vocabulary. This will allow us to use the whole artillery of results from [**Part 1**, Chapter III] on all structures.

DEFINITION II.13. Given a class Γ of $\mathcal{L}_{\omega_1,\omega}$ formulas (for example $\Sigma^{\mathtt{in}}_{\alpha}$ or $\Pi^{\mathtt{c}}_{\alpha}$), a structure $\mathcal{A}$ is said to be Γ-*atomic* if every automorphism orbit is definable by a formula in Γ without parameters.

EXAMPLE II.14. $(\mathbb{Q}; \leq)$ is quantifier-free-atomic, as the automorphism type of a tuple is determined by the order of its elements.

[†] For the ($\Leftarrow$) direction, prove it for tuples within the structures and use transfinite induction on the rank of the formula.

$(\mathbb{Z}; \mathsf{Adj})$ is $\exists$-atomic,[‡] as the automorphism type of a tuple is determined by the order of its elements and the distance between the elements. $(\mathbb{Z} + \mathbb{Z} + \mathbb{Z}; \mathsf{Adj})$ is $\exists$-atomic over a finite set of parameters (three actually). $(\mathbb{Z}; \leq)$ and $(\mathbb{N}; \leq)$ are $\Sigma^{\mathtt{in}}_2$-atomic but not $\Sigma^{\mathtt{in}}_1$-atomic as follows from the next observation and [**Part 1**, Exercise III.4].

Observation II.15. If $\mathcal{A}$ is $\Sigma^{\mathtt{in}}_1$-atomic, it is also $\exists$-atomic. This is because if $\bigvee\!\!\!\bigvee_{i\in\mathbb{N}} \psi_i(\bar{x})$ defines the automorphism orbit of a tuple $\bar{a}$, where all the formulas $\psi_i(\bar{x})$ are existential, then one of these disjuncts must be true about $\bar{a}$ too — say ψ_{i_0}. But, since $\psi_{i_0}(\bar{x})$ alone implies the whole disjunction $\bigvee\!\!\!\bigvee_{i\in\mathbb{N}} \psi_i(\bar{x})$, $\psi_{i_0}(\bar{x})$ can only be true on tuples automorphic to $\bar{a}$. It follows that the automorphism orbit of $\bar{a}$ is existentially definable by $\psi_{i_0}(\bar{x})$.

Definition II.16. We define the *parameterless Scott rank* of $\mathcal{A}$ to be the least ordinal $\alpha > 0$ such that $\mathcal{A}$ is $\Sigma^{\mathtt{in}}_\alpha$-atomic. We define the *parametrized Scott rank* of $\mathcal{A}$ to be the least ordinal $\alpha > 0$ such that, for some finite tuple of parameters $\bar{a} \in \mathcal{A}^{<\mathbb{N}}$, $(\mathcal{A}, \bar{a})$ is $\Sigma^{\mathtt{in}}_\alpha$-atomic. In this book we use *Scott rank* to mean *parametrized* Scott rank.

Observation II.17. If every orbit is $\Sigma^{\mathtt{in}}_\alpha$-definable, then so is every automorphism-invariant relation, as these are countable unions of automorphism orbits. The complements of automorphism-invariant relations are also automorphism invariant, and hence are also $\Sigma^{\mathtt{in}}_\alpha$-definable. Therefore, all automorphism-invariant relations are $\Delta^{\mathtt{in}}_\alpha$-definable, including all orbits. The Scott rank is, thus, the least α such that, over some finite tuple of parameters, every automorphism-invariant relation is $\Delta^{\mathtt{in}}_\alpha$-definable.

So, for instance, from the example above we get that $(\mathbb{Q}; \leq)$, $(\mathbb{Z}; \mathsf{Adj})$, and $(\mathbb{Z} + \mathbb{Z} + \mathbb{Z}; \mathsf{Adj})$ have Scott rank 1. $(\mathbb{Z}; \leq)$ and $(\mathbb{N}; \leq)$ have Scott rank 2.

Lemma II.18. *ω^α has a Scott rank at most 2α.*

We will prove in Corollary II.40 that ω^α has Scott rank exactly 2α.

Proof. Since ω^α is rigid, i.e., has no non-trivial automorphisms, we need to find formulas defining each element of ω^α.

Let $\varphi_{\omega^\beta}(x, y)$ be the $\Sigma^{\mathtt{in}}_{2\beta}$ formula from Lemma II.5 that says that the interval between x and y has order type less than ω^β. There is a $\Pi^{\mathtt{in}}_{2\beta+1}$ sentence $\psi_{\omega^\beta}(x, y)$ that says that an interval (x, y) is isomorphic

[‡] By $\exists$-*atomic* we mean Γ-atomic where Γ is the set of finitary existential formulas.

to ω^β, namely

$$\neg\varphi_{\omega^\beta}(x,y) \wedge (\forall z(x<z<y \Rightarrow \varphi_{\omega^\beta}(x,z)))$$

Now consider $\gamma \in \omega^\alpha$. By taking its Cantor normal form, we can write γ as $\omega^{\beta_1} + \omega^{\beta_2} + \cdots + \omega^{\beta_k}$ with $\alpha > \beta_1 \geq \beta_2 \geq \cdots \geq \beta_k$. We can then write a formula $\psi_\gamma(y)$ that is only true of γ within the structure $(\omega^\alpha; \leq)$:

$$\exists z_1, ...z_k \left(z_1 < \cdots < z_k = y \wedge \bigwedge_{i<k} \psi_{\omega^{\beta_{i+1}}}(z_i, z_{i+1}) \right),$$

where the conjunct $\psi_{\omega^{\beta_1}}(z_0, z_1)$ corresponding to $i=0$ is read as saying that the interval to the left of z_1 has order type ω^{β_1}. This formula is $\Sigma^{\mathtt{in}}_{2\beta_1+2}$ and in particular $\Sigma^{\mathtt{in}}_{2\alpha}$. □

Exercise II.19. Prove that the Scott rank is preserved under $\Delta^{\mathtt{in}}_1$-bi-interpretability, where $\Delta^{\mathtt{in}}_1$-bi-interpretability is as in [**Part 1**, Definition VI.25], but using $\Sigma^{\mathtt{in}}_1$ formulas instead of $\Sigma^{\mathtt{c}}_1$ ones.

Exercise II.20. In a linear ordering, we say that x is an α-left limit if it is a left limit of β-left limit points for all $\beta < \alpha$. All points are 0-left limits. Show that the relation of being an α-left limit is $\Pi^{\mathtt{in}}_{2\alpha}$ definable. (Exercise II.42 asks to prove sharpness.)

We will see that the Scott rank is not only a measure of the complexity of the automorphism orbits of a structure, but is also a measure of how difficult it is to distinguish the structure from others, and also a measure of how difficult it is to find isomorphisms between different representations of the structure.

Let us remark that, since Scott's original definition in 1965 [**Sco65**], there have been many definitions of Scott rank — and I do not mean equivalent definitions. These different definitions may, depending on the structure, be off by 1, by ω, or even by a multiplicative factor of ω. They are not even off by the same amount on all structures; how off they are depends on the structure. The reason we prefer our definition is that it is more robust and we get equivalence theorems like II.23, VII.21, and VII.25 tying up various measures of complexity very neatly, while, for the previous notions, we did not get exact equivalences.

II.4. The type-omitting theorem

A *type-omitting theorem* is one that claims the existence of structures that satisfy certain sentences but omit certain types. By *type* here, we mean a type as in model theory, namely a set of formulas with a shared tuple of free variables, and by *omitting* a type we mean that the structure has no element satisfying all the formulas in the type. Type-omitting theorems are extremely useful in model theory, and they are useful in infinitary logic too. The original version is due to Henkin and Orey who used it for omitting the type of a non-standard natural number. See Lemma II.28 for the statement of the type-omitting theorem of finitary first-order logic.

There are various versions of the type-omitting theorem for infinitary logic, and, in most cases, their proofs are not too different from the original finitary version. The instance we will see here, where we need a sharp count of the alternations of quantifiers, is from [**Mon15b**], while other versions in the literature are too coarse for our purposes. Once the statement is set up correctly, the idea of the proof is not new, and is based on ideas the author learned from conversations with Julia Knight and Sy Friedman. The reader may consult Keisler's book [**Kei71**] or Barwise's book [**Bar75**] for other versions and other proof techniques, as for instance the use of Makkai's consistency properties.

We have already proved the cases $\alpha = 1$ of the results in this chapter back in [**Part 1**, Chapter III] using slightly simpler, but similar proofs. For general α we can take two possible approaches. We will take them both, and we will give two proofs. First, in this section, we modify the proofs in [**Part 1**, Chapter III], but we do not rely on them, so the reader who did not read [**Part 1**, Chapter III] can follow them without problem. Next, we will introduce the technique of Morleyization, which will allow us to lift the results from [**Part 1**, Chapter III] directly without re-doing the proofs.

Definition II.21. A set of infinitary formulas $\Phi(\bar{x})$ is $\Sigma^{\mathtt{in}}_{\alpha}$-*supported* in $\mathcal{A}$ if there exists a $\Sigma^{\mathtt{in}}_{\alpha}$ formula $\varphi(\bar{x})$ such that

$$\mathcal{A} \models \exists \bar{x}\big(\varphi(\bar{x})\big) \quad \wedge \quad \forall \bar{x}\big(\varphi(\bar{x}) \Rightarrow \bigwedge\!\!\!\bigwedge_{\psi \in \Phi} \psi(\bar{x})\big).$$

Lemma II.22 (Type-omitting lemma (Version from [**Mon15b**])). *Let $\mathcal{A}$ be a structure and φ be a $\Pi^{\mathtt{in}}_{\alpha+1}$ sentence true of $\mathcal{A}$. Let $\Phi(\bar{x})$ be a partial $\Pi^{\mathtt{in}}_{\alpha}$-type which is not $\Sigma^{\mathtt{in}}_{\alpha}$-supported in $\mathcal{A}$. Then there exists a structure $\mathcal{B}$ which models φ and omits Φ.*

By a *partial* $\Pi^{\mathtt{in}}_{\alpha}$*-type* we just mean a set of $\Pi^{\mathtt{in}}_{\alpha}$ formulas all sharing the same finite set of free variables. By omitting Φ we mean that no tuple from $\mathcal{B}$ satisfies Φ.

Proof. Write φ as $\bigwedge\!\!\!\bigwedge_j \forall \bar{y}_j \varphi_j(\bar{y}_j)$, where each φ_j is $\Sigma^{\mathtt{in}}_{\alpha}$. Let $C = \{\mathsf{c}_0, \mathsf{c}_1, ...\}$ be a set of fresh constants. Using a Henkin-type construction, we will build a set S of $\Sigma^{\mathtt{in}}_{\alpha}$ sentences over the vocabulary $\tau \cup C$ satisfying the following properties:

(A): If $\bigvee\!\!\!\!\bigvee \theta_i \in S$, then $\theta_i \in S$ for some i.

(B): If $\exists \bar{y}\theta(\bar{y}) \in S$, then $\theta(\bar{\mathsf{c}}) \in S$ for some tuple of constants $\bar{\mathsf{c}}$ from C.

(C): If $\bigwedge\!\!\!\!\bigwedge \theta_i \in S$, then $\theta_i \in S$ for all i.

(D): If $\forall \bar{y}\theta(\bar{y}) \in S$, then $\theta(\bar{\mathsf{c}}) \in S$ for all $\bar{\mathsf{c}}$ from C.

(E): For every atomic sentence θ over $\tau \cup C$, either $\theta \in S$ or $\neg\theta \in S$, but not both.

(F): For every j and every tuple $\bar{\mathsf{c}}$ from C of length $|\bar{y}_j|$, $\varphi_j(\bar{\mathsf{c}}) \in S$.

(G): For every tuple $\bar{\mathsf{c}}$ from C of length $|\bar{x}|$, there is a formula $\psi \in \Phi$ such that $\neg\psi(\bar{\mathsf{c}}) \in S$.

Furthermore, we will make sure along the way that every finite subset of S is satisfiable. Then, once we have S satisfying (E), we can build a structure $\mathcal{B}$ as usual: We let $\mathcal{B}$ have domain C, and we use the atomic sentences in S to define a congruence τ-presentation $\mathcal{B}$.[§] By induction on formulas, using properties (A)-(E), we get that $\mathcal{B} \models \theta$ for every $\theta \in S$. From (F) we get that $\mathcal{B} \models \varphi$, and from (G) we get that $\mathcal{B}$ omits Φ.

The construction of S is by stages as in the usual Henkin construction. At stage s, we define a finite set of $\Sigma^{\mathtt{in}}_\alpha$ sentences S_s, and we will define $S = \bigcup_{s\in\omega} S_s$ at the end. Each S_s mentions at most finitely many of the constants from C. To ensure consistency, we make sure that, at each s, there is an assignment v_s that assigns values in $\mathcal{A}$ to the constants that appear in S_s in a way that S_s holds in $\mathcal{A}$. That is, if S_s mentions the constants $\mathsf{c}_0, ..., \mathsf{c}_n$, and v_s maps c_i to $a_i \in \mathcal{A}$, then for each formula $\theta(\mathsf{c}_0, ..., \mathsf{c}_n) \in S_s$, $\mathcal{A} \models \theta(a_0, ..., a_n)$.

At each stage, we take care of a new instance of one of the requirements. Instances of the requirements (A)-(F) can all be satisfied in a straightforward way without modifying the values in the assignment v_s. For instance, suppose that at stage $s+1$ we want to satisfy requirement (B) for the sentence $\exists\bar{y}\theta(\mathsf{c}_0, ..., \mathsf{c}_n, \bar{y}) \in S_s$, and suppose v_s maps c_i to $a_i \in \mathcal{A}$. Since $\mathcal{A} \models \exists\bar{y}\theta(a_0, ..., a_n, \bar{y})$, we have that for some $\bar{b} \in A^{<\mathbb{N}}$, $\mathcal{A} \models \theta(a_0, ..., a_n, \bar{b})$. Let $\bar{\mathsf{c}}$ be a tuple of new constants, let v_{s+1} be the extension of v_s which maps $\bar{\mathsf{c}}$ to $\bar{b}$, and let $S_{s+1} = S_s \cup \{\theta(\bar{\mathsf{c}})\}$. We leave the requirements (A), (C), (D), (E) and (F) to the reader.

Requirement (G) is a standard type-omitting argument: Take a tuple $\bar{\mathsf{c}}$ from C of the same length as $\bar{x}$, and suppose we have already built S_s. Let $\varphi(\bar{\mathsf{c}}, \bar{\mathsf{d}}) = \bigwedge S_s$, where $\bar{\mathsf{d}}$ is the tuple of constants from C that occur in S_s but are not present in $\bar{\mathsf{c}}$. So $\exists\bar{y}\ \varphi(\bar{x}, \bar{y})$ is a $\Sigma^{\mathtt{in}}_\alpha$ formula realized in $\mathcal{A}$. Since Φ is not $\Sigma^{\mathtt{in}}_\alpha$-supported, there is a formula $\theta(\bar{x}) \in \Phi$

[§] I.e., if the sentence '$\mathsf{c}_i = \mathsf{c}_j$' is in S, we let c_i and c_j be equivalent in $\mathcal{B}$.

such that $\mathcal{A} \models \neg\forall\bar{x}(\exists\bar{y}\varphi(\bar{x},\bar{y}) \to \theta(\bar{x}))$. That is, there are tuples $\bar{a},\bar{b} \in \mathcal{A}^{<\mathbb{N}}$ such that $\mathcal{A} \models \varphi(\bar{a},\bar{b}) \wedge \neg\theta(\bar{a})$. Let $S_{s+1} = S_s \cup \{\neg\theta(\bar{\mathsf{c}})\}$, and let v_{s+1} map $\bar{\mathsf{c}}\bar{\mathsf{d}}$ to $\bar{a}\bar{b}$. □

We will now use the type-omitting theorem to show how Scott ranks and Scott sentences are connected.

Theorem II.23. *Let $\mathcal{A}$ be a countable structure and α be a countable ordinal. The following are equivalent:*

(U1) *Every automorphism orbit is $\Sigma^{\mathtt{in}}_{\alpha}$-definable without parameters.*
(U2) *$\mathcal{A}$ has a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence.*
(U3) *Every $\Pi^{\mathtt{in}}_{\alpha}$-type realized in $\mathcal{A}$ is $\Sigma^{\mathtt{in}}_{\alpha}$-supported within $\mathcal{A}$.*

This theorem is one of the first results in this book showing the robustness of our notion of Scott rank introduced in [**Mon15b**]. Earlier definitions of Scott rank did not produce such sharp equivalences.

Proof. We already saw how (U1) implies (U2) in Observation II.11.

Let us now prove that (U2) implies (U3). Let φ be a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence for $\mathcal{A}$. Suppose, towards a contradiction, that there is a $\Pi^{\mathtt{in}}_{\alpha}$ type $p(\bar{x})$ realized in $\mathcal{A}$ by some tuple $\bar{a}$ which is not $\Sigma^{\mathtt{in}}_{\alpha}$ supported within $\mathcal{A}$. By Lemma II.22, there is a structure $\mathcal{B}$ which models φ and omits $p(\bar{x})$. The structure $\mathcal{B}$ cannot be isomorphic to $\mathcal{A}$, as it omits $p(\bar{x})$, and hence this contradicts that φ is a Scott sentence for $\mathcal{A}$.

Let us now prove that (U3) implies (U1). For each tuple $\bar{a}$ in $\mathcal{A}$, let $\varphi_{\bar{a}}(\bar{x})$ be a $\Sigma^{\mathtt{in}}_{\alpha}$ formula that supports $\Pi^{\mathtt{in}}_{\alpha}\text{-tp}_{\mathcal{A}}(\bar{a})$. We will show that $\varphi_{\bar{a}}(\bar{x})$ defines the automorphism orbit of $\bar{a}$.

First, note that $\varphi_{\bar{a}}$ is true of $\bar{a}$, as otherwise $\neg\varphi_{\bar{a}}$ would belong to $\Pi^{\mathtt{in}}_{\alpha}\text{-tp}_{\mathcal{A}}(\bar{a})$, and it would be implied by $\varphi_{\bar{a}}$. Second, we need to observe that if $\mathcal{A} \models \varphi_{\bar{a}}(\bar{b})$, then $\mathcal{A} \models \varphi_{\bar{b}}(\bar{a})$ too. Suppose not, and that $\mathcal{A} \models \varphi_{\bar{a}}(\bar{b}) \wedge \neg\varphi_{\bar{b}}(\bar{a})$. We would then have that $\neg\varphi_{\bar{b}}(\bar{x}) \in \Pi^{\mathtt{in}}_{\alpha}\text{-tp}_{\mathcal{A}}(\bar{a})$, and hence that $\varphi_{\bar{a}}(\bar{x})$ implies $\neg\varphi_{\bar{b}}(\bar{x})$, which we know is not true, as $\mathcal{A} \models \varphi_{\bar{a}}(\bar{b}) \wedge \varphi_{\bar{b}}(\bar{b})$.

Consider the set of pairs

$$P = \{\langle\bar{a},\bar{b}\rangle \in (\mathcal{A}^{<\mathbb{N}})^2 : \mathcal{A} \models \varphi_{\bar{a}}(\bar{b})\}.$$

We claim that P has the back-and-forth property. This would imply that $\bar{a}$ and $\bar{b}$ are automorphic whenever $\mathcal{A} \models \varphi_{\bar{a}}(\bar{b})$, and hence that $\varphi_{\bar{a}}(\bar{x})$ defines the automorphism orbit of $\bar{a}$. Suppose $\langle\bar{a},\bar{b}\rangle \in P$. Clearly $D_{\mathcal{A}}(\bar{a}) = D_{\mathcal{A}}(\bar{b})$ as $D_{\mathcal{A}}(\bar{a})$ is determined by $\Pi^{\mathtt{in}}_{\alpha}\text{-tp}_{\mathcal{A}}(\bar{a})$. Let $d \in A$; we want to show that there exists $c \in A$ such that $\langle\bar{a}c,\bar{b}d\rangle \in P$. Thus, we need to show that $\mathcal{A} \models \exists y\ \varphi_{\bar{b},d}(\bar{a},y)$. Suppose not. Then $\forall y \neg\varphi_{\bar{b},d}(\bar{a},y)$ is part of the $\Pi^{\mathtt{in}}_{\alpha}$-type of $\bar{a}$, and hence implied by $\varphi_{\bar{a}}$. But then, since

$\mathcal{A} \models \varphi_{\bar{a}}(\bar{b})$, we would have $\mathcal{A} \models \forall y \neg \varphi_{\bar{b},d}(\bar{b}, y)$, contradicting that $\mathcal{A} \models \varphi_{\bar{b}d}(\bar{b}, d)$. □

II.5. Morleyizations

In [**Part 1**, Chapter III], we showed that a structure is $\exists$-atomic if and only if it has a $\Pi^{\mathtt{in}}_2$ Scott sentence. In this section, we use the technique of Morleyization to lift that result to $\Sigma^{\mathtt{in}}_\alpha$-atomic structures and show that those are exactly the ones that have a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence. We will also use Morleyizations to prove an α-level version of the type-omitting theorem. Most results we prove here using Morleyizations were already proved in the previous sections using different proofs.

Definition II.24. Consider a vocabulary τ and a countable set Ψ of $\mathcal{L}_{\omega_1,\omega}$ τ-formulas. The *Morleyization* of τ with respect to Ψ refers to the following expansion $\check{\tau}$ of the vocabulary. Let us assume that Ψ is closed under taking sub-formulas — if not, close it. For each formula $\psi(\bar{x})$ in Ψ, consider a new relation symbol R_ψ of arity $|\bar{x}|$. Let $\check{\tau} = \tau \cup \{\mathsf{R}_\psi : \psi \in \Psi\}$.

For each τ structure $\mathcal{A}$, the *Morley expansion* of $\mathcal{A}$ is the $\check{\tau}$ structure $\check{\mathcal{A}} = (\mathcal{A}, R^{\mathcal{A}}_\psi : \psi \in \Psi)$, where $R^{\mathcal{A}}_\psi = \{\bar{a} \in A^{|\bar{x}|} : \mathcal{A} \models \psi(\bar{a})\}$.

The objective of Morleyization is to simplify the complexity of formulas. For starters, all the formulas in Ψ become atomic. When studying theories, we need to ensure the new relations have the right meanings. However, adding the definitions of the new relations directly, namely $\forall \bar{x}(\mathsf{R}_\psi(\bar{x}) \leftrightarrow \psi(\bar{x}))$ for $\psi \in \Psi$, has the great disadvantage that we are adding formulas that are as complex as the formulas in Ψ, which defeats the purpose of simplifying formulas. There is a way around this.

Definition II.25. For each formula ψ, we consider a sentence φ_ψ that defines R_ψ recursively:

(1) If $\psi(\bar{x})$ is atomic, then let φ_ψ be $\forall \bar{x}(\mathsf{R}_\psi(\bar{x}) \leftrightarrow \psi(\bar{x}))$.
(2) If $\psi(\bar{x})$ is $\neg\theta(\bar{x})$, then let φ_ψ be $\forall \bar{x}(\mathsf{R}_\psi(\bar{x}) \leftrightarrow \neg \mathsf{R}_\theta(\bar{x}))$.
(3) If $\psi(\bar{x})$ is $\exists y \theta(\bar{x}, y)$, then let φ_ψ be $\forall \bar{x}(\mathsf{R}_\psi(\bar{x}) \leftrightarrow \exists y \mathsf{R}_\theta(\bar{x}, y))$.
(4) If $\psi(\bar{x})$ is $\bigvee\!\!\!\bigvee_i \theta_i(\bar{x})$, then let φ_ψ be $\forall \bar{x}(\mathsf{R}_\psi(\bar{x}) \leftrightarrow \bigvee\!\!\!\bigvee_i \mathsf{R}_{\theta_i}(\bar{x}))$.

Let M_Ψ be $\bigwedge\!\!\!\bigwedge_{\psi \in \Psi} \varphi_\psi$.

Note that M_Ψ is $\Pi^{\mathtt{in}}_2$ and that

$$M_\Psi \quad \iff \quad \bigwedge\!\!\!\bigwedge_{\psi(\bar{x}) \in \Psi} \forall \bar{x}(\mathsf{R}_\psi(\bar{x}) \leftrightarrow \psi(\bar{x})).$$

Also note that the Morley expansion $\breve{\mathcal{A}}$ with respect to Ψ is the unique $\breve{\tau}$-expansion of $\mathcal{A}$ that satisfies M_Ψ.

For our first application of Morleyization, let us consider [**Part 1**, Theorem III.34], which says that a structure is $\exists$-atomic if and only if it has a $\Pi^{\mathtt{in}}_2$ Scott sentence, and [**Part 1**, Lemma III.35], which says that a structure is $\exists$-atomic over a finite tuple of parameters if and only if it has a $\Sigma^{\mathtt{in}}_3$ Scott sentence.

Proposition II.26. *For a structure $\mathcal{A}$ and an ordinal $\alpha > 0$, the following are equivalent:*

(1) *$\mathcal{A}$ is $\Sigma^{\mathtt{in}}_\alpha$-atomic.*
(2) *$\mathcal{A}$ has a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence.*

If we consider parameters, we get that the following are equivalent:

(1) *$\mathcal{A}$ is $\Sigma^{\mathtt{in}}_\alpha$-atomic over a finite tuple of parameters.*
(2) *$\mathcal{A}$ has a $\Sigma^{\mathtt{in}}_{\alpha+2}$ Scott sentence.*

Proof. Let us consider the first part of the theorem — the proof of the second part is essentially the same.

(1) implies (2): We say that a formula is $\Pi^{\mathtt{in}}_{<\alpha}$ if it is $\Pi^{\mathtt{in}}_\beta$ for some $\beta < \alpha$. Consider the set of all $\Sigma^{\mathtt{in}}_\alpha$ formulas that define automorphism orbits of tuples in $\mathcal{A}$. Let Ψ be the set of $\Pi^{\mathtt{in}}_{<\alpha}$ formulas that appear as sub-formulas of those $\Sigma^{\mathtt{in}}_\alpha$ formulas. Notice that these $\Sigma^{\mathtt{in}}_\alpha$ formulas are $\Sigma^{\mathtt{in}}_1$ over Ψ, meaning that if we replace the formulas in Ψ with the corresponding relations, we are left with a $\Sigma^{\mathtt{in}}_1$ formula. Let $\breve{\mathcal{A}}$ be the Morley expansion of $\mathcal{A}$ with respect to Ψ. Since every relation added to the language of $\breve{\mathcal{A}}$ was already definable in $\mathcal{A}$, all automorphisms of $\mathcal{A}$ remain automorphisms of $\breve{\mathcal{A}}$, and hence both structures have the same automorphism orbits. These automorphism orbits in $\breve{\mathcal{A}}$ are now definable by $\Sigma^{\mathtt{in}}_1$ $\breve{\tau}$-formulas. By Observation II.15, this implies that all automorphism orbits are actually definable by $\exists$-$\breve{\tau}$-formulas. In other words, $\breve{\mathcal{A}}$ is $\exists$-atomic. By [**Part 1**, Theorem III.34], $\breve{\mathcal{A}}$ has a $\Pi^{\mathtt{in}}_2$ $\breve{\tau}$-Scott sentence $\breve{\varphi}$. Let φ be defined by replacing each occurrence of R_ψ in $\breve{\varphi}$ by ψ for each $\psi \in \Psi$. We claim that φ is the desired $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence for $\mathcal{A}$. It is $\Pi^{\mathtt{in}}_{\alpha+1}$ because $\breve{\varphi}$ is $\Pi^{\mathtt{in}}_2$ and each ψ being replaced is $\Pi^{\mathtt{in}}_{<\alpha}$. Let $\mathcal{B}$ be another structure satisfying φ and let $\breve{\mathcal{B}}$ be its Morley expansion with respect to Ψ. Then $\breve{\mathcal{B}} \models M_\Psi$, and hence $\breve{\mathcal{B}} \models \breve{\varphi}$, as φ and $\breve{\varphi}$ are equivalent over M_Ψ. Thus, $\breve{\mathcal{A}}$ and $\breve{\mathcal{B}}$ must be isomorphic. Their τ-reducts, namely $\mathcal{A}$ and $\mathcal{B}$, must then be isomorphic too.

(2) implies (1): Let φ be the $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence for $\mathcal{A}$. Let Ψ be the set of $\Pi^{\mathtt{in}}_{<\alpha}$ sub-formulas of φ, and consider the corresponding Morleyization. Within φ, replace each maximal $\Pi^{\mathtt{in}}_{<\alpha}$ sub-formula ψ for R_ψ. We get a $\Pi^{\mathtt{in}}_2$ $\breve{\tau}$-sentence $\breve{\varphi}$. If we assume M_Ψ, $\breve{\varphi}$ is equivalent to

φ. We thus get that $\check{\varphi} \wedge M_\Psi$ is a $\Pi^{\mathtt{in}}_2$ Scott sentence for the Morley expansion $\check{\mathcal{A}}$ of $\mathcal{A}$. By [**Part 1**, Theorem III.34], every automorphism orbit in $\check{\mathcal{A}}$ is definable by a $\exists$-$\check{\tau}$-formula. Replacing R_ψ for ψ within each of these definitions, we get equivalent formulas in $\check{\mathcal{A}}$, and hence we get $\Sigma^{\mathtt{in}}_\alpha$ τ-definitions for all the automorphism orbits in $\mathcal{A}$. □

Corollary II.27. *The parameterless Scott rank of $\mathcal{A}$ is the least ordinal α such that $\mathcal{A}$ has a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence. The parametrized Scott rank of $\mathcal{A}$ is the least ordinal α such that $\mathcal{A}$ has a $\Sigma^{\mathtt{in}}_{\alpha+2}$ Scott sentence.*

We can use the same technique to lift other results from [**Part 1**, Chapter III]. For instance, we can lift the $\forall$-type-omitting theorem we proved in [**Part 1**, Lemma III.31] and make it a $\Pi^{\mathtt{in}}_\alpha$ type-omitting theorem. Let us first recall that [**Part 1**, Lemma III.31] says that if $\mathbb{K}$ is a $\Pi^{\mathtt{in}}_2$ class of structures and $\{p_i(\bar{x}_i) : i \in \mathbb{N}\}$ a sequence of $\forall$-types which are not $\exists$-supported in $\mathbb{K}$, then there is a structure $\mathcal{A} \in \mathbb{K}$ that omits all the types $p_i(\bar{x}_i)$ for $i \in \mathbb{N}$. Recall that a type $p(\bar{x})$ is Γ-*supported* in a class $\mathbb{K}$ if there is a Γ formula $\varphi(\bar{x})$ realizable in $\mathbb{K}$ which implies all the formulas in $p(\bar{x})$ within $\mathbb{K}$. First, let us deduce the classical finitary type-omitting theorem.

Lemma II.28. *Let T be a finitary first-order theory, and let $\{p_i : i \in \mathbb{N}\}$ be a list of finitary first-order types that are not elementary supported over T. Then T has a model that omits all the p_i's.*

In the context of finitary first-order logic, types that are elementary supported are called *principal types.* Recall that an *elementary formula* is a finitary first-order formula.

Proof. Let Ψ be the set of all finitary first-order formulas, and consider the corresponding Morleyization $\check{\tau}$. Then T is equivalent to a $\Pi^{\mathtt{in}}_1$ $\check{\tau}$-sentence, each p_i is a quantifier free type (and in particular a $\forall$-type), and no p_i is $\exists$-supported over $T \wedge M_\Psi$, as otherwise they would be elementary supported over T. We can then apply [**Part 1**, Lemma III.31] to get a $\check{\tau}$-model of $T \wedge M_\Psi$ which does not realize any p_i. □

Theorem II.29. *Let $\mathbb{K}$ be the class of models of a $\Pi^{\mathtt{in}}_{\alpha+1}$ sentence φ, and let $\{p_i : i \in \mathbb{N}\}$ be a list of $\Pi^{\mathtt{in}}_\alpha$ types that are not $\Sigma^{\mathtt{in}}_\alpha$ supported in $\mathbb{K}$. Then there is a structure in $\mathbb{K}$ that omits all the p_i's.*

Proof. The proof is essentially the same as that of the lemma above. Let Ψ be the set of all $\Pi^{\mathtt{in}}_{<\alpha}$ sub-formulas of φ and of the formulas that appear in the types p_i for $i \in \mathbb{N}$. Then φ is equivalent to a $\Pi^{\mathtt{in}}_2$ $\check{\tau}$-sentence $\check{\varphi}$, each p_i is a $\Pi^{\mathtt{in}}_1$ $\check{\tau}$-type (and in particular a $\forall$-type), and no p_i is $\exists$-supported over $\check{\varphi} \wedge M_\Psi$, as otherwise they would be $\Sigma^{\mathtt{in}}_\alpha$-supported over φ. We can then apply [**Part 1**, Lemma III.31] to get a $\check{\tau}$-model of $\varphi \wedge M_\Psi$ which does not realize any p_i. □

The type-omitting theorem for fragments of infinitary logic is due to Keisler [**Kei71**]. Our formulation above, which is from [**Mon15b**], is more subtle than Keisler's original, as Keisler was not worried about the complexity of the formulas, and the fragments he used were coarser than the ones we use here.

Exercise II.30. Use Morleyization on [**Part 1**, Theorem III.22] to prove that a countable structure is $\Sigma^{\mathtt{in}}_\alpha$-atomic if and only if every $\Pi^{\mathtt{in}}_\alpha$ type realized in $\mathcal{A}$ is $\Sigma^{\mathtt{in}}_\alpha$ supported in $\mathcal{A}$.

Remark II.31. Let us briefly mention how Scott ranks work for uncountable structures. The correct definition in this setting is based on the previous exercise. First, we need to consider the language $\mathcal{L}_{\infty,\omega}$, instead of $\mathcal{L}_{\omega_1,\omega}$. The Scott rank of a structure is the least α such that, over a finite tuple of parameters, every $\Pi^{\mathtt{in}}_\alpha$ type is $\Sigma^{\mathtt{in}}_\alpha$-supported. One can then prove that the Scott rank is also the least α such that there is a $\Sigma^{\mathtt{in}}_{\alpha+2}$ sentence that determines the structure up to $\mathcal{L}_{\infty,\omega}$-elementary equivalence.

II.6. Back-and-forth relations

The back-and-forth relations measure how hard it is to differentiate two structures, or two tuples from the same structure or from different structures. They are a combinatorial device used to study $\Sigma^{\mathtt{in}}_\alpha$ elementary equivalence. The rough idea is that two tuples are n-back-and-forth equivalent if we cannot differentiate them using only n Turing jumps.

With the techniques we have seen so far, we can prove upper bounds on Scott ranks by either giving $\Sigma^{\mathtt{in}}_\alpha$ definitions of all orbits or exhibiting a $\Sigma^{\mathtt{in}}_{\alpha+2}$ Scott sentence. What we do not have yet is a technique for showing that these formulas are as simple as possible. That is where the back-and-forth relations step in.

Definition II.32. For each ordinal α, we define a pre-order $\leq_\alpha$ on the tuples of all the τ-structures by transfinite recursion. Given an ordinal α, τ-structures $\mathcal{A}$ and $\mathcal{B}$, and tuples $\bar{a} \in \mathcal{A}^{<\mathbb{N}}$ and $\bar{b} \in \mathcal{B}^{<\mathbb{N}}$, let

$$\begin{aligned}(\mathcal{A},\bar{a}) \leq_\alpha (\mathcal{B},\bar{b}) \quad \iff \quad & \forall \gamma < \alpha \\ & \forall \bar{d} \in \mathcal{B}^{<\mathbb{N}} \;\; \exists \bar{c} \in \mathcal{A}^{<\mathbb{N}} \; (\mathcal{A},\bar{a}\bar{c}) \geq_\gamma (\mathcal{B},\bar{b}\bar{d}).\end{aligned}$$

For the base case, we let $(\mathcal{A}, \bar{a}) \leq_0 (\mathcal{B}, \bar{b})$ if $\bar{a}$ and $\bar{b} \upharpoonright |\bar{a}|$ satisfy the same quantifier-free $\tau_{|\bar{a}|}$-formulas, or equivalently, if $D_{\mathcal{A}}(\bar{a}) \subseteq D_{\mathcal{B}}(\bar{b})$.[¶] We will sometimes write $\bar{a} \leq_\alpha \bar{b}$ instead of $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b})$ if one can easily deduce from context where the tuples are coming from.

Observation II.33. In most cases, one considers back-and-forth relations only between tuples of the same length, and the reader may imagine that is the case for now. For tuples of different lengths, one can show by transfinite induction that $\bar{a} \leq_\alpha \bar{b}$ if and only if $|\bar{a}| \leq |\bar{b}|$ and $\bar{a} \leq_\alpha \bar{b} \upharpoonright |\bar{a}|$.

Observation II.34. Back-and-forth relations are preserved under taking sub-tuples. That is, if $(\mathcal{A}, \bar{a}, \bar{a}') \leq_\alpha (\mathcal{B}, \bar{b}, \bar{b}')$, then $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b})$. This can be proved by an easy transfinite induction too.

Observation II.35. It is easy to see that the α-back-and-forth relations get finer as α grows. Furthermore, $(\mathcal{A}, \bar{a}) \leq_{\alpha+1} (\mathcal{B}, \bar{b})$ not only implies $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b})$, but also $(\mathcal{A}, \bar{a}) \geq_\alpha (\mathcal{B}, \bar{b})$. This, again, can be proved by an easy transfinite induction argument.

The back-and-forth relations can be visualized in terms of a game where player I is trying to show $(\mathcal{A}, \bar{a}) \not\leq_\alpha (\mathcal{B}, \bar{b})$ by challenging player II to come up with matchings for player I's moves. This is a *clopen game*, that is, a finitely terminating game where there are infinitely many possibilities for each move. Fix τ-structures $\mathcal{A}$ and $\mathcal{B}$, and tuples $\bar{a} \in \mathcal{A}^{<\mathbb{N}}$ and $\bar{b} \in \mathcal{B}^{<\mathbb{N}}$ of the same length. The game $\mathsf{G}(\alpha, (\mathcal{A}, \bar{a}), (\mathcal{B}, \bar{b}))$ starts with player I playing a tuple $\bar{b}_1 \in B^{<\mathbb{N}}$ and an ordinal $\gamma_1 < \alpha$, and player II responding with a tuple $\bar{a}_1 \in A^{<\mathbb{N}}$ of the same length. They then continue playing the game $\mathsf{G}(\gamma_1, (\mathcal{B}, \bar{b}, \bar{b}_1), (\mathcal{A}, \bar{a}, \bar{a}_1))$, where now player I is trying to show $(\mathcal{B}, \bar{b}, \bar{b}_1) \not\leq_{\gamma_1} (\mathcal{A}, \bar{a}, \bar{a}_1)$. That is, for the second move, and for subsequent even-numbered moves, player I plays a tuple $\bar{a}_k \in A^{<\mathbb{N}}$ and an ordinal $\gamma_k < \gamma_{k-1}$, and player II plays a tuple $\bar{b}_k \in B^{<\mathbb{N}}$ of the same length. At odd-numbered moves, I plays a tuple $\bar{b}_k \in B^{<\mathbb{N}}$ and an ordinal $\gamma_k < \gamma_{k-1}$, and player II plays a tuple $\bar{a}_k \in A^{<\mathbb{N}}$ of the same length.

Player I	$\bar{b}_1, \gamma_1$		$\bar{a}_2, \gamma_2$		$\bar{b}_3, \gamma_3$	$\cdots$	$\cdots$	$\bar{b}_k, \gamma_k$	
Player II		$\bar{a}_1$		$\bar{b}_2$		$\cdots$	$\cdots$		$\bar{a}_k$

The game ends when they reach $\gamma_k = 0$. Player II wins the game if $D_{\mathcal{A}}(\bar{a}, \bar{a}_1, ..., \bar{a}_k) = D_{\mathcal{B}}(\bar{b}, \bar{b}_1, ..., \bar{b}_k)$, and player I wins otherwise. One can show by transfinite induction that player II has a winning strategy for the game $\mathsf{G}(\alpha, (\mathcal{A}, \bar{a}), (\mathcal{B}, \bar{b}))$ if and only if $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b})$.

[¶] Recall that τ_s refers to the step s approximation to the vocabulary τ. Recall that to have $D_{\mathcal{A}}(\bar{a})$ be a finite string, we defined $D_{\mathcal{A}}(\bar{a})$ so that it only contains the truth values of the $\tau_{|\bar{a}|}$-formulas. See page xxiii.

There is yet a third way of visualizing back-and-forth relations. The following theorem, due to Carol Karp, characterizes the back-and-forth relations in terms of $\Pi^{\mathtt{in}}_\alpha$-types.

Theorem II.36 (Karp [**Kar65**]). *Let α be a nonzero ordinal, $\mathcal{A}$ and $\mathcal{B}$ τ-structures, and $\bar a$ and $\bar b$ tuples in $A^{<\mathbb{N}}$ and $B^{<\mathbb{N}}$. The following are equivalent:*

(1) $(\mathcal{A},\bar a) \leq_\alpha (\mathcal{B},\bar b)$.
(2) $\Pi^{\mathtt{in}}_\alpha$-$tp_{\mathcal{A}}(\bar a) \subseteq \Pi^{\mathtt{in}}_\alpha$-$tp_{\mathcal{B}}(\bar b)$*, that is, every* $\Pi^{\mathtt{in}}_\alpha$ *formula true about* $\bar a$ *in* $\mathcal{A}$ *is true about* $\bar b$ *in* $\mathcal{B}$.

Proof. The proof is by transfinite induction on α.

The theorem was stated for $\alpha > 0$ because for $\alpha = 0$ we have that $(\mathcal{A},\bar a) \leq_\alpha (\mathcal{B},\bar b)$ if and only if $D_{\mathcal{A}}(\bar a) \subseteq D_{\mathcal{B}}(\bar b)$, and recall that $D_{\mathcal{A}}(\bar a)$ only deals with atomic formulas over the finite sub-vocabulary $\tau_{|\bar a|}$. This small discrepancy disappears at higher levels.

For the downward direction, consider a $\Pi^{\mathtt{in}}_\alpha$ formula $\bigwedge\!\!\!\bigwedge_{i\in\mathbb{N}} \forall \bar y_i \varphi_i(\bar x, \bar y_i)$ true of $\bar a$ in $\mathcal{A}$, where each φ_i is $\Sigma^{\mathtt{in}}_{\alpha_i}$ and $\alpha_i < \alpha$ — we need to show this $\Pi^{\mathtt{in}}_\alpha$ formula holds of $\bar b$ in $\mathcal{B}$. Take $i \in \mathbb{N}$ and $\bar d \in B^{|\bar y_i|}$ — we need to show that $\mathcal{B} \models \varphi_i(\bar b, \bar d)$. Since $(\mathcal{A},\bar a) \leq_\alpha (\mathcal{B},\bar b)$, there is a tuple $\bar c \in A^{|\bar y_i|}$ such that $(\mathcal{A},\bar a,\bar c) \geq_{\alpha_i} (\mathcal{B},\bar b,\bar d)$. Since $\mathcal{A} \models \forall \bar y_i \varphi_i(\bar a, \bar y_i)$, $\mathcal{A} \models \varphi_i(\bar a,\bar c)$. By the induction hypothesis, applied to $\neg\varphi_i$, we get $\mathcal{B} \models \varphi_i(\bar b,\bar d)$ as needed. (When $\alpha_i = 0$, we need to extend $\bar d$ to any string $\bar d'$ that is long enough so that all the symbols in φ_i are in the finite approximation $\tau_{|\bar d'|}$ to the vocabulary τ. We would then get $\bar c'$ with $(\mathcal{A},\bar a,\bar c') \geq_0 (\mathcal{B},\bar b,\bar d')$, and hence $\mathcal{A} \models \varphi_i(\bar a,\bar c)$ implies $\mathcal{B} \models \varphi_i(\bar b,\bar d)$.)

For the upward direction, we prove the contra-positive. Suppose $(\mathcal{A},\bar a) \not\leq_\alpha (\mathcal{B},\bar b)$, and let $\bar d \in B^{<\mathbb{N}}$ and $\beta < \alpha$ be such that, for all $\bar c \in A^{<\mathbb{N}}$, $(\mathcal{A},\bar a,\bar c) \not\geq_\beta (\mathcal{B},\bar b,\bar d)$. By the induction hypothesis, for each $\bar c \in A^{|\bar d|}$, there is a $\Pi^{\mathtt{in}}_\beta$ formula $\psi_{\bar c}$ true of $\bar b\bar d$ in $\mathcal{B}$, but not of $\bar a\bar c$ in $\mathcal{A}$. Then

$$\forall \bar y \bigvee\!\!\!\bigvee_{\bar c\in A^{|\bar d|}} \neg\psi_{\bar c}(\bar a,\bar y)$$

is a $\Pi^{\mathtt{in}}_\alpha$ formula true of $\bar a$ in $\mathcal{A}$, but not of $\bar b$ in $\mathcal{B}$ as witnessed by $\bar d$. □

We will see later in Theorem VII.30 that the back-and-forth relations can also be characterized in descriptive set theoretic terms: $\mathcal{A} \leq_\alpha \mathcal{B}$ if and only if distinguishing the copies of $\mathcal{A}$ from the copies of $\mathcal{B}$ is $\boldsymbol{\Sigma}^0_\alpha$ hard.

In the literature one can find definitions of what are also called back-and-forth relations, but which are not equivalent to the one we give here. The key advantage of the definition we use here is Karp's characterization in terms of $\Pi^{\mathtt{in}}_\alpha$ types, Theorem II.36. As we mentioned before, there are also various different non-equivalent definitions of Scott rank in the literature. Most of them are based on some notion of back-and-forth relation. We will see how our definition of Scott rank can be defined in terms of this back-and-forth relation, and compare it to Ash and Knight's [**AK00**, Section 6.7] definition of Scott rank in Section II.9

II.6.1. Example: Linear Orderings.

There are various classes of structures whose back-and-forth relations have been thoroughly analyzed: The back-and-forth relations of interval Boolean algebras of ordinals are calculated in [**AK00**, Proposition 15.14]; The back-and-forth relations on F-vector spaces are calculated in [**AK00**, Section 15.3.2]; The back-and-forth relations on linear orderings are simple up to level two, but they get messy after that. The most comprehensive analysis of the back-and-forth relations on scattered linear orders to date can be found in Alvir and Rossegger's paper [**AR20**].

Linear orderings are a good case study for playing with back-and-forth calculations. The first level only involves the order among the different elements of the tuple:

$$(\mathcal{A}, a_0, ..., a_k) \leq_0 (\mathcal{B}, b_0, ..., b_k) \iff \\ a_i \leq_A a_j \leftrightarrow b_i \leq_B b_j \quad \text{for all } i, j \leq k.$$

At the next level, we compare sizes:

$$\mathcal{A} \leq_1 \mathcal{B} \quad \iff \quad |A| \geq |B|,$$

where $|A|$ is the cardinality of A, which is either a finite number or ∞. This is because for every $n \leq |B|$, if one chooses a tuple of different elements $\bar{b} \in B^n$, one has to be able to match it in $\mathcal{A}$, and hence $\mathcal{A}$ needs to have size at least n. To decide if $(\mathcal{A}, \bar{a}) \leq_1 (\mathcal{B}, \bar{b})$, one needs to look inside the segments defined by the tuples. The following lemma shows how, in linear orderings, back-and-forth calculation can be vastly simplified by comparing segments.

Lemma II.37. *(See* [**AK00**, Lemma 15.7]*) For $\alpha > 0$, when comparing tuples on linear orderings under $\leq_\alpha$, it is enough to compare the segments determined by them. That is, if $\mathcal{A}$ and $\mathcal{B}$ are linear orderings, and we have tuples $a_1 \leq_A a_2 \leq_A \cdots \leq_A a_k$ and $b_1 \leq_B b_2 \leq_B \cdots \leq_B b_k$, then*

$$(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b}) \quad \iff \quad (a_i, a_{i+1})_\mathcal{A} \leq_\alpha (b_i, b_{i+1})_\mathcal{B} \text{ for all } i \leq k,\text{[‖]}$$

interpreting a_0 and b_0 as $-\infty$, and a_{k+1} and b_{k+1} as $+\infty$.

[‖] Recall that $(a, b)_\mathcal{A}$ denotes the open interval $\{x \in A : a < x < b\}$.

Proof. The proof is a straightforward transfinite induction. □

The next level up, namely $\leq_2$, is a bit more complicated, but it can still be reasonably well understood. See [**Mon10**, Section 4.1]. The relations $\leq_3$ get much messier, except when we restrict ourselves to particular classes of linear orderings as, for instance, the class of ordinals.

Lemma II.38. *Let $\mathcal{A}$ and $\mathcal{B}$ be linear orderings, and assume both have a first element. Let α be an ordinal. Then*

$$\omega^\alpha \cdot \mathcal{A} \quad \leq_{2\alpha+1} \quad \omega^\alpha \cdot \mathcal{B}, \qquad \iff \qquad |\mathcal{A}| \geq |\mathcal{B}|.$$

Proof. This is a purely combinatorial proof, and the reader should work it out with pen and paper while reading the details.

The proof is by transfinite induction. In the case $\alpha = 0$, we have $2\alpha + 1 = 1$ and $\omega^\alpha = 1$, which puts us in the setting we already mentioned above.

For the ($\Leftarrow$) direction, assume $|\mathcal{A}| \geq |\mathcal{B}|$. Consider a tuple

$$\underbrace{\langle\gamma_{1,1}, b_1\rangle, ..., \langle\gamma_{1,\ell_1}, b_1\rangle}_{\in\omega^\alpha\times\{b_1\}}, \underbrace{\langle\gamma_{2,1}, b_2\rangle, ..., \langle\gamma_{2,\ell_2}, b_2\rangle}_{\in\omega^\alpha\times\{b_2\}}, \dots \underbrace{..., \langle\gamma_{k,\ell_k}, b_k\rangle}_{\in\omega^\alpha\times\{b_k\}}$$

from $\omega^\alpha \cdot \mathcal{B}$, where the $\gamma_{i,j}$'s belong to ω^α and the b_i's to B. Assume the tuple is given in increasing order. Also, by adding elements to the tuple if necessary, we may assume that $\gamma_{i,1}$ is the first element of ω^α for each $i \leq k$, and that b_1 is the first element of $\mathcal{B}$.[**] We need to find a matching tuple in $\omega^\alpha \cdot \mathcal{A}$. Using that $|\mathcal{A}| \geq |\mathcal{B}| \geq k$, we can pick a tuple $a_1 <_A \cdots <_A a_k$ from A, where a_1 is the first element of $\mathcal{A}$. We keep the $\gamma_{i,j}$'s unchanged. Thus, our matching tuple looks like this:

$$\underbrace{\langle\gamma_{1,1}, a_1\rangle, ..., \langle\gamma_{1,\ell_1}, a_1\rangle}_{\in\omega^\alpha\times\{a_1\}}, \underbrace{\langle\gamma_{2,1}, a_2\rangle, ..., \langle\gamma_{2,\ell_2}, a_2\rangle}_{\in\omega^\alpha\times\{a_2\}}, \dots \underbrace{..., \langle\gamma_{k,\ell_k}, a_k\rangle}_{\in\omega^\alpha\times\{a_k\}}.$$

We now need to verify that each of the intervals in $\omega^\alpha \cdot \mathcal{A}$ determined by this tuple is $\geq_{2\alpha}$-above the corresponding interval on the $\omega^\alpha \cdot \mathcal{B}$ side. There are two types of intervals. First, we have the intervals of the form $(\langle\gamma_{i,j}, b_i\rangle, \langle\gamma_{i,j+1}, b_i\rangle)_{\omega^\alpha\mathcal{B}}$, which are contained in a copy of ω^α and are isomorphic to their corresponding intervals $(\langle\gamma_{i,j}, a_i\rangle, \langle\gamma_{i,j+1}, a_i\rangle)_{\omega^\alpha\mathcal{A}}$, and hence $\geq_{2\alpha}$-back-and-forth related. Second, we have the intervals of the form $(\langle\gamma_{i,\ell_i}, b_i\rangle, \langle\gamma_{i+1,1}, b_{i+1}\rangle)_{\omega^\alpha\mathcal{B}}$ and their corresponding intervals $(\langle\gamma_{i,\ell_i}, a_i\rangle, \langle\gamma_{i+1,1}, a_{i+1}\rangle)_{\omega^\alpha\mathcal{A}}$, which are isomorphic to intervals of the form $\omega^\alpha \cdot \mathcal{B}_i$ and $\omega^\alpha \cdot \mathcal{A}_i$ respectively, where $\mathcal{B}_i = [b_i, b_{i+1})_\mathcal{B}$ and $\mathcal{A}_i = [a_i, a_{i+1})_\mathcal{A}$ are linear orderings with first elements. Note

[**] Notice that when proving that a back-and-forth relation holds, we can add elements to the tuples without loss of generality by Observation II.34.

that this is also the case for the last intervals $(\langle\gamma_{k,\ell_k}, b_k\rangle, +\infty)_{\omega^\alpha\mathcal{B}}$ and $(\langle\gamma_{k,\ell_k}, a_k\rangle, +\infty)_{\omega^\alpha\mathcal{A}}$. To prove that these intervals are $\geq_{2\alpha}$-back-and-forth related as needed, it is enough to show the following: If $\mathcal{A}$ and $\mathcal{B}$ are linear orderings with first elements (and no assumptions on their sizes), then $\omega^\alpha \cdot \mathcal{A} \geq_{2\alpha} \omega^\alpha \cdot \mathcal{B}$.

The proof starts pretty much the same way as the paragraph above. Consider a tuple $c_1, ..., c_k$ from $\omega^\alpha \cdot \mathcal{A}$ and an ordinal $\beta < \alpha$. Adding elements if necessary, assume that if an element from a copy of ω^α is one of the c_i's, so is the first element of that copy. This way, the intervals we get are either isomorphic to an ordinal smaller than ω^α, or of the form $\omega^\alpha \cdot \widetilde{\mathcal{A}}$, where $\widetilde{\mathcal{A}}$ is a segment of $\mathcal{A}$ with a first element. The last segment $(c_k, +\infty)_{\omega^\alpha\mathcal{A}}$ is always of the latter form. We now need to match these elements to elements from $\omega^\alpha \cdot \mathcal{B}$. We proceed as follows. All the c_i's will be matched to elements d_i in the first copy of ω^α. We do it in a step-by-step fashion. Map the intervals which are isomorphic to ordinals below ω^α to isomorphic copies of them. Map the intervals of the form $\omega^\alpha \cdot \widetilde{\mathcal{A}}$ to intervals of the form ω^β. Since ω^α is closed under addition, all these intervals can be found one after the other within the first copy of ω^α in $\omega^\alpha \cdot \mathcal{B}$. By the inductive hypothesis, we know that $\omega^\alpha \cdot \widetilde{\mathcal{A}}$, which is isomorphic to $\omega^\beta \cdot \omega^{\alpha-\beta} \cdot \widetilde{\mathcal{A}}$, is $\leq_{2\beta+1} \omega^\beta$. The last interval $(c_k, +\infty)_{\omega^\alpha\mathcal{A}}$, which is of the form $\omega^\alpha \cdot \widetilde{\mathcal{A}}$, is matched with the last interval of $\omega^\alpha \cdot \mathcal{B}$. Both last intervals are infinite multiples of ω^β. So all the matching intervals we defined are $\leq_{2\beta+1}$-less than their corresponding intervals in $\mathcal{A}$.

For the $(\Rightarrow)$ direction, assume $|A| < |B|$ — we need to pick a tuple in $\omega^\alpha \cdot \mathcal{B}$ without a matching tuple in $\omega^\alpha \cdot \mathcal{A}$. For this, let $b_0 < \cdots < b_{|A|}$ be $|A| + 1$ distinct elements from $\mathcal{B}$, and consider the tuple

$$\langle 0, b_0\rangle, \langle 0, b_1\rangle, ..., \langle 0, b_{|A|}\rangle$$

from $\omega^\alpha \cdot \mathcal{B}$. All the intervals are isomorphic to $\omega^\alpha \cdot [b_i, b_{i+1})_{\mathcal{B}}$. Consider a matching tuple in $\omega^\alpha \cdot \mathcal{A}$. By the pigeon-hole principle, two elements of this tuple must come from the same copy of ω^α. The interval between those two elements is then isomorphic to some ordinal below ω^α — say γ. We now need to prove that for all $\gamma < \omega^\alpha$ and all $\widetilde{\mathcal{B}}$, $\gamma \not\geq_{2\alpha} \omega^\alpha \cdot \widetilde{\mathcal{B}}$.

To prove this, consider the partition of γ into two intervals splitting γ as $\gamma_0 + \omega^\delta$, where ω^δ is the last term in the Cantor normal form of γ. If γ is already of the form ω^δ, let $\gamma_0 = 0$. Consider now a potential matching partition of $\omega^\alpha \cdot \widetilde{\mathcal{B}}$ into two intervals. The second interval must be isomorphic to $\omega^\alpha \cdot \check{\mathcal{B}} \cong \omega^\delta \cdot (\omega^{\alpha-\delta}) \cdot \check{\mathcal{B}}$ for some end segment $\check{\mathcal{B}}$ of $\widetilde{\mathcal{B}}$. Since $1 < |\omega^{\alpha-\delta} \cdot \check{\mathcal{B}}|$, we get from the induction hypothesis that

$\omega^\delta \not\leq_{2\delta+1} \omega^\alpha \cdot \check{\mathcal{B}}$. So, there is no way to match the partition of γ in $\omega^\alpha \cdot \widetilde{\mathcal{B}}$, showing that $\gamma \not\geq_{2\alpha} \omega^\alpha \cdot \widetilde{\mathcal{B}}$. □

Corollary II.39. *Let $\mathcal{A}$ be any linear ordering. Then*

$$\omega^\alpha \geq_{2\alpha+1} \omega^\alpha + \omega^\alpha \cdot \mathcal{A}, \quad \textit{but} \quad \omega^\alpha \not\leq_{2\alpha+1} \omega^\alpha + \omega^\alpha \cdot \mathcal{A}.$$

We are now ready to calculate the precise Scott rank of an ordinal [**Mil83**, Lemma 3.5]. Given an ordinal δ, define $\log_\omega(\delta)$ to be the ordinal α such that $\omega^\alpha \leq \delta < \omega^{\alpha+1}$.

Corollary II.40. *The parametrized Scott rank of an ordinal δ is $2\log_\omega(\delta)$.*

Thus, in particular, the Scott rank of ω^α is 2α.

Proof. We already know from Lemma II.18 that ω^α has Scott rank at most 2α. If δ is of the form $\delta = \omega^{\alpha_0} + \omega^{\alpha_1} + \cdots + \omega^{\alpha_k}$, where $\alpha_0 \geq \alpha_1 \geq \cdots \geq \alpha_k$, then it has Scott rank at most $2\alpha_0$ (which equals $2\log_\omega(\delta)$), as one can add parameters to separate the summands of the form ω^{α_i}.

For the lower bound, by the lemmas above, $\delta \geq_{2\alpha_0+1} \omega^\beta + \omega^{\alpha_1} + \cdots + \omega^{\alpha_k}$ for any $\beta > \alpha_0$.[††] It follows that every $\Sigma^{\mathtt{in}}_{2\alpha_0+1}$ sentence true about δ is also true about $\omega^\beta + \omega^{\alpha_1} + \cdots + \omega^{\alpha_k}$, and hence it cannot be a Scott sentence for δ. Thus, using Proposition II.26, the Scott rank of δ must be at least $2\alpha_0$. □

Corollary II.41 (Morley [**Mor65**], Lopez-Escobar [**LE66**]). *There is no $\mathcal{L}_{\omega_1,\omega}$ sentence whose countable models are exactly the countable well-orderings.*

Proof. Suppose φ is a $\Sigma^{\mathtt{in}}_\alpha$ sentence true of all ordinals. Since it is true of ω^α, it is also true of $\omega^\alpha \cdot \mathcal{A}$ for any linear ordering $\mathcal{A}$ with a first element, and hence in particular of $\omega^\alpha + \omega^\alpha \cdot \mathbb{Q}$, which is not well-founded. □

Exercise II.42. Show that the $\Pi^{\mathtt{in}}_{2\alpha}$ formula defining the relation of α-left limit in Exercise II.20 is best possible in the sense that there is no $\Sigma^{\mathtt{in}}_{2\alpha}$ formula defining the α-left limit relation in all linear orderings. See hint in footnote.[‡‡]

[††] We are using here that if $\mathcal{A} \leq_\alpha \mathcal{B}$, then $\mathcal{A} + \mathcal{C} \leq_\alpha \mathcal{A} + \mathcal{C}$. This can be proved easily by transfinite induction on α. For the case when $\mathcal{C}$ has a least element, which is the situation we are applying it to, it just follows from Lemma II.37. We are also using Corollary II.39 to get that $\omega^{\alpha_0} \geq_{2\alpha_0+1} \omega^\beta$, noting that $\omega^\beta = \omega^{\alpha_0} + \omega^{\alpha_0} \cdot \omega^{\beta-\alpha_0}$.

[‡‡] Consider $\omega^\alpha + \omega^\alpha$, and show that every $\Pi^{\mathtt{in}}_{<2\alpha}$ formula that holds of some tuple also holds of some tuple contained in the left copy of ω^α.

Exercise II.43. (a) Show that the $\Sigma^{\mathtt{in}}_{2\alpha}$ sentence φ_{ω^α} from Lemma II.5 that says that a linear ordering is strictly less than ω^α is the best possible in the sense that there is no $\Pi^{\mathtt{in}}_{2\alpha}$ sentence expressing the same thing. See hint in footnote.[*]

(b) Write a $\Pi^{\mathtt{in}}_{2\alpha+1}$ sentence that is true exactly of the well-orderings less than or equal to ω^α. Show that there is no such $\Sigma^{\mathtt{in}}_{2\alpha+1}$ sentence.

Exercise II.44. ([**Ash86a**, Lemma 7]) This exercise provides a complete description of the back-and-forth relations on ordinals. Given different ordinals β and γ, decompose them as follows:

$$\beta = \omega^\alpha \beta_1 + \delta, \quad \& \quad \gamma = \omega^\alpha \gamma_1 + \delta,$$

where $\beta_1, \gamma_1 \neq 0$, $\delta < \omega^\alpha$, and α is the largest for which such a decomposition exists. To find such a decomposition, one needs to look for the rightmost term in the Cantor normal forms of β and γ that is different. Prove:

(a) Let m and n be the remainders of β_1 and γ_1 in the division over ω. (I.e., $\beta_1 = \omega \cdot \beta_2 + m$, and $\gamma_1 = \omega \cdot \gamma_2 + n$.) Prove that either $|\beta_1| \neq |\gamma_1|$ or $m \neq n$, where $|\beta|$ represents the size of β, that is, $|\beta| = \beta$ if $\beta < \omega$ and $|\beta| = \infty$ if $\beta \geq \omega$.

(b) $\beta \leq_{2\alpha+1} \gamma$ if and only if $|\beta_1| \geq |\gamma_1|$.

(c) If β_1 and γ_1 are both infinite, then $\beta \leq_{2\alpha+2} \gamma$ if and only if $n \geq m$.

Goncharov, Harizanov, Knight, McCoy, and R. Miller [**GHK$^+$05**] proved that

$$\mathbb{Z}^\alpha \cdot \omega \equiv_{2\alpha+1} \mathbb{Z}^\alpha \cdot \omega^*, \quad \text{but} \quad \mathbb{Z}^\alpha \cdot \omega \not\leq_{2\alpha+2} \mathbb{Z}^\alpha \cdot \omega^*,$$

and gave a complete analysis of the back-and-forth tuples within these structures.

Exercise II.45. Prove that the Scott rank of $\mathbb{Z}^\alpha \cdot \mathcal{A}$ is 2α plus the Scott rank of $\mathcal{A}$.

Exercise II.46. Show that the parameterless Scott rank of an ordinal δ is either $2\log_\omega(\delta)$ or $2\log_\omega(\delta) + 1$, depending on whether the Cantor normal form of δ starts with only one copy of $\omega^{\log_\omega(\delta)}$ and then continues with smaller terms, or starts with at least two copies of $\omega^{\log_\omega(\delta)}$.

Exercise II.47. What are the possible parametrized and parameterless Scott ranks of equivalence structures?

[*] Show that if a $\Pi^{\mathtt{in}}_{<2\alpha}$ formula is true about some tuple in ω^α, then it also true of some tuple inside a smaller ordinal.

Exercise II.48. What are the possible parametrized and parameterless Scott ranks of $\mathbb{Q}$-vector spaces?

II.6.2. Σ^0_1- and Σ^0_2-hardness. As we mentioned before, another way of defining the back-and-forth relations is in terms of how difficult it is to distinguish the ω-presentations of one structure from the ω-presentations of the other. In this section we work out for the cases of $\leq_1$ and $\leq_2$, as to give the reader an idea of how these work. To get the general characterization of $\leq_\alpha$ we will need to use the technique of forcing for the boldface case, which we will see in Chapter VII (Theorem VII.30), and to use the technique of iterated true stages for the lightface case, which we will see in Chapter IX (Theorems VIII.7 and IX.10).

Let us start with the $\leq_1$-case.

Definition II.49. Given computable ω-presentations $\mathcal{A}$ and $\mathcal{B}$, we say that *distinguishing $\mathcal{A}$ from $\mathcal{B}$ is Σ^0_1-hard* if there is a computable operator Γ such that, for all $X \in 2^{\mathbb{N}}$, Γ^X is an ω-presentation of a structure, and

$$\Gamma^X \cong \begin{cases} \mathcal{A} & \text{if } X \neq 0^\infty \\ \mathcal{B} & \text{if } X = 0^\infty, \end{cases}$$

where 0^∞ is the infinite sequence of all zeros.

Theorem II.50. *Let $\mathcal{A}_0$ and $\mathcal{A}_1$ be computable ω-presentations such that $\mathcal{A}_1 \leq_1 \mathcal{A}_0$. Then distinguishing $\mathcal{A}_1$ from $\mathcal{A}_0$ is Σ^0_1-hard.*

Proof. Given $X \in 2^{\mathbb{N}}$, we build a structure $\mathcal{B}^X$ uniformly computable from X such that $\mathcal{B}^X \cong \mathcal{A}_1$ if X contains some 1 and $\mathcal{B}^X \cong \mathcal{A}_0$ if X is always 0. Let $\widehat{X} \in 2^{\mathbb{N}}$ be defined as follows:

$$\widehat{X}(s) = 1 \iff \exists t \leq s \ \ (X(t) = 1).$$

Thus, if $X = 0^\infty$, then $\widehat{X} = 0^\infty$, and if $X \neq 0^\infty$, then $\widehat{X}$ starts with a few zeros, and then it is all ones from some point on. At each stage s we will define a finite tuple $\bar{a}_s$ of length s that belongs to $\mathcal{A}_0$ if $\widehat{X}(s) = 0$, and to $\mathcal{A}_1$ if $\widehat{X}(s) = 1$. Let $\mathcal{B}_s$ be the finite structure with domain $\{0, ..., s-1\}$ obtained as the pull back of $\bar{a}_s$. That is $D(\mathcal{B}_s) = D_{\mathcal{A}_{\widehat{X}(s)}}(\bar{a}_s)$. We will define $\mathcal{B}^X$ as the limit of the structures $\mathcal{B}_s$, so we need to ensure they are nested. Equivalently, we need to ensure that, for all $s \in \mathbb{N}$,

$$D_{\mathcal{A}_{\widehat{X}(s)}}(\bar{a}_s) \subseteq D_{\mathcal{A}_{\widehat{X}(s+1)}}(\bar{a}_{s+1}).$$

We will end up with

$$D(\mathcal{B}^X) = \bigcup_s D(\mathcal{B}_s) = \bigcup_s D_{\mathcal{A}_{\widehat{X}(s)}}(\bar{a}_s).$$

In the construction, while $\widehat{X}(s) = 0$, we let $\bar{a}_s$ consist of the first s elements of $\mathcal{A}_0$. This way, if $X = 0^\infty$, we will end up with $\mathcal{B}^X = \mathcal{A}_0$ as wanted.

Otherwise, there is a point at which $\widehat{X}$ changes its value from $\widehat{X}(s) = 0$ to $\widehat{X}(s+1) = 1$, and we need to switch and start taking tuples from $\mathcal{A}_1$ instead of $\mathcal{A}_0$. Since $\mathcal{A}_1 \leq_1 \mathcal{A}_0$, there is a tuple $\bar{b} \in A_1^s$ so that $(\mathcal{A}_1, \bar{b}) \geq_0 (\mathcal{A}_0, \bar{a}_s)$, i.e., $D_{\mathcal{A}_0}(\bar{a}_s) = D_{\mathcal{A}_1}(\bar{b})$. Define $\bar{a}_{s+1}$ to be $\bar{b}$, together with some extra element from the domain of $\mathcal{A}_1$ so that it has length $s+1$. From this point on we have that $\widehat{X}$ is all ones. When we have $\widehat{X}(s) = \widehat{X}(s+1) = 1$, we define $\bar{a}_{s+1}$ extending $\bar{a}_s$ in $\mathcal{A}_1$ making sure all elements of $\mathcal{A}_1$ will eventually show up. This way, we will end up with $\mathcal{B}^X \cong \mathcal{A}_1$. □

Let us now consider the $\leq_2$ case.

Definition II.51. Given computable ω-presentations $\mathcal{A}$ and $\mathcal{B}$, we say that *distinguishing $\mathcal{A}$ from $\mathcal{B}$ is Σ_2^0-hard* if there is a computable operator Γ such that, for all $X \in 2^{\mathbb{N}}$, Γ^X is an ω-presentation of a structure and

$$\Gamma^X \cong \begin{cases} \mathcal{A} & \text{if } X \text{ has finitely many zeros} \\ \mathcal{B} & \text{if } X \text{ has infinitely many zeros.} \end{cases}$$

Theorem II.52. *Let $\mathcal{A}_0$ and $\mathcal{A}_1$ be computable ω-presentations such that $\mathcal{A}_1 \leq_2 \mathcal{A}_0$, and such that the set*

$$\{\langle \bar{a}, \bar{b} \rangle \in A_0^{<\mathbb{N}} \times A_1^{<\mathbb{N}} : (\mathcal{A}_0, \bar{a}) \leq_1 (\mathcal{A}_1, \bar{b})\}$$

is c.e. Then distinguishing $\mathcal{A}_1$ from $\mathcal{A}_0$ is Σ_2^0-hard.

Proof. Given $X \in 2^{\mathbb{N}}$, we build a structure $\mathcal{B}^X$ uniformly computable from X such that $\mathcal{B}^X \cong \mathcal{A}_0$ if X has infinitely many zeros and $\mathcal{B}^X \cong \mathcal{A}_1$ if X is always one from some point on. As above, at each stage s we will define a tuple $\bar{a}_s \in A_{X(s)}$ of length s satisfying $D_{\mathcal{A}_{X(s-1)}}(\bar{a}_{s-1}) \subseteq D_{\mathcal{A}_{X(s)}}(\bar{a}_s)$. As above, we can then define $\mathcal{B}^X$ as the limit of a nested sequence of finite structures $\mathcal{B}_s$ defined as the pull-back through $\bar{a}_s$ of $\mathcal{A}_{X(s)}$. We end up with

$$D(\mathcal{B}^X) = \bigcup_s D_{\mathcal{A}_{X(s)}}(\bar{a}_s).$$

In the case when there is a point after which $X(s)$ is always 1, if we ensure that after that point we have $\bar{a}_s \subseteq \bar{a}_{s+1}$, we will end up with $\mathcal{B}^X \cong \mathcal{A}_1$. This will follow from the case $X(s-1) = X(s)$ in the construction below.

In the case when X has infinitely many zeros, to ensure that the limit structure is isomorphic to $\mathcal{A}_0$, we will require that

- if $t < s$ and $X(t) = X(s) = 0$, then $\bar{a}_t \subseteq \bar{a}_s \in \mathcal{A}_0^{<\mathbb{N}}$.

We will then have that $g = \bigcup_{s:X(s)=0} \bar{a}_s$ is a bijection from $\mathcal{B}^X$ to $\mathcal{A}_0$. Suppose that we have $X(t) = X(s) = 0$ and that $X(r) = 1$ for all r between t and s. Suppose we have already defined $\bar{a}_{s-1} \in A_1^{<\mathbb{N}}$ and we want to define $\bar{a}_s$. We need to find $\bar{a}_s \in A_0$ extending $\bar{a}_t$ and satisfying that $D_{\mathcal{A}_1}(\bar{a}_{s-1}) \subseteq D_{\mathcal{A}_0}(\bar{a}_s)$. What condition would guarantee the existence of such $\bar{a}_s$? The answer is: $(\mathcal{A}_0, \bar{a}_t) \leq_1 (\mathcal{A}_1, \bar{a}_{s-1})$. So let us impose that condition too:

- if $t < r$, $X(t) = 0$ and $X(r) = 1$, then $(\mathcal{A}_0, \bar{a}_t) \leq_1 (\mathcal{A}_1, \bar{a}_r)$.

This brings an extra burden in the construction of $\bar{a}_r$. The condition that guarantees the existence of such $\bar{a}_r$ is $\mathcal{A}_0 \geq_2 \mathcal{A}_1$.

Let us be more precise in the construction of $\bar{a}_s$:

(1) If $X(s-1) = X(s)$, let $\bar{a}_s$ extend $\bar{a}_{s-1}$ by adding one element, say the least element in the ω-presentation $\mathcal{A}_{X(s)}$ that is not already in $\bar{a}_{s-1}$.
(2) If $X(s) = 1$ and $X(s-1) = 0$, define $\bar{a}_s \in \mathcal{A}_1^s$ so that $(\mathcal{A}_0, \bar{a}_{s-1}) \leq_1 (\mathcal{A}_1, \bar{a}_s)$. The existence of such $\bar{a}_s$ follows from the hypothesis that $\mathcal{A}_0 \geq_2 \mathcal{A}_1$. We can find $\bar{a}_s$ computably by our effectiveness condition on $\leq_1$.
(3) If $X(s) = 0$ and $X(s-1) = 1$, let $t < s-1$ be the largest with $X(t) = 0$. Because of the way we have carried out the constructions far, we have that
$$\bar{a}_t \leq_1 \bar{a}_{t+1} \subseteq \bar{a}_{t+2} \subseteq \cdots \subseteq \bar{a}_{s-1}.$$
Thus $(\mathcal{A}_0, \bar{a}_t) \leq_1 (\mathcal{A}_1, \bar{a}_{s-1})$. We then know there exists $\bar{a}_s \in \mathcal{A}_0^s$ with $\bar{a}_s \supseteq \bar{a}_t$ and $(\mathcal{A}_0, \bar{a}_s) \geq_0 (\mathcal{A}_1, \bar{a}_{s-1})$.

This finishes the construction of the sequence of tuples $\langle \bar{a}_s : s \in \mathbb{N} \rangle$ satisfying the conditions we imposed above. □

II.7. Scott sentence complexity

The Scott rank of a structure was defined in Section II.3 as a measure of the complexity of the automorphism orbits of tuples in the structure. We then saw in Proposition II.26 that the Scott rank also measures the complexity of the Scott sentences for the structure: A structure is $\Sigma^{\mathtt{in}}_{\alpha}$-atomic over parameters (i.e., it has Scott rank α) if and only if it has a $\Sigma^{\mathtt{in}}_{\alpha+2}$-Scott sentence. The former is a measure of complexity from within, measuring the difficulty of distinguishing tuples within the structure. The latter is a measure of complexity from the outside, measuring the difficulty of distinguishing the structure from other structures.

In this section, we analyze the second approach further and look for the simplest Scott sentences. We will see that when the Scott rank of a structure is a successor ordinal, using the parameterless Scott rank and the parametrized Scott rank of a structure we can deduce its Scott-sentence complexity, and vice versa, as in Table 1 below. If the Scott rank of a structure is a limit ordinal, we get new interesting cases.

We use d-$\Sigma^{\mathtt{in}}_{\alpha}$ to denote $\Sigma^{\mathtt{in}}_{\alpha} \wedge \Pi^{\mathtt{in}}_{\alpha}$, that is, the class of formulas of the form $\varphi \wedge \psi$, where φ is $\Sigma^{\mathtt{in}}_{\alpha}$ and ψ is $\Pi^{\mathtt{in}}_{\alpha}$. (The '$d$' is for difference, as these formulas can be viewed as the difference of two $\Sigma^{\mathtt{in}}_{\alpha}$ formulas.) As we will see in Theorem II.57 below, if a structure has both a $\Sigma^{\mathtt{in}}_{\alpha+1}$ Scott sentence and a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence, then it has a d-$\Sigma^{\mathtt{in}}_{\alpha}$ Scott sentence.

Remark II.53. Alvir and Harrison-Trainor [**AGHTT**] showed that the Wadge degree of the set of ω-presentations of a structure can only be $\mathbf{\Sigma}^0_{\alpha}$, $\mathbf{\Pi}^0_{\alpha}$, or d-$\mathbf{\Sigma}^0_{\alpha}$. Alvir, Greenberg, Harrison-Trainor, and Turetsky [**AGHTT**] do a deep analysis of the landscape of Scott sentence complexities.

Definition II.54. The *Scott-sentence complexity* of a structure $\mathcal{A}$ is the complexity of the simplest Scott sentence for $\mathcal{A}$, which could be $\Sigma^{\mathtt{in}}_{\alpha}$, $\Pi^{\mathtt{in}}_{\alpha}$, or d-$\Sigma^{\mathtt{in}}_{\alpha}$ for some ordinal α.

Scott sentence complexity	parametrized Scott rank	parameterless Scott rank	complexity of parameters
$\Sigma^{\mathtt{in}}_{\alpha+2}$	α	$\alpha+2$	$\Pi^{\mathtt{in}}_{\alpha+1}$
d-$\Sigma^{\mathtt{in}}_{\alpha+1}$	α	$\alpha+1$	$\Pi^{\mathtt{in}}_{\alpha}$
$\Pi^{\mathtt{in}}_{\alpha+1}$	α	α	none
α limit ordinal:			
$\Sigma^{\mathtt{in}}_{\alpha+1}$	α	$\alpha+1$	$\Pi^{\mathtt{in}}_{\alpha}$
$\Pi^{\mathtt{in}}_{\alpha}$	α	α	none

Table 1. This table shows all the possible Scott-sentence complexities for structures of Scott rank α. The first three rows are for all $\alpha \geq 1$, and the last two lines occur only when α is a limit ordinal. All these cases are attainable. The left column reflects the Scott-sentence complexity, the second column the Scott rank, the third column the parameterless Scott rank, and the last column the complexity of the parameters over which the structure is $\Sigma^{\mathtt{in}}_{\alpha}$-atomic.

Let us start by ruling out a few cases. Harrison-Trainor [**AGHTT**] (and previously Arnold Miller [**Mil83**] for relational languages only) showed that no infinite structure has a $\Sigma_2^{\mathtt{in}}$ Scott sentence.

Finite structures have d-$\Sigma_1^{\mathtt{in}}$ Scott sentences, but we will not worry about them. Thus, the simplest Scott-sentence complexity of an infinite structure is $\Pi_2^{\mathtt{in}}$, which is the Scott-sentence complexity of $\exists$-atomic structures [**Part 1**, Theorem III.34]. We can also rule out $\Sigma_\alpha^{\mathtt{in}}$ and d-$\Sigma_\alpha^{\mathtt{in}}$ for limit ordinals α as possible Scott-sentence complexities: This is because if a structure satisfies a $\Sigma_\alpha^{\mathtt{in}}$ formula, it must satisfy one of its disjuncts which is $\Sigma_\beta^{\mathtt{in}}$ for some $\beta < \alpha$. Therefore, if a structure has a $\Sigma_\alpha^{\mathtt{in}}$ Scott sentence, it has a simpler one. Also, if a structure has a d-$\Sigma_\alpha^{\mathtt{in}}$ Scott sentence, the $\Sigma_\alpha^{\mathtt{in}}$-conjunct could be simplified to $\Sigma_\beta^{\mathtt{in}}$ for some $\beta < \alpha$, and hence the structure would have a $\Pi_\alpha^{\mathtt{in}}$ Scott sentence. All other Scott-sentence complexities are attainable — we will give examples or references below.

Suppose we have a structure $\mathcal{A}$ of Scott rank α. We dedicate the rest of this section to analyzing the possible Scott-sentence complexities of $\mathcal{A}$. We know from Proposition II.26 that $\mathcal{A}$ has a $\Sigma_{\alpha+2}^{\mathtt{in}}$ Scott sentence and no $\Sigma_{\beta+2}^{\mathtt{in}}$ Scott sentence for any $\beta < \alpha$. This does not say anything about whether $\mathcal{A}$ has a $\Pi_{\alpha+1}^{\mathtt{in}}$ or $\Pi_{\alpha+2}^{\mathtt{in}}$ Scott sentence, which, as we will see, will depend on the complexity of the parameters over which $\mathcal{A}$ is $\Sigma_\alpha^{\mathtt{in}}$-atomic. Also, when α is a limit ordinal, this does not rule out $\mathcal{A}$ having a $\Sigma_{\alpha+1}^{\mathtt{in}}$ Scott sentence and still having Scott rank α.

Let $\bar{p} \in A^{<\mathbb{N}}$ be such that $(\mathcal{A}, \bar{p})$ is $\Sigma_\alpha^{\mathtt{in}}$-atomic. The first observation is that the orbit of these parameters must be $\Pi_{\alpha+1}^{\mathtt{in}}$-definable: We know that $(\mathcal{A}, \bar{p})$ has a $\Pi_{\alpha+1}^{\mathtt{in}}$ Scott sentence $\varphi(\bar{p})$ (Proposition II.26), and hence $\varphi(\bar{x})$ is a $\Pi_{\alpha+1}^{\mathtt{in}}$ formula defining the automorphism orbit of $\bar{p}$. Let us now consider three cases depending on the complexity of these parameters.

Case 1. *The automorphism orbit of $\bar{p}$ is not $\Sigma_{\alpha+1}^{\mathtt{in}}$ definable.* In this case we know from Proposition II.26 that $\mathcal{A}$ does not have a $\Pi_{\alpha+2}^{\mathtt{in}}$ Scott sentence, and hence its Scott-sentence complexity must be $\Sigma_{\alpha+2}^{\mathtt{in}}$. Here is an example.

Exercise II.55. (due to A. Miller) Show that the adjacency linear ordering $(2 \cdot \mathbb{Q} + 1 + \mathbb{Q}; \leq, \mathsf{Adj})$ has Scott rank 1 and Scott-sentence complexity $\Sigma_3^{\mathtt{in}}$. See hint in footnote.[†]

Case 2. *The automorphism orbit of $\bar{p}$ is $\Sigma_{\alpha+1}^{\mathtt{in}}$ definable, but not $\Sigma_\alpha^{\mathtt{in}}$ definable.* As we prove below, in this case, the structure must

[†] Show that it is $\exists$-atomic over the middle '1,' but that the middle '1' is not $\Sigma_2^{\mathtt{in}}$-definable as it is $\geq_2$ all elements to its right.

have a d-$\Sigma^{\mathtt{in}}_{\alpha+1}$ Scott sentence. We know from Proposition II.26 that $\mathcal{A}$ does not have a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence. This implies that the Scott-sentence complexity must be either d-$\Sigma^{\mathtt{in}}_{\alpha+1}$ or $\Sigma^{\mathtt{in}}_{\alpha+1}$. When α is a successor ordinal, the latter case would imply that the structure has Scott rank $\alpha-1$, and hence the only possibility is to have Scott-sentence complexity d-$\Sigma^{\mathtt{in}}_{\alpha+1}$.

Exercise II.56. Show that $\omega^\alpha + \omega^\alpha$ has Scott sentence complexity d-$\Sigma^{\mathtt{in}}_{2\alpha+1}$. See hint in footnote.[‡]

When α is a limit ordinal, an example of a structure with Scott-sentence complexity $\Sigma^{\mathtt{in}}_{\alpha+1}$ was recently built by Alvir, Greenberg, Harrison-Trainor, and Turetsky [**AGHTT**].

Theorem II.57 (A. Miller [**Mil83**], D.Miller [**Mil78**]). *Let $\mathcal{A}$ be a structure and α an ordinal. The following are equivalent:*

(1) *$\mathcal{A}$ has both a $\Sigma^{\mathtt{in}}_{\alpha+2}$ Scott sentence and a $\Pi^{\mathtt{in}}_{\alpha+2}$ Scott sentence.*
(2) *$\mathcal{A}$ is $\Sigma^{\mathtt{in}}_{\alpha}$-atomic over a tuple of parameters whose automorphism orbit is $\Sigma^{\mathtt{in}}_{\alpha+1}$-definable.*
(3) *$\mathcal{A}$ has a d-$\Sigma^{\mathtt{in}}_{\alpha+1}$ Scott sentence.*

Proof. (Alvir [**AKM20**]) Start by assuming (1) and let us prove (2). Since $\mathcal{A}$ has a $\Sigma^{\mathtt{in}}_{\alpha+2}$ Scott sentence, $\mathcal{A}$ must be $\Sigma^{\mathtt{in}}_{\alpha}$-atomic over some tuple of parameters. Since $\mathcal{A}$ also has a $\Pi^{\mathtt{in}}_{\alpha+2}$ Scott sentence, the automorphism orbit of every tuple is definable by a $\Sigma^{\mathtt{in}}_{\alpha+1}$ formula as in Proposition II.26.

Let us now assume (2) and prove (3). Let $\bar{p}$ be the parameters over which $\mathcal{A}$ is Σ_α-atomic. Let $\varphi(\bar{p})$ be a $\Pi_{\alpha+1}$ Scott sentence for $(\mathcal{A}, \bar{p})$, and let $\gamma(\bar{x})$ be a $\Sigma^{\mathtt{in}}_{\alpha+1}$ formula defining the automorphism orbit of $\bar{p}$. The following formula is a $\Sigma^{\mathtt{in}}_{\alpha+1} \wedge \Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence for $\mathcal{A}$:

$$\exists \bar{x} \gamma(\bar{x}) \quad \wedge \quad \forall \bar{x}(\gamma(\bar{x}) \rightarrow \varphi(\bar{x})).$$

To see that this is a Scott sentence for $\mathcal{A}$, suppose it is true about $\mathcal{B}$. Let $\bar{b} \in B^{<\mathbb{N}}$ be such that $\mathcal{B} \models \gamma(\bar{b})$. We then have that $(\mathcal{B}, \bar{b}) \models \varphi(\bar{b})$, and hence that $(\mathcal{A}, \bar{p}) \cong (\mathcal{B}, \bar{b})$.

The implication (3) ⇒ (1) is straightforward. □

If we keep on simplifying the parameters, the next step is when the parameters are $\Pi^{\mathtt{in}}_{\alpha}$, which turns out to be equivalent to the case above.

Lemma II.58. *In the theorem above, we have a fourth equivalent statement*

(4) *$\mathcal{A}$ is $\Sigma^{\mathtt{in}}_{\alpha}$-atomic over a tuple of parameters whose automorphism orbit is $\Pi^{\mathtt{in}}_{\alpha}$-definable.*

[‡] Use Exercise II.42 on α-limits.

Proof. It is clear that (4) implies (2). For the converse, assume the statements in the theorem are true about $\mathcal{A}$. We then have that $\mathcal{A}$ is Σ_α-atomic over a tuple $\bar{p}$ of parameters which is $\Sigma^{\mathtt{in}}_{\alpha+1}$ definable. If an automorphism is $\Sigma^{\mathtt{in}}_{\alpha+1}$ definable, one of the disjuncts must be true about the tuple, and hence that disjunct must define its automorphism orbit too. We thus have that $\bar{p}$ is definable by a formula of the form $\exists \bar{y}\ \gamma(\bar{x}, \bar{y})$, where γ is $\Pi^{\mathtt{in}}_\alpha$. Let $\bar{b} \in A^{<\mathbb{N}}$ be a witness for $\mathcal{A} \models \gamma(\bar{p}, \bar{b})$. Recall that since every automorphism orbit is definable by a $\Sigma^{\mathtt{in}}_\alpha$ formula over $\bar{p}$, so is every automorphism invariant relation (as an automorphism invariant relation is a union of automorphism orbits). Taking complements, we get that all automorphism invariant sets are $\Pi^{\mathtt{in}}_\alpha$ definable, and in particular so are all automorphism orbits. We thus get that the automorphism orbit of $\bar{b}$ is $\Pi^{\mathtt{in}}_\alpha$ definable over $\bar{p}$; let $\delta(\bar{x}, \bar{y})$ be such that if $\mathcal{A} \models \delta(\bar{p}, \bar{b}')$, then $\bar{b}'$ is automorphic to $\bar{b}$ via some automorphism that fixes $\bar{p}$. We claim now that the automorphism orbit of $\bar{p}\bar{b}$ is $\Pi^{\mathtt{in}}_\alpha$ definable without parameters by the formula $\gamma(\bar{x}, \bar{y}) \wedge \delta(\bar{x}, \bar{y})$. This would finish the proof of the theorem because $\mathcal{A}$ is $\Sigma^{\mathtt{in}}_\alpha$-atomic over $\bar{p}\bar{b}$. To prove the claim, suppose $\mathcal{A} \models \gamma(\bar{p}', \bar{b}') \wedge \delta(\bar{p}', \bar{b}')$ — we need to show that $\bar{p}'\bar{b}'$ is automorphic to $\bar{p}\bar{b}$. First, since $\mathcal{A} \models \exists \bar{y}\ \gamma(\bar{p}', \bar{y})$, we get that $\bar{p}$ and $\bar{p}'$ are automorphic. Let $\bar{b}''$ be the tuple matching $\bar{b}'$ under this automorphism so that $\bar{p}'\bar{b}'$ is automorphic to $\bar{p}\bar{b}''$. Then, since $\mathcal{A} \models \delta(\bar{p}', \bar{b}')$, we also have that $\mathcal{A} \models \delta(\bar{p}, \bar{b}'')$ and then that $\bar{p}\bar{b}''$ is automorphic to $\bar{p}\bar{b}$. □

Case 3. *The orbit of the parameters $\bar{p}$ is $\Sigma^{\mathtt{in}}_\alpha$ definable.* In that case, all orbits would be $\Sigma^{\mathtt{in}}_\alpha$ definable without parameters, and $\mathcal{A}$ would be Σ_α-atomic over no parameters. Thus, $\mathcal{A}$ would have a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence. In the case when α is a successor ordinal, $\mathcal{A}$ does not have a $\Sigma^{\mathtt{in}}_{\alpha+1}$ Scott sentence, as otherwise it would have Scott rank $\alpha - 1$, so the Scott sentence complexity must be $\Pi^{\mathtt{in}}_{\alpha+1}$. An example of a structure of Scott sentence complexity $\Pi^{\mathtt{in}}_{2\alpha+1}$ is the linear ordering ω^α (see Exercise II.46). In the cases when α is a limit ordinal, the structure could have Scott sentence complexity $\Pi^{\mathtt{in}}_\alpha$. (Recall that d-$\Sigma^{\mathtt{in}}_\alpha$ and $\Sigma^{\mathtt{in}}_\alpha$ are not possible.) An example of a structure with Scott-sentence complexity $\Pi^{\mathtt{in}}_\alpha$ for limit α is given by the disjoint union of structures of Scott ranks α_n, where $\sup_{n\in\mathbb{N}} \alpha_n = \alpha$, having unary predicates to distinguish the domains of the different structures.

Historical Remark II.59. *Most of the analysis of the last section was also done independently and almost simultaneously by Rachel Alvir* [**Alv**].

The original proof of Theorem II.57 by A. Miller [**Mil83**] *was no more than an observation using D. Miller's descriptive set theoretic result* [**Mil78**] *that when we have a Polish group acting continuously on a Polish space, two disjoint $\Pi^0_{\alpha+1}$ invariant sets of reals can be separated by a countable union of invariant $\Sigma^0_\alpha \wedge \Pi^0_\alpha$ sets of reals. A. Miller's paper* [**Mil83**] *analyzes which Scott-sentence complexities are possible by studying the Borel complexity of the sets of ω-presentations. A. Miller also proves that $\Sigma^{\mathtt{in}}_2$ is not a possible Scott-sentence complexity when the vocabulary is relational. Matthew Harrison-Trainor* [**AGHTT**] *then proves this for all vocabularies. A. Miller shows $\Pi^{\mathtt{in}}_\omega$ is a possible Scott-sentence complexity, and claims his proof can be extended to $\Pi^{\mathtt{in}}_\lambda$ for all limit ordinals λ, but it is not clear how to do that. However, our construction above (due to Harrison-Trainor) easily works for all λ. A. Miller left open whether $\Sigma^{\mathtt{in}}_{\lambda+1}$ for λ limit is a possible Scott sentence complexity. Alvir, Greenberg, Harrison-Trainor and Turetsky's have recently shown it is* [**AGHTT**].

The proof of Theorem II.57 given above is quite recent and due to Rachel Alvir [**AKM20**]. *In that paper, they also prove a computability theoretic version: If $\mathcal{A}$ has both a computable $\Sigma_{\alpha+1}$ and a computable $\Pi_{\alpha+1}$ Scott sentence, then it has a computable $\Sigma_\alpha \wedge \Pi_\alpha$ one.*

The most comprehensive analysis of the Scott sentence complexity of structures within a class of structures is Alvir and Rossegger study of scattered linear orderings [**AR**].

II.8. The Löwenheim-Skolem theorem

We say that an $\mathcal{L}_{\omega_1,\omega}$ sentence is *satisfiable* if it is true in some structure. In finitary first-order logic, this is equivalent to being consistent. Versions of this equivalence have been proved for infinitary logic once the correct notion of infinitary proof is defined. We will not get into infinitary proofs in this book — the interested reader may consult [**Bar75**, Chapter III]. However, we would still like to understand the complexity of the satisfiability predicate. As we defined it, it uses an existential quantifier over models of arbitrary size — this is way too complex for us. Fortunately, the Löwenheim-Skolem theorem from finitary first-order logic works for infinitary logic too, as we will see below. This implies that an $\mathcal{L}_{\omega_1,\omega}$ sentence is satisfiable if and only if it is true in some countable structure, which will allow us to conclude that the satisfiability predicate is Σ^1_1.

Lemma II.60 (Vaught's criterion). *Let Ψ be a set of $\mathcal{L}_{\omega_1,\omega}$ formulas closed under taking sub-formulas. Consider structures $\mathcal{A} \subseteq \mathcal{B}$ such that, for every $\psi(\bar{x}, y) \in \Psi$ and $\bar{a} \in A^{|\bar{x}|}$, if $\mathcal{B} \models \exists y\ \psi(\bar{a}, y)$, then there exists a $c \in A$ such that $\mathcal{B} \models \psi(\bar{a}, c)$. Then, for every $\theta(\bar{x}) \in \Psi$ and $\bar{a} \in A^{|\bar{x}|}$, $\mathcal{A} \models \theta(\bar{a}) \iff \mathcal{B} \models \theta(\bar{a})$.*

Proof. The proof is by induction on formulas the same way as in the standard proof of Vaught's criterion. The only difference is that now we need to use well-founded induction. For atomic formulas this is immediate. For negations too. For infinitary conjunctions, apply the inductive hypothesis to each conjunct. Do the same for infinitary disjunctions. Lastly, given an existential formula $\exists y\ \psi(\bar{a}, y)$ and $\bar{a} \in A^{<\mathbb{N}}$, $\mathcal{B} \models \exists y\ \psi(\bar{a}, y)$ if and only if $\mathcal{B} \models \psi(\bar{a}, c)$ for some $c \in A$ by our assumptions on $\mathcal{A}$ and $\mathcal{B}$, which by the induction hypothesis holds if and only if $\mathcal{A} \models \psi(\bar{a}, c)$ for some $c \in A$, which is equivalent to $\mathcal{A} \models \exists y\ \psi(\bar{a}, y)$. For universal formulas, negate existential ones. □

Theorem II.61 (Löwenheim-Skolem). *If an $\mathcal{L}_{\omega_1,\omega}$ sentence is satisfied in some model of any cardinality, then it is satisfied in a countable structure.*

Proof. Let θ be an $\mathcal{L}_{\omega_1,\omega}$ sentence and $\mathcal{B}$ an uncountable model of θ. Let Ψ be the set of all sub-formulas of θ, including θ itself. We will build a countable sub-structure $\mathcal{A}$ of $\mathcal{B}$ satisfying Vaught's criterion for Ψ. Since $\theta \in \Psi$ and $\mathcal{B} \models \theta$, this will imply that $\mathcal{A} \models \theta$.

The construction of $\mathcal{A}$ is a standard closure argument. Let A_0 be the countable sub-structure of $\mathcal{B}$ generated by the constants in the vocabulary τ. Given A_n, we define A_{n+1} with $A_n \subseteq A_{n+1} \subseteq B$ by first adding a witness $c \in B$ for each formula $\psi(\bar{x}, y) \in \Psi$ and tuple $\bar{a} \in A_n^{|\bar{x}|}$ such that $\mathcal{B} \models \exists y\ \psi(\bar{a}, y)$ and then closing under the functions of the vocabulary to obtain a sub-structure $\mathcal{A}_{n+1}$. Note that since A_n and Ψ are countable, we are adding at most countably many witnesses, keeping $\mathcal{A}_{n+1}$ countable. Finally, let $A = \bigcup_{n \in \mathbb{N}} A_n$, and observe that the sub-structure $\mathcal{A}$ of $\mathcal{B}$ with domain A satisfies the hypothesis of Vaught's criterion for Ψ, and hence satisfies θ. □

II.9. Scott rank via back-and-forth relations

In this last section we will see how the Scott rank can be defined in terms of the back-and-forth relations using the notion of α-free tuple. This will allow us to calculate the Scott rank of a structure if we know how to calculate the back-and-forth relations on it. This section is a bit technical, so some readers may want to skip it. We will use results from this section in Theorem VII.21 later in the book.

As we mentioned before, there are various non-equivalent definitions of Scott rank in the literature. Most of them are defined out of some notion of back-and-forth relation, of which there are also various non-equivalent definitions. The closest definition to ours is from Ash and Knight [**AK00**, Section 6.7], who use the same back-and-forth relations we use but a slightly different definition of rank. They define $r(\mathcal{A})$ to be the least α for which the relation $\leq_\alpha$ coincides with the automorphism relation on $\mathcal{A}$, that is, the least α such that, for all $\bar{a}, \bar{b} \in \mathcal{A}^{<\mathbb{N}}$, $\bar{a} \leq_\alpha \bar{b}$ implies $\bar{a} \cong \bar{b}$. We will prove below that

$$r(\mathcal{A}) \quad \leq \quad \mathrm{SR}_{\text{p-less}}(\mathcal{A}) \quad \leq \quad r(\mathcal{A}) + 1$$

for all structures $\mathcal{A}$, where $\mathrm{SR}_{\text{p-less}}(\mathcal{A})$ denotes the parameterless Scott rank of $\mathcal{A}$.

We start with a lemma that shows that all $\Pi^{\mathtt{in}}_\alpha$ types realized in a structure $\mathcal{A}$ are $\Pi^{\mathtt{in}}_\alpha$-principal within the structure.

Lemma II.62. *For every $\bar{a} \in A^{<\mathbb{N}}$ and every ordinal α, there is a $\Pi^{\mathtt{in}}_\alpha$ formula $\varphi_{\bar{a}}(\bar{x})$ true about $\bar{a}$ which, within $\mathcal{A}$, implies all other $\Pi^{\mathtt{in}}_\alpha$ formulas true about $\bar{a}$. In other words*

$$\mathcal{A} \models \forall \bar{x} \quad \big(\varphi_{\bar{a}}(\bar{x}) \quad \leftrightarrow \bigwedge\!\!\!\bigwedge_{\psi \in \Pi^{\mathtt{in}}_\alpha\text{-}tp_{\mathcal{A}}(\bar{a})} \psi(\bar{x})\big),$$

or equivalently, for all $\bar{b} \in A^{|\bar{a}|}$,

$$\mathcal{A} \models \varphi_{\bar{a}}(\bar{b}) \qquad \Longleftrightarrow \qquad \bar{a} \leq_\alpha \bar{b}.$$

Proof. About the equivalence of the last two statements, recall from Theorem II.36 that $\bar{a} \leq_\alpha \bar{b}$ if and only if $\bar{b}$ satisfies all the formulas in $\Pi^{\mathtt{in}}_\alpha$-$tp_{\mathcal{A}}(\bar{a})$.

We know from Theorem II.36 that for every $\bar{c} \in A^{|\bar{a}|}$ with $\bar{a} \not\leq_\alpha \bar{c}$ there is a $\Pi^{\mathtt{in}}_\alpha$ formula $\psi_{\bar{c}}(\bar{x})$ true about $\bar{a}$, false about $\bar{c}$. It follows that

$$\bigwedge\!\!\!\bigwedge_{\substack{\bar{c} \in A^{|\bar{a}|} \\ \bar{a} \not\leq_\alpha \bar{c}}} \psi_{\bar{c}}(\bar{x})$$

is true about $\bar{a}$ and false about any $\bar{c} \not\geq_\alpha \bar{a}$. Since this formula is $\Pi^{\mathtt{in}}_\alpha$, again by Theorem II.36, it must also be true about all $\bar{b} \geq_\alpha \bar{a}$. □

It follows that if $\leq_\alpha$ coincides with the automorphism relation on $\mathcal{A}$, then every automorphism orbit is $\Pi^{\mathtt{in}}_\alpha$-definable, as every automorphism orbit is of the form $\{\bar{b} \in A^{|\bar{a}|} : \bar{a} \leq_\alpha \bar{b}\}$ for some $\bar{a} \in \mathcal{A}^{<\mathbb{N}}$. Conversely, if every automorphism orbit is $\Pi^{\mathtt{in}}_\alpha$-definable, every automorphism orbit is of the form $\{\bar{b} \in A^{|\bar{a}|} : \bar{a} \leq_\alpha \bar{b}\}$, and hence $\leq_\alpha$ coincides with the automorphism relation on $\mathcal{A}$. Therefore, $r(\mathcal{A})$ is the least ordinal α such that every automorphism orbit in $\mathcal{A}$ is $\Pi^{\mathtt{in}}_\alpha$-definable. Since $\Pi^{\mathtt{in}}_\alpha$-definable implies $\Sigma^{\mathtt{in}}_{\alpha+1}$-definable, we get that $\mathrm{SR}_{\text{p-less}}(\mathcal{A}) \leq r(\mathcal{A}) + 1$. Since having all orbits $\Sigma^{\mathtt{in}}_\alpha$-definable implies that all automorphism invariant sets are also $\Sigma^{\mathtt{in}}_\alpha$-definable, and hence $\Pi^{\mathtt{in}}_\alpha$-definable (by taking complements), it follows that $\mathrm{SR}_{\text{p-less}}(\mathcal{A}) \leq r(\mathcal{A})$ as we had claimed above. In any case, we get that if $\beta > \mathrm{SR}_{\text{p-less}}(\mathcal{A})$ then $\leq_\beta$ coincides with the automorphism relation on $\mathcal{A}$, and if $\beta < \mathrm{SR}_{\text{p-less}}(\mathcal{A})$ then $\leq_\beta$ does not coincide with the automorphism relation on $\mathcal{A}$.

For some structures we have $r(\mathcal{A}) = \mathrm{SR}_{\text{p-less}}(\mathcal{A})$ while for other structures we have $r(\mathcal{A}) = \mathrm{SR}_{\text{p-less}}(\mathcal{A}) + 1$.

Exercise II.63. Give an example of a structure with $r(\mathcal{A}) = \mathrm{SR}_{\text{p-less}}(\mathcal{A})$ and another example with $r(\mathcal{A}) = \mathrm{SR}_{\text{p-less}}(\mathcal{A}) + 1$.

To distinguish between these two cases, we need to introduce the notion of an α-free tuple.

Definition II.64. (Ash and Knight [**AK00**, Section 17.4]) We say that a tuple $\bar{a}$ is *α-free* in $\mathcal{A}$ if for every tuple $\bar{b} \in \mathcal{A}^{|\bar{a}|}$ and every $\beta < \alpha$, there are tuples $\bar{a}', \bar{b}'$ such that

$$\begin{array}{rcl} \bar{a}\bar{b} & \leq_\beta & \bar{a}'\bar{b}' \qquad \text{and} \\ \bar{a} & \not\leq_\alpha & \bar{a}'. \end{array}$$

Lemma II.65. *A tuple $\bar{a}$ is α-free if and only if its $\Pi^{\mathtt{in}}_\alpha$ type is not $\Sigma^{\mathtt{in}}_\alpha$ supported with $\mathcal{A}$.*[§]

Proof. For the ($\Rightarrow$) direction, suppose that the $\Pi^{\mathtt{in}}_\alpha$ type of $\bar{a}$ is $\Sigma^{\mathtt{in}}_\alpha$ supported within $\mathcal{A}$ by the formula $\exists \bar{y} \varphi(\bar{x}, \bar{y})$ where φ is $\Pi^{\mathtt{in}}_\beta$ for some $\beta < \alpha$. (Recall that if the $\Pi^{\mathtt{in}}_\alpha$ type of $\bar{a}$ is supported by a formula of the form $\bigvee\!\!\!\bigvee_i \ \exists \bar{y} \varphi_i(\bar{x}, \bar{y})$, then whichever of these disjuncts is true about $\bar{a}$ would also support its $\Pi^{\mathtt{in}}_\alpha$ type.) Let $\bar{b}$ be a witness to this formula, i.e., $\mathcal{A} \models \varphi(\bar{a}, \bar{b})$. Now, for every $\bar{a}', \bar{b}'$, if $\bar{a}\bar{b} \leq_\beta \bar{a}'\bar{b}'$, then $\mathcal{A} \models \varphi(\bar{a}', \bar{b}')$ as φ is $\Pi^{\mathtt{in}}_\beta$. Since $\exists \bar{y} \varphi(\bar{x}, \bar{y})$ supports $\Pi^{\mathtt{in}}_\alpha$-$tp_{\mathcal{A}}(\bar{a})$, we get that $\bar{a}'$ satisfies all the formulas in $\Pi^{\mathtt{in}}_\alpha$-$tp_{\mathcal{A}}(\bar{a})$ and hence that $\bar{a} \leq_\alpha \bar{a}'$. This shows that $\bar{a}$ is not α-free.

Conversely, suppose that $\bar{a}$ is not α-free, and that $\bar{b}$ and $\beta < \alpha$ are such that for every pair of tuples $\bar{a}', \bar{b}'$, if $\bar{a}\bar{b} \leq_\beta \bar{a}'\bar{b}'$ then $\bar{a} \leq_\alpha \bar{a}'$. Let $\varphi(\bar{x}, \bar{y})$ be the $\Pi^{\mathtt{in}}_\beta$-formula given by the previous lemma which implies the whole $\Pi^{\mathtt{in}}_\beta$ type of $\bar{a}\bar{b}$. We claim that $\exists \bar{y}\ \varphi(\bar{x}, \bar{y})$ supports the $\Pi^{\mathtt{in}}_\alpha$-type of $\bar{a}$. To see this, suppose that $\mathcal{A} \models \exists \bar{y}\ \varphi(\bar{a}', \bar{y})$ for some tuple $\bar{a}'$ — we need to show that $\bar{a} \leq_\alpha \bar{a}'$. Let $\bar{b}'$ be such that $\mathcal{A} \models \varphi(\bar{a}', \bar{b}')$. It follows that $\bar{a}\bar{b} \leq_\beta \bar{a}'\bar{b}'$, and hence that $\bar{a} \leq_\alpha \bar{a}'$. □

Theorem II.66 (Ash and Knight [**AK00**, Proposition 6.11]). *The parameterless Scott rank of $\mathcal{A}$ is the least α for which no tuple is α-free.*

Proof. Follows directly from Theorem II.23. □

We can use this characterization of Scott rank to build infinitary sentences that are true of structures with certain Scott ranks.

[§] Supported types were defined in II.21.

Lemma II.67. *For each vocabulary τ and ordinal α, there is a $\Pi^{\mathtt{in}}_{2\alpha+2}$ sentence ρ_α such that*

$$\mathcal{A} \models \rho_\alpha \iff SR(\mathcal{A}) \geq \alpha$$

for all τ-structures $\mathcal{A}$.

Proof. The idea is for ρ_α to say that for every possible tuple of parameters $\bar{z}$ there is no tuple $\bar{x}$ that is α-free over $\bar{z}$. (For the parameterless Scott rank just omit the parameters.) Thus, we can define ρ_α as

$$\forall \bar{z}\ \forall \bar{x}\ (\bar{x} \text{ is not } \alpha\text{-free over } \bar{z}),$$ [¶]

and we can write down $\bar{x}$ being α-free over $\bar{z}$ as

$$\bigwedge\!\!\!\bigwedge_{\beta<\alpha} \forall \bar{y}\ \exists \bar{x}'\bar{y}' \ \ (\bar{z}\bar{x}\bar{y} \leq_\beta \bar{z}\bar{x}'\bar{y}' \wedge \bar{z}\bar{x} \not\leq_\alpha \bar{z}\bar{x}').$$

We need to show that the back-and-forth relations $\leq_\beta$ are $\mathcal{L}_{\omega_1,\omega}$-definable. In other words, we need formulas $\varphi_\beta(\bar{x},\bar{y})$ for $\beta \leq \alpha$ such that

$$\mathcal{A} \models \varphi_\beta(\bar{a},\bar{b}) \quad\iff\quad (\mathcal{A},\bar{a}) \leq_\beta (\mathcal{A},\bar{b}).$$

These formulas can be easily defined by transfinite recursion by spelling out the definition of $\leq_\beta$ from Definition II.32. That is, define

$$\varphi_\beta(\bar{x},\bar{y}) \quad \text{as} \quad \bigwedge\!\!\!\bigwedge_{\gamma<\beta} \forall \bar{w}\ \exists \bar{z}\ \varphi_\gamma(\bar{y}\bar{w},\bar{x}\bar{z}).$$

The base case needs to say that $\bar{x}$ and $\bar{y}$ have the same diagrams: $\varphi_0(\bar{x},\bar{y})$ is the formula $\bigvee_{\sigma\in 2^{\ell_{|\bar{x}|}}} D(\bar{x}) = \sigma \wedge D(\bar{y}) = \sigma$.

To see that ρ_α is $\Pi^{\mathtt{in}}_{2\alpha+2}$, we first need to observe that φ_β is $\Pi^{\mathtt{in}}_{2\beta}$. To see this, we need to change the definition of φ_β that we give above in the case when β is a limit ordinal. When β is a limit ordinal, we can define φ_β as $\bigwedge\!\!\!\bigwedge_{\gamma<\beta}\varphi_\gamma$. We can then use transfinite induction to show that φ_β is $\Pi^{\mathtt{in}}_{2\beta}$. Then, we get that the formula saying that $\bar{x}$ is α free is $\Pi^{\mathtt{in}}_{2\alpha+1}$, and that ρ_α is $\Pi^{\mathtt{in}}_{2\alpha+2}$. □

[¶] A tuple $\bar{a}$ is *α-free over* a tuple $\bar{b}$ in $\mathcal{A}$, if α is α-free in $(\mathcal{A},\bar{b})$.

CHAPTER III

Computably Infinitary Languages

To study the computational properties of structures syntactically the appropriate language is the computable infinitary language, as first noticed by Chris Ash in [**Ash86b**]. We are referring to the subset of $\mathcal{L}_{\omega_1,\omega}$ that consists of the infinitary formulas that have computable representations. It can also be defined as the set of $\mathcal{L}_{\omega_1,\omega}$ formulas where the infinitary conjunctions and disjunctions must be taken over computable lists of formulas. We have already worked with the first few levels of the computable infinitary language in [**Part 1**]. The main result connecting these formulas with computational complexity is the Ash-Knight-Manasse-Slaman–Chisholm Theorem [**Part 1**, Theorem II.16], which states that a relation is r.i.c.e. if and only if it is $\Sigma^{\mathtt{c}}_1$ definable over parameters. We will see in Theorem VII.18 that this result extends through the arithmetic and hyperarithmetic hierarchies.

III.1. Representing infinitary formulas as trees

When we defined infinitary formulas in the past chapter, we did not really represent them as concrete objects — such formality was not necessary. However, now that we want to talk about computable representations of formulas, we need to settle on some way of representing them. We will represent infinitary formulas with trees, where each node is labeled with either $\bigdoublevee$, $\bigdoublewedge$, $\forall\mathtt{x}$, or $\exists\mathtt{y}$, and each leaf of the tree is labeled with a quantifier-free formula.

Definition III.1. A *tree representation* for a τ-$\mathcal{L}_{\omega_1,\omega}$ formula consists of

(1) a well-founded tree T,
(2) a labeling function ℓ that assigns to each node of T a string of characters satisfying that, if σ is a leaf of T, then $\ell(\sigma)$ is a finitary quantifier-free τ-formula, and if σ is not a leaf, then $\ell(\sigma)$ can be one of: $\bigdoublevee$, $\bigdoublewedge$, $\forall\mathtt{x}$, or $\exists\mathtt{y}$, where $\mathtt{x}$ and $\mathtt{y}$ can be any variable symbols. When $\ell(\sigma)$ is either $\forall\mathtt{x}$ or $\exists\mathtt{y}$, σ has a unique child in the tree T.

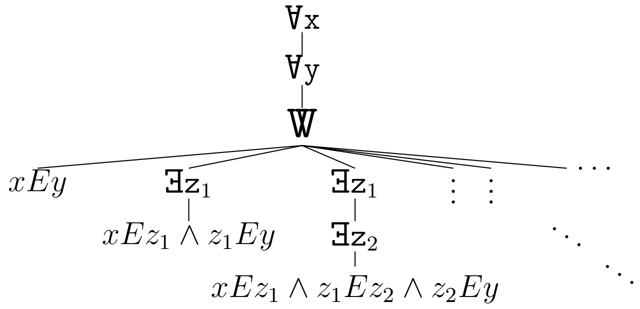


FIGURE III.1. Tree for the infinitary sentence that says that a graph is connected.

(3) a free-variable function $\mathrm{var}(\cdot)$ that assigns to each node of T a finite set of variables satisfying that, if $\ell(\sigma^-) = \mathbb{W}$ or $\ell(\sigma^-) = \mathbb{M}$, then $\mathrm{var}(\sigma) \subseteq \mathrm{var}(\sigma^-)$; if $\ell(\sigma^-) = \forall\mathtt{y}$ or $\ell(\sigma^-) = \exists\mathtt{y}$, then $\mathrm{var}(\sigma) \subseteq \mathrm{var}(\sigma^-) \cup \{\mathtt{y}\}$; and if σ is a leaf of the tree, then the quantifier-free formula $\ell(\sigma)$ only uses variables from $\mathrm{var}(\sigma)$.[*]

Now that we know what a formula is, the next step is to describe what it does. That is, we need to define the *satisfiability relation* $\models$ that, given a formula $\varphi(\bar{x})$, a structure $\mathcal{A}$, and a tuple $\bar{a}$, decides if φ is true of $\bar{a}$ in $\mathcal{A}$, written $\mathcal{A} \models \varphi(\bar{a})$. For this, we need to define the notion of *valuation*, which is a function that assigns a truth value to every sub-formula of φ with every possible interpretation for their variables.

DEFINITION III.2. Consider an infinitary formula φ as in the definition above with free variables $\bar{x} = \mathrm{var}(\langle\rangle)$, a structure $\mathcal{A}$, and a tuple $\bar{a} \in A^{|\bar{x}|}$. A *valuation* for φ and $\mathcal{A}$ is a function v that assigns to each $\sigma \in T$ and each variable assignment $\bar{p}\colon \mathrm{var}(\sigma) \to A$, a truth value $v(\sigma, p)$ in $\{\mathtt{True}, \mathtt{False}\}$. A valuation v is *valid* if it satisfies the obvious rules of logic, that is:

- If $\ell(\sigma) = \mathbb{W}$, then $v(\sigma, p) = \mathtt{True}$ if and only if, for some i with $\sigma^\frown i \in T$, $v(\sigma^\frown i, p) = \mathtt{True}$.
- If $\ell(\sigma) = \mathbb{M}$, then $v(\sigma, p) = \mathtt{True}$ if and only if, for all i with $\sigma^\frown i \in T$, $v(\sigma^\frown i, p) = \mathtt{True}$.
- If $\ell(\sigma) = \exists\mathtt{x}$ and τ is the unique child of σ in T, then $v(\sigma, p) = \mathtt{True}$ if and only if, for some $b \in A$, $v(\tau, p[\mapsto b]) = \mathtt{True}$.
- If $\ell(\sigma) = \forall\mathtt{x}$ and τ is the unique child of σ in T, then $v(\sigma, p) = \mathtt{True}$ if and only if, for all $b \in A$, $v(\tau, p[x \mapsto b]) = \mathtt{True}$.

[*] We use σ^- to denote σ without its last entry.

- If σ is a leaf of the tree, then $v(\sigma, p) = \texttt{True}$ if and only if $\mathcal{A}$ satisfies the quantifier-free formula $\ell(\sigma)$ with the variables in $\mathrm{var}(\sigma)$ assigned according to p.[†]

It can be shown by transfinite recursion that, for every structure $\mathcal{A}$ and formula φ as above, a valid valuation exists and is unique.

Definition III.3. We let $\mathcal{A} \models \varphi(\bar{a})$ if $v(\langle\rangle, p) = \texttt{True}$, where v is the unique valid valuation v for φ and $\mathcal{A}$, and p is the variable assignment mapping $\bar{x}$ to $\bar{a}$.

Observation III.4. We will introduce Π^1_1 and Σ^1_1 sets in the next chapter, but for those readers already familiar with these notions, let us observe that $\mathcal{A} \models \varphi(\bar{a})$ is a Σ^1_1 property of $\mathcal{A}$, φ, and $\bar{a}$: one needs a 2nd-order existential quantifier to say that there exists a valuation and then checking that a valuation is valid is arithmetical. By the uniqueness of valuations, it is also a Π^1_1 property.

Definition III.5. The *computable infinitary formulas* are the ones with computable tree representations, meaning that the tree T and the functions $\ell(\cdot)$ and $\mathrm{var}(\cdot)$ are all computable. We use $\mathcal{L}_{\mathsf{c},\omega}$ to denote the set of all computable infinitary formulas.

Example III.6. The formulas for torsion, connectedness, and finitely apart from Section II.1.1 are all computable. So are the formulas that give bounds for well-founded ranks and well-orderings from Lemmas II.4 and II.5 when the given ordinal is computable. To see this, one has to use effective transfinite recursion (Theorem I.33). Let us look, for instance, at the formula $\psi_\alpha(x)$ from Section II.1.3 that expresses that the well-founded rank of x in a partial ordering is at most α. Recall that we defined

$$\psi_\alpha(x) \quad \text{as} \quad \forall y < x \bigvee\!\!\!\!\bigvee_{\gamma<\alpha} \psi_\gamma(y).$$

Suppose we were already given a computable ω-presentation β of an ordinal, and we are thinking of α as a member of β. We need to define a computable function with domain β, such that for every γ in β, it gives us an index for a computable tree-representation of the formula $\psi_\gamma(x)$. This is a direct application of effective transfinite recursion (Theorem I.33): If we are given a function that gives us the indices for the tree-representations of $\psi_\gamma(y)$ for $\gamma < \alpha$, we can easily build a computable tree-representation of $\forall y < x \bigvee\!\!\!\!\bigvee_{\gamma<\alpha} \psi_\gamma(y)$.

[†] By $p[x \mapsto b]$ we mean the variable assignment that maps x to b and behaves like p on the variables different from x.

The sentence from Lemma II.67 that hold on a structure if and only if the structure has Scott rank at least α is also computable provided α is a computable ordinal.

We now want to classify the computable infinitary formulas according to their alternation-of-quantifier complexity. The process of counting alternations of quantifiers in infinitary formulas is not necessarily computable. Thus, for technical reasons, in the definition below we ask for the existence of a computable function that counts alternations.

Definition III.7. The *computable infinitary Σ_α formulas*, which we denote by $\Sigma^{\mathtt{c}}_\alpha$, are the computable infinitary formulas for which we can computably witness that they are $\Sigma^{\mathtt{in}}_\alpha$. That is, given a computable ordinal α, a computable infinitary formula is $\Sigma^{\mathtt{c}}_\alpha$ if there is a computable ranking function that assigns $\Sigma^{\mathtt{c}}_\alpha$ to the root of the tree. A *ranking function* assigns to each node in the tree representation a symbol of the form $\Sigma^{\mathtt{c}}_\beta$ or $\Pi^{\mathtt{c}}_\beta$ for $\beta \in \alpha + 1$ following the obvious rules: Formulas that start with $\forall$ and $\bigwedge\mkern-15mu\bigwedge$ are assigned $\Pi^{\mathtt{c}}$'s, and formulas that start with $\exists$ and $\bigvee\mkern-15mu\bigvee$ are assigned $\Sigma^{\mathtt{c}}$'s; every time a node switches with respect to its parent node from either $\forall$ or $\bigwedge\mkern-15mu\bigwedge$ to either $\exists$ and $\bigvee\mkern-15mu\bigvee$ or vice versa, its ranking goes down;[‡] and the finitary quantifier-free sub-formulas may be assigned either $\Sigma^{\mathtt{c}}_0$ or $\Pi^{\mathtt{c}}_0$.

Every computable infinitary formula is $\Sigma^{\mathtt{c}}_\alpha$ or $\Pi^{\mathtt{c}}_\alpha$ for some computable ordinal α: Given a formula φ as above, let α be the Kleene-Brouwer ordering on T, and assign to each node σ of T either $\Sigma^{\mathtt{c}}_\sigma$ or $\Pi^{\mathtt{c}}_\sigma$ according to whether $\ell(\sigma)$ is $\exists$ or $\bigvee\mkern-15mu\bigvee$, or $\forall$ or $\bigwedge\mkern-15mu\bigwedge$. Let us note that this is far from the optimal ranking function for φ.

III.2. Representations from the bottom up

Another way of defining computable infinitary formulas is by requiring the infinitary conjunctions and disjunctions to be over lists of formulas that are computable. For this to make sense, we need to have already defined indices for the formulas of smaller rank, so that we can talk about conjunctions and disjunctions over a c.e. set of indices. We then need to define indices for computable infinitary formulas by effective transfinite recursion. The idea is that a $\Sigma^{\mathtt{c}}_\alpha$ formula with index e is the disjunction of all the formulas with indices in W_e, the e-th c.e. set. We use the same idea as when we defined indices for the $\Sigma^{\mathtt{c}}_1$ formulas in

[‡] We do not ask for the ordinal assigned to a node to be the least one with these properties. Thus, this ranking function does not need to be the least ranking function.

[**Part 1**, Section II.1.3]. Let $\varphi^{\mathrm{qf}}_{i,j}(\bar{x})$ for $i, j \in \mathbb{N}$ be an effective enumeration of the quantifier-free finitary τ-formulas, where j is the number of free variables (i.e. $j = |\bar{x}|$). Let $\varphi^{\Pi^{\mathtt{c}}_0}_{i,j}(\bar{x}) = \varphi^{\Sigma^{\mathtt{c}}_0}_{i,j}(\bar{x}) = \varphi^{\mathrm{qf}}_{i,j}(\bar{x})$. Given a computable ordinal α, we define $\varphi^{\Sigma^{\mathtt{c}}_\alpha}_{e,j}(x_1, ..., x_j)$, the e-th $\Sigma^{\mathtt{c}}_\alpha$ formula with j free variables as follows:

$$\varphi^{\Sigma^{\mathtt{c}}_\alpha}_{e,j}(x_1, ..., x_j) \quad \text{is} \quad \mathop{\bigvee\mkern-15mu\bigvee}_{\substack{\langle i,k,\beta\rangle \in W_e \\ \beta \in \alpha}} \exists y_1, ..., y_k \;\; \varphi^{\Pi^{\mathtt{c}}_\beta}_{i,j+k}(\bar{x}, \bar{y}),$$

and define

$$\varphi^{\Pi^{\mathtt{c}}_\alpha}_{e,j}(x_1, ..., x_j) \quad \text{as} \quad \mathop{\bigwedge\mkern-15mu\bigwedge}_{\substack{\langle i,k,\beta\rangle \in W_e \\ \beta \in \alpha}} \forall y_1, ..., y_k \;\; \varphi^{\Sigma^{\mathtt{c}}_\beta}_{i,j+k}(\bar{x}, \bar{y}).$$[§]

By effective transfinite recursion on a computable well-ordering α, one can define a function that, given $\beta < \alpha$, an index e, and a number j, produces computable tree representations for the formulas $\varphi^{\Sigma^{\mathtt{c}}_\beta}_{e,j}$ and $\varphi^{\Pi^{\mathtt{c}}_\beta}_{e,j}$, and computable ranking functions. Conversely, again by effective transfinite recursion, given a tree representation with a computable ranking function for a $\Sigma^{\mathtt{c}}_\alpha$ formula, we can effectively find an index for it.

[§] When we write $\Pi^{\mathtt{c}}_\beta$ for $\beta \in \alpha$, we are identifying the ordinal β with the corresponding element of the given ω-presentation for α.

CHAPTER IV

Pi-one-one Sets

In this chapter, we explore the tight connection between Π^1_1-ness and well-orderness. This connection is one of the pillars of higher recursion theory.

Recall that a formula in the language of second-order arithmetic is *arithmetic* if it has no quantifiers over second-order objects (see page xx). Throughout this section, we will use the variables F and G to range over functions $\mathbb{N} \to \mathbb{N}$. We call them *second-order variables.* We call the elements of $2^{\mathbb{N}}$ and $\mathbb{N}^{\mathbb{N}}$ *reals.* We use n, m, x, y, z, etc. for variables that range over numbers in $\mathbb{N}$. We call them *first-order variables.*

Definition IV.1. A Π^1_1 *formula* is one of the form

$$\forall F \in \mathbb{N}^{\mathbb{N}} \ \varphi(F),$$

where φ is an arithmetic formula which may have both first-order and second-order free variables other than F. A Σ^1_1 *formula* is one of the form $\exists F \in \mathbb{N}^{\mathbb{N}} \ \varphi(F)$, where φ is an arithmetic formula.

A subset of either $\mathbb{N}$ or $\mathbb{N}^{\mathbb{N}}$ is said to be Π^1_1 if it can be defined by a Π^1_1 formula.

Observation IV.2. Standard arguments show that Π^1_1 formulas are closed under conjunctions and disjunctions. It is not hard to see that they are also closed under first-order universal quantification: $\forall x \forall F \ \theta(F, x)$ is equivalent to $\forall F \forall x \ \theta(F, x)$. They are also closed under first-order existential quantification, but this requires an argument; one has to observe that

$$\exists n \in \mathbb{N} \ \forall F \in \mathbb{N}^{\mathbb{N}} \ \theta(F, n) \quad \iff \quad \forall F \in \mathbb{N}^{\mathbb{N}} \ \exists n \in \mathbb{N} \ \theta(F^{[n]}, n),$$

where $F^{[n]}(m) = F(\langle n, m\rangle)$. The ($\Rightarrow$) direction is straightforward. For the ($\Leftarrow$) direction, prove the contra-positive as follows: If $\forall n \ \exists F_n \neg\theta(F_n, n)$, then $F = \bigoplus_n F_n$ witnesses that $\exists F \ \forall n \ \neg\theta(F^{[n]}, n)$.

IV.1. Well-orders

Recall from Definition I.30 that *Kleene's* $\mathcal{O}_{wo}$ is the set of indices for computable well-orderings, using the indexing from Lemma I.29 that assigns a linear ordering $\mathcal{L}_e$ to each natural number e.

Observation IV.3. Kleene's $\mathcal{O}_{\text{wo}}$ is a Π^1_1 subset of $\mathbb{N}$. Just write down its definition and count quantifiers: $e \in \mathcal{O}_{\text{wo}}$ if and only if no function from $\mathbb{N}$ to L_e is a descending sequence in $\mathcal{L}_e$, that is, if $\forall F\ \exists n\ (F(n+1) \not<_{\mathcal{L}_e} F(n))$.

Similarly, $\mathbb{WO}$, the set of ($\subseteq\omega$)-presentations of well-orderings, is a Π^1_1 subset of $2^{\mathbb{N}}$.

Definition IV.4. A set $X \subseteq \mathbb{N}$ is Π^1_1-*complete* if it is Π^1_1 and every other Π^1_1 set $Y \subseteq \mathbb{N}$ m-reduces to it. A set $\mathfrak{X} \subseteq \mathbb{N}^{\mathbb{N}}$ is Π^1_1-*complete* if it is Π^1_1 and every other Π^1_1 set $\mathfrak{Y} \subseteq \mathbb{N}^{\mathbb{N}}$ effectively Wadge[*] reduces to it.

We will show that $\mathcal{O}_{\text{wo}}$ is Π^1_1-complete as a set of numbers and that $\mathbb{WO}$ is Π^1_1-complete as a set of reals.

Lemma IV.5 (Kleene normal form). *Every Σ^1_1 formula of arithmetic is equivalent to one of the form $\exists G \in \mathbb{N}^{\mathbb{N}}\ \varphi(G)$, where φ is Π^0_1.*

Proof. Let ψ be a formula of the form

$$\exists F \forall n_1 \exists m_1 \forall n_2 \exists m_2 \dots \forall n_k \exists n_k \quad \theta(F, n_1, m_1, n_2, m_2, ..., n_k, m_k),$$

where θ is a bounded formula of arithmetic.[†] We will prove that ψ is equivalent to a formula of the form $\exists G \in \mathbb{N}^{\mathbb{N}}\ \forall n \in \mathbb{N}\ \varphi(G,n)$, where ψ is a bounded formula. The key point is that a formula of the form $\forall n\ \exists m\ \theta(n,m)$ is equivalent to $\exists G \in \mathbb{N}^{\mathbb{N}} \forall n\ \theta(n, G(n))$ — the function G is called a *Skolem function* for θ. Iterating this idea, we get that ψ is equivalent to

$$\begin{aligned}&\exists F, G_1, ..., G_k \in \mathbb{N}^{\mathbb{N}}\ \forall n_1, n_2, ..., n_k \\ &\qquad\qquad \theta(F, n_1, G_1(n_1), n_2, G_2(n_1, n_2), ..., n_k, G_k(n_0, ..., n_k)),\end{aligned}$$

which is equivalent to

$$\exists G \forall n \big(\forall n_1, ..., n_k < n \quad \theta(G^{[0]}, n_1, G^{[1]}(n_1), ..., n_k, G^{[k]}(n_1, ..., n_k))\big). \qquad \square$$

Recall that for every Π^0_1 formula $\psi(F)$, there is a computable tree T such that $\psi(F)$ holds if and only if F is a path through T. (See page xx or [**Part 1**, Definition V.17].)

[*] See Definition I.22.

[†] Recall from page xix that a *bounded formula* of arithmetic is one where all quantifiers are of the form $\forall x < a$ or $\exists y < b$.

Corollary IV.6. *(1) Let $\mathcal{S} \subseteq \mathbb{N}^{\mathbb{N}}$ be a Σ^1_1 set of reals. There is a computable tree T such that*

$$X \in \mathcal{S} \quad\iff\quad \exists F\ (X \oplus F \in [T]).$$

(2) Let $S \subseteq \mathbb{N}$ be a Σ^1_1 set of numbers. There is a computable sequence of trees $\{T_m : m \in \omega\}$ such that $m \in S$ if and only if T_m is ill-founded.

Proof. For the first part, write the formula defining $\mathcal{S}$ in the form $\exists F \varphi(X, F)$, where φ is Π^0_1. Let T be a computable tree such that $\varphi(X, F)$ holds if and only if $X \oplus F$ is a path through T.

For the second part, the formula defining S is of the form $\exists F \varphi(m, F)$, where φ is Π^0_1. Let T be a computable tree such that $\varphi(m, F)$ holds if and only if $m^\frown F$ is a path through T, and let $T_m = \{\sigma \in \mathbb{N}^{<\mathbb{N}} : m^\frown\sigma \in T\}$. □

Corollary IV.7. *$\mathbb{WO}$ is a Π^1_1-complete set of reals.*

Proof. Given a Π^1_1 set of reals $\mathfrak{Y} \subseteq \mathbb{N}^{\mathbb{N}}$, let T be as in the corollary above for the complement of $\mathfrak{Y}$. For each $X \in \mathbb{N}^{\mathbb{N}}$, let

$$T^X = \{\sigma \in \mathbb{N}^{<\mathbb{N}} : (X \upharpoonright |\sigma|) \oplus \sigma \in T\}.$$

Note that T^X is a tree and that it can be built computably from X. We then have that $X \in \mathfrak{Y}$ if and only if $\forall F\ (X \oplus F \notin [T])$, which holds if and only if T^X is well-founded. Thus,

$$X \in \mathfrak{Y} \iff (\mathcal{T}^X; \leq_{\text{KB}}) \in \mathbb{WO}. \qquad \square$$

Theorem IV.8. *Kleene's $\mathcal{O}_{wo}$ is a Π^1_1-complete set of numbers.*

Proof. Consider a Π^1_1 set $S \subseteq \mathbb{N}$. By Corollary IV.6 applied to the complement of S, there is a computable sequence of trees $\{T_m : m \in \omega\}$ such that $m \in S$ if and only if T_m is well-founded. Let f be a computable function that, given m, outputs an index for the computable linear ordering $(T_m; \leq_{\text{KB}})$. We then have that $m \in S$ if and only if T_m is well-founded, if and only if $f(m) \in \mathcal{O}_{\text{wo}}$. □

To emphasize such an important theorem, let us state it again: A set of numbers is Π^1_1 if and only if it is many-one reducible to $\mathcal{O}_{\text{wo}}$. This is the defining property of $\mathcal{O}_{\text{wo}}$ as a many-one degree. As a corollary, we get that the first step of the projective hierarchy is proper.

Corollary IV.9. *Kleene's $\mathcal{O}_{wo}$ is not Σ^1_1.*

Proof. If $\mathcal{O}_{\text{wo}}$ were Σ^1_1, the set

$$R = \{e : \Phi_e(e)\downarrow \wedge\ \Phi_e(e) \notin \mathcal{O}_{\text{wo}}\}$$

would be Π^1_1. But then there would be a total computable function f such that $e \in R \iff f(e) \in \mathcal{O}_{\text{wo}}$. Let e_0 be a computable index for f. We would then have that

$$e_0 \in R \iff f(e_0) \in \mathcal{O}_{\text{wo}} \iff \Phi_{e_0}(e_0) \in \mathcal{O}_{\text{wo}} \iff e_0 \notin R. \qquad \square$$

In terms of its Turing degree, the main use for Kleene's $\mathcal{O}_{\text{wo}}$ is that it computes paths through ill-founded trees:

Lemma IV.10. *$\mathcal{O}_{wo}$ can compute paths through every computable tree that has a path.*

Proof. Let T be a computable tree with a path. Let $S \subseteq T$ be the set of $\sigma \in T$ for which T_σ is not well-founded, where T_σ is the subtree of T extending σ. Notice that S is computable from $\mathcal{O}_{\text{wo}}$. Since T is ill-founded, so is S. Furthermore, S has no end nodes, so one can climb it straight up in a step-by-step way without ever getting stuck. This process produces an S-computable path. $\square$

Exercise IV.11. Given $X \in 2^{\mathbb{N}}$, let $\mathcal{O}^X_{\text{wo}}$ be Kleene's $\mathcal{O}_{\text{wo}}$ relativized to X, that is, the set of e's such that $\mathcal{L}^X_e$ is well-ordered, where $\mathcal{L}^X_e$ is the eth X-computable linear ordering (as in Lemma I.29).

Prove that $\mathfrak{A} \subseteq 2^{\mathbb{N}}$ is Π^1_1 if and only if there exists an $n \in \mathbb{N}$ such that, for all $X \in 2^{\mathbb{N}}$, $X \in \mathfrak{A} \iff n \in \mathcal{O}^X_{\text{wo}}$.

IV.2. Sigma-one-one bounding

In this section, we prove an extremely useful lemma called Σ^1_1 bounding. An important property of ω_1 is that every countable set of countable well-orderings has a least upper bound in ω_1. The same is true for ω_1^{CK} if we consider Σ^1_1 sets of computable well-orderings. There are two versions, one for sets of indices of computable well-orderings and one for sets of ω-presentations of well-orderings.

Theorem IV.12 (Σ^1_1 bounding for numbers). *For every Σ^1_1 subset $A \subseteq \mathcal{O}_{wo}$, there is an $\alpha < \omega_1^{CK}$ such that each $e \in A$ is an index for a well-ordering smaller than α.*

We give two proofs. This first is a short application of the fact that $\mathcal{O}_{\text{wo}}$ is not Σ^1_1. The second is more hands-on and shows us how to obtain the upper bound α effectively from a Σ^1_1 index for A.

Proof. Let

$$B = \{e : \exists n\ (n \in A\ \ \&\ \text{ there exists an embedding } \mathcal{L}_e \to \mathcal{L}_n)\}.$$

Note that $\mathcal{B}$ is Σ^1_1 and that $B \subseteq \mathcal{O}_{\text{wo}}$. Since $\mathcal{O}_{\text{wo}}$ is not Σ^1_1 itself, there must be an $e \in \mathcal{O} \smallsetminus B$. Let α be the order type of $\mathcal{L}_e$. Then $\alpha \not\preccurlyeq \mathcal{L}_n$

for all $\mathcal{L}_n$ for $n \in A$. We then have that $\alpha = \mathcal{L}_e$ is an upper bound for all $\mathcal{L}_n$ for $n \in A$. □

Theorem IV.13 (Σ^1_1 bounding for sets of reals). *Let $\mathfrak{A}$ be a Σ^1_1 set of atomic diagrams of ω-presentations of well-orderings. There is an $\alpha < \omega_1^{CK}$ such that every $\beta \in \mathfrak{A}$ is below α.*

Proof. The proof is the same as that of the theorem above. Let

$$B = \{e : \exists \mathcal{L}\ (\mathcal{L} \in \mathfrak{A}\ \ \&\ \ \text{there exists an embedding } \mathcal{L}_e \to \mathcal{L})\}.$$

Note that $\mathcal{B}$ is Σ^1_1 and that $B \subseteq \mathcal{O}_{\text{wo}}$. Since $\mathcal{O}_{\text{wo}}$ is not Σ^1_1 itself, there must be an $e \in \mathcal{O} \smallsetminus B$. We then have that $\alpha = \mathcal{L}_e$ is an upper bound for all $\mathcal{L} \in \mathfrak{A}$. □

The proofs above do not say how to construct the upper bounds. However, in both cases, the upper bound α can be computed from an index for the Σ^1_1 set A or $\mathfrak{A}$, as we will see in the proofs below. The ideas in these proofs are useful tools for other results in the literature too. A key operation used in the proof is the product of trees, whose rank is the minimum of the ranks of the input trees:

Definition IV.14. The *merging of strings* $\sigma = \langle a_0, ..., a_k\rangle$ and $\tau = \langle b_0, ..., b_k\rangle$ of the same length is defined as follows:

$$\sigma * \tau = \langle\langle a_0, b_0\rangle, ..., \langle a_k, b_k\rangle\rangle.$$

We define the *product of trees* S and T as

$$S * T = \{\sigma * \tau : \sigma \in S, \tau \in T, |\sigma| = |\tau|\}.$$

A path through $S*T$ is obtained by merging a path through S and a path through T. Thus, $S*T$ is ill-founded if and only if both S and T are ill-founded. Much more can be said about $S*T$:

Lemma IV.15. *For all trees T and S,*

$$\text{rk}(S * T) = \min\{\text{rk}(S), \text{rk}(T)\}.$$

Proof. To see that $\text{rk}(S * T) \leq \text{rk}(S)$, consider the $\subsetneq$-preserving map $\pi_1 \colon S * T \to S$ given by $\pi_1(\sigma * \tau) = \sigma$, and apply Lemma I.21. Do the same with T to get $\text{rk}(S * T) \leq \text{rk}(T)$. It follows that $\text{rk}(S * T) \leq \min\{\text{rk}(S), \text{rk}(T)\}$. Suppose now that $\text{rk}(S) \leq \text{rk}(T)$ and hence that $\min\{\text{rk}(S), \text{rk}(T)\} = \text{rk}(S)$ — the case where $\text{rk}(T) \leq \text{rk}(S)$ is completely symmetric. By Lemma I.21, there is a $\subsetneq$-preserving map $f \colon S \to T$. Define $g \colon S \to S*T$ by $g(\sigma) = \sigma*(f(\sigma)\restriction|\sigma|)$, and note that g is $\subsetneq$-preserving. It follows that $\min\{\text{rk}(S), \text{rk}(T)\} \leq \text{rk}(S * T)$. □

Uniform proof of Theorem IV.12. Since A is Σ^1_1 and $\mathcal{O}_{\text{wo}}$ is Π^1_1-complete, there is a computable f such that

$$e \in A \iff f(e) \notin \mathcal{O}_{\text{wo}} \iff \mathcal{L}_{f(e)} \notin \mathbb{WO}.$$

For each $e \in \mathbb{N}$, consider the tree

$$S_e = T_{\mathcal{L}_e} * T_{\mathcal{L}_{f(e)}},$$

where $T_{\mathcal{L}}$ is the tree of finite descending sequences of $\mathcal{L}$ as defined on page 12. Since $A \subseteq \mathcal{O}_{\text{wo}}$, for every e, either $e \in \mathcal{O}_{\text{wo}}$ or $e \notin A$. It follows that one of $\mathcal{L}_e$ or $\mathcal{L}_{f(e)}$ must be well-founded, and thus S_e is well-founded for all e. Recall from Observation I.28 that if $\mathcal{L}$ is well-ordered, then $\mathrm{rk}(T_{\mathcal{L}}) \cong \mathcal{L}$, and if $\mathcal{L}$ is not well-ordered, then $\mathrm{rk}(T_{\mathcal{L}}) = \infty$. If $e \in A$, then $\mathrm{rk}(T_{L_{f(e)}}) = \infty$, and hence $\mathrm{rk}(S_e) = \mathrm{rk}(T_{\mathcal{L}_e}) \cong \mathcal{L}_e$. Recall from Exercise I.27 that $\mathrm{rk}(T) < (T; \leq_{\text{KB}})$ for every well-founded tree T. (We include a proof in this footnote.[‡]) What we have so far is that the linear ordering $(S_e; \leq_{\text{KB}})$ is always well-ordered, and for $e \in \mathcal{A}$, we have

$$\mathcal{L}_e \;\cong\; \mathrm{rk}(T_{\mathcal{L}_e}) \;=\; \mathrm{rk}(S_e) \;\preccurlyeq\; (S_e; \leq_{\text{KB}}).$$

Finally, add together all the linear orderings and define

$$\mathcal{L} = \sum_{e \in \omega} (S_e; \leq_{\text{KB}}).$$

It follows that $\mathcal{L}$ is a computable well-ordering that is longer than $\mathcal{L}_e$ for all $e \in A$. □

Uniform proof of Theorem IV.13. In the previous theorem, we added up all the linear orderings $(S_e; \leq_{\text{KB}})$ for $e \in \mathbb{N}$, but that is not possible in this proof, as there are continuum many linear orderings to consider. Instead, we will merge them all together.

Since we are talking about ω-presentations of linear orderings, the only important part of the diagram is the ordering, which is a subset of $\mathbb{N}^2$. So, let us assume that $\mathfrak{A}$ is a set of orderings $<_{\mathcal{L}}$ on $\mathbb{N}$, all of which happen to be well-ordered.

Since $\mathfrak{A}$ is Σ^1_1, there is a computable tree S such that $\mathcal{L} \in \mathfrak{A} \iff \exists X \in \mathbb{N}^{\mathbb{N}}\ \mathcal{L} \oplus X \in [S]$. Consider the Π^0_1 class $\mathcal{P}$ of triples $\mathcal{L} \oplus X \oplus Z$, where $\mathcal{L} \in 2^{\mathbb{N}^2}$ is an ω-presentation of a linear ordering, X is a witness that $\mathcal{L} \in \mathfrak{A}$ (i.e., $\mathcal{L} \oplus X \in [S]$), and $Z \in \mathbb{N}^{\mathbb{N}}$ is a descending sequence in the linear ordering with diagram $\mathcal{L}$. Since $\mathfrak{A}$ contains only well-orderings, if there exists a witness X that $\mathcal{L} \in \mathfrak{A}$, then no descending sequence Z exists. Let us consider the tree T associated with this Π^0_1

[‡] The proof is again by transfinite induction. Show that for each $\tau \in T$, $\mathrm{rk}(T_\tau) < (T_\tau; \leq_{\text{KB}})$ by observing that $(T_\tau; \leq_{\text{KB}}) \cong (\sum_{n \in \mathbb{N}} (T_{\tau^\frown n}; \leq_{\text{KB}})) + 1 \geq \sup_{n \in \mathbb{N}}((T_{\tau^\frown n}; \leq_{\text{KB}}) + 1)$.

class; let T be the set of all strings σ such that if we write σ as $\lambda \oplus \xi \oplus \zeta$, then $\lambda \oplus \xi \in S$ and ζ appears to be a descending sequence according to λ, that is, $\lambda(\langle \zeta(i+1), \zeta(i) \rangle) = 1$ for all i with $\langle \zeta(i+1), \zeta(i) \rangle < |\lambda|$. It is easy to see that $[T] = \mathcal{P}$. Now, since $\mathfrak{A}$ consists only of well-orderings, this Π^0_1 class is empty, and T has no paths. Thus, T is a computable well-founded tree. We now claim the rank of T is a bound for $\mathfrak{A}$, that is, that for every $\mathcal{L} \in \mathfrak{A}$, the order type of $\mathcal{L}$ is below the rank of T. Fix $\mathcal{L} \in \mathfrak{A}$ and a witness Y that $\mathcal{L} \in \mathfrak{A}$. Note that $(\mathcal{L} \upharpoonright n) \oplus (Y \upharpoonright n) \in S$ for every n. Let $T_{\mathcal{L}}$ be the tree of descending sequences through $\mathcal{L}$. $T_{\mathcal{L}}$ has rank $\mathcal{L}$ (Observation I.28). We can easily embed $T_{\mathcal{L}}$ into T by $\zeta \mapsto (\mathcal{L} \upharpoonright |\zeta|) \oplus (Y \upharpoonright |\zeta|) \oplus \zeta$, getting that the rank of T is greater than that of $T_{\mathcal{L}}$. □

Let $\mathcal{O}_{\mathrm{wo}\leq\alpha} = \{e : \mathcal{L}_e \preccurlyeq \alpha\}$, where $\mathcal{A} \preccurlyeq \mathcal{B}$ if there is an embedding from $\mathcal{A}$ to $\mathcal{B}$. Σ^1_1 bounding can be stated as saying that if $A \subseteq \mathcal{O}_{\mathrm{wo}}$ is Σ^1_1, then $A \subseteq \mathcal{O}_{\mathrm{wo}\leq\alpha}$ for some $\alpha < \omega_1^{CK}$. Notice that the sets $\mathcal{O}_{\mathrm{wo}\leq\alpha}$ are Δ^1_1 (that is, both Π^1_1 and Σ^1_1): The definition we gave is Σ^1_1, and also $e \in \mathcal{O}_{\mathrm{wo}\leq\alpha} \iff e \in \mathcal{O}_{\mathrm{wo}}$ & $\alpha + 1 \not\preccurlyeq \mathcal{L}_e$, which is Π^1_1. This observation can be stated more generally as follows:

Theorem IV.16 (Σ^1_1 separation). *Let A and B be disjoint Σ^1_1 sets. There exists a Δ^1_1 set C such that $A \subseteq C \subseteq B^c$.*

Proof. Let f be an m-reduction from B^c to $\mathcal{O}_{\mathrm{wo}}$. By Σ^1_1 bounding, since $f(A)$ is a Σ^1_1 subset of $\mathcal{O}_{\mathrm{wo}}$, there is an $\alpha \in \mathcal{O}_{\mathrm{wo}}$ such that $\mathcal{L}_{f(e)} \preccurlyeq \alpha$ for all $e \in A$. Let $C = \{e \in \mathbb{N} : \mathcal{L}_{f(e)} \preccurlyeq \alpha\} = f^{-1}(\mathcal{O}_{\mathrm{wo}\leq\alpha})$. It is clear that $A \subseteq C \subseteq B^c$. Since $\mathcal{O}_{\mathrm{wo}\leq\alpha}$ is Δ^1_1, so is C. □

Theorem IV.17 (Σ^1_1 separation for sets of reals). *Let $\mathfrak{A}$ and $\mathfrak{B}$ be disjoint Σ^1_1 subsets of $\mathbb{N}^{\mathbb{N}}$. There exists a Δ^1_1 set $\mathfrak{C}$ such that $\mathfrak{A} \subseteq \mathfrak{C} \subseteq \mathfrak{B}^c$.*

Proof. Apply Corollary IV.6 to $\mathfrak{B}$ to get a computable tree T such that

$$X \in \mathfrak{B} \iff \exists F\ (X \oplus F \in [T]).$$

For each X, let

$$T^X = \{\sigma \in \mathbb{N}^{<\mathbb{N}} : (X \upharpoonright |\sigma|) \oplus \sigma \in T\}.$$

So, we have that $X \in \mathfrak{B}^c$ if and only if T^X is well-founded, and only if $(\mathcal{T}^X; \leq_{\mathrm{KB}})$ is well-ordered. The set $\{(T^X; \leq_{KB}) : X \in \mathfrak{A}\}$ is a Σ^1_1 set of well-orderings, and hence from Σ^1_1-bounding (Theorem IV.13), we get an ordinal α such that for every $X \in \mathfrak{A}$, $(T^X; \leq_{KB})$ is below α. Let

$$\mathfrak{C} = \{X \in 2^{\mathbb{N}} : (T^X; \leq_{KB}) \preccurlyeq \alpha\}.$$

It is clear that $\mathfrak{A} \subseteq \mathfrak{C} \subseteq \mathcal{B}^c$. Since the existence of embeddings is Σ^1_1, $\mathfrak{C}$ is Σ^1_1. Notice that $X \notin \mathfrak{C}$ if and only if, either T^X is ill-founded, or $\alpha + 1 \preccurlyeq (T^X; \leq_{KB})$. Thus, $\mathfrak{C}^c$ is also Σ^1_1, and hence $\mathfrak{C}$ is Δ^1_1. □

Corollary IV.18. *A set $C \subseteq \mathbb{N}$ is Δ^1_1 if and only if $C \leq_m \mathcal{O}_{wo \leq \alpha}$ for some $\alpha < \omega_1^{CK}$.*

Proof. The ($\Leftarrow$) direction follows from the observation above that the sets $\mathcal{O}_{\text{wo} \leq \alpha}$ are Δ^1_1. For the ($\Rightarrow$) direction, we have to look at the proof of the theorem above applied to $A = C$ and $B = C^c$. We get that the only separator, namely C, is equal to $f^{-1}(\mathcal{O}_{\text{wo} \leq \alpha})$, and hence $C \leq_m \mathcal{O}_{\text{wo} \leq \alpha}$. □

Another corollary of Σ^1_1 bounding is Spector's theorem:

Theorem IV.19 (Spector [**Spe55**]). *Every Σ^1_1 well-order $\mathcal{L} = (L; \leq$) is isomorphic to a computable one.*[§]

Proof. Let

$$B = \{e : \text{ there exists an embedding } \mathcal{L}_e \to \mathcal{L}\}.$$

Note that B is Σ^1_1 and that $B \subseteq \mathcal{O}_{\text{wo}}$. By Σ^1_1 bounding, there is a bound $\alpha < \omega_1^{CK}$ for B. We must have $\mathcal{L} \leq \alpha < \omega_1^{CK}$, and hence $\mathcal{L}$ has a computable presentation. □

Corollary IV.20. *Let $T \subseteq \mathbb{N}^{<\mathbb{N}}$ be a Σ^1_1 well-founded tree. Then* $\text{rk}(T) < \omega_1^{CK}$.

Proof. Consider $(T; \leq_{KB})$ and apply the previous theorem. □

IV.3. Gandy basis theorem

This is another extremely useful theorem. It is often the case that we want to find reals with a certain property that are not too complex. For instance, if the property is Π^0_1, the low basis theorem of Jockusch and Soare [**JS72**] states there has to be a low X with that property (i.e., an X with $X' \equiv_T 0'$). We now consider the case when the property is Σ^1_1. First, let us look at the limitations we may have in finding such a real. The following lemma shows that there are Σ^1_1 sets of reals without easily definable members. Recall that a set is Δ^1_1 if it has both a Π^1_1 definition and a Σ^1_1 definition.

Lemma IV.21. *The class $\mathfrak{D} = \{Y \subseteq \mathbb{N} : Y$ is $\Delta^1_1\}$ is Π^1_1.*

[§] By Σ^1_1 well-order, we mean an ω-presentation of a well-ordering $(L; \leq)$ where both L and $\leq$ are Σ^1_1.

We thus have a Σ^1_1 class of reals without Δ^1_1 members, namely the class of all non-Δ^1_1 reals.

PROOF. We claim that Y is Δ^1_1 if and only if there exist computable sequences $\{T_n : n \in \mathbb{N}\}$ and $\{S_n : n \in \mathbb{N}\}$ of trees such that if $n \in Y$ then $S_n \not\preccurlyeq T_n$, and if $n \notin Y$ then $T_n \not\preccurlyeq S_n$, where $T \preccurlyeq S$ means that there is a $\subsetneq$-preserving map from T to S. Note that the existence of such a sequence of trees is a Π^1_1 statement about Y, as the existence of $\subsetneq$-preserving maps is Σ^1_1. Thus, it will follow from the claim that $\mathfrak{D}$ is Π^1_1.

To show the claim, let us first recall that there is a $\subsetneq$-preserving map from T to S if and only if $\mathrm{rk}(T) \leq \mathrm{rk}(S)$ by Lemma I.21. It follows that for any two trees T and S, either $T \preccurlyeq S$, or $S \preccurlyeq T$, or both. Then, if we have sequences of trees $\{T_n : n \in \mathbb{N}\}$ and $\{S_n : n \in \mathbb{N}\}$ as above, we have that, for every $n \in \mathbb{N}$,

$$n \in Y \iff S_n \not\preccurlyeq T_n \iff T_n \preccurlyeq S_n.$$

This gives us a Δ^1_1 definition of Y.

Conversely, if we know Y is Δ^1_1, by Corollary IV.6, there exist two computable sequences of computable trees — $\{T_n : n \in \mathbb{N}\}$ and $\{S_n : n \in \mathbb{N}\}$ — such that, for every $n \in \mathbb{N}$,

$$n \in Y \qquad \iff \qquad T_n \text{ is well-founded} \qquad \iff \qquad S_n \text{ is ill-founded.}$$

Thus, $S_n \not\preccurlyeq T_n$ if $n \in Y$, and $T_n \not\preccurlyeq S_n$ if $n \notin Y$, as needed. □

We need to go higher up in the complexity hierarchy to find a member of a Σ^1_1 class.

LEMMA IV.22. *Kleene's $\mathcal{O}_{wo}$ computes a member of every non-empty Σ^1_1 class of reals.*

PROOF. Let $\mathfrak{G} \subseteq \mathbb{N}^{\mathbb{N}}$ be a non-empty Σ^1_1 class of reals. As in Corollary IV.6, let T be a computable tree such that, for all $X \in \mathbb{N}^{\mathbb{N}}$,

$$X \in \mathfrak{G} \qquad \iff \qquad T^X = \{\tau \in \mathbb{N}^{<\mathbb{N}} : X \upharpoonright |\tau| \oplus \tau \in T\} \text{ is ill-founded.}$$

Since $\mathfrak{G}$ is non-empty, T is ill-founded. Kleene's $\mathcal{O}_{\mathrm{wo}}$ can then compute a path $X \oplus Y \in [T]$, as proved in Lemma IV.10. $\mathcal{O}_{\mathrm{wo}}$ then computes $X \in \mathfrak{G}$. □

Not only does $\mathcal{O}_{\mathrm{wo}}$ compute members of every non-empty Σ^1_1 class, it computes members that are *low* in a sense we need to specify.

DEFINITION IV.23. Given $X \in 2^{\mathbb{N}}$, let ω_1^X be ω_1^{CK} relativized to X, that is, ω_1^X is the least ordinal that does not have an X-computable ω-presentation.

Definition IV.24. A set $X \subseteq \mathbb{N}$ is *low for* ω_1 if $\omega_1^X = \omega_1^{CK}$.

Lemma IV.25. *For every $X \subseteq \mathbb{N}$, $\omega_1^X > \omega_1^{CK}$ if and only if $\mathcal{O}_{wo}$ is Δ_1^1 relative to X.*

Proof. For the ($\Rightarrow$) direction, suppose that $\omega_1^X > \omega_1^{CK}$ and hence that there is an X-computable presentation of ω_1^{CK}. Since the $\mathcal{L}_e$'s are computable, we have that $\mathcal{L}_e$ is well-ordered if and only if there exists an embedding from $\mathcal{L}_e$ to ω_1^{CK}. The existence of such an embedding can be expressed with a Σ_1^1 in X formula that uses X to describe the presentation of ω_1^{CK}. Therefore, $\mathcal{O}_{\text{wo}}$ is Σ_1^1 in X. Since $\mathcal{O}_{\text{wo}}$ is Π_1^1, this implies it is Δ_1^1 in X.

For the ($\Leftarrow$) direction, suppose $\mathcal{O}_{\text{wo}}$ is Σ_1^1 relative to X. Consider the linear ordering

$$\mathcal{L} = \sum_{e \in \mathcal{O}_{\text{wo}}} \mathcal{L}_e,$$

which, as we saw in page 15, is isomorphic to ω_1^{CK}. We claim that $\mathcal{L}$ is Σ_1^1 in X. The domain is $\{\langle e, n\rangle : e \in \mathcal{O}_{\text{wo}}, n \in L_e\}$, which is Σ_1^1 in X. The ordering is given by $\langle e_0, n_0\rangle \leq_{\mathcal{L}} \langle e_1, n_1\rangle$ if $e_0 <_{\mathbb{N}} e_1$, or $e_0 =_{\mathbb{N}} e_1$ and $n_0 <_{\mathcal{L}_e} n_1$, which is computable. We thus have a Σ_1^1-in-X ω-presentation of $\mathcal{L}$. By Spector's theorem (Theorem IV.19), $\mathcal{L}$ is isomorphic to an X-computable well-ordering. □

The following proof of Gandy's theorem is more tricky than it is informative. There is a more informative proof using Gandy–Harrington forcing, but since this type of technique is not central to this book, we include only the shorter proof.

Theorem IV.26 (Gandy basis theorem). *Every non-empty Σ_1^1 set $\mathfrak{S}$ of reals has a member that is computable in $\mathcal{O}_{wo}$ and low for ω_1.*

Proof. Consider the set $\mathfrak{R}$ of pairs $X \oplus Y$ such that $X \in \mathfrak{S}$ and Y is not Δ_1^1 in X. Relativizing Lemma IV.21, one can see that the set of pairs $\{Y \oplus X : Y \text{ is } \Delta_1^1 \text{ in } X\}$ is Π_1^1. Thus, $\mathfrak{R}$ is Σ_1^1. $\mathfrak{R}$ is non-empty because once you pick $X \in \mathfrak{S}$, you can pick any Y that is not Δ_1^1 in X. Then $\mathcal{O}_{\text{wo}}$ computes a member $X \oplus Y$ of $\mathfrak{R}$ (Lemma IV.22). Since Y is computable in $\mathcal{O}_{\text{wo}}$ and not Δ_1^1 in X, $\mathcal{O}_{\text{wo}}$ cannot be Δ_1^1 in X either. From the previous lemma, we then get that $\omega_1^X = \omega_1^{CK}$. Putting it all together, $X \in \mathfrak{S}$, $X \leq_T \mathcal{O}_{\text{wo}}$, and $\omega_1^X = \omega_1^{CK}$ as needed. □

Exercise IV.27. Prove that if $A \subseteq \mathbb{N}$ is Π_1^1 but not Δ_1^1, then $\mathcal{O}_{\text{wo}}$ is Δ_1^1 in A.

IV.4. An application of the Gandy basis theorem

Let us build some interesting-looking structures.

Theorem IV.28. *If a Π^1_1 set of indices of computable infinitary sentences has a model, it has one with an ω-presentation that is low for ω_1.*

Proof. As we mentioned before (Observation III.4), the satisfiability predicate $\mathcal{A} \models \varphi$ is a Σ^1_1 property of $\mathcal{A}$ and φ. In other words, there is a Σ^1_1 formula $\psi(X, x)$ such that, if $D(\mathcal{A})$ is the diagram of some structure and e is the index for an $\mathcal{L}_{\mathsf{c},\omega}$ sentence φ_e (as in Section III.2), then $\psi(D(\mathcal{A}), e) \iff \mathcal{A} \models \varphi_e$. Now, if S is a Π^1_1 set of indices of computable infinitary sentences, then the set of ω-presentations $\mathcal{A}$ such that $\forall e\ (e \in S \to \mathcal{A} \models \varphi_e)$ is Σ^1_1. It is also non-empty, as we are assuming that this set of sentences has a model, and by the Löwenheim-Skolem theorem, Theorem II.61, it must have a countable one. By the Gandy basis theorem (Theorem IV.26), there is an ω-presentation $\mathcal{A}$ in that set with $\omega_1^{D(\mathcal{A})} = \omega_1^{CK}$. □

In Section VI.2, we will study structures of *high Scott rank*. These are structures whose Scott rank is an ordinal they cannot compute. We give a proof of their existence here.

Corollary IV.29. *There is an ω-presentation $\mathcal{A}$ whose Scott rank is an ordinal that is not computable in $\mathcal{A}$.*

Proof. Consider the set of sentences that say that “$SR(\mathcal{A}) \geq \mathcal{L}_e$” for $e \in \mathcal{O}_{\text{wo}}$, as defined in Lemma II.67, say for the vocabulary τ of linear orderings. This is a Π^1_1 set of computable infinitary sentences, and it has a model — as, for instance, the linear ordering ω_1^{CK} viewed as a structure has rank ω_1^{CK}. By the previous theorem, it has a model $\mathcal{A}$ with $\omega_1^{D(\mathcal{A})} = \omega_1^{CK}$. Since $\mathcal{A}$ satisfies all these sentences, $\mathcal{A}$ must have Scott rank at least ω_1^{CK}. □

We will improve this corollary later on and show there is a computable structure whose Scott rank is not computable (Lemma VI.9). We will also show that the Scott rank of such a structure can be at most $\omega_1^{CK} + 1$ (Corollary VI.19).

The following corollary assumes ZFC is ω-consistent, i.e., that it has a model where the ω of the model looks exactly like the standard $\mathbb{N}$. The reader not comfortable with this assumption may take a fragment of ZFC instead.

Corollary IV.30. *(Assume ZFC is ω-consistent.) There is a countable model $\mathcal{M}$ of ZFC for which the chain of ordinals $(ON^{\mathcal{M}}; \in_{\mathcal{M}})$ is ill-founded and has a well-founded part isomorphic to ω_1^{CK}.*

Proof. Let τ be the vocabulary of set theory $\{\in\}$. Let Γ be a sentence that consists of the infinitary conjunction of all the axioms of ZFC plus one more sentence that says that the natural numbers look like $\mathbb{N}$. To say that the natural numbers in the model are like the standard natural numbers, one first has to observe that ω, zero, and the successor function $S(\cdot)$ are definable in ZFC. Then, using these definitions, we can write down the formula $\forall x \in \omega \bigvee\!\!\!\bigvee_{n \in \mathbb{N}} x = \underbrace{S(S(....S}_{n}(0)))$.

The assumption that ZFC is ω-consistent says that Γ has a model and, by the Löwenheim-Skolem theorem (Theorem II.61), a countable one. The theorem above then implies that Γ has a countable model $\mathcal{M}$ with $\omega_1^{D(\mathcal{M})} = \omega_1^{CK}$. Since $\omega^{\mathcal{M}} \cong \mathbb{N}$, everything that can be defined in arithmetic can be defined in $\mathcal{M}$. In particular, every computable well-ordering of ω has an ω-presentation in $\mathcal{M}$. Since $\mathcal{M}$ satisfies ZFC, every well-ordering of ω is isomorphic to an ordinal, and hence $\mathcal{M}$ contains ordinals isomorphic to every computable well-ordering. It follows that all computable ordinals are initial segments of $\mathrm{ON}^{\mathcal{M}}$. In other words, ω_1^{CK} is an initial segment of $\mathrm{ON}^{\mathcal{M}}$. However, there cannot be an element in $\mathrm{ON}^{\mathcal{M}}$ isomorphic to ω_1^{CK}, as otherwise we could use the diagram $D(\mathcal{M})$ of $\mathcal{M}$ to compute an ω-presentation of ω_1^{CK}, contradicting that $\omega_1^{D(\mathcal{M})} = \omega_1^{CK}$. Thus, $\mathrm{ON}^{\mathcal{M}} \smallsetminus \omega_1^{CK}$ has no least element, and hence the well-founded part of $\mathrm{ON}^{\mathcal{M}}$ is exactly ω_1^{CK}. □

On the one hand $\mathcal{M}$ is a model of ZFC satisfying all true Π_1^1 sentences, and on the other hand $\mathcal{M}$ believes $\mathrm{ON}^{\mathcal{M}}$ is well-ordered while in reality it is not. This has some interesting consequences that are often quite useful in proofs.

CHAPTER V

Hyperarithmetic Sets

The hyperarithmetic hierarchy extends the arithmetical hierarchy through the computable ordinals, giving us new complexity levels that are sometimes necessary to describe the complexity of relations or isomorphisms on structures.

V.1. Computably infinitary definable sets

A set $A \subseteq \mathbb{N}$ is *arithmetic* if it can be defined in

$$\mathcal{N} = (\mathbb{N}; +, \times, 0, 1, <)$$

by a finitary first-order formula. We now take a step beyond the arithmetic.

Definition V.1. A set $A \subseteq \mathbb{N}$ is *hyperarithmetic* if it can be defined in $\mathcal{N} = (\mathbb{N}; +, \times, 0, 1, <)$ by a computable infinitary formula, that is, if there is a computable infinitary formula $\varphi(x)$ in the vocabulary of arithmetic such that

$$A \quad = \quad \{n \in \mathbb{N} : \mathcal{N} \models \varphi(n)\}.$$

For example, $0^{(\omega)} = \bigoplus_{n\in\mathbb{N}} 0^{(n)}$ is not arithmetic, but it is hyperarithmetic:

$$\langle n, m\rangle \in 0^{(\omega)} \iff \bigvee\!\!\!\bigvee_{k\in\mathbb{N}} n = \mathbf{k} \ \wedge \ m \in 0^{(k)},$$

where $\mathbf{k}$ is shorthand for $1 + \cdots + 1$ k times, and $0^{(k)}$ is shorthand for the Σ^0_k formula defining $0^{(k)}$.

Observation V.2. The hyperarithmetic sets are closed downward under Turing reducibility and closed under Turing jumps: Suppose that $X \subseteq \mathbb{N}$ is hyperarithmetic and definable by $\varphi(x)$. Then, if Y is computable from X via the eth Turing functional,

$$n \in Y \iff \bigvee\!\!\!\bigvee_{\substack{\sigma\in 2^{<\mathbb{N}} \\ \Phi_e^\sigma(n)=1}} \sigma \subseteq X$$

and

$$n \in X' \iff \bigvee\!\!\!\bigvee_{\substack{\sigma \in 2^{<\mathbb{N}} \\ \Phi_n^\sigma(n)\downarrow}} \sigma \subseteq X,$$

where $\sigma \subseteq X$ is shorthand for

$$\bigwedge_{\substack{i<|\sigma| \\ \sigma(i)=1}} \varphi(i) \wedge \bigwedge_{\substack{i<|\sigma| \\ \sigma(i)=0}} \neg\varphi(i).$$

Lemma V.3. *The following are equivalent:*

(1) *A is hyperarithmetic.*

(2) *There is a computable list $\{\varphi_n : n \in \mathbb{N}\}$ of computable infinitary sentences in the empty vocabulary such that*

$$n \in A \iff \varphi_n \text{ holds.}$$

For (2), we allow for the use of symbols $\top$ and $\bot$, representing propositions that are always true and always false respectively.[*]

Proof. To prove that (2) implies (1), consider the formula $\varphi(x)$ defined as $\bigwedge\!\!\!\bigwedge_{n\in\mathbb{N}}(x = \mathbf{n} \to \varphi_n)$, where $\mathbf{n}$ is short for $\overbrace{\mathbf{1} + \cdots + \mathbf{1}}^{n \text{ times}}$.

The interesting direction is (1) implies (2). Let A be definable in $(\mathbb{N}; +, \times, 0, 1, <)$ by a computable infinitary formula. As an intermediate step, we show that A is computably infinitary definable by a formula $\psi(w)$ in the structure $(\mathbb{N}; \mathbf{0}, \mathbf{1}, \mathbf{2}, ...)$ over the vocabulary that only contains constants naming each natural number, but does not contain any relation or operation. For this, replace each sub-formula $x + y = z$ by

$$\bigvee\!\!\!\bigvee_{\substack{c,d,e\in\mathbb{N}, \\ c+d=e}} x = \mathbf{c} \wedge y = \mathbf{d} \wedge z = \mathbf{e}.$$

Do the same for each sub-formula of the form $x \times y = z$ and $x \leq y$. This way we obtain an equivalent formula which does not use the symbols $+$, $\times$, or $\leq$.

Now, replace each universal quantifier $\forall x$ by $\bigwedge\!\!\!\bigwedge_{m\in\mathbb{N}}$ and, within the disjunct corresponding to m, replace x with $\mathbf{m}$. The same way, replace existential quantifiers with infinitary disjunctions. That is, if we have a sub-formula of the form $\exists x\ \psi(x)$, replace it with $\bigvee\!\!\!\bigvee_{m\in\mathbb{N}} \psi(\mathbf{m})$.

We now have an equivalent formula $\varphi(w)$ which mentions no variables other than w. For the last step, for each $n \in \mathbb{N}$, consider the

[*] We may use a conjunction over the empty set to represent $\top$ and a disjunction over the empty set to represent $\bot$. In terms of complexity, count $\top$ and $\bot$ as both $\Sigma_0^{\mathtt{c}}$ and $\Pi_0^{\mathtt{c}}$ formulas.

formula φ_n, where the free variable w is replaced by $\mathbf{n}$. This way we eliminate all the variables, and all the atomic sub-formulas are of the form $\mathbf{a} = \mathbf{b}$ for some $a, b \in \mathbb{N}$. Replace each of those atomic formulas with either $\top$ or $\perp$ depending on whether the equality is true or false.

We are now left with computable infinitary formulas φ_n whose only symbols are $\bigvee\!\!\!\bigvee$, $\bigwedge\!\!\!\bigwedge$, $\perp$, and $\top$, and such that $n \in A$ if and only if φ_n holds. $\square$

We call these formulas which only use the symbols $\bigvee\!\!\!\bigvee$, $\bigwedge\!\!\!\bigwedge$, $\perp$, and $\top$ *infinitary propositional sentences.*

Observation V.4. It is not hard to see from the proof above that the complexity of the formulas is preserved. That is, that, for $\alpha > 0$, A can be defined by a $\Sigma^{\mathsf{c}}_{\alpha}$ formula of arithmetic if and only if there is a computable sequence of $\Sigma^{\mathsf{c}}_{\alpha}$ formulas over the empty language as in part (2) of the lemma.

Definition V.5. We say that a set A is Σ^0_{α} if it is definable by a $\Sigma^{\mathsf{c}}_{\alpha}$ formula of arithmetic.

For $n \in \mathbb{N}$, this definition of Σ^0_n set coincides with the one we gave in the background section on page xix.

Lemma V.6. *Let $\mathcal{A}$ be a computable ω-presentation of a τ-structure and $\varphi(\bar{x})$ a $\Sigma^{\mathsf{c}}_{\alpha}$ τ-formula for $\alpha \geq 1$. The set $\{\bar{a} : \mathcal{A} \models \varphi(\bar{a})\} \subseteq \mathbb{N}^{|\bar{a}|}$ is Σ^0_{α}.*

Proof. Each atomic formula about $\mathcal{A}$ can be replaced by its computable definition in $\mathcal{N} = (\mathbb{N}; +, \times, 0, 1, <)$, which can be chosen to be Σ^0_1 or Π^0_1, depending on whether the atomic formula appears negatively or positively and what complexity is wanted for it.

For instance, suppose $\varphi^{\tau}(\bar{x})$ is a Σ^{c}_1 τ-formula of the form

$$\bigvee\!\!\!\bigvee_{i \in \mathbb{N}} \exists \bar{y}\ \left(\psi^{\tau}_i(\bar{x}, \bar{y}) \wedge \theta^{\tau}_i(\bar{x}, \bar{y})\right),$$

where ψ^{τ}_i is a conjunction of atomic τ-formulas and θ^{τ}_i is a conjunction of negations of τ-atomic formulas. Since each atomic τ-formula is computable in this particular ω-presentation of $\mathcal{A}$, each atomic τ-formula is equivalent to both a Σ^{c}_1 $\mathbb{N}$-formula about $(\mathbb{N}; +, \times, 0, 1, <)$ and a Π^{c}_1 $\mathbb{N}$-formula about $(\mathbb{N}; +, \times, 0, 1, <)$.[†] If we replace each atomic τ-formula in ψ^{τ}_i by its equivalent Σ^{c}_1 $\mathbb{N}$-formula, we get that ψ^{τ}_i is equivalent to a Σ^{c}_1 $\mathbb{N}$-formula $\psi^{\mathcal{N}}_i$. If we replace each atomic τ-formula in θ^{τ}_i by its equivalent Π^{c}_1 $\mathbb{N}$-formula, we get that θ^{τ}_i is equivalent to a Σ^{c}_1 $\mathbb{N}$-formula $\theta^{\mathcal{N}}_i$. We then get that $\varphi^{\tau}(\bar{x})$ is itself equivalent to a Σ^{c}_1

[†] By $\mathbb{N}$-*formula*, we mean a formula in the vocabulary of arithmetic.

$\mathbb{N}$-formula $\varphi^{\mathcal{N}}(\bar{x})$ given by $\bigvee\!\!\!\bigvee_{i\in\mathbb{N}} \exists \bar{y}\ \big(\psi_i^{\mathcal{N}}(\bar{x},\bar{y}) \wedge \theta_i^{\mathcal{N}}(\bar{x},\bar{y})\big)$. That is, if $\bar{a} \in A^{<\mathbb{N}} = \mathbb{N}^{<\mathbb{N}}$, then

$$\mathcal{A} \models \varphi^{\tau}(\bar{a}) \quad \iff \quad (\mathbb{N}; +, \times, 0, 1, <) \models \varphi^{\mathcal{N}}(\bar{a}).$$

If we start with a $\Sigma^{\mathtt{c}}_{\alpha}$ formula instead of a $\Sigma^{\mathtt{c}}_{1}$ formula, apply the same procedure to the maximal $\Sigma^{\mathtt{c}}_{1}$ and $\Pi^{\mathtt{c}}_{1}$ sub-formulas of φ. □

Lemma V.7. *Given a $\Sigma^{\mathtt{c}}_{\alpha}$ τ-sentence φ and an index e for a computable structure $\mathcal{A}_e$, deciding if $\mathcal{A}_e$ satisfies φ is Σ^0_{α}.*

We remark that for e to be the index of some computable structure we must have that Φ_e is total and that it is of the form $D(\mathcal{A}_e)$ of some structure $\mathcal{A}_e$, meaning that the function symbols are represented by functions, etc. This is a Π^0_2 property about e.

Proof. Let φ^{τ} be a $\Sigma^{\mathtt{c}}_{\alpha}$ τ-sentence. In the proof of the lemma above, we described a procedure to go from an index e for a computable ω-presentation $\mathcal{A}_e$ with diagram $D(\mathcal{A}_e) = \Phi_e \in 2^{\mathbb{N}}$ to a $\Sigma^{\mathtt{c}}_{\alpha}$ $\mathcal{N}$-sentence $\varphi^{\mathcal{N},e}$ such that

$$\mathcal{A}_e \models \varphi^{\tau} \iff (\mathbb{N}; +, \times, 0, 1, <) \models \varphi^{\mathcal{N},e}.$$

We then have that the set of indices of structures satisfying φ^{τ} can be defined in $(\mathbb{N}; +, \times, 0, 1, <)$ by the formula $\psi(x)$ given by

$$\bigvee\!\!\!\bigvee_{e\in\mathbb{N}} (x = \mathbf{e}) \ \wedge\ \varphi^{\mathcal{N},e}.$$ □

Lemma V.8. *Given a computable ordinal α, the set of indices of computable well-orderings less than α and the set of indices of computable well-founded trees of rank less than α are hyperarithmetic.*

Furthermore, if $\alpha = \omega^{\beta}$, the former set is $\Sigma^0_{2\beta}$, and if $\alpha = \omega\gamma$, the latter set is $\Sigma^0_{2\gamma}$.

Proof. This follows immediately from the previous lemma using the computable infinitary formulas we defined in Lemmas II.4 and II.5. □

Theorem V.9. *(Kleene) Let A be a subset of $\mathbb{N}$. The following are equivalent:*

(1) *A is hyperarithmetic.*
(2) *A is Δ^1_1.*
(3) *$A \leq_m \mathcal{O}_{wo\leq\alpha}$ for some $\alpha < \omega_1^{CK}$.*

Recall that $\mathcal{O}_{\mathrm{wo}\leq\alpha} = \{e : \mathcal{L}_e \preccurlyeq \alpha\}$ and that $\{\mathcal{L}_e : e \in \mathbb{N}\}$ is a computable enumeration of the computable linear orderings defined in Lemma I.29.

Proof. For (1)⇒(2), recall from Observation III.4 that there is a Σ^1_1 formula that decides if an infinitary sentence is true on an ω-presentation. Thus, hyperarithmetic sets are Σ^1_1 sets. Since the complement of a hyperarithmetic set is also hyperarithmetic, they are also Π^1_1.

That (2)⇒(3) was proved in Corollary IV.18.

That (3)⇒(1) follows from the previous lemma and Observation V.2 that hyperarithmetic sets are closed under many-one reducibility. □

Lemma V.10. *A Σ^0_α disjunction of $\Sigma^{\mathtt{c}}_\alpha$ formulas is equivalent to a $\Sigma^{\mathtt{c}}_\alpha$ formula. A Σ^0_α conjunction of $\Pi^{\mathtt{c}}_\alpha$ formulas is equivalent to a $\Pi^{\mathtt{c}}_\alpha$ formula.*

By "Σ^0_α disjunction" we mean an infinitary disjunction of formulas whose indices come from a Σ^0_α set.

Proof. Consider a formula φ of the form $\bigvee\!\!\!\bigvee_{e\in I} \varphi_e^{\Sigma^{\mathtt{c}}_\alpha}$, where I is Σ^0_α. By Lemma V.3, there is a computable sequence $\{\psi_n : n \in \mathbb{N}\}$ of $\Sigma^{\mathtt{c}}_\alpha$ propositional sentences such that $n \in I \iff \psi_n$. Then φ is equivalent to the following $\Sigma^{\mathtt{c}}_\alpha$ formula:

$$\bigvee\!\!\!\bigvee_{e\in\mathbb{N}} \left(\psi_e \wedge \varphi_e^{\Sigma^{\mathtt{c}}_\alpha}\right).$$

For the second part, $\bigwedge\!\!\!\bigwedge_{e\in I} \varphi_e^{\Pi^{\mathtt{c}}_\alpha}$ is equivalent to $\bigwedge\!\!\!\bigwedge_{e\in\mathbb{N}}(\psi_e \to \varphi_e^{\Pi^{\mathtt{c}}_\alpha})$, which is $\Pi^{\mathtt{c}}_\alpha$. □

Historical Remark V.11. The hyperarithmetic sets were introduced independently in the early 1950s by Martin Davis, Andrej Mostowski and Stephen Cole Kleene.

V.2. The jump hierarchy

Another way of defining the hyperarithmetic hierarchy is using transfinite iterates of the Turing jump. We know that a set $A \subseteq \mathbb{N}$ is arithmetic if and only if it is computable in $0^{(n)}$ for some $n \in \mathbb{N}$ (page xix). Correspondingly, we will see that a set is hyperarithmetic if and only if it is computable in $0^{(\alpha)}$ for some computable ordinal α.

Definition V.12. Given a computable linear ordering $\mathcal{L}$, a *jump hierarchy* on $\mathcal{L}$ is a set $H \subseteq L \times \mathbb{N}$ such that

$$(\forall a \in \mathcal{L}) \quad H^{[a]} = (H^{[<_{\mathcal{L}} a]})', \tag{JH}$$

where

$$\begin{aligned} H^{[a]} &= \{n \in \omega : \langle a, n\rangle \in H\} \quad \text{and} \\ H^{[<_{\mathcal{L}} a]} &= \{\langle b, n\rangle \in L \times \omega : b <_{\mathcal{L}} a \;\&\; \langle b, n\rangle \in H\} \\ &= H \cap (L_{(<\alpha)} \times \omega). \end{aligned}$$

If $\mathcal{L}$ is a computable well-ordering, we use $0^{(\mathcal{L})}$ to denote the jump hierarchy corresponding to $\mathcal{L}$. If $\alpha \in \mathcal{L}$, we often write $0^{(\alpha)}$ as shorthand for $0^{(\mathcal{L}\restriction\alpha)}$. Recall that $\mathcal{L} \restriction \alpha$ is the same as $\mathcal{L}_{(<\alpha)}$, the restriction of the linear ordering to the elements below α.

Given a well-ordering $\mathcal{L}$, it is not hard to prove by transfinite recursion that $\mathcal{L}$ admits a jump hierarchy, and then by transfinite induction that such a jump hierarchy is unique. We will consider ill-founded linear orderings in future chapters. We will see that in the ill-founded case, jump hierarchies may or may not exist, and if they exist, they need not be unique. For now, let us concentrate on the case when $\mathcal{L}$ is well-ordered.

Suppose that $0_{\mathcal{L}}, 1_{\mathcal{L}}, 2_{\mathcal{L}}, ...$ are the first elements of $\mathcal{L}$, and H is the jump hierarchy along $\mathcal{L}$. Then $H^{[0_{\mathcal{L}}]} = H^{[<0_{\mathcal{L}}]\prime} = \emptyset'$. We then have that $H^{[1_{\mathcal{L}}]} \cong_1 0''$, where $\cong_1$ means computably isomorphic or 1-equivalent. We did not write 'equals' because $H^{[<1_{\mathcal{L}}]}$ is not equal to $0'$ but to $\{0_{\mathcal{L}}\} \times 0'$. Continuing on, we see that $H^{[n_{\mathcal{L}}]} \cong_1 0^{(n+1)}$ for all $n \in \mathbb{N}$:

$$\begin{aligned} H^{[n_{\mathcal{L}}]} &= H^{[<_{\mathcal{L}} n_{\mathcal{L}}]\prime} \\ &= \Big(\bigcup_{i<n} \{i_{\mathcal{L}}\} \times H^{[i_{\mathcal{L}}]}\Big)' \\ &\cong_1 \Big(\bigcup_{i<n} \{i_{\mathcal{L}}\} \times 0^{(i+1)}\Big)' \\ &\cong_1 \Big(\bigoplus_{i<n} 0^{(i+1)}\Big)' \\ &\cong_1 (0^{(n)})' \\ &= 0^{(n+1)}, \end{aligned}$$

and hence

$$H^{[<_{\mathcal{L}} m_{\mathcal{L}}]} \cong_1 0^{(m)} \qquad \text{for all } m \in \mathbb{N}.$$

In particular, if $\mathbf{m}$ is the finite linear ordering with m elements, then $0^{(\mathbf{m})}$ is Turing equivalent to the m-th iterate of the Turing jump.

Observation V.13. If $\mathcal{L}$ is a computable well-ordering, the set $0^{(\mathcal{L})}$ is Δ^1_1 (and hence hyperarithmetic). This is because, for $k \in \mathcal{L} \times \mathbb{N}$,

$$\begin{aligned} k \in 0^{(\mathcal{L})} \quad &\iff \quad (\exists H \subseteq L \times \mathbb{N}) \;\; H \text{ is a jump hierarchy on } \mathcal{L} \text{ and } k \in H \\ &\iff \quad (\forall H \subseteq L \times \mathbb{N}) \;\; \text{if } H \text{ is a jump hierarchy on } \mathcal{L}, \text{ then } k \in H, \end{aligned}$$

and H being a jump hierarchy on $\mathcal{L}$ is a Π^0_2 property of H and $\mathcal{L}$ (see equation JH).

Observation V.14. If we want to define the $\mathcal{L}$th jump of a real X, we need to modify the definition of jump hierarchy at the start and let $H^{[0_\mathcal{L}]} = X'$. We then define $X^{(\mathcal{L})}$ to be the unique such jump hierarchy.

V.2.1. Jump hierarchies and $\mathcal{L}_{\mathsf{c},\omega}$. We can pinpoint the complexity of $0^{(\alpha)}$ much better than just saying that it is Δ^1_1. We will prove in Theorem V.16 below that $0^{(\alpha+1)}$ is a complete $\Sigma^0_{1+\alpha}$ set for all computable well-orderings α.[‡] We start by proving the easier direction of completeness.

Lemma V.15. *For each computable ordinal α, $0^{(\alpha+1)}$ is $\Sigma^0_{1+\alpha}$.*

Proof. Let $\mathcal{L}$ be a computable well-ordering extending α, so that we can think of α as a member of $\mathcal{L}$. Let H be the jump hierarchy along $\mathcal{L}$. We need to show that for each $\alpha \in \mathcal{L}$, $H^{[\leq_\mathcal{L}\alpha]}$ is $\Sigma^0_{1+\alpha}$. Notice that $H^{[\leq_\mathcal{L}\alpha]}$ is the same thing as $0^{(\alpha+1)}$.

The first idea is to use induction on $\alpha \in \mathcal{L}$. One has to be careful with the limit cases though, because, to prove that $H^{[\leq_\mathcal{L}\lambda]}$ is $\Sigma^0_{1+\lambda}$ for λ limit, we will need more than just knowing that $H^{[\leq_\mathcal{L}\beta]}$ is $\Sigma^0_{1+\beta}$ for all $\beta < \lambda$: We will need to know that this happens uniformly.

What we will do is use effective transfinite recursion (Theorem I.33) to define a computable function $f\colon \mathcal{L} \to \mathbb{N}$, such that for each $\gamma \in \mathcal{L}$, $f(\gamma)$ is an index for a $\Sigma^{\mathsf{c}}_{1+\gamma}$ formula of arithmetic defining $H^{[\gamma]}$. This will give that $H^{[\leq_\mathcal{L}\gamma]} = \bigoplus_{\beta\in\mathcal{L}_{\leq\gamma}} H^{[\beta]}$ is $\Sigma^{\mathsf{c}}_{1+\gamma}$ too. We are now ready to get into the details to define f.

If a set X is $\Delta^{\mathsf{c}}_\gamma$-definable, its jump is $\Sigma^{\mathsf{c}}_\gamma$ definable. To see this, use that

$$x \in X' \iff \bigvee\!\!\!\bigvee_{\substack{\sigma\in 2^{<\mathbb{N}} \\ \Phi^\sigma_x(x)\downarrow}} \sigma \subseteq X$$

[‡] The indices $\alpha+1$ and $1+\alpha$ may seem to be off. Unfortunately, the $0^{(\beta)}$ and the Σ^0_β hierarchies were historically defined in a way that causes this mismatch. For finite n, $0^{(n)}$ is Σ^0_n complete, while it is $0^{(\omega+1)}$ which is Σ^0_ω complete. What we can say about $0^{(\omega)}$ is that it is Δ^0_ω Turing-complete. For infinite α, the complete Σ^0_α set is $0^{(\alpha+1)}$ (Theorem V.16).

to produce an index for the $\Sigma^{\mathsf{c}}_{\gamma}$ formula ψ defining X' from indices for the $\Sigma^{\mathsf{c}}_{\gamma}$ and $\Pi^{\mathsf{c}}_{\gamma}$ formulas θ^{Σ} and θ^{Π} defining X:

$$\psi(x) \quad \text{is} \quad \mathop{\bigvee\mkern-15mu\bigvee}_{\substack{\sigma\in 2^{<\mathbb{N}} \\ \Phi^{\sigma}_{x}(x)\downarrow}} \left(\bigwedge_{\substack{i<|\sigma| \\ \sigma(i)=1}} \theta^{\Sigma}(i) \wedge \bigwedge_{\substack{i<|\sigma| \\ \sigma(i)=0}} \neg\theta^{\Pi}(i) \right).$$

We can use this to calculate $f(\gamma)$, the $\Sigma^{\mathsf{c}}_{1+\gamma}$-index for $H^{[\gamma]}$, using a $\Delta^{\mathsf{c}}_{1+\gamma}$ index for $H^{[<_{\mathcal{L}}\gamma]}$. To get a $\Delta^{\mathsf{c}}_{1+\gamma}$ index for

$$H^{[<_{\mathcal{L}}\gamma]} = \bigoplus_{\beta\in\mathcal{L}\restriction\gamma} H^{[\beta]},$$

recall that we are using effective transfinite recursion, and we have access to a computable index for $f \restriction \mathcal{L}_{(<\gamma)}$ to get $\Sigma^{\mathsf{c}}_{1+\beta}$ indices for each $H^{[\beta]}$ for $\beta < \gamma$. We can easily transform a $\Sigma^{\mathsf{c}}_{1+\beta}$ index to both a $\Sigma^{\mathsf{c}}_{1+\gamma}$ index and a $\Pi^{\mathsf{c}}_{1+\gamma}$ index for each $\beta < \gamma$, and thus obtain a $\Delta^{\mathsf{c}}_{1+\gamma}$ index for $H^{[<_{\mathcal{L}}\gamma]}$. □

THEOREM V.16. *For each computable well-ordering α, $0^{(\alpha+1)}$ is a complete $\Sigma^{0}_{1+\alpha}$ set.*

PROOF. Again, let $\mathcal{L}$ be a computable well-ordering extending α, so that we can think of α as a member of $\mathcal{L}$.[§]

We will use effective transfinite recursion (Theorem I.33) to define a computable function $f\colon \mathcal{L}\times\mathbb{N}\to\mathbb{N}$ that assigns to each $\Sigma^{\mathsf{c}}_{1+\alpha}$-propositional sentence $\varphi^{\Sigma^{\mathsf{c}}_{1+\alpha}}_{e}$ over the empty language (as in Theorem V.3) a number $f(\alpha, e)$ such that

$$\varphi^{\Sigma^{\mathsf{c}}_{1+\alpha}}_{e} \text{ holds if and only if } \langle\alpha, f(\alpha,e)\rangle \in 0^{(\mathcal{L})}.$$

The case $\alpha = 0$ just says that $0'$ is Σ^{0}_{1}-complete, which we already know, and we know how to define $f(0,e)$.[¶] Let us now define $f(\alpha,e)$ assuming we have access to a computable index for $f \restriction \alpha\times\omega$.

Recall that $\varphi^{\Sigma^{\mathsf{c}}_{1+\alpha}}_{e}$, the eth $\Sigma^{\mathsf{c}}_{1+\alpha}$-sentence over the empty language, was defined as[‖]

$$\mathop{\bigvee\mkern-15mu\bigvee}_{\substack{\langle m,1+\gamma\rangle\in W_e \\ 1+\gamma<1+\alpha}} \varphi^{\Pi^{\mathsf{c}}_{1+\gamma}}_{m}.$$

[§] We can view $1+\mathcal{L}$ as a computable well-ordering too, and when we write $1+\alpha$, we are thinking of an initial segment of $1+\mathcal{L}$.

[¶] By $\alpha = 0$, we just mean that α is the first element of $\mathcal{L}$.

[‖] We are assuming that $\alpha \neq 0$, so we may assume that Π^{c}_{0} formulas do not show up in the disjunction.

Thus, $\varphi_e^{\Sigma^{\mathsf{c}}_{1+\alpha}}$ holds if and only if

$$\exists \gamma \in \mathcal{L} \restriction \alpha \; \exists m \in \omega \quad (\langle m, 1+\gamma\rangle \in W_e \text{ and } \langle \gamma, f(\gamma, m)\rangle \notin 0^{(\alpha)}).$$

There is a number k such that this holds if and only if $k \in 0^{(\alpha)\prime}$. Let $f(\alpha, e)$ be that number k. Then, we have that $\varphi_e^{\Sigma^{\mathsf{c}}_{1+\alpha}}$ holds if and only if $\langle \alpha, f(\alpha, e)\rangle \in 0^{(\mathcal{L})}$, as needed. □

This theorem gets us a new characterization of the hyperarithmetic sets:

Corollary V.17. *A set $A \subseteq \mathbb{N}$ is hyperarithmetic if and only if $A \leq_T 0^{(\mathcal{L})}$ for some computable well-ordering $\mathcal{L}$.*

V.2.2. Independence on presentation.

Given an ω-presentation of a well-ordering α, there is a unique jump hierarchy along α. But different ω-presentations of α would give different jump hierarchies. The goal of this section is to show that, for computable ordinals α, the Turing degree of $0^{(\alpha)}$ is independent of the ω-presentation of α.

When we have a computable isomorphism between two different ω-presentations of α, it is not too difficult to show that the respective jump hierarchies are Turing equivalent (Lemma V.18). However, the isomorphism between two ω-presentations of an ordinal may be quite hard to compute. We will see that $0^{(\alpha)}$ itself can compute such isomorphisms. We will see how this is just good enough to show that the jump hierarchies along such different ω-presentations of α are still Turing equivalent.

Lemma V.18. *Let α and β be computably isomorphic computable well-orderings. Then $0^{(\alpha)} \equiv_T 0^{(\beta)}$.*

Proof. Let H_α and H_β be the jump hierarchies along α and β, respectively. Let f be the computable isomorphism from α to β. We will use effective transfinite recursion on $a \in \alpha$ to define a computable sequence of indices i_a for Turing reductions such that

$$H_\beta^{[f(a)]} \leq_{T \text{ via } i_a} H_\alpha^{[a]},$$

where $X \leq_{T \text{ via } i} Y$ is shorthand for $\Phi_i^Y = X$. Observe that

$$H_\beta^{[<_\beta f(a)]} = \bigcup_{d \in \beta \restriction f(a)} \{d\} \times H_\beta^{[d]} = \bigcup_{c \in \alpha \restriction a} \{f(c)\} \times \Phi_{i_c}^{H_\alpha^{[c]}}.$$

Since we are using transfinite recursion, we can assume we have access to an index for the computable function $c \mapsto i_c$ for $c \in \alpha \restriction a$. We can then find an index e for the Turing reduction

$$H_\beta^{[<_\beta f(a)]} = \bigcup_{c\in\alpha\restriction a} \{f(c)\} \times \Phi_{i_c}^{H_\alpha^{[c]}} \leq_{T \text{ via } e} \bigcup_{c\in\alpha\restriction a} \{c\} \times H_\alpha^{[c]} = H_\alpha^{[<_\alpha a]}.$$

(Notice that this would not work if we did not assume f was computable, as it will be the case later on.) Once we have e, let i_a be an index for the following Turing reduction

$$H_\beta^{[<_\beta f(a)]\prime} \leq_{T \text{ via } i_a} H_\alpha^{[<_\alpha a]\prime}. \qquad \square$$

In the next lemma, we will show that $0^{(\alpha)}$ can compute the isomorphism between α and another computable copy of α. However, if we want uniformity, we need an extra jump:s

Lemma V.19. *Let α and β be isomorphic computable ω-presentations of an ordinal and let $f\colon \alpha \to \beta$ be the isomorphism between them. Let H_α be the jump hierarchy along α. Then, for every $a \in \mathcal{A}$, $f \restriction \alpha_{(<a)}$ is uniformly computable from $H_\alpha^{[a]}$.*[**]

Proof. We use effective transfinite induction on $a \in \alpha$ to define a computable sequence of indices e_a for Turing reductions such that

$$f \restriction \alpha_{(<a)} \leq_{T \text{ via } e_a} H_\alpha^{[a]}.$$

Consider $a \in \alpha$. We want to find e_a using an index for the computable sequence $\{e_c : c \in \alpha \restriction a\}$.

If a is the first element of α, $f \restriction \alpha_{(<a)}$ is the empty function. Let us assume a is not the first element of α. We split the construction into three cases:

(1) a is a limit ordinal;
(2) $a = b + 1$ and b is a limit ordinal;
(3) $a = b + 1$ and $b = c + 1$.

Use $0''$, which is computable from $H_\alpha^{[a]}$, to determine which case we are in and to find b and c.

Case (1): If a is a limit ordinal, then $f \restriction \alpha_{(<a)} = \bigcup_{c\in\alpha\restriction a} f \restriction \alpha_{(<c)}$. So, using an index for the sequence $\{e_c : c \in \alpha \restriction a\}$, we can figure out an index for

$$f \restriction \alpha_{(<a)} \leq_T H_\alpha^{[<_\alpha a]} \leq_T H_\alpha^{[a]}.$$

(Notice that in this limit case we did not need the full power of $H_\alpha^{[a]}$ and that $H_\alpha^{[<_\alpha a]}$ was enough. We will use this a few times later.)

[**] Recall that $f \restriction D$ is the partial function obtained by restricting f to the domain D.

Case (2): If $a = b+1$ and b is a limit ordinal, then $f \restriction \alpha_{(<a)} = f \restriction \alpha_{(<b)} \cup \{\langle b, f(b)\rangle\}$. We saw before that we can compute $f \restriction \alpha_{(<b)}$ from $H_\alpha^{[<_\alpha b]}$. We now use oracle $H_\alpha^{[a]} \equiv_T H_\alpha^{[<_\alpha b]\prime\prime} \geq_T (f \restriction \alpha_{(<b)})''$ to find $f(b)$, which is the least element of β that is not in the image of $f \restriction \alpha_{(<b)}$. That is, use $(f \restriction \alpha_{(<b)})''$ to find $d \in \beta$ such that

- for all $c \in \alpha \restriction b$, $d \neq f(c)$, and
- for all $e \in \beta \restriction d$, there is some $h \in \alpha \restriction b$ such that $e = f(h)$.

Case (3): Suppose now that $a = b+1 = c+2$. We can use $H_\alpha^{[b]}$ to get an index for $f \restriction \alpha_{(<b)}$. To find $f(b)$, just use $0''$ to find the successor of $f(c)$ in β. □

Theorem V.20. *If α and β are isomorphic computable ω-presentations of an ordinal, then $0^{(\alpha)} \equiv_T 0^{(\beta)}$.*

Proof. Let H_α and H_β be the jump hierarchies along α and β, respectively. Let f be the isomorphism from α to β. From the lemma above, we have a computable sequence of indices e_a such that $f \restriction \alpha_{(<a)} \leq_{T \text{ via } e_a} H_\alpha^{[a]}$.

As in Lemma V.18, we will use effective transfinite recursion on $a \in \alpha$ to define a computable sequence of indices i_a for Turing reductions such that

$$H_\beta^{[f(a)]} \leq_{T \text{ via } i_a} H_\alpha^{[a]}.$$

This time we will have to be a bit more careful. Since we are using transfinite recursion, we can assume we have access to an index for the computable function $c \mapsto i_c$ for $c \in \alpha \restriction a$. First, we want to use $H_\alpha^{[a]}$ to find an index for the Turing reduction

$$H_\beta^{[<_\beta f(a)]} \leq_T H_\alpha^{[<_\alpha a]}.$$

Recall from the proof of Lemma V.18 that

$$H_\beta^{[<_\beta f(a)]} = \bigcup_{c \in \alpha \restriction a} \{f(c)\} \times \Phi_{i_c}^{H_\alpha^{[c]}}.$$

So, using $f \restriction \alpha_{(<a)}$ and the sequence $\{i_c : c \in \alpha \restriction a\}$, we can compute $H_\beta^{[<_\beta f(a)]}$ from $H_\alpha^{[<_\alpha a]}$. However, we know that $f \restriction \alpha_{(<a)}$ is computable from $H_\alpha^{[a]}$ but not necessarily from $H_\alpha^{[<_\alpha a]}$ — it is close though.

We split the construction into two cases:

(1) a is a limit ordinal;
(2) $a = b+1$.

Use $0''$, which is computable from $H_\alpha^{[a]}$, to determine which case we are in and to find b in the latter case.

Case (1): If a is a limit ordinal, we saw in the proof of Lemma V.19 that $f \restriction \alpha_{(<a)} \leq_T H_\alpha^{[<_\alpha a]}$.

Case (2): If not, and $a = b+1$, then we know that $f \restriction \alpha_{(<b)} \leq_T H_\alpha^{[b]} \leq_T H_\alpha^{[<_\alpha a]}$. We are missing the value of $f(b)$ which $H_\alpha^{[a]}$ can compute. Using the value of $f(b)$ as a parameter, we can find an index for the reduction

$$\bigcup_{c \in \alpha \restriction a} \{f(c)\} \times \Phi_{i_c}^{H_\alpha^{[c]}} \leq_T H_\alpha^{[<_\alpha a]}.$$

One way or another, we have shown that $H_\beta^{[<_\beta f(a)]} \leq_T H_\alpha^{[<_\alpha a]}$, and we have used $H_\alpha^{[a]}$ to find an index for that reduction. We can then use $H_\alpha^{[a]}$ to find an index for $H_\beta^{[<_\beta f(a)]\prime} \leq_T H_\alpha^{[<_\alpha a]\prime}$ and thus computably find an index i_a for

$$H_\beta^{[f(a)]} \leq_{T \text{ via } i_a} H_\alpha^{[a]}. \qquad \square$$

Corollary V.21. *If α and β are isomorphic computable ω-presentations of a successor ordinal, then $0^{(\alpha)}$ and $0^{(\beta)}$ are computably isomorphic.*

Proof. Recall that if two sets are Turing equivalent, their jumps are computably isomorphic. □

V.3. Hyperarithmetically infinitary formulas

An infinitary formula is said to be *hyperarithmetically infinitary* if it has a hyperarithmetic tree representation as in Definition III.1.

In this section, we show an important closure property of the hyperarithmetic sets: a set defined in $(\mathbb{N}; +, \times, 0, 1, <)$ by a hyperarithmetically infinitary formula is still hyperarithmetic. If in a rush, the reader may skip this section, as we will not use this result in the rest of the book.

Theorem V.22. *Every hyperarithmetically infinitary formula is equivalent to a computable infinitary formula.*

The rest of this section is dedicated to proving this theorem.

First, every hyperarithmetically infinitary formula is an X-computable infinitary formula for some hyperarithmetic $X \in 2^{\mathbb{N}}$. As in Section III.2, if a formula has an X-computable tree representation, it has a Σ_α^{cX} index for some X-computable well-ordering α. Recall that the Σ_α^{cX}

formula with index e (denoted $\varphi_e^{\Sigma^{cX}_\alpha}$) is the disjunction of the $\exists$-over-$\Pi^{cX}_{<\alpha}$ formulas[††] with indices in W_e^X. That is,

$$\varphi_e^{\Sigma^{cX}_\alpha} \quad \text{is} \quad \bigvee\!\!\!\!\bigvee_{\substack{\langle i,\beta\rangle\in W_e^X\\ \beta<\alpha}} \exists\bar{y}\ \ \varphi_i^{\Pi^{cX}_\beta}(\bar{x},\bar{y}).$$

(In Section III.2, we also used a sub-index j describing the arity of the formula. We omit it here to simplify the notation.)

Let $\mathcal{L}$ be a hyperarithmetic well-ordering extending α, so that we can think of α as a member of $\mathcal{L}$. We want to show that every Σ^{cX}_β formula, for $\beta\in\mathcal{L}$, is equivalent to a computable one. There are two obstacles. The first obstacle is that the infinitary disjunctions and conjunctions are not c.e. but X-c.e. The second obstacle is that the ordinals $\mathcal{L}\upharpoonright\beta$ indexing the complexity classes are also not computable but X-computable. We will resolve the first issue by recursively applying Lemma V.10, which states that a Σ^0_α disjunction of Σ^c_α formulas is equivalent to a Σ^c_α formula. We will resolve the second issue Spector's theorem (Theorem IV.19), which states that every hyperarithmetic well-ordering has a computable copy.

By Spector's theorem, there is a computable well-ordering $\mathcal{K}$ isomorphic to $\mathcal{L}$. Furthermore, the isomorphism $h\colon\mathcal{L}\to\mathcal{K}$ is hyperarithmetic: Recall from Lemma II.18 that there are computable infinitary formulas $\psi_\gamma(x)$ for $\gamma\in\mathcal{K}$ such that

$$\mathcal{L}\models\psi_\gamma(\alpha) \iff \mathcal{L}\upharpoonright\alpha\cong\mathcal{K}\upharpoonright\gamma \iff h(\alpha)=\gamma.$$

The formulas ψ_γ are defined computably uniformly in $\gamma\in\mathcal{K}$. Since $\mathcal{L}$ and $\mathcal{K}$ are ω-presentations, we can think of h as a hyperarithmetic function $\mathbb{N}\to\mathbb{N}$. Let Z be a hyperarithmetic real that computes X and computes the isomorphism h from $\mathcal{L}$ to $\mathcal{K}$.

Lemma V.23. *Every $\Sigma^{cX}_{<\mathcal{L}}$ formula is equivalent to a $\Sigma^{cZ}_{<\mathcal{K}}$ formula.*[‡‡]

Proof. Using Z-effective transfinite recursion (Theorem I.33), define a Z-computable function $g\colon\mathcal{L}\times\mathbb{N}\to\mathbb{N}$ that, for each $\alpha\in\mathcal{L}$ and $e\in\mathbb{N}$, produces an index $g(\alpha,e)$ for a $\Sigma^{cZ}_{h(\alpha)}$ formula equivalent to the eth Σ^{cX}_α formula. That is, $g(\alpha,e)$ will be defined so that

$$\varphi_{g(\alpha,e)}^{\Sigma^{cZ}_{h(\alpha)}} \iff \varphi_e^{\Sigma^{cX}_\alpha}.$$

[††] The $\exists$-over-$\Pi^{cX}_{<\alpha}$ formulas are the ones generated from the $\Pi^{cX}_{<\alpha}$ and $\Sigma^{cX}_{<\alpha}$ formulas using finitary conjunctions and disjunctions and existential quantifiers.

[‡‡] Recall that a $\Sigma^{cX}_{<\mathcal{L}}$ formula is a Σ^{cX}_β formula for some $\beta\in\mathcal{L}$.

For this, let $g(\alpha, e)$ be an index for the Z-c.e. set given by

$$W^Z_{g(\alpha,e)} \quad = \quad \{\langle g(\beta, i), h(\beta)\rangle : \beta \in \mathcal{L} \restriction \alpha \text{ and } \langle i, \beta\rangle \in W^X_e\},$$

so we get that

$$\varphi_{g(\alpha,e)}^{\Sigma^{\mathsf{c}Z}_{h(\alpha)}} \quad \text{is} \quad \mathop{\bigvee\!\!\!\!\bigvee}_{\substack{\langle i,\beta\rangle\in W^X_e\\ \beta<\alpha}} \exists \bar{y}\ \ \varphi_{g(\beta,i)}^{\Pi^{\mathsf{c}Z}_{h(\beta)}}(\bar{x}, \bar{y}),$$

which by transfinite induction is equivalent to $\varphi_e^{\Sigma^{\mathsf{c}X}_\alpha}$. □

The next step is to show that every $\Sigma^{\mathsf{c}Z}_{<\mathcal{K}}$ formula is equivalent to a computable infinitary formula — now knowing that $\mathcal{K}$ is computable. Let π be a computable ordinal that is large enough that Z is Δ^0_π.*

Lemma V.24. *For every $\gamma \in \mathcal{K}$, every $\Sigma^{\mathsf{c}Z}_\gamma$ formula is equivalent to a $\Sigma^{\mathsf{c}}_{\pi+\gamma}$ formula.*

Proof. We use effective transfinite recursion (Theorem I.33) to define a function $f : \mathcal{K} \times \mathbb{N} \to \mathbb{N}$ such that, for $\gamma \in \mathcal{K}$ and $e \in \mathbb{N}$, $f(\gamma, e)$ is an index for a $\Sigma^{\mathsf{c}}_{\pi+\gamma}$ formula equivalent to the eth $\Sigma^{\mathsf{c}Z}_\gamma$ formula; that is, $f(\gamma, e)$ will be defined so that

$$\varphi_{f(\gamma,e)}^{\Sigma^{\mathsf{c}}_{\pi+\gamma}} \iff \varphi_e^{\Sigma^{\mathsf{c}Z}_\gamma}.$$

Recall that we defined

$$\varphi_e^{\Sigma^{\mathsf{c}Z}_\gamma} \qquad \text{as} \qquad \mathop{\bigvee\!\!\!\!\bigvee}_{\substack{\langle i,\delta\rangle\in W^Z_e\\ \delta\in\mathcal{K}\restriction\gamma}} \exists \bar{y}\ \ \varphi_i^{\Pi^{\mathsf{c}Z}_\delta}.$$

Using the same idea as in Lemma V.10, this is equivalent to

$$\mathop{\bigvee\!\!\!\!\bigvee}_{\langle i,\delta\rangle\in\mathbb{N}\times\mathcal{K}\restriction\gamma} \langle i, \delta\rangle \in W^Z_e \ \wedge\ \exists \bar{y}\ \ \varphi_i^{\Pi^{\mathsf{c}Z}_\delta}. \tag{3}$$

Recall that we chose π so that W^Z_e is Σ^0_π. Thus, the formula "$\langle i, \delta\rangle \in W^Z_e$" can be replaced by a Σ^{c}_π sentence $\psi_{e,\langle i,\delta\rangle}$ over the empty vocabulary (Lemma V.3), defined uniformly on e and $\langle i, \delta\rangle$.

To define $f(\gamma, e)$ recursively, we define an auxiliary function $\tilde{f}$. Let $\tilde{f}(\delta, i)$ be the index of the $\Pi^{\mathsf{c}}_{\pi+\delta}$ formula "$\langle i, \delta\rangle \in W^Z_e \ \wedge\ \varphi_{f(\delta,i)}^{\Pi^{\mathsf{c}}_{\pi+\delta}}$." By the induction hypothesis, the formula (3) is equivalent to

$$\mathop{\bigvee\!\!\!\!\bigvee}_{\langle i,\delta\rangle\in\mathbb{N}\times\mathcal{K}\restriction\gamma} \exists \bar{y}\ \big(\underbrace{\langle i, \delta\rangle \in W^Z_e \ \wedge\ \varphi_{f(\delta,i)}^{\Pi^{\mathsf{c}}_{\pi+\delta}}}_{\varphi_{\tilde{f}(\delta,i)}^{\Pi^{\mathsf{c}}_{\pi+\delta}}}\big).$$

[*] So that W^Z_e is Σ^0_π for all e.

Finally, we define $f(\gamma, e)$ as the index for the c.e. set

$$W_{f(\gamma,e)} \quad = \quad \{\langle \tilde{f}(i,\delta), \pi + \delta\rangle : \langle i, \delta\rangle \in \mathbb{N} \times \mathcal{K} \restriction \gamma\}. \qquad \square$$

V.4. Complexity classes in Cantor Space

Infinitary formulas can also be used to define sets of reals. Consider formulas $\varphi(D)$ in the language of arithmetic with an extra unary relation symbol D, which we treat as a second-order variable.[†] Let us call these, $\mathbb{N}$-*formulas*. If Γ is a class of $\mathbb{N}$-formulas, a set $\mathfrak{A} \subseteq 2^{\mathbb{N}}$ is said to be Γ-definable if there is an $\mathbb{N}$-formula $\varphi(D)$ in Γ such that $\mathfrak{A} = \{A \in 2^{\mathbb{N}} : \varphi(A)\}$. As in Lemma V.3, one can show that every $\mathbb{N}$-$\mathcal{L}_{\omega_1,\omega}$-formula $\varphi(D)$ is equivalent to a quantifier-free one in the vocabulary containing only the relation D, where, by 'equivalent,' we mean equivalent when evaluated within the structure of the natural numbers. To see this, replace sub-formulas of the form $\forall x \varphi(x)$ and $\exists x(\varphi(x))$ by $\bigwedge\!\!\!\bigwedge_{n\in\mathbb{N}} \varphi(\mathbf{n})$ and $\bigvee\!\!\!\bigvee_{n\in\mathbb{N}} \varphi(\mathbf{n})$ respectively, where $\mathbf{n}$ is short for $\overbrace{1 + \cdots + 1}^{n \text{ times}}$, and then replace atomic sentences that do not use D by $\top$ or $\bot$ depending on whether they are true or false.

The standard topology on $2^{\mathbb{N}}$ has a sub-base of open sets that consist of the sets of the form

$$O_{n,i} = \{A \in 2^{\mathbb{N}} : A(n) = i\},$$

for $n \in \mathbb{N}$ and $i \in 2$. With this topology, the open sets are the countable unions of finite intersections of sub-basic open sets, which are exactly those defined by infinitary disjunctions of finite conjunctions of formulas of the form $D(\mathbf{n})$ or $\neg D(\mathbf{n})$, namely the $\Sigma^{\mathtt{in}}_1$-formulas. The closed sets are the $\Pi^{\mathtt{in}}_1$ definable sets.

Definition V.25. The class of *Borel sets* is the smallest class of subsets of $2^{\mathbb{N}}$ which contains all the sub-basic open sets and is closed under countable unions, countable intersections, and complements.

Theorem V.26. *A set $\mathfrak{A}$ is Borel if and only if it is* $\mathbb{N}$-$\mathcal{L}_{\omega_1,\omega}$-*definable.*

Proof. First observe that the class of $\mathbb{N}$-$\mathcal{L}_{\omega_1,\omega}$-definable sets contains all the basic open sets is closed under countable unions, countable intersections, and complements, because the $\mathcal{L}_{\omega_1,\omega}$-formulas include the quantifier-free formulas and are closed under countable disjunctions,

[†] This is still an infinitary first-order formula which, on top of the standard vocabulary of first-order arithmetic, has atomic sub-formulas of the form $D(x)$ for x ranging over the natural numbers.

countable conjunctions, and negations. So, all Borel sets are $\mathcal{L}_{\omega_1,\omega}$-definable.

The other direction is proved by transfinite induction on the complexity of the $\mathcal{L}_{\omega_1,\omega}$ definition of the set. Suppose that $\mathfrak{A}$ is $\Sigma^{\mathtt{in}}_{\alpha}$ definable, say by the formula $\bigvee\!\!\!\bigvee_{i\in I} \exists \bar{y}_i(\varphi_i(\bar{y}_i, D))$ where each φ_i is $\Pi^{\mathtt{in}}_{<\alpha}$. We can then write $\mathfrak{A}$ as the countable union of the sets defined by the formulas $\varphi_i(\bar{\mathbf{n}}, D)$ for $i \in I$ and $\bar{n} \in \mathbb{N}^{|\bar{y}_i|}$, and by the induction hypothesis, we may assume each of these sets is Borel. It follows that $\mathfrak{A}$ is Borel too. □

Theorem V.27. *(Souslin) A set $\mathfrak{A}$ is Borel if and only if it Δ^1_1 relative to some oracle $A \in 2^{\mathbb{N}}$.*

Proof. For the ($\Rightarrow$) direction, suppose that $\mathfrak{A}$ is Borel and hence $\mathcal{L}_{\omega_1,\omega}$-definable. Let A be such that $\mathfrak{A}$ is $\mathcal{L}_{\mathsf{c},\omega}$-definable relative to A. Recall from Observation III.4 that there is a Σ^1_1 formula that decides if an infinitary sentence is true. We thus have that $\mathfrak{A}$ is Σ^1_1 relative to A. Applying the same argument on its complement, we get that $\mathfrak{A}$ is Π^1_1, and thus Δ^1_1, relative to A.

For the ($\Leftarrow$) direction, apply Corollary IV.6 to the complement of $\mathfrak{A}$ to get an A-computable tree T such that

$$X \in \mathfrak{A} \quad\iff\quad \forall F\ (X \oplus F \notin [T]).$$

For each X, let

$$T^X = \{\sigma \in \mathbb{N}^{<\mathbb{N}} : (X \upharpoonright |\sigma|) \oplus \sigma \in T\}.$$

So, we have that $X \in \mathfrak{A}$ if and only if T^X is well-founded, and only if $(\mathcal{T}^X; \leq_{\text{KB}})$ is well-ordered. The set $\{(T^X; \leq_{KB}) : X \in \mathfrak{A}\}$ is a $\Sigma^1_1(A)$ set of well-orderings, and hence from the Σ^1_1-bounding theorem (Theorem IV.13), we get an A-computable ordinal α such that for every $X \in \mathfrak{A}$, $(T^X; \leq_{KB})$ is below α. Using Lemma II.4, and using T as an oracle, one can write a $\mathbb{N}$-$\mathcal{L}_{\omega_1,\omega}$-formula $\varphi(X)$ which says that $(T^X; \leq_{KB})$ is below α. This shows that $\mathfrak{A}$ is Borel. □

Remark V.28. If $\mathfrak{A}$ is Δ^1_1, it is not just $\mathbb{N}$-$\mathcal{L}_{\omega_1,\omega}$-definable, but also $\mathbb{N}$-$\mathcal{L}_{\mathsf{c},\omega}$-definable. To see this notice in the ($\Leftarrow$) direction of the proof above, when $A = \emptyset$, the tree T and the ordinal α can be taken to be computable, and hence the formula φ is computable too.

V.4.1. The space of presentations. Fix a vocabulary τ. Let Mod_τ be the set of all ω-presentations of all τ-structures. Each τ-structure is determined by its diagram, so Mod_τ is essentially $2^{\mathbb{N}}$, with the difference that we think of the elements Mod_τ as ω-presentations of τ-structures instead of binary sequences. We equip Mod_τ with the same topology as $2^{\mathbb{N}}$. Notice that the basic open sets are now of the form

$$\{\mathcal{K} \in Mod_\tau : \mathcal{K} \models \varphi_n^{\mathrm{at}}[x_j \mapsto j : j \in \mathbb{N}]\}$$

and

$$\{\mathcal{K} \in Mod_\tau : \mathcal{K} \models \neg\varphi_n^{\mathrm{at}}[x_j \mapsto j : j \in \mathbb{N}]\},$$

where φ_n^{at} is the nth atomic τ-formula.

Given a τ-sentence φ, we let

$$Mod(\varphi) = \{\mathcal{K} \in Mod_\tau : \mathcal{K} \models \varphi\}.$$

As in Lemmas V.6 and V.7 one can see that, if φ is a $\Sigma_\alpha^{\mathtt{c}}$ sentence, then $Mod(\varphi)$ is Σ_α^0. We will see in our chapter on forcing that the converse is also true (Theorem VII.25).

CHAPTER VI

Overspill

VI.1. Non-standard jump hierarchies

We saw in Section V.2 that over every computable well-ordering we have a jump hierarchy, and that it is unique. The definition was for jump hierarchies over linear orderings in general, but we did not say much about what happens when the linear ordering is not well-ordered. The following lemma uses an overspill argument to show that there are jump hierarchies over non-well-ordered computable linear orderings.

Lemma VI.1. *There is a non-well-ordered computable linear ordering over which there exists a jump hierarchy.*

Proof. Let J be the set of indices of computable linear orderings over which there exists a jump hierarchy.

$$J = \{e \in \mathbb{N} : \exists H \subseteq L_e \times \mathbb{N}\ \forall a \in L_e\ (H^{[a]} = (H^{[<a]})')\}.$$

Deciding if a set H is a jump hierarchy over a linear ordering $\mathcal{L}_e$ is a Π^0_2 property of H and e. Thus, J is Σ^1_1. As we saw in the previous chapter, over every well-ordering there is a jump hierarchy. So we have that

$$\mathcal{O}_{\text{wo}} \subseteq J.$$

We proved in IV.9 that $\mathcal{O}_{\text{wo}}$ is not Σ^1_1 and that it is actually Π^1_1-complete. So, J cannot be equal to $\mathcal{O}_{\text{wo}}$; it must *overspill*! That is, $\mathcal{O}_{\text{wo}}$ must be a proper subset of J, and there must exist some $e \in J \smallsetminus \mathcal{O}_{\text{wo}}$ which is an index for a non-well-ordered computable linear ordering over which there is a jump hierarchy. □

These jump hierarchies are hard to visualize, as there does not seem to be a way to build them. The lemma above just says they exist. The next lemma shows that indeed, they cannot be hyperarithmetic.

Lemma VI.2. *Let $\{X_i : i \in \mathbb{N}\}$ be a sequence of reals such that $X'_{i+1} \leq_T X_i$ for every i. Then all the X_i's compute all the hyperarithmetic sets.*

Proof. We prove that, for every $\alpha < \omega_1^{CK}$, every X_i computes $0^{(\alpha)}$ by transfinite induction on α. This is obvious for $\alpha = 0$. Assume this is true for α. Then for every i, since X_{i+1} computes $0^{(\alpha)}$, X_i computes $0^{(\alpha)\prime} \equiv_T 0^{(\alpha+1)}$, and hence it is true for $\alpha+1$. For a limit ordinal λ, suppose that every X_i computes every $0^{(\beta)}$ for $\beta < \lambda$. Observe that $0^{(\lambda)} \equiv_T \oplus_{\beta<\lambda} 0^{(\beta)}$. The fact that X_i computes each $0^{(\beta)}$ does not mean that it computes them uniformly – we need a couple of jumps to get that uniformity: Given e, X'' can check if φ_e^X is a jump hierarchy along β (recall that checking this is Π^0_2). That is, X_i can compute the set of pairs $\langle \beta, e\rangle$ such that $\varphi_e^{X_{i+2}}$ is a jump hierarchy along β. It can then compute their join and hence compute $0^{(\lambda)}$. □

Let HYP be the class of all hyperarithmetic sets.

Theorem VI.3. *(Spector–Gandy* [**Spe60, Gan60**]*) If $\psi(X)$ is a Π^1_1 formula of arithmetic, then*

$$\exists X \in \mathit{HYP}\ \psi(X)$$

is equivalent to a Π^1_1 formula.

Conversely, every Π^1_1 formula $\varphi(Y)$ is equivalent to one of the form $\exists X \in \mathit{HYP}\ (\psi(X))$, where ψ is Π^0_2.

The formulas above may have 1st- or 2nd-order free variables.

Proof. For the first part, the idea is to replace the second-order quantifier "$\exists X \in$ HYP" with a first-order quantifier over the indices of hyperarithmetic sets. Let $\varphi_e^{\Sigma^{\mathtt{c}}_a}(x)$ be the eth $\Sigma^{\mathtt{c}}_{\mathcal{L}_a}$-formula with one free variable x, where $\mathcal{L}_a$ is the computable linear ordering with index a (as in Lemma I.29). Notice that for $\varphi_e^{\Sigma^{\mathtt{c}}_a}(x)$ to be an $\mathcal{L}_{\mathtt{c},\omega}$ formula, we need to have $a \in \mathcal{O}_{\text{wo}}$. Therefore, the set of pairs $\langle a, e\rangle$ which can be used as indices for $\mathcal{L}_{\mathtt{c},\omega}$ formulas is Π^1_1. We then have that

$$\exists X \in \text{HYP}\ \psi(X) \iff \exists a, e \in \mathbb{N}\ (a \in \mathcal{O}_{\text{wo}}\ \&\ \forall X(\text{if } \varphi_e^{\Sigma^{\mathtt{c}}_a}(x) \text{ defines } X \to \psi(X))).$$

Recall that satisfaction of $\mathcal{L}_{\omega_1,\omega}$ formulas is Δ^1_1 (Observation III.4) and hence that saying that a formula $\varphi_e^{\Sigma^{\mathtt{c}}_a}(x)$ defines a set X, namely

$$\forall n\ (n \in X \leftrightarrow \varphi_e^{\Sigma^{\mathtt{c}}_a}(n)),$$

is a Δ^1_1 property of X, a and e. So the right-hand-side of the equivalence above is Π^1_1.

For the second part, let $\mathcal{L}_\varphi$ be a linear ordering such that φ holds if and only if $\mathcal{L}_\varphi$ is well-ordered. We can build $\mathcal{L}$ uniformly from φ and the parameters in φ. We know from Section V.2 that if $\mathcal{L}_\varphi$ is well-ordered, there exists a jump hierarchy on it and this hierarchy is hyperarithmetic. Conversely, if $\mathcal{L}_\varphi$ is not well-ordered, then either there is no jump hierarchy over it, or if there is one, it cannot be hyperarithmetic by the previous lemma. We then get that

$$\varphi \iff \exists H \in \text{HYP } (H \text{ is a jump hierarchy over } \mathcal{L}_\varphi). \qquad \square$$

Jump hierarchies over ill-ordered linear orderings produce $\leq_T$-descending sequences $\{X_i : i \in \omega\}$ satisfying $X'_{i+1} \leq_T X_i$ for all i. As we see in the following lemma, such sequences cannot be uniform.

Lemma VI.4 (Steel [**Ste75**]). *There is no sequence $\{X_i : i \in \mathbb{N}\}$ where X_i computes X'_{i+1} for all i uniformly, that is, where for some computable operator Γ, $X'_{i+1} = \Gamma(X_i)$ for all $i \in \mathbb{N}$.*

Proof. Assume such a sequence exists. Using the recursion theorem, we will find an $e_0 \in \mathbb{N}$ such that for all $i \in \mathbb{N}$,

$$e_0 \in X'_i \iff (\exists j > i)\ e_0 \notin X'_j.$$

Before showing the details of how to find such an e_0, let us show how we get a contradiction from it. If for some $i_0 \in \mathbb{N}$ we have $e_0 \in X'_{i_0}$, then for some $i_1 > i_0$, $e_0 \notin X'_{i_1}$. Thus, one way or another, there exists i_1 with $e_0 \notin X'_{i_1}$. Then, for all $j > i_1$, $e_0 \in X'_j$. But if $e_0 \in X'_{i_1+1}$, there must exits $i_2 > i_1$ with $e_0 \notin X'_{i_2}$, contradicting the previous line.

Let us now prove that such an e_0 exists. Using Γ, find a computable operator Φ such that for all X_j, $\Phi(X_j) = X_{j+1}$. Given k, let $\Gamma_k = \Gamma \circ \Phi^{k-1}$. This way, we have that, for $i < j$, $X'_j = \Gamma_{j-i}(X_i)$. Now, to apply the recursion theorem, we define a computable function f on indices of computable operators as follows: Given e, let $f(e)$ be the index of a computable operator such that, for all n,

$$\Phi^X_{f(e)}(n)\downarrow \iff \exists k > 0\ e \notin \Gamma_k(X).$$

Using the recursion theorem, let e_0 be such that $\Phi^X_{f(e_0)} = \Phi^X_{e_0}$ for all X. Substituting X_i for X and e_0 for e, $f(e)$, and n above, we get

$$e_0 \in X'_i \iff \Phi^{X_i}_{e_0}(e_0)\downarrow \iff \exists k > 0\ e_0 \notin X'_{i+k}. \qquad \square$$

Exercise VI.5. Prove that if $\mathcal{L}$ is a computable linear ordering over which there is a jump hierarchy, then $\mathcal{L}$ has no hyperarithmetic descending sequences. See hint in footnote.[*]

[*] Use the same idea as in the previous lemma, using the fact that the columns along a hyperarithmetic descending sequence compute the sequence.

VI.1.1. Harrison's linear ordering. The Harrison linear ordering is one of the most interesting objects in higher computable structure theory. It is a computable linear ordering with an initial segment isomorphic to ω_1^{CK}. It will thus allow us to fix indices for all ordinals below ω_1^{CK}. It follows from Exercise VI.5 that there is a computable, non-well-founded linear ordering without hyperarithmetic descending sequences. We give a more direct proof:

THEOREM VI.6. *(Harrison* [**Har68**]*) There is a computable linear ordering that is not well-ordered but has no hyperarithmetic descending sequences.*

PROOF. Let S be the set of indices of computable linear orderings without hyperarithmetic descending sequences. Since well-orders have no descending sequences, S contains all of Kleene's $\mathcal{O}_{\mathrm{wo}}$. The set S is definable by a formula of the form $\neg\exists X \in \mathrm{HYP}\ \phi(X)$ where ϕ is arithmetic, and hence by the Spector–Gandy Theorem VI.3 applied to the negation of this formula, the set S is Σ_1^1. Therefore, the set S must *overspill*. That is, since $S \supseteq \mathcal{O}_{\mathrm{wo}}$ and S is Σ_1^1, we must have $\mathcal{O}_{\mathrm{wo}} \subsetneq S$. Any element of $S \smallsetminus \mathcal{O}_{\mathrm{wo}}$ is an index for a computable, ill-founded linear ordering without hyperarithmetic descending sequences. □

Note that a computable linear ordering has no hyperarithmetic descending sequences if and only if every hyperarithmetic subset has a least element. This is because, given a hyperarithmetic descending sequence $\{a_n : n \in \mathbb{N}\}$, the set $\{b \in L : \exists n\ (a_n <_L b)\}$ is hyperarithmetic and has no least element, and conversely, given a hyperarithmetic set A with no least element, the sequence defined by a_{n+1} as the $\leq_{\mathbb{N}}$-least $b \in A$ with $b <_L a_n$ is a hyperarithmetic descending sequence.

THEOREM VI.7. *(*[**Har68**]*) Every computable linear ordering without hyperarithmetic descending sequences is isomorphic to an initial segment of $\omega_1^{CK} + \omega_1^{CK} \cdot \mathbb{Q}$.*

PROOF. Let $\mathcal{L}$ be a computable linear ordering without hyperarithmetic descending sequences. Consider the equivalence relation on $\mathcal{L}$ defined by $a \sim b$ if the interval $[a, b]$ in $\mathcal{L}$ is well-ordered. This is, of course, a *convex* equivalence relation in the sense that if $a < b < c$ and $a \sim c$, then $a \sim b \sim c$. We will prove the following three facts about $\sim$ that together imply that $\mathcal{L}$ is isomorphic to an initial segment of $\omega_1^{CK}(1 + \mathbb{Q})$:

(1) Every equivalence class is well-ordered.
(2) The quotient is isomorphic to either 1, $1 + \mathbb{Q}$, or $1 + \mathbb{Q} + 1$.

(3) Every equivalence class has order type ω_1^{CK} except possibly the last one. If there is a last one, it must be isomorphic to a proper initial segment of ω_1^{CK}.

Part (1) is the crux of the proof. Pick an element $b \in L$. It is easy to see that the upper half of b's equivalence class, $\{c \in L : c >_L b \wedge b \sim c\}$, is well-ordered by $\mathcal{L}$. To show that the bottom half, $\{a \in L : a <_L b \wedge a \sim b\}$, is well-ordered, we must show it has a first element. Notice that as a decreases within the $\sim$-equivalence class of b, the order type of $[a, b]$ either grows or stays the same. If there is a maximum order type among the order types of $[a, b]$ for $a \sim b$, say α, then we would have that, for $a <_L b$, $a \sim b$ if and only if $[a, b] \preccurlyeq \alpha$, which we know we can decide in a hyperarithmetic way by Theorem V.9. We could then conclude that there is a least such a, as every hyperarithmetic subset of $\mathcal{L}$ as a least element.

Suppose now that is not the case. That is, suppose that there is no maximum order type among the order types of $[a, b]$ for $a \sim b$. This will lead us to a contradiction. Let us define a sequence of elements $b >_L a_0 >_L a_1 >_L \cdots$, all equivalent to b as follows. Let a_0 be the least among all the elements $a <_L b$ for which the order type $[a, b]$ is least possible. That is, $a_0 <_L b$ satisfies that, for all $c \in [a_0, b)$ we have $[a_0, b] \cong [c, b]$, and for all $c < a_0$, $[a_0, b] + 1 \preccurlyeq [c, b]$. We continue on by recursion. Given a_n, let a_{n+1} be the least among all the elements $a <_L a_n$ for which the order type $[a, b]$ is least possible. Since we are assuming there is no maximum order type among the order types of $[a, b]$ for $a \sim b$, this process will continue for all $n \in \mathbb{N}$. Notice that a_{n+1} is defined from a_n by the following Σ_1^1 property: for all $c \in [a_{n+1}, a_n)$ we have $[a_{n+1}, b] \cong [c, b]$, and for all $c < a_{n+1}$, $[a_{n+1}, b] + 1 \preccurlyeq [c, b]$. One can then show that the set $\{a_n : n \in \mathbb{N}\}$ is Σ_1^1, as it can be defined as the set of all a's for which there is a finite sequence $a = a_n <_L a_{n-1} <_L \cdots <_L a_0 <_L b$, where each a_{i+1} is defined from a_i as above. The set of computable well-orderings $\{[a_n, b] : n \in \mathbb{N}\}$ is then Σ_1^1, and hence bounded below ω_1^{CK}. Let $\alpha \in \omega_1^{CK}$ be the supremum of this set. Then, we would have that, for $c <_L b$,

$$(\exists n \in \mathbb{N})\ a_n <_L c \quad \iff \quad [c, b] + 1 \preccurlyeq \alpha.$$

The set of all these c would then be hyperarithmetic and have no least element, contradicting our assumptions on $\mathcal{L}$.

For part (2), first observe that since $\mathcal{L}$ must have a first element, so must its quotient. What is left to prove is that the quotient has no adjacent classes: If $a <_L b$ were in adjacent equivalence classes, $[a, b]$ would be the sum of two well-orders and hence well-ordered itself, and a and b would actually be equivalent.

For part (3), if a belonged to a class isomorphic to some $\alpha < \omega_1^{CK}$ but not to the last class, then the set of $c >_L a$ such that $\alpha + 1 \preccurlyeq [a, c]$ would be hyperarithmetic and have no least element.

If there is a last equivalence class, since it has a first element, the class would be computable. Since it is well-order, it must be isomorphic to some $\alpha < \omega_1^{CK}$. □

We call $\omega_1^{CK} + \omega_1^{CK} \cdot \mathbb{Q}$ the *Harrison linear ordering*. By the previous theorems, it has a computable copy which does not have any hyperarithmetic descending sequence. We denote it by $\mathcal{H}$.

Exercise VI.8. Show that if $\mathcal{L}$ is a computable linear ordering with an initial segment isomorphic to ω_1^{CK}, it must have an initial segment isomorphic to $\omega_1^{CK} + \omega_1^{CK} \cdot \mathbb{Q}$. See hint in footnote.[†]

Lemma VI.9. *$\mathcal{H}$ has Scott rank at least $\omega_1^{CK} + 1$.*

We will show in Corollary VI.19 that $\omega_1^{CK} + 1$ is the largest Scott rank a computable structure can have and hence that $SR(\mathcal{H}) = \omega_1^{CK} + 1$.

Proof. Let a be an element that is the first in a copy of ω_1^{CK} other than the first copy.[‡] The automorphism orbit of a consists of all the elements that are the first in a copy of ω_1^{CK} other than the first copy. We claim this orbit is not $\Sigma^{\mathtt{in}}_{\omega_1^{CK}}$ definable. Recall that if an orbit is $\Sigma^{\mathtt{in}}_{\omega_1^{CK}}$ definable, it must be $\Sigma^{\mathtt{in}}_{\alpha}$ definable for some $\alpha < \omega_1^{CK}$. But the orbit of a cannot be $\Sigma^{\mathtt{in}}_{\alpha}$-definable, because if we let $b = a + \omega^\alpha$, then the intervals above a and b are isomorphic to each other, and the intervals below are isomorphic to $\omega^\alpha \cdot \omega_1^{CK} \cdot (1 + \mathbb{Q})$ and $\omega^\alpha(\cdot\omega_1^{CK} \cdot (1 + \mathbb{Q}) + 1)$, respectively. By Lemma II.38, these are 2α-back-and-forth equivalent and hence satisfy the same $\Sigma^{\mathtt{in}}_{\alpha}$ formulas.

We have proved that $\mathcal{H}$ is not $\Sigma^{\mathtt{in}}_{\omega_1^{CK}}$-atomic. If we add parameters $p_1 < \cdots < p_k$, $(\mathcal{H}, \bar{p})$ is still not $\Sigma^{\mathtt{in}}_{\omega_1^{CK}}$-atomic because, for some $i \in \{0, ..., k\}$, the interval $[p_i, p_{i+1}]$ (where $p_0 = -\infty$ and $p_{k+1} = +\infty$) is isomorphic to $\mathcal{H}$ and hence not $\Sigma^{\mathtt{in}}_{\omega_1^{CK}}$-atomic. □

Exercise VI.10. Prove that all the automorphism orbits in the ill-founded part of $\mathcal{H}$ are $\Pi^{\mathtt{in}}_{\omega_1^{CK}}$.

Lemma VI.11. *There is a computable operator $\mathcal{H}$ such that for every $X \in 2^{<\mathbb{N}}$, $\mathcal{H}^X$ is an ω-presentation of the Harrison linear ordering*

[†] Show that the set of $b \in L$ for which there is a hyperarithmetic descending sequence starting at b is Π^1_1.

[‡] By a copy of ω_1^{CK} within $\mathcal{H}$, we mean a maximal interval isomorphic to ω_1^{CK}.

relative to X. That is, it has order type $\omega_1^X \cdot (1+\mathbb{Q})$ and has no X-hyperarithmetic descending sequences.

Proof. The set of $Y \in 2^{\mathbb{N}}$ which are not hyperarithmetic in X is Σ_1^1 in X (Lemma IV.21). Thus, we can build a tree $\mathcal{T}^X$ whose paths are of the form $Y \oplus Z$, where Y is not hyperarithmetic in X, and Z is a witness that Y is not. Furthermore, $\mathcal{T}^X$ is computably uniform in X. In other words, we consider the tree $\mathcal{T}$ corresponding to the Π_1^0 class of reals $X \oplus Y \oplus Z \in \omega^\omega$ such that Z is a witness for the Σ_1^1 formula that says that Y is not hyperarithmetic in X (Corollary IV.6), and then let $\mathcal{T}^X$ be as in the proof of Corollary IV.7 or Theorem IV.17. This tree $\mathcal{T}^X$ is not well-founded for any X, as there are lots of Y's which are not hyperarithmetic in X. But it has no path hyperarithmetic in X. The Kleene–Brouwer ordering of this tree is then ill-founded (Definition I.24). Furthermore, if we look into the proof of Theorem I.26, we can see that if f is a descending sequence in the Kleene–Brouwer ordering of a tree, its jump, f', can compute a path through the tree (as it is obtained using a limit). Thus, in the current case, our Kleene–Brouwer ordering has no X-hyperarithmetic descending sequence. $(\mathcal{T}^X, \leq_{\text{KB}})$ is thus isomorphic to an initial segment of $\omega_1^X(1+\mathbb{Q})$. Let $\mathcal{H}^X = (\mathcal{T}^X, \leq_{\text{KB}}$ $) \times \omega$. $\mathcal{H}^{\mathcal{X}}$ still has no hyperarithmetic descending sequences and it is now isomorphic to $\omega_1^X \cdot (1+\mathbb{Q})$. □

We can even assume that in $\mathcal{H}^X$, the basic operations on ordinals, like successor, addition, and deciding if an element is a limit or a successor, are all computable. To see this, we have to observe that if $\mathcal{A}$ is any ordinal, $\omega^{\mathcal{A}}$ is an ordinal where all these operations are computable. It is not hard to see that $\omega^{\mathcal{H}} \cong \mathcal{H}$. If we also want multiplication to be computable, one would need to consider $\omega^{\omega^{\mathcal{A}}}$. If we also want exponentiation to be computable, we would need to consider $\epsilon_{\mathcal{A}}$ as in [**MM11**].

Exercise VI.12 (Jockusch [**Joc68**, Theorem 4.1(3) and Corollary 4.3]). The ω_1^{CK} initial segment of $\mathcal{H}$ is clearly Π_1^1 and not Σ_1^1. Prove that it is not Π_1^1-complete. See hint in footnote. [§]

VI.2. Structures of high Scott rank

If a structure is computable, does it have a computable Scott sentence? The answer is no, and the Harrison linear ordering is the main counterexample. We show below that a computable structure has a computable Scott sentence if and only if its Scott rank is computable.

[§] Use a priority argument to diagonalize against all computable many-one reductions from a Π_1^1 set you build. It is enough to build a Σ_2^0 set.

Definition VI.13. A computable structure whose Scott rank is not a computable ordinal is said to have *high Scott rank.*

More generally, we define $\omega_1^{\mathcal{A}} = \min\{\omega_1^{D(\mathcal{B})} : \mathcal{B} \cong \mathcal{A}\}$. Thus, if $\mathcal{A}$ has a computable ω-presentation, $\omega_1^{\mathcal{A}} = \omega_1^{CK}$. A structure, computable or not, is said to have *high Scott rank* if $SR(\mathcal{A}) \geq \omega_1^{\mathcal{A}}$.

Since the Harrison linear ordering has Scott rank $\omega_1^{CK}+1$ (Lemma VI.9), it is a structure of high Scott rank. In this section, we prove that the computable structures of high Scott rank are the ones which do not have computable Scott sentences. Before that, we prove a lemma that shows that every $\Pi^{\mathtt{in}}_{\alpha}$-type realized in a computable structure is equivalent to a $\Pi^{\mathtt{c}}_{2\alpha}$-formula.

Lemma VI.14. *Let $\mathcal{A}$ be a computable τ-structure. For every $\bar{a} \in A^{<\mathbb{N}}$ and every computable ordinal α, there is a $\Pi^{\mathtt{c}}_{2\alpha}$ formula $\varphi_{\bar{a},\alpha}$ such that, for any other τ-structure $\mathcal{B}$ and tuple $\bar{b}$,*

$$\mathcal{B} \models \varphi_{\bar{a},\alpha}(\bar{b}) \iff (\mathcal{A},\bar{a}) \leq_\alpha (\mathcal{B},\bar{b}).$$

Furthermore, we can find $\varphi_{\bar{a},\alpha}$ uniformly in $\bar{a}$ and α.

Proof. For the transfinite recursion to work, we also need to define $\psi_{\bar{a},\alpha}(\bar{x})$ such that

$$\mathcal{B} \models \psi_{\bar{a},\alpha}(\bar{b}) \iff (\mathcal{A},\bar{a}) \geq_\alpha (\mathcal{B},\bar{b}).$$

The definitions of $\varphi_{\bar{a},\alpha}$ and $\psi_{\bar{a},\alpha}$ are by simultaneous effective transfinite recursion: Let $\varphi_{\bar{a},\alpha}(\bar{x})$ be the formula

$$\bigwedge\!\!\!\bigwedge_{\beta<\alpha} \forall \bar{y} \bigvee\!\!\!\bigvee_{\bar{c}\in A^{<\mathbb{N}}} \psi_{\bar{a}\bar{c},\beta}(\bar{x},\bar{y})$$

and $\psi_{\bar{a},\alpha}$ be the formula

$$\bigwedge\!\!\!\bigwedge_{\beta<\alpha} \bigwedge\!\!\!\bigwedge_{\bar{c}\in A^{<\mathbb{N}}} \exists \bar{y}\ \varphi_{\bar{a}\bar{c},\beta}(\bar{x},\bar{y}).$$

It is not hard to prove by transfinite induction that these formulas are as needed. □

Theorem VI.15 (Nadel [**Nad74**]). *A computable structure has a computable infinitary Scott sentence if and only if its Scott rank is a computable ordinal.*

Proof. For the ($\Rightarrow$) direction, if $\mathcal{A}$ has a computable infinitary Scott sentence, that sentence must be $\Sigma^{\mathtt{c}}_{\alpha}$ or $\Pi^{\mathtt{c}}_{\alpha}$ for some $\alpha < \omega_1^{CK}$, and hence $\mathcal{A}$ has Scott rank below ω_1^{CK}.

For the ($\Leftarrow$) direction, let $\alpha < \omega_1^{CK}$ be the Scott rank of $\mathcal{A}$. Then, all automorphism orbits are $\Sigma^{\mathtt{in}}_{\alpha}$ definable over parameters, and all of them are $\Sigma^{\mathtt{in}}_{\alpha+2}$ definable with no parameters. We thus have that, given $\bar{a} \in A^{<\mathbb{N}}$, another tuple $\bar{b}$ is automorphic to $\bar{a}$ if and only if $\bar{a} \leq_{\alpha+3} \bar{b}$. From Lemma VI.14, we get a $\Pi^{\mathtt{c}}_{2\alpha+6}$ formula $\varphi_{\bar{a},\alpha+3}$ that defines the automorphism orbit of $\bar{a}$. Once we have all these formulas, we can build a Scott sentence exactly as in Theorem II.9. □

The computable Scott sentence we defined in the previous theorem is not of optimal complexity.

Exercise VI.16. (Alvir, Knight, McCoy [**AKM20**]) Prove that if $\mathcal{A}$ has a *computably* infinitary Π_α-Scott sentence, then $\mathcal{A}$ is $\Sigma^{\mathtt{c}}_{<\alpha}$-atomic (not necessarily uniformly so). See hint in footnote.[¶]

VI.2.1. Structures of high Scott rank. We already saw that Scott theorems do not effectivize, in the sense that computable structures do not need to have computable Scott ranks or computable Scott sentences. However, Lemma II.7, which states that $\mathcal{L}_{\omega_1,\omega}$-elementary countable structures are isomorphic, does effectivize: Computable structures that satisfy the same computable infinitary sentences are isomorphic (Corollary VI.18).

Theorem VI.17. *If $\mathcal{A}$ and $\mathcal{B}$ are computable ω-presentations, and $\mathcal{A} \equiv_\alpha \mathcal{B}$ for all $\alpha < \omega_1^{CK}$, then $\mathcal{A}$ and $\mathcal{B}$ are isomorphic.*

Proof. We say that a family $\{E_\xi : \xi \leq \alpha\}$ of sets $E_\xi \subseteq A^{<\mathbb{N}} \times B^{<\mathbb{N}}$ for $\alpha \in \mathcal{H}$ is a *bf-family* if it satisfies the properties of the back-and-forth relations, that is, $\bar{a}E_0\bar{b}$ if and only if $D_{\mathcal{H}}(\bar{a}) \subseteq D_{\mathcal{H}}(\bar{b})$ and $\bar{a}E_\xi\bar{b} \leftrightarrow \forall\zeta < \xi\forall\bar{d}\exists\bar{c}\ (\bar{b}\bar{d}E_\zeta\bar{a}\bar{c})$. Consider the set of $\alpha \in \mathcal{H}$ for which such an E exists and the empty tuples of $\mathcal{A}$ and $\mathcal{B}$ are E_α-related (i.e., $(\langle\rangle, \langle\rangle) \in E_\alpha$). This set is Σ^1_1 and contains ω_1^{CK} — it must overspill. We have some $\alpha^* \in \mathcal{H} \smallsetminus \omega_1^{CK}$ for which we have a bf-family $\{E_\xi : \xi < \alpha^*\}$ with $\langle\rangle E_{\alpha^*} \langle\rangle$.

Now consider the set

$$I = \{(\bar{a}, \bar{b}) \in A^{<\mathbb{N}} \times B^{<\mathbb{N}} : \exists\xi \in \mathcal{H} \smallsetminus \omega_1^{CK}, (\bar{a}, \bar{b}) \in E_\xi\}.$$

Since E satisfies the property of a back-and-forth relation, one can easily show that I has the back-and-forth property (Definition II.6) and hence that $\mathcal{A}$ and $\mathcal{B}$ are isomorphic [**Part 1**, Lemma III.15]. □

The following corollary is a particular case of a more general result due to Ressayre [**Res73, Res77**].

Corollary VI.18. *If two computable structures satisfy the same computable infinitary sentences, they are isomorphic.*

[¶] Use Morleyization as in Proposition II.26.

Proof. Recall from Lemma VI.14 that for each computable structure $\mathcal{A}$ and ordinal $\alpha < \omega_1^{CK}$, there is a $\Pi^{\mathtt{c}}_{2\alpha}$ sentence $\varphi_{\mathcal{A},\alpha}$ such that for any other structure $\mathcal{B}$,

$$\mathcal{B} \models \varphi_{\mathcal{A},\alpha} \iff \mathcal{A} \leq_\alpha \mathcal{B}.$$

Therefore, if $\mathcal{A}$ and $\mathcal{B}$ are computable structures and satisfy the same computable infinitary sentences, they must be α-back-and-forth equivalent for all $\alpha < \omega_1^{CK}$. □

Corollary VI.19. *[Nad74] The Scott rank of a computable structure is at most $\omega_1^{CK} + 1$.*

Proof. Every automorphism orbit is determined by the conjunction of all the computable infinitary formulas true about it. This is a $\Pi^{\mathtt{in}}_{\omega_1^{CK}}$ formula. Thus, every computable structure is $\Pi^{\mathtt{in}}_{\omega_1^{CK}}$-atomic. □

This leaves two possible ranks for computable structures of high Scott rank: ω_1^{CK} and $\omega_1^{CK} + 1$. In the former case, every orbit is $\Sigma^{\mathtt{in}}_{\omega_1^{CK}}$ definable over parameters, and hence $\Sigma^{\mathtt{in}}_{<\omega_1^{CK}}$-definable. In the latter case, there must exist at least one orbit that is not $\Sigma^{\mathtt{in}}_{<\omega_1^{CK}}$-definable.

We already saw an example of a computable structure of Scott rank $\omega_1^{CK} + 1$, namely the Harrison linear ordering, from which we can build the Harrison tree, the Harrison Boolean algebra, and the Harrison p-group, all of high Scott rank: the *Harrison tree* is just the tree of descending sequences of $\mathcal{H}$ (see page 12); the *Harrison Boolean algebra* is the interval algebra of $\mathcal{H}$; and the *Harrison abelian p-group* has one generator for each node in the Harrison tree, the root of the tree being the identity of the group, and these generators satisfy that if σ is a node of the tree, σ time p is equal to the parent of σ. For a while, these were the only examples of computable structures of Scott rank $\omega_1^{CK} + 1$. A conceptually different example of a structure of Scott rank $\omega_1^{CK} + 1$ was recently built by Harrison-Trainor [**HT18**].

A computable structure of Scott rank ω_1^{CK} was built by Knight and Millar [**KM10**], improving a construction of an arithmetical structure of Scott rank ω_1^{CK} by Makkai [**Mak81**].

Theorem VI.20. *There is a computable structure of Scott rank ω_1^{CK}.*

Proof. ([**CKM06**]) We start by defining a computable sequence of sets

$$A_0 \subseteq A_1 \subseteq \cdots \subseteq \mathcal{H}$$

satisfying the following properties:

(1) Each A_n has order type at most ω^{n+1}.

(2) A_0 is cofinal in $\mathcal{H}$.
(3) For each $n \in \mathbb{N}$ and $a \in A_n$, $a = \sup\{b + 1 \in A_{n+1} : b < a\}$. In other words, if a is a successor, then $a - 1 \in A_{n+1}$, and if a is a limit, then there exists $b_0 < b_1 < b_2 < \cdots \in A_{n+1}$ with limit a.
(4) $\bigcup_{n\in\mathbb{N}} A_n = \mathcal{H}$.

It is not hard to build the sets A_n by recursion on n: For each $a \in A_{n-1}$, add to A_n a sequence $b^a_0 < b^a_1 < b^a_2 < \cdots \to a$ that may be finite or infinite, where b^a_n is the $\leq_{\mathbb{N}}$-least element b such that $b^a_{n-1} < b < a$, if such an element exists. If a is a successor ordinal, we will eventually have $b^a_n = a - 1$ and stop finding new elements in the sequence. If a is a limit ordinal, this sequence will be infinite, and for every $c < a$ we must have $c < b^a_n$ for some n, because for the least n with $b^a_n \geq_{\mathbb{N}} c$ (in the ordering of $\mathbb{N}$), if $c \geq_{\mathcal{H}} b^a_{n-1}$ then c would be chosen as b^a_n. We claim that $\bigcup_n A_n = \mathcal{H}$: Otherwise take $h \in \mathcal{H} \smallsetminus \bigcup_n A_n$ and, for each n, let a_n be the least element of A_n greater than h, which exits because A_n is computable and all hyperarithmetic subsets of $\mathcal{H}$ have a least element. Note that $0'$ can compute a_n, and by (3), $a_n < a_{n-1}$ for all n, contradicting that $\mathcal{H}$ has no $0'$-computable descending sequences.

Now that we have the sets A_n, let us define a tree $T \subseteq (H \times \mathbb{N})^{<\mathbb{N}}$, which we will prove has Scott rank ω_1^{CK}:

$$T = \{\langle\langle h_0, n_0\rangle, \langle h_1, n_1\rangle, ..., \langle h_k, n_k\rangle\rangle \in (H \times \mathbb{N})^{<\mathbb{N}} : \\ (\forall i \leq k)\ h_i \in A_i\ \ \&\ \ h_i <_H h_{i-1}\}.$$

Notice the second coordinate of each entry of the tuple is ignored, and it is the first coordinate that must be a decreasing sequence in $\mathcal{H}$ and belong to the right set A_i. The second coordinate is only there to make sure that each branch is repeated infinitely often. Let us use $h(\sigma)$ to denote the first coordinate of the last entry of σ. We view T as a graph with a special constant denominating the root. That is, we are considering the structure

$$\mathcal{T} = (T; \langle\rangle, R),$$

where $R = \{\langle\sigma, \sigma^-\rangle : \sigma \in T \smallsetminus \{\langle\rangle\}\}$ is the parent relation in the tree. It is $\mathcal{T}$ that we claim has Scott rank ω_1^{CK}. When we view $\mathcal{T}$ as a structure, we erase the information about the sequence of pairs which constitutes each element of T. We will be able to more or less recover some of that information — but at a cost.

It is not hard to show by transfinite induction using (3) that, for every $\sigma \in T$ with $h(\sigma) \in \omega_1^{CK}$, $\mathrm{rk}(T_\sigma) = h(\sigma)$.[‖] For σ with $h(\sigma) \in \mathcal{H} \setminus \omega_1^{CK}$, T_σ is ill-founded. Furthermore, it is not hard to see that, given $\sigma, \tau \in T$,

- if $h(\sigma), h(\tau) < \omega_1^{CK}$, then $T_\sigma \cong T_\tau$ if and only if and $h(\sigma) = h(\tau)$, and
- if $h(\sigma), h(\tau) \in \mathcal{H} \setminus \omega_1^{CK}$, using a back-and-forth proof, one can show that $T_\sigma \cong T_\tau$, independently of the value of $h(\sigma)$ and $h(\tau)$.

Thus, we can tell if two nodes are automorphic as follows:

$$(\mathcal{T}, \sigma) \cong (\mathcal{T}, \tau) \quad \iff \quad |\sigma| = |\tau| \;\;\&\;\; \forall i \leq |\sigma| \; (\mathrm{rk}(T_{\sigma\restriction i}) = \mathrm{rk}(T_{\tau\restriction i})),$$

including the possibility of $\mathrm{rk}(T_{\sigma\restriction i}) = \infty$. What we need to be able to tell is when two tuples of nodes are automorphic, not just single nodes. Given tuples of nodes $\bar{\sigma} = \langle \sigma_1, ..., \sigma_\ell \rangle$, we let $\bar{\sigma}\downarrow$ be its downward closure. That is, let $\bar{\sigma}\downarrow$ be the tuple which contains all the initial segments of the nodes in $\bar{\sigma}$, i.e., all the nodes of the form $\sigma_j \restriction i$ for $j \leq \ell$ and $i \leq |\sigma_j|$. We then have that, for two tuples of nodes $\bar{\sigma}$ and $\bar{\tau}$ of length ℓ,

$$\begin{aligned}&(\mathcal{T}, \bar{\sigma}) \cong (\mathcal{T}, \bar{\tau}) \iff \\ &\quad (\mathcal{T}, \bar{\sigma}\downarrow) \equiv_0 (\mathcal{T}, \bar{\tau}\downarrow) \;\;\&\;\; \forall j \leq \ell \; \forall i \leq |\sigma_j| \; (\mathrm{rk}(T_{\sigma_j\restriction i}) = \mathrm{rk}(T_{\tau_j\restriction i})).\end{aligned}$$

Thus, to define the automorphism orbit of any tuple, we need to find the ranks of the branches of the trees below the elements of the tuple: If $h(\sigma) \in \omega_1^{CK}$, then we know from Lemma II.4 that there is a computable infinitary sentence that is true only for trees of rank $h(\sigma)$. If $h(\sigma) \in \mathcal{H} \setminus \omega_1^{CK}$, then there is no infinitary formula that says that a tree has infinite rank (see Corollary II.41). However, if we know the length of σ, say n, all we need to say is that the rank of T_σ is not in $A_n \cap \omega_1^{CK}$. Let α_n be the supremum of $A_n \cap \omega_1^{CK}$, which, since A_n has order-type at most ω^{n+1}, has to be an ordinal in ω_1^{CK}. (This is because if a_n is the least element of $A_n \setminus \omega_1^{CK}$, then $A_n \cap \omega_1^{CK} = A_n \cap (\mathcal{H} \restriction a_n)$ is computable, and hence it must be bounded below ω_1^{CK}.) Then $\mathrm{rk}(T_\sigma) = \infty$ if and only if $\mathrm{rk}(T_\sigma) > \alpha_n$, and we know from Lemma II.4 that there is a computable infinitary sentence that is true only for trees of rank greater than α_n. We conclude that T has Scott rank at most ω_1^{CK}.

To prove that it does not have Scott rank below ω_1^{CK}, we need to show that there is no bound below ω_1^{CK} on the complexity of the

[‖] Recall that we use ω_1^{CK} to denote the well-ordered initial segment of $\mathcal{H}$. Recall also that $T_\sigma = \{\gamma : \sigma^\frown\gamma \in T\}$.

formulas defining the automorphism orbits. That is, we need to show that for every $\alpha < \omega_1^{CK}$, there exists $\sigma, \tau \in T$ which satisfy the same $\Pi^{\mathtt{in}}_\alpha$-formulas, i.e., such that σ and τ are α-back-and-forth equivalent in T but not automorphic. This will follow from the following claim:

Claim VI.20.1. If $|\sigma| = |\tau| = n$ and $\omega \cdot \alpha < h(\sigma), h(\tau)$, then $T_\sigma \equiv_\alpha T_\tau$.

To work out the back-and-forth relations on T, we need a few basic observations. The first is that it is enough to consider tuples and extensions of tuples which are closed downward in the tree (in other words, that are finite subtrees, all of the form $\bar{\sigma}\downarrow$). The second key observation is that given finite tuples $\bar{\sigma}$, $\bar{\tau}$ which are closed downwards, $\bar{\sigma} \leq_\alpha \bar{\tau}$ if and only if, for every $i < |\bar{\sigma}|$, $T_{\sigma_i \setminus \bar{\sigma}} \leq_\alpha T_{\tau_i \setminus \bar{\tau}}$, where $T_{\sigma_i \setminus \bar{\sigma}}$ is the tree of all $\gamma \supseteq \sigma_i$ such that for no j different from i, $\sigma_i \subsetneq \sigma_j \subseteq \gamma$. This is because the sets $T_{\sigma_i \setminus \bar{\sigma}}$ for $i = 0, ..., |\bar{\sigma}| - 1$ partition T into completely independent pieces with no interaction between them. Thus, when you consider a tuple extending $\bar{\sigma}$, you can consider the parts of the tuple inside each $T_{\sigma_i \setminus \bar{\sigma}}$ independently. The third observation is that $T_{\sigma_i \setminus \bar{\sigma}} \cong T_{\sigma_i}$, because each branch repeats infinity often and removing a few branches does not affect the isomorphism type.

The proof of the claim is by transfinite induction of α. We recommend the reader try it with pencil and paper before reading these details. The case $\alpha = 0$ is trivial. Let us move to the general case. By symmetry, it is enough to show that $T_\sigma \leq_\alpha T_\tau$. Let $\bar{b}$ be a tuple in T_τ that is closed downwards, and let $\beta < \alpha$. We need to find a tuple $\bar{a}$ such that for each $i < |\bar{a}|$, $|a_i| = |b_i|$ and either $h(a_i) = h(b_i) < \omega \cdot \beta$ or $\omega \cdot \beta < h(a_i), h(b_i)$. This would imply that $T_{a_i} \geq_\beta T_{b_i}$ for all $i < |\bar{b}|$, as needed. Let k be the length of the longest tuple in $\bar{b}$, and let $\gamma_0, ..., \gamma_k$ be such that $\omega \cdot \beta < \gamma_k < \gamma_{k-1} < \cdots < \gamma_0 < \omega \cdot \alpha \leq h(\sigma)$ and $\gamma_i \in A_{|\sigma|+i}$, which we can do by (3), making sure at each step that $\gamma_i > \omega \cdot \beta + k - i$. Define $\bar{a}$ starting from the shortest nodes in the sub-tree to the longest according to the following rule: If $h(b_i) < \omega \cdot \beta$, let $h(a_i) = h(b_i)$; and if $h(b_i) \geq \omega \cdot \beta$, let $h(a_i) = \gamma_{|a_i|}$. Of course, you must also preserve lengths: $|a_i| = |b_i|$.

This finishes the proof of the claim. It follows that for no $\alpha < \omega_1^{CK}$ we have that all orbits are $\Sigma^{\mathtt{in}}_\alpha$-definable and hence that T must have high Scott rank. □

Observation VI.21. The Scott-sentence complexity of the tree above is $\Pi^{\mathtt{in}}_{\omega_1^{CK}}$. The Scott sentence for T says the following: For every n and every σ in T of length n, T_σ has rank either in $A_n \cap \omega_1^{CK}$ or greater than α_n. If $\mathrm{rk}(T_\sigma) = \gamma \in A_n \cap \omega_1^{CK}$, then σ has children of all ranks in $A_n \cap \gamma$, each rank appearing infinitely often. If $\mathrm{rk}(T_\sigma) > \alpha_n$, then σ has children of all ranks in $A_n \cap \alpha_n$, each rank appearing infinitely often, and also has infinitely many children of rank greater than α_{n+1}.

New structures of high Scott rank have been built recently. Harrison-Trainor, Igusa, and Knight [HTIK18] proved that there is a structure that is not $\aleph_0$-categorical for the computable infinitary theory. Alvir, Greenberg, Harrison-Trainor, and Turetsky [**AGHTT**] have since then built new examples and done a deep analysis of the possible Scott sentence complexities of the computable structures of high Scott rank.

VI.2.2. Barwise-Kreisel compactness. Recall that a set S of $\mathcal{L}_{\omega_1,\omega}$ sentences is said to be *satisfiable* if it has a model. For countable S, from the Löwenheim-Skolem Theorem II.61 we get that if S has a model, it must have a countable one.

The most important tool in model theory of finitary first-order logic is compactness: If every finite subset of a set of sentences is satisfiable, then the whole set is satisfiable. This is not true of infinitary logic. Here is an example. In the vocabulary with constants a, b and a unary function S, the set

$$\{\text{“}\overbrace{\mathsf{S}(\mathsf{S}(\cdots\mathsf{S}(}^{n}\mathsf{a})\cdots)) \neq \mathsf{b}\text{”} : n \in \mathbb{N}\} \cup \{\text{“}\bigvee\!\!\!\bigvee_{n\in\mathbb{N}} \overbrace{\mathsf{S}(\mathsf{S}(\cdots\mathsf{S}(}^{n}\mathsf{a})\cdots)) = \mathsf{b}\text{”}\}$$

is not satisfiable, but every finite subset of it is.

However, in the computable infinitary language, there is a version of compactness that turns out to be extremely useful.

Theorem VI.22 (Barwise [**Bar67, Bar69**]). *Let $\{\varphi_\xi : \xi \in \omega_1^{CK}\}$ be a computable sequence of computable infinitary formulas. If for each $\alpha < \omega_1^{CK}$, the set $\{\varphi_\xi : \xi < \alpha\}$ is satisfiable, then the whole set $\{\varphi_\xi : \xi \in \omega_1^{CK}\}$ is satisfiable.*

When we say that $\{\varphi_\xi : \xi \in \omega_1^{CK}\}$ is a computable sequence of computable infinitary formulas, we mean that there is a partial computable function f such that, for all α in ω_1^{CK}, which we view as the well-founded part of a given ω-presentation of $\mathcal{H}$, $f(\alpha)$ is defined and gives the index for a computable infinitary formula, and we do not care what f does on $\mathcal{H} \smallsetminus \omega_1^{CK}$. Recall from Section III.2 that an index for a computable infinitary formula consists of a quadruple $\langle \Gamma, \beta, i, j\rangle$ where $\Gamma \in \{\Sigma, \Pi\}$, $\beta < \omega_1^{CK}$, and $i, j \in \mathbb{N}$, a formula which we denote by $\varphi_{i,j}^{\Gamma_\beta}(x_1, ..., x_j)$.

Proof. There is a Σ_1^1 $\mathbb{N}$-formula χ that, given an ω-presentation of a structure $\mathcal{A}$ and an index e for a computable infinitary sentence

φ_e, $\chi(\mathcal{A}, e)$ holds if and only if $\mathcal{A} \models \varphi_e$. Consider the set of $\zeta \in \mathcal{H}$ for which $\{\varphi_\xi : \xi < \alpha\}$ is satisfiable. That is, let

$$Z = \{\zeta \in \mathcal{H} : \exists \mathcal{A}\ \forall \xi < \zeta\ (f(\xi)\downarrow\ \wedge\ \chi(\mathcal{A}, f(\xi))).$$

Z is Σ^1_1 and contains ω_1^{CK} — it must overspill. There is some $\zeta^* \in Z \smallsetminus \omega_1^{CK}$.[**] We then have that for some structure $\mathcal{A}$, for every $\xi < \omega_1^{CK} < \zeta^*$, $f(\xi)\downarrow\ \wedge\ \chi(\mathcal{A}, f(\xi))$. Thus, $\mathcal{A}$ is a model of $\{\varphi_\xi : \xi \in \omega_1^{CK}\}$. □

Barwise's version of the theorem above was in terms of admissible sets. If one considers the right setting, the theorem above can be seen as a particular case of Barwise compactness. The corollary below is attributed to Kreisel [**Kre61**] in [**AK00**, Page 123].

Corollary VI.23. *(Barwise-Kreisel Compactness Theorem) Let S be a Π^1_1 set of indices of computable infinitary formulas. If every hyperarithmetic subset of S is satisfiable, then so is S.*

Proof. The first step is to notice that every Π^1_1 set can be decomposed as the union $\bigcup_{\xi\in\omega_1^{CK}} S_\xi$ of an increasing sequence of sets, where S_ξ is $\Sigma^0_{2\xi}$ for each $\xi \in \omega_1^{CK}$: Given an m-reduction from S to $\mathcal{O}_{\text{wo}}$, let

$$S_\xi = \{e \in \mathbb{N} : \mathcal{L}_{f(e)} \prec \omega^\xi\}.$$

The sets S_ξ are $\Sigma^0_{2\xi}$ (by Lemma II.5).

We showed in Lemma V.10 that a $\Sigma^0_{2\xi}$ conjunction of computable infinitary sentences is equivalent to a computable infinitary sentence, and we can find this equivalent formula uniformly, given an index for the $\Sigma^0_{2\xi}$ set. Let ψ_ξ be a computable infinitary sentence equivalent to the conjunction of the formulas with indices in S_ξ. That is,

$$\psi_\xi \quad \equiv \quad \bigwedge\!\!\!\bigwedge_{e\in S_\xi} \varphi_e.$$

For each α, since S_α is Δ^1_1, the set $\{\varphi_\xi : \xi < \alpha\}$ is satisfiable. By the previous theorem, the whole set $\{\varphi_\xi : \xi \in \omega_1^{CK}\}$, which is equivalent to $\bigwedge\!\!\!\bigwedge_{e\in S} \varphi_e$, is satisfiable. □

Exercise VI.24. Prove that the theorem above still holds if the function $f : \xi \mapsto \varphi_\xi$ is Σ^1_1, that is, there is a Σ^1_1 formula $\theta(x, y)$ such that, for every $e \in \omega_1^{CK}$, $f(e) = d$ if and only if $\theta(e, d)$.

[**] For $\zeta^* \in \mathcal{H} \smallsetminus \omega_1^{CK}$, $f(\zeta^*)$ might be undefined, or if it is defined, it might output a quadruple that is an index of a computable infinitary sentence or not. Independently of whether $k \in \mathbb{N}$ is an index for a computable infinitary formula, $\chi(\mathcal{A}, k)$ is either true or false. The truth value of $\chi(\mathcal{A}, k)$ is meaningless if k is not an index for a computable infinitary sentence.

The following is a version of Barwise-Kreisel compactness where we consider satisfaction only by computable models.

Corollary VI.25. *Let $\{\varphi_\xi : \xi \in \omega_1^{CK}\}$ be a computable sequence of computable infinitary formulas. If for each $\alpha < \omega_1^{CK}$, the set $\{\varphi_\xi : \xi < \alpha\}$ is satisfiable in a computable structure, then the whole set $\{\varphi_\xi : \xi \in \omega_1^{CK}\}$ is satisfiable in a computable structure.*

Proof. The proof is almost exactly the same as that of Theorem VI.22, with the difference being that we consider only computable structures $\mathcal{A}$. The set Z is still Σ^1_1 and must overspill. □

Corollary VI.26. *If a computable infinitary sentence T has computable models of arbitrarily high Scott rank below ω_1^{CK}, it has a computable model of high Scott rank.*

Proof. Consider the sequence $\{\varphi_\xi : \xi \in \omega_1^{CK}\}$ defined as follows: φ_0 is just T. For $\xi > 0$, φ_ξ is the sentence ρ_ξ that says that the model has Scott rank greater than or equal to ξ and in Lemma II.67. The corollary then follows directly from the theorem. □

The following result is due to Morley and Barwise independently. See Keisler's book [**Kei71**, Chapters 15 and 16]. The version for infinitary sentences is due to Morley [**Mor**] and Barwise [**Bar69**]. The boldface versions are due to Morley [**Mor65**] and López-Escobar [**LE66**]. (They show the Hanf number of $\mathcal{L}_{\omega_1,\omega}$ is $\beth_{\omega_1}$ and that of $\mathcal{L}_{\mathtt{c},\omega}$ is $\beth_{\omega_1^{CK}}$. They use the Erdös-Rado theorem to build an order-indiscernible sequence over a language with added Skolem functions.)

The following result was one of the key ingredients in the proof that there is a structure Muchnik equivalent to its own jump that we gave in [**Part 1**, Chapter IX].

Theorem VI.27. *If a computable infinitary τ-sentence T has a model of size $\beth_{\omega_1^{CK}}$, it has a countable model with a non-trivial automorphism.*

Recall that $\beth_\alpha$ is the cardinal obtained by iterating the power set operation α times.

Proof. We consider structures with two sorts, one of which we call $\mathcal{M}$ and is a model of T and the other is a linear ordering $\mathcal{L}$ with a first element 0 and a last element ℓ, which we should think of as a well-order (though it does not need to be). These two-sorted structures also have a relation $E \subseteq \mathcal{L} \times M^{<\mathbb{N}} \times M^{<\mathbb{N}}$ which encodes the back-and-forth relations in $\mathcal{M}$ indexed by elements of $\mathcal{L}$, which are treated as ordinals. That is, if $\mathcal{L}$ were actually well-ordered, then $E(\alpha, \bar{a}, \bar{b})$ would hold if and only if $(\mathcal{M},\bar{a}) \leq_\alpha (\mathcal{M},\bar{b})$. We also need two different elements c and d from M which are E_ℓ-equivalent, that is, such that $E(\ell,c,d)$ holds. The idea is to prove that there exists such a model where $\mathcal{L}$ is ill-founded and prove that, in that case, c and d are automorphic.

Concretely, let τ' be a vocabulary that consists of $\tau \cup \{M, L, \leq_L, E, 0, \ell, \mathsf{c}, \mathsf{d}\}$.[††] Let S be the computable infinitary τ'-sentence saying the following:

(1) M and L partition the domain.
(2) $\mathcal{M} \models T$, and c and d are two different elements from M.
(3) $(L;\leq_L)$ is a linear ordering with first element 0 and last element ℓ.
(4) For $\bar{a},\bar{b} \in M^{<\mathbb{N}}$ of the same length, $E(0,\bar{a},\bar{b})$ holds if $\bar{a}$ and $\bar{b}$ satisfy the same atomic $\tau_{|\bar{a}|}$-formulas in $\mathcal{M}$.
(5) For $\alpha \in L$ and $\bar{a},\bar{b} \in M^{<\mathbb{N}}$ of the same length, $E(\alpha,\bar{a},\bar{b})$ holds if and only if, for every $\beta <_L \alpha$ and every $\bar{f} \in M^{<\mathbb{N}}$, there exists $\bar{e} \in M^{|\bar{d}|}$ such that $E(\beta,\bar{b}\bar{f},\bar{a}\bar{e})$ holds.
(6) $E(\ell,\mathsf{c},\mathsf{d})$.

We claim that, if $\mathcal{L}$ is a computable well-ordering and $\mathcal{M}$ is a model of T of size $\beth_{\omega_1^{CK}}$, then $\mathcal{M}$ and $\mathcal{L}$ can be put together to build a model of S. The first step is to define E, but since $\mathcal{L}$ is well-ordered, E is uniquely defined by the rules above and we must have $E(\alpha,\bar{a},\bar{b}) \iff (\mathcal{M},\bar{a}) \leq_\alpha (\mathcal{M},\bar{b})$. The crux is to show that one can name two elements of M, c and d, so that $c \leq_\ell d$. To show this, we claim that for each $\alpha \in \mathcal{L}$, there are at most $\beth_{\alpha+1}$ many $\equiv_\alpha$-equivalence classes. This is true of $\alpha = 0$, as there are countably many possible values for $D_{\mathcal{A}}(\bar{a})$. The $\equiv_{\alpha+1}$-equivalence class of a tuple $\bar{a}$ is determined by the set of possible $\equiv_\alpha$-equivalence classes of tuples of the form $\bar{a}\bar{e}$. If there are at most $\beth_{\alpha+1}$ $\equiv_\alpha$-equivalence classes, then there are at most $2^{\beth_{\alpha+1}} = \beth_{\alpha+2}$ sets of $\equiv_\alpha$-equivalence classes and hence at most $\beth_{\alpha+2}$ $\equiv_{\alpha+1}$-equivalence classes. For limit ordinals λ, a $\equiv_\lambda$-equivalence class is determined by the α-equivalence classes for $\alpha < \lambda$. Each $\equiv_\lambda$-equivalence class can thus be represented by a function with domain λ which assigns an α-equivalence to each $\alpha \in \lambda$. The number of such functions is bounded by $|\lambda|^{\sup_{\alpha<\lambda}\beth_{\alpha+1}} = \omega^{\beth_\lambda} = \beth_{\lambda+1}$.

Now, if $\mathcal{M}$ has size larger than $\beth_{\mathcal{L}}$, there must be at least one $\equiv_\ell$-equivalence class with at least two elements — call them c and d.

For each computable ordinal ξ, consider the sentence ψ_ξ that says that $\mathcal{L}$ does not embed in ξ (see Lemma II.5). By the previous claim, for every $\alpha < \omega_1^{CK}$, the theory $S \cup \{\psi_\xi : \xi < \alpha\}$ is satisfiable by a

[††] E is a actually a sequence of relations $\{E_n : n \in \mathbb{N}\}$, where E_n has arity $2n+1$ and applies to triples $\alpha,\bar{a},\bar{b}$, with $\alpha \in L$ and $\bar{a},\bar{b} \in M^n$.

model where $\mathcal{L}$ is computable. From the Barwise-Kreisel compactness theorem (Theorem VI.22), $S \cup \{\psi_\xi : \xi < \omega_1^{CK}\}$ is satisfiable by a model where $\mathcal{L}$ is computable. Since $\mathcal{L} \not\preccurlyeq \xi$ for any computable ordinal ξ, $\mathcal{L}$ cannot be well-ordered. Split $\mathcal{L}$ into $\mathcal{L}_0 + \mathcal{L}_1$ where $\mathcal{L}_0$ is well-ordered and $\mathcal{L}_1$ has no least element. It follows from (5) that the set

$$\{\langle \bar{a}, \bar{b} \rangle : E(\alpha^*, \bar{a}, \bar{b}) \text{ for some } \alpha^* \in L_1\}$$

has the back-and-forth property (Definition II.6), and hence any pair in it is a pair of automorphic elements. It follows from [**Part 1**, Lemma III.15] that there is an automorphism mapping c to d. □

CHAPTER VII

Forcing

Forcing was introduced by Paul Cohen to prove that the continuum hypothesis does not follow from the ZFC axioms of set theory. Soon after, it became one of the main tools in set theory to prove independence results of all kinds. The objects produced by this technique are called *generics*.

There are also many applications of forcing in computability theory. In computable structure theory, forcing is used as a tool to translate computational properties of ω-presentations to structural properties of structures. In this book, we will look at generic enumerations of structures and generic presentations of structures. The special feature of these generic presentations is that there is nothing special about them — they are generic. Thus, if a generic ω-presentation has some particular computational property, that property is not special to this ω-presentation and there must be some structural reason for it.

We introduced the first ideas of forcing in computable structure theory in [**Part 1**, Chapter IV]. There, we only considered 1-generics, which decide only $\Sigma^{\mathtt{c}}_1$ relations. Now, we will consider $\mathcal{L}_{\mathtt{c},\omega}$-generics, which decide all $\mathcal{L}_{\mathtt{c},\omega}$-definable relations. An understanding of [**Part 1**, Chapter IV], while recommended, is not required to read this chapter.

The first one to use forcing in computable structure theory were Ash, Knight, Manasse, and Slaman [**AKMS89**], and independently Chisholm [**Chi90**]. The notion of forcing we introduce here is aesthetically different than theirs, but the ideas are the same.

VII.1. Generic enumerations and generic presentations

Let $A^\star$ be the set of all finite tuples from A whose entries are all different. We will use the partial ordering $(A^\star; \supseteq)$ as what set theorists call *the forcing notion.* We say that a subset $R \subseteq A^\star$ is *dense* if, for every $\bar{r} \in A^\star$, there is a $\bar{p} \supseteq \bar{r}$, $\bar{p} \in R$. Given an injective enumeration[*] g of A, we say that g *meets* R if g has some initial segment $\bar{p} \subset g$ in R.

Consider the topological space of all injective enumerations of a structure $\mathcal{A}$, viewed as a subspace of $A^{\mathbb{N}}$, which in turn is homeomorphic

[*] Recall that an enumeration of A is an onto map from $\mathbb{N}$ to A.

to Baire space $\mathbb{N}^{\mathbb{N}}$. The set of injective enumerations that meet a set $R \subseteq A^\star$, denoted $[R]^\subset$, is an open set (see Section V.4). Actually, all open sets are of this form. If R is dense as defined above, then $[R]^\subset$ is dense in the topological sense too (i.e., it intersects every open set). Topologically speaking, dense open sets are considered large sets, and belonging to the complement of a dense open set is thus a special property. These are the kind of special properties generic enumerations do not have. When forcing in set theory, one considers all dense open sets. Here, we only consider the ones definable by computable infinitary formulas.

We say that a relation $R \subseteq A^{<\mathbb{N}}$ is *computably infinitary definable*, or $\mathcal{L}_{\mathsf{c},\omega}$*-definable*, if each set $R \cap A^n$ is $\mathcal{L}_{\mathsf{c},\omega}$-definable uniformly in n, in other words, if there is a computable list of computable infinitary formulas $\{\varphi_n(x_1, ..., x_n) : n \in \mathbb{N}\}$ such that $R \cap A^n = \{\bar{a} \in A^n : \mathcal{A} \models \varphi_n(\bar{a})\}$ for all $n \in \mathbb{N}$.

Definition VII.1. An injective enumeration $g \colon \omega \to A$ of a structure $\mathcal{A}$ is $\mathcal{L}_{\mathsf{c},\omega}$*-generic* if it meets every dense $\mathcal{L}_{\mathsf{c},\omega}$-definable subset of $A^\star$.

Lemma VII.2. *For every structure $\mathcal{A}$, every $\bar{p} \in A^\star$ can be extended to an $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration $g \colon \omega \to A$.*

Recall that, unless stated otherwise, all structures we consider are countable.

Proof. This is essentially a corollary of the Baire category theorem, which says that the intersection of a countable collection of dense open sets is never empty. We give a direct proof.

There are countably many $\mathcal{L}_{\mathsf{c},\omega}$-definable relations, so one can build g meeting one $\mathcal{L}_{\mathsf{c},\omega}$-definable relation at the time as follows. We build g as the limit of a nested sequence $\bar{p}_0 \subseteq \bar{p}_1 \subseteq \cdots$. Let $\bar{p}_0 = \bar{p}$. To make sure g is generic, define p_{2n} so that it extends p_{2n-1} and meets the n-th dense $\mathcal{L}_{\mathsf{c},\omega}$-definable relation. To make sure g is onto, define p_{2n+1} so that it extends p_{2n} and contains the n-th element of $\mathcal{A}$. □

Definition VII.3. An ω-presentation $\mathcal{G}$ is an $\mathcal{L}_{\mathsf{c},\omega}$*-generic presentation* of a structure $\mathcal{A}$ if $\mathcal{G}$ can be obtained as the pull-back $g^{-1}(\mathcal{A})$ through an $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration g.[†]

[†] Recall that the pull-back of a structure $\mathcal{A}$ through an enumeration $g \colon \mathbb{N} \to \mathcal{A}$ is the unique ω-presentation $\mathcal{G}$ that makes g an isomorphism from $\mathcal{G}$ to $\mathcal{A}$. (See page xxiv.)

The notion of $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration depends only on the structural properties of $\mathcal{A}$, as it only depends on which relations are $\mathcal{L}_{\mathsf{c},\omega}$-definable in $\mathcal{A}$. It follows that the notion of $\mathcal{L}_{\mathsf{c},\omega}$-generic presentation of a structure $\mathcal{A}$ is independent of the given ω-presentation of $\mathcal{A}$. More concretely, if $h\colon \mathcal{A} \to \mathcal{B}$ is an isomorphism, then $g\colon \mathbb{N} \to \mathcal{A}$ is $\mathcal{L}_{\mathsf{c},\omega}$-generic if and only if $h \circ g\colon \mathbb{N} \to \mathcal{B}$ is $\mathcal{L}_{\mathsf{c},\omega}$-generic.

Let us consider the particular situation where $\mathcal{B} = \mathcal{G}$, just for a minute. If $\mathcal{G}$ is the pull-back $g^{-1}(\mathcal{A})$ of $\mathcal{A}$ through some $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration $g\colon \mathbb{N} \to \mathcal{A}$, and we let $\mathcal{B} = \mathcal{G}$ and h be $g^{-1}\colon \mathcal{A} \to \mathcal{G}$, then $id = g^{-1} \circ g\colon \mathcal{G} \to \mathcal{G}$ is also $\mathcal{L}_{\mathsf{c},\omega}$-generic. It follows that an ω-presentation $\mathcal{G}$ is an $\mathcal{L}_{\mathsf{c},\omega}$-generic presentation if and only if the identity function on $\mathbb{N}$ is an $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration of $\mathcal{G}$ itself.

Exercise VII.4. Prove that if g is $\mathcal{L}_{\mathsf{c},\omega}$-generic, it also meets all dense subsets of $A^\star$ that are $\mathcal{L}_{\mathsf{c},\omega}$-definable over parameters.

Exercise VII.5. Show that if $\mathcal{A}$ is computable, Kleene's $\mathcal{O}$ can compute an $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration of $\mathcal{A}$.

VII.2. The forcing relation

An ω-presentation $\mathcal{G}$ is characterized by its atomic diagram $D(\mathcal{G}) \in 2^{\mathbb{N}}$ (see page xxi). Thus, to talk about properties of presentations, we use the vocabulary $\{0, 1, +, \times, \leq, X(\cdot)\}$ of 1st order arithmetic with an extra unary relation X to represent the diagram of the structure. We will think of X as a 2nd-order variable and write $\varphi(X)$ to emphasize this. We call these formulas $\mathbb{N}$-*formulas*. Given an ω-presentation $\mathcal{G}$, we will write $\varphi(\mathcal{G})$ to mean $\varphi(D(\mathcal{G}))$. Instead of X, we will use the set theoretic dot notation $\dot{\mathcal{G}}$, as a 2nd-order variable symbol to emphasize that we are talking about the diagram of an ω-presentation. The dot on top of $\dot{\mathcal{G}}$ means that we are not talking about a particular presentation, but about a name for a generic presentation that we will obtain after forcing. Recall from Section V.4 that the $\mathbb{N}$-$\mathcal{L}_{\omega_1,\omega}$-definable subsets of $2^{\mathbb{N}}$ are exactly the Borel subsets of $2^{\mathbb{N}}$.

Recall from Section V.4 that every $\mathbb{N}$-$\mathcal{L}_{\omega_1,\omega}$-formula is equivalent, when evaluated in the structure of the natural numbers, to an infinitary quantifier-free one, where the quantifiers $\forall n$ and $\exists n$ are replaced by the infinitary connectives $\bigwedge\!\!\!\bigwedge_{n\in\mathbb{N}}$ and $\bigvee\!\!\!\bigvee_{n\in\mathbb{N}}$. Recall that we can also replace the atomic formulas that do not mention $\dot{\mathcal{G}}$ by their truth values $\top$ or $\bot$. Furthermore, this transformation can be done without changing the complexity of the formula. We will thus assume that our $\mathbb{N}$-$\mathcal{L}_{\omega_1,\omega}$-formulas $\varphi(\dot{\mathcal{G}})$ are always infinitary quantifier free and that the literal[‡]

[‡] Recall that a *literal* is a formula that is either atomic or negation of atomic.

sub-formulas are of the form $\top$, $\bot$, $\dot{\mathcal{G}}(n)$ or $\neg\dot{\mathcal{G}}(n)$. By deMorgan's laws, we may also assume that all negations appear only next to atomic formulas.

Definition VII.6. We say that a tuple $\bar{p} \in A^\star$ *semantically forces* a computable infinitary $\mathbb{N}$-formula $\varphi(\dot{\mathcal{G}})$ if $\varphi(g^{-1}(\mathcal{A}))$ holds for every $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration g of $\mathcal{A}$ extending $\bar{p}$.

The forcing-equals-truth theorem (Theorem VII.13 below) will formalize the idea that generics have no special properties by showing that an $\mathcal{L}_{\mathsf{c},\omega}$-generic presentation $g^{-1}(\mathcal{A})$ satisfies an $\mathbb{N}$-formula $\varphi(\dot{\mathcal{G}})$ if and only if g has an initial segment $\bar{p}$ that forces it. Thus, if $g^{-1}(\mathcal{A})$ satisfies $\varphi(\dot{\mathcal{G}})$, then all other generic enumerations extending $\bar{p}$ do too. So, all the information needed to guarantee $\varphi(\dot{\mathcal{G}})$ must be encoded in $\bar{p}$ and $\mathcal{A}$, and is independent of what the generic enumeration g does beyond $\bar{p}$. To formalize this last statement, we will define a relation $\Vdash$ in a purely syntactical way, and we will show that it mostly coincides with semantical forcing.

Let us assume the vocabulary τ is relational (see page xxii).

Definition VII.7 (The strong *forcing relation*). Given a structure $\mathcal{A}$, a tuple $\bar{p}$ from $\mathcal{A}^\star$, and an infinitary $\mathbb{N}$-formula $\varphi(\dot{\mathcal{G}})$, we define a predicate $\bar{p} \Vdash_{\mathcal{A}} \varphi$ by recursion on φ as follows:

- $\bar{p} \Vdash_{\mathcal{A}} \top$ holds and $\bar{p} \Vdash_{\mathcal{A}} \bot$ does not.
- $\bar{p} \Vdash_{\mathcal{A}} \dot{\mathcal{G}}(\mathbf{n}) \iff D_{\mathcal{A}}(\bar{p})(n)\downarrow = 1.$[§]
- $\bar{p} \Vdash_{\mathcal{A}} \neg\dot{\mathcal{G}}(\mathbf{n}) \iff D_{\mathcal{A}}(\bar{p})(n)\downarrow = 0.$
- $\bar{p} \Vdash_{\mathcal{A}} \bigvee\!\!\!\bigvee_{i\in I} \psi_i$ if there exists $i \in I$ such that $\bar{p} \Vdash_{\mathcal{A}} \psi_i$.
- $\bar{p} \Vdash_{\mathcal{A}} \bigwedge\!\!\!\bigwedge_{i\in I} \psi_i$ if, for all $i \in I$ and $\bar{q} \in A^\star$ with $\bar{q} \supseteq \bar{p}$, there exists $\bar{r} \supseteq \bar{q}$ such that $\bar{r} \Vdash_{\mathcal{A}} \psi_i$.

We will often omit the sub-index $\mathcal{A}$ and write $\bar{p} \Vdash \varphi$ if it is clear which structure we are talking about.

By writing down this definition formally, we can see that for every $\mathbb{N}$-formula $\varphi(\dot{\mathcal{G}})$, there is a τ-formula *Force*$_\varphi$ such that

$$\mathcal{A} \models \mathit{Force}_\varphi(\bar{p}) \quad\iff\quad \bar{p} \Vdash_{\mathcal{A}} \varphi.$$

Definition VII.8. Here is the definition in detail:[¶]

- *Force*$_\top$ is $\top$ and *Force*$_\bot$ is $\bot$.

[§] Recall that $D_{\mathcal{A}}(\bar{p})$ is the atomic diagram of the tuple $\bar{p}$ in the structure $\mathcal{A}$ (see page xxiii). It is a finite binary string. By $D_{\mathcal{A}}(\bar{p})(n)\downarrow$ we just mean that $n < |D_{\mathcal{A}}(\bar{p})|$.

[¶] *Force*$_\varphi$ is not exactly a single formula, but a computable sequence of formulas, one of each arity.

- $\mathit{Force}_{\dot{\mathcal{G}}(\mathbf{n})}(\bar{p})$ is $\varphi^{\mathrm{at}}_n(\bar{p})$.[‖]
- $\mathit{Force}_{\neg\dot{\mathcal{G}}(\mathbf{n})}(\bar{p})$ is $\neg\varphi^{\mathrm{at}}_n(\bar{p})$.
- $\mathit{Force}_{\bigdoublevee_{i\in I}\psi_i}(\bar{p})$ is $\bigdoublevee_{i\in I}\mathit{Force}_{\psi_i}(\bar{p})$.
- $\mathit{Force}_{\bigdoublewedge_{i\in I}\psi_i}(\bar{p})$ is $\bigdoublewedge_{i\in I}\forall\bar{q}\supseteq\bar{p}\;\exists\bar{r}\supseteq\bar{q}\;\mathit{Force}_{\psi_i}(\bar{r})$.

It is not hard to prove by induction that if the formula φ is $\mathbb{N}$-$\Pi^{\mathtt{c}}_{\alpha}$, then Force_φ is τ-$\Pi^{\mathtt{c}}_{\alpha}$, and that if φ is $\mathbb{N}$-$\Sigma^{\mathtt{c}}_{\alpha}$, then Force_φ is τ-$\Sigma^{\mathtt{c}}_{\alpha}$. There is one case, namely the $\Pi^{\mathtt{c}}_1$ case, that requires a little observation: For an atomic formula of the form $\dot{\mathcal{G}}(\mathbf{n})$, we have that if $|\bar{q}|\geq n$, then either $\bar{q}\Vdash\dot{\mathcal{G}}(\mathbf{n})$ or $\bar{q}\Vdash\neg\dot{\mathcal{G}}(\mathbf{n})$. This is because of our old convention that $\varphi^{\mathrm{at}}_n(\bar{p})$ has no free variables beyond $x_0,\dots,x_{n-1}$ (see page xxiii). Thus, when we have an $\mathbb{N}$-$\Pi^{\mathtt{c}}_1$-formula $\bigdoublewedge_{i\in I}\psi_i$ where each ψ_i is a literal, we get that $\bar{p}\Vdash\bigdoublewedge_{i\in I}\psi_i$ if and only if

- there are no $\bot$'s among the ψ_i for $i\in I$,
- for all ψ_i of the form $\dot{\mathcal{G}}(\mathbf{n})$ and all $\bar{q}\supseteq\bar{p}$ of length greater than n, $\varphi^{\mathrm{at}}_n(\bar{q})$, and
- for all ψ_i of the form $\neg\dot{\mathcal{G}}(\mathbf{n})$ and all $\bar{q}\supseteq\bar{p}$ of length greater than n, $\neg\varphi^{\mathrm{at}}_n(\bar{q})$.

So, $\mathit{Force}_{\bigdoublewedge_{i\in I}\psi_i}$ is τ-$\Pi^{\mathtt{c}}_1$.

We will call Σ-*formulas* the formulas that start with $\bigdoublevee$, and call Π-*formulas* the ones that start with $\bigdoublewedge$. The definition of $\Vdash$ for Π-formulas can be restated as follows: $\bar{p}\Vdash\bigdoublewedge_{i\in I}\psi_i$ if and only if, for all $i\in I$, the relation $\{\bar{r}\in A^\star:\bar{r}\Vdash\psi_i\}$ is dense above $\bar{p}$.[**] It follows that for a Π-formula φ, if the set $\{\bar{r}\in A^\star:\bar{r}\Vdash\varphi\}$ is dense above $\bar{p}$, then $\bar{p}\Vdash\varphi$. This is not necessarily true for Σ-formulas, as one could have that the set $\{\bar{r}\in A^\star:\bar{r}\Vdash\varphi\}$ is dense, but each such $\bar{r}$ forces φ via a different witness i. This causes a minor annoyance: The truth of the forcing relation depends on how the formula φ is written. For instance, if φ is a Σ-formula and $\bigdoublewedge\varphi$ is built by adding a dummy conjunction in front, then we may have some $\bar{p}$ that forces $\bigdoublewedge\varphi$ but not φ. Fortunately, this can only happen for Σ-formulas, as we will see in Corollary VII.15.

Lemma VII.9. *Let φ be an $\mathbb{N}$-$\mathcal{L}_{\mathtt{c},\omega}$-formula and $\bar{p},\bar{q}$ injective tuples from a structure $\mathcal{A}$.*

- *If $\bar{p}\subseteq\bar{q}$ and $\bar{p}\Vdash\varphi$, then $\bar{q}\Vdash\varphi$.*
- *For no $\bar{p}$ and φ we have $\bar{p}\Vdash\varphi$ and $\bar{p}\Vdash\neg\varphi$.*

[‖] Recall that φ^{at}_n is the nth atomic τ-formula. By $\varphi^{\mathrm{at}}_n(\bar{p})$, we mean $\varphi^{\mathrm{at}}_n[x_i\mapsto p_i]$. If φ^{at}_n has some free variable x_j with $j\geq|\bar{p}|$, we let both $\varphi^{\mathrm{at}}_n(\bar{p})$ and $\neg\varphi^{\mathrm{at}}_n(\bar{p})$ be false.

[**] We say that a relation $R\subseteq A^\star$ is *dense above* $\bar{p}$ if $\forall q\supseteq\bar{p}\;\exists\bar{r}\supseteq\bar{q}\;(\bar{r}\in R)$.

Proof. The first part is a straightforward induction on φ.

The second part is also by induction on φ. It is trivial for literals. Suppose φ is of the form $\bigvee\!\!\!\bigvee_{i\in I}\psi_i$ and that $\bar{p}$ forces φ and $\neg\varphi$. Let i_0 be such that $\bar{p} \Vdash \psi_{i_0}$. Since $\bar{p} \Vdash \bigwedge\!\!\!\bigwedge_{i\in I}\neg\psi_i$, there is a $\bar{q} \supseteq \bar{p}$ which forces $\neg\psi_{i_0}$. By the previous part of the lemma, $\bar{q}$ also forces ψ_{i_0}. This contradicts the induction hypothesis applied to ψ_{i_0}. □

Lemma VII.10. *For every* $\mathbb{N}$-$\mathcal{L}_{\mathsf{c},\omega}$*-sentence* φ*, the set*

$$\mathcal{D}_\varphi = \{\bar{p} \in A^\star : \bar{p} \Vdash \varphi \ \vee \ \bar{p} \Vdash \neg\varphi\}$$

is dense in $A^\star$*.*

Proof. If φ is atomic, then every $\bar{p}$ whose length is large enough forces either φ or $\neg\varphi$.

Suppose now that φ is of the form $\bigwedge\!\!\!\bigwedge_{i\in I}\psi_i$. Pick a tuple $\bar{p} \in A^\star$. If $\bar{p} \nVdash \varphi$, there is some $\bar{q} \supseteq \bar{p}$ and $i_0 \in I$ such that for every $\bar{r} \supseteq \bar{q}$, $\bar{r} \nVdash \psi_{i_0}$. By the induction hypothesis, we get that the set of $\bar{r}$'s extending $\bar{q}$ and forcing either ψ_{i_0} or $\neg\psi_{i_0}$ is dense above $\bar{q}$. Since no $\bar{r} \supseteq \bar{q}$ forces ψ_{i_0}, there exist plenty of $\bar{r}$'s extending $\bar{q}$ such that $\bar{r} \Vdash \neg\psi_{i_0}$. For any such $\bar{r}$, we have $\bar{r} \Vdash \neg\varphi$. Summing up, either $\bar{p} \Vdash \varphi$ or there is some $\bar{r}$ extending $\bar{p}$ such that $\bar{r} \Vdash \neg\varphi$. □

We say that $\bar{p}$ *decides* φ if either $\bar{p} \Vdash \varphi$ or $\bar{p} \Vdash \neg\varphi$.

Corollary VII.11. *If* g *is* $\mathcal{L}_{\mathsf{c},\omega}$*-generic, for every* $\mathbb{N}$-$\mathcal{L}_{\mathsf{c},\omega}$*-formula* φ*, there is a* $\bar{p} \subset g$ *that decides* φ*.*

Proof. The relation $\mathcal{D}_\varphi \subseteq A^\star$ from the lemma above is $\mathcal{L}_{\mathsf{c},\omega}$-definable and dense, so it has to be met by g. □

Lemma VII.12. *(Forcing-implies-truth) If* $\bar{p} \Vdash \varphi$*, then* $\bar{p}$ *semantically forces* φ*.*

Proof. The proof is again by induction on φ. The result is immediate when φ is atomic. When φ is a Σ-formula, the induction step is also quite easy. When φ is a Π-formula of the form $\bigwedge\!\!\!\bigwedge_{i\in I}\psi_i$, we have that for each $i \in I$, the set of $\bar{q} \supseteq \bar{p}$ forcing ψ_i is dense above $\bar{p}$, and hence g must meet it. That is, for each $i \in I$, g has an initial segment that forces ψ_i, and then, by the induction hypothesis, ψ_i must be true of $g^{-1}(\mathcal{A})$. Then so is $\bigwedge\!\!\!\bigwedge_{i\in I}\psi_i$. □

Theorem VII.13. *(Forcing-equals-truth) If* g *is an* $\mathcal{L}_{\mathsf{c},\omega}$*–generic enumeration of* $\mathcal{A}$*,* $\mathcal{G} = g^{-1}(\mathcal{A})$*, and* $\varphi(\dot{\mathcal{G}})$ *is an* $\mathbb{N}$-$\mathcal{L}_{\mathsf{c},\omega}$*-formula , then*

$$\varphi(\mathcal{G}) \quad \iff \quad (\exists \bar{p} \subset g)\ \bar{p} \Vdash \varphi$$

Proof. The ($\Leftarrow$) direction follows from Lemma VII.12. For the ($\Rightarrow$) direction, we get from Corollary VII.11 that some initial segment of g must force either φ or $\neg\varphi$. But, because of Lemma VII.12, it cannot force $\neg\varphi$. □

Corollary VII.14. *For all $\bar{p} \in A^\star$ and $\mathcal{L}_{\mathsf{c},\omega}$-formula $\varphi(\dot{\mathcal{G}})$, $\bar{p}$ semantically forces φ if and only if $\{\bar{q} \in A^\star : \bar{q} \Vdash \varphi\}$ is dense above $\bar{p}$.*

Proof. For the ($\Leftarrow$) direction, if $\{\bar{q} \in A^\star : \bar{q} \Vdash \varphi\}$ is dense above $\bar{p}$ and g is a $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration extending $\bar{p}$, then g must meet $\{\bar{q} \in A^\star : \bar{q} \Vdash \varphi\}$ and hence, by the theorem, $\mathcal{G}$ must satisfy φ.

For the ($\Rightarrow$) direction, suppose that $\bar{p}$ semantically forces φ and consider $\bar{r} \supseteq \bar{p}$. It has an $\mathcal{L}_{\mathsf{c},\omega}$-generic extension. That generic extension satisfies φ, and hence it has an initial segment $\bar{q}$ which forces φ, which we may take to be longer than $\bar{r}$. This shows that $\{\bar{q} \in A^\star : \bar{q} \Vdash \varphi\}$ is dense above $\bar{p}$. □

Corollary VII.15. *For Π-formulas, $\bar{p} \Vdash \varphi$ if and only if $\bar{p}$ semantically forces φ.*

Proof. This follows from our observation that for Π-formulas, $\bar{p} \Vdash \varphi$ if and only if $\{\bar{r} \in A^\star : \bar{r} \Vdash \varphi\}$ is dense above $\bar{p}$. □

The "semantically forces" relation is thus slightly weaker than the $\Vdash$ relation. It is sometimes called *weak forcing* and denoted $\Vdash^w$. Similarly, the $\Vdash$ relation as defined here is sometimes called *strong forcing*. In set theory, as for instance in [**Kun80**], the standard notion is that of weak forcing. For computability theorists, though, $\Vdash^w$ has a problem: It does not preserve complexity. For Π-formulas it does, but weakly forcing a $\Sigma^{\mathsf{c}}_\alpha$ formula is $\Pi^{\mathsf{c}}_{\alpha+1}$, and that messes up our complexity considerations.

The semantical forcing can also be defined syntactically. One needs the following modification in the definition of the forcing relation:

- $\bar{p} \Vdash^w \bigvee\!\!\!\bigvee_{i\in I} \psi_i$ if $\forall \bar{q} \supseteq \bar{p}\ \bigvee\!\!\!\bigvee_{i\in I} \exists \bar{r} \supseteq \bar{q}\ (\bar{r} \Vdash^w \psi_i)$.
- $\bar{p} \Vdash^w \bigwedge\!\!\!\bigwedge_{i\in I} \psi_i$ if $\bigwedge\!\!\!\bigwedge_{i\in I} \big(\bar{p} \Vdash^w \psi_i\big)$.

VII.3. The Ash-Knight-Manasse-Slaman–Chisholm theorem

This is one of the most classic theorems of computable structure theory. It shows why the computable infinitary language is so important in computable structure theory.

Definition VII.16. A relation $R \subseteq A^n$ is said to be *relatively intrinsically* Σ^0_α if, on every copy $(\mathcal{B}, R^{\mathcal{B}})$ of $(\mathcal{A}, R)$, $R^{\mathcal{B}}$, viewed as a subset of $\mathbb{N}^n$, is Σ^0_α relative to the oracle $D(\mathcal{B}) \in 2^{\mathbb{N}}$.[††]

[††] Recall that a *copy* of a structure is an ω-presentation isomorphic to it.

Let us remark that this is a purely computability theoretic notion defined in terms of the ω-presentations of the structure $\mathcal{A}$ and the arithmetic and hyperarithmetic hierarchies. The Ash-Knight-Manasse-Slaman–Chisholm theorem shows that it can be characterized in a purely structural way.

Theorem VII.17. *(Ash-Knight-Manasse-Slaman* [**AKMS89**] *— Chisholm* [**Chi90**]*) A relation $R \subseteq A^n$ is relatively intrinsically Σ^0_α if and only if it is τ-$\Sigma^{\mathsf{c}}_\alpha$-definable in $\mathcal{A}$ over a finite tuple of parameters.*

Proof. The ($\Leftarrow$) direction is straightforward from Lemma V.6.

For the other direction, we need to find a τ-$\Sigma^{\mathsf{c}}_\alpha$-definition of R. Let g be an $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration of $(\mathcal{A}, R)$ and $(\mathcal{G}, R^{\mathcal{G}})$ be the corresponding $\mathcal{L}_{\mathsf{c},\omega}$-generic presentation. Since $R^{\mathcal{G}} \subseteq \mathbb{N}^n$ is $\Sigma^{0,D(\mathcal{G})}_\alpha$, there is an $\mathbb{N}$-$\Sigma^{\mathsf{c}}_\alpha$-formula $\varphi(\bar{k}, \dot{\mathcal{G}})$ such that, for all $\bar{k} \in \mathbb{N}^n$,

$$\bar{k} \in R^{\mathcal{G}} \iff \varphi(\bar{k}, \mathcal{G}).$$

Whether "$\bar{k} \in R^{\mathcal{G}}$" is true can be read off from the atomic diagram of $(\mathcal{G}, R^{\mathcal{G}})$, so the formula

$$\bigwedge\!\!\!\bigwedge_{\bar{k} \in \mathbb{N}^n} (\bar{k} \in R^{\dot{\mathcal{G}}} \iff \varphi(\bar{k}, \dot{\mathcal{G}}))$$

is an $\mathbb{N}$-$\mathcal{L}_{\mathsf{c},\omega}$ formula that, by the forcing-equals-truth theorem, must be forced by some $\bar{p} \subset g$.

Consider a tuple $\bar{a} \in A^n$. Each line below can be easily seen to be equivalent to the previous one:

- $\bar{a}$ belongs to R.
- for some $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration g of $(\mathcal{A}, R)$ extending $\bar{p}$, $g^{-1}(\bar{a}) \in R^{\mathcal{G}}$.
- for some $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration g of $(\mathcal{A}, R)$ extending $\bar{p}$ and for $\bar{k}$ such that $g(\bar{k}) = \bar{a}$, we have $\varphi(\bar{k}, \mathcal{G})$.
- for some tuple $\bar{q} \supseteq \bar{p}$ which contains $\bar{a}$ in its image and for $\bar{k}$ such that $\bar{q}(\bar{k}) = \bar{a}$, we have $\bar{q} \Vdash \varphi(\bar{k}, \dot{\mathcal{G}})$.

Rewriting once more, we get:

$$\bar{a} \in R \quad \iff \quad \exists \bar{q} \supseteq \bar{p} \bigvee_{\bar{k} \in |\bar{q}|^n} (\bar{a} = \bar{q}(\bar{k}) \ \&\ \mathit{Force}_{\varphi(\bar{k}, \dot{\mathcal{G}})}(\bar{q})).^{\ddagger\ddagger}$$

Notice that $\varphi(\bar{k}, \dot{\mathcal{G}})$ is a formula only about $D(\mathcal{G})$, and it does not use $R^{\mathcal{G}}$. So any atomic sub-formula of $\varphi(\bar{k}, \dot{\mathcal{G}})$ must be of the form $\dot{\mathcal{G}}(n)$ or $\neg\dot{\mathcal{G}}(n)$, and n is such that the symbol R does not appear in

[‡‡] By $|\bar{q}|^n$ we mean the set of n tuples of numbers below the length of $\bar{q}$. If $\bar{q} = \langle q_0, ..., q_{|\bar{q}|-1}\rangle$ and $\bar{k} = \langle k_0, ..., k_{n-1}\rangle \in |\bar{q}|^n$, then $\bar{q}(\bar{k})$ is the n-tuple $\langle q_{k_0}, ..., q_{k_{n-1}}\rangle$.

φ_n^{at}. It follows that $\mathit{Force}_{\varphi(\bar{k},\dot{\mathcal{G}})}$ is a τ-$\Sigma^{\mathsf{c}}_{\alpha}$-formula that does not use R (see discussion after Definition VII.8). Also, let us recall that $\bar{a} = \bar{q}(\bar{k})$ is shorthand for $\bigwedge_{i<|\bar{k}|} \bar{q}(k_i) = a_i$. We have thus obtained a τ-$\Sigma^{\mathsf{c}}_{\alpha}$ definition of R using $\bar{p}$ as parameters. □

As in [**Part 1**], we extend the notion of *relation* to mean any subset of $A^{<\mathbb{N}}$, instead of just a subset of A^n. Furthermore, we will sometimes be interested in uniformly definable sequences of relations $\langle R_0, R_1, ...\rangle$, which can be seen as a single subset of $\mathbb{N} \times A^{<\mathbb{N}}$. Since most of our complexity-related results apply to subsets of A^n as well as subsets of $\mathbb{N} \times A^{<\mathbb{N}}$, we will consider the latter as our notion of relation, which is general enough to encapsulate all other notions of relation. We say that $R \subseteq \mathbb{N} \times A^{<\mathbb{N}}$ is $\Sigma^{\mathsf{c}}_{\alpha}$-*definable* in $\mathcal{A}$ if there is a computable sequence $\langle \varphi_{m,n} : m, n \in \mathbb{N}\rangle$ of $\Sigma^{\mathsf{c}}_{\alpha}$ formulas such that each relation $R_{m,n}$ given by $\{\bar{a} \in A^n : \langle m, \bar{a}\rangle \in R\}$ is definable by $\varphi_{m,n}$. A relation $R \subseteq \mathbb{N} \times A^{<\mathbb{N}}$ is *relatively intrinsically* Σ^0_{α} if, on every copy $(\mathcal{B}, R^{\mathcal{B}})$ of $(\mathcal{A}, R)$, $R^{\mathcal{B}}$, viewed as a subset of $\mathbb{N} \times \mathbb{N}^{<\mathbb{N}}$, is Σ^0_{α} relative to the oracle $D(\mathcal{B}) \in 2^{\mathbb{N}}$.

THEOREM VII.18. *A relation $R \subseteq \mathbb{N} \times A^{<\mathbb{N}}$ is relatively intrinsically Σ^0_{α} if and only if it is τ-$\Sigma^{\mathsf{c}}_{\alpha}$-definable in $\mathcal{A}$ over a finite tuple of parameters.*

PROOF. All we have to do is observe that the previous proof works uniformly. The ($\Leftarrow$) direction is again straightforward from Lemma V.6. For the other direction, we need to define a τ-$\Sigma^{\mathsf{c}}_{\alpha}$-definition of R, that is, a τ-$\Sigma^{\mathsf{c}}_{\alpha}$-definition of $R_{m,n}$ for each m and n, computably uniformly in m and n.

Let g be an $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration of $(\mathcal{A}, R)$ and $\mathcal{G}$ the corresponding $\mathcal{L}_{\mathsf{c},\omega}$-generic presentation. Since R is $\Sigma^{0,D(\mathcal{G})}_{\alpha}$, there is a computable sequence of $\mathbb{N}$-$\Sigma^{\mathsf{c}}_{\alpha}$-formulas $\varphi_{m,n}(\bar{k}, \dot{\mathcal{G}})$ such that

$$\bigwedge\!\!\!\bigwedge_{m,n\in\mathbb{N}} \bigwedge\!\!\!\bigwedge_{\bar{k}\in\mathbb{N}^n} ((m, \bar{k}) \in R^{\dot{\mathcal{G}}} \iff \varphi_{m,n}(\bar{k}, \dot{\mathcal{G}})). \tag{4}$$

This $\mathbb{N}$-$\mathcal{L}_{\mathsf{c},\omega}$ formula must be forced by some $\bar{p} \subset g$.

The rest of the proof is the same as the one above. We end up obtaining

$$\bar{a} \in R_{m,n} \quad\iff\quad \exists \bar{q} \supseteq \bar{p} \bigvee_{\bar{k}\in|\bar{q}|^n} (\bar{a} = \bar{q}(\bar{k})\ \&\ \mathit{Force}_{\varphi_{m,n}(\bar{k},\dot{\mathcal{G}})}(\bar{q})).$$

The right-hand side is the desired τ-$\Sigma^{\mathsf{c}}_{\alpha}$ definition of $R_{m,n}$, which can be computed uniformly from m and n. □

We say that R is *uniformly relatively intrinsically* Σ^0_{α} if there is a Σ^0_{α} operator Γ such that on every copy $(\mathcal{B}, R^{\mathcal{B}})$ of $(\mathcal{A}, R)$, $R^{\mathcal{B}} = \Gamma^{D(\mathcal{B})}$. For these relations, we can get rid of the parameters in the theorem above. That is, a relation $R \subseteq \mathbb{N} \times A^{<\mathbb{N}}$ is uniformly relatively intrinsically Σ^0_α if and only if it is τ-$\Sigma^{\mathsf{c}}_\alpha$-definable in $\mathcal{A}$ without parameters. To see this, notice that all generic presentations $\mathcal{G}$ would use the same $\mathbb{N}$-Σ^0_α-formulas $\varphi_{m,n}$ for $R^{\mathcal{G}}_{m,n}$. The formula (4) is thus true for all generics and hence forced by the empty tuple $\langle\rangle$. The rest of the proof continues with $\bar{p} = \langle\rangle$.

If we consider relations that are subsets of $\mathbb{N} \times A^0$, we can look at the binary information that is coded in a structure. We say that a set $X \subseteq \mathbb{N}$ is Σ^0_α-*coded* in a structure $\mathcal{A}$ if it is Σ^0_α relative to the diagram of any copy of $\mathcal{A}$.

Corollary VII.19 (Knight [**Kni86**, Theorem 1.4'], see also [**AK00**, Theorem 10.17]). *A set X is Σ^0_α-coded in a structure if and only if it is many-one reducible to the $\Sigma^{\mathsf{c}}_\alpha$-type of some tuple in $\mathcal{A}$.*

Proof. Suppose X is many-one reducible to a $\Sigma^{\mathsf{c}}_\alpha$-type of some tuple $\bar{p} \in A^{<\mathbb{N}}$. The $\Sigma^{\mathsf{c}}_\alpha$-type of $\bar{p}$ is Σ^0_α relative to the diagram of any copy of $\mathcal{A}$. Since a set that is many-one reducible to a Σ^0_α set is also Σ^0_α, we have that X is Σ^0_α relative to the diagram of any copy of $\mathcal{A}$ too.

Conversely, if $X \subseteq \mathbb{N}$ is Σ^0_α-coded in a structure $\mathcal{A}$, then the relation $X \times \langle\rangle \subseteq \mathbb{N} \times A^0 \subseteq \mathbb{N} \times A^{<\mathbb{N}}$ is relatively intrinsically $\Sigma^{\mathsf{c}}_\alpha$ in $\mathcal{A}$, and hence $\Sigma^{\mathsf{c}}_\alpha$-definable in $\mathcal{A}$ over some parameters $\bar{p}$. Recall that, as we defined in page 115, a relation $R \subseteq \mathbb{N} \times A^{<\mathbb{N}}$ is $\Sigma^{\mathsf{c}}_\alpha$-definable in $\mathcal{A}$ over parameters $\bar{p}$ if and only if there exists a computable list $\{\varphi_{n,k}(\bar{x}, \bar{y}) : n, k \in \mathbb{N}\}$ of $\Sigma^{\mathsf{c}}_\alpha$ formulas such that $\langle n, \bar{a}\rangle \in R \iff \mathcal{A} \models \varphi_{n,|\bar{a}|}(\bar{p}, \bar{a})$. In the case when we have a relation $X \times \langle\rangle \subseteq \mathbb{N} \times A^0$, what we have is a computable list $\{\varphi_{n,0}(\bar{x}) : n \in \mathbb{N}\}$ of $\Sigma^{\mathsf{c}}_\alpha$ formulas such that $n \in X \iff \mathcal{A} \models \varphi_{n,0}(\bar{p})$. X is then m-reducible to the $\Sigma^{\mathsf{c}}_\alpha$-type of $\bar{p}$ as follows:

$$n \in X \quad \iff \quad \varphi_{n,0} \in \Sigma^{\mathsf{c}}_\alpha\text{-}tp(\bar{p}). \qquad \square$$

VII.4. Relative Δ^0_α-categoricity

Computably categorical structures are the ones for which all computable ω-presentations have the same computational properties. We studied them in detail in [**Part 1**, Chapter VIII]. Most structures are not computably categorical, but instead, one needs a certain number of Turing jumps to compute isomorphisms between different presentations.

Definition VII.20 ([**AKMS89**, Section 4][**Chi90**, Definition V.9]). A computable structure $\mathcal{A}$ is *relatively Δ^0_α categorical* if, for every copy $\mathcal{B}$ of $\mathcal{A}$, there is an isomorphism between $\mathcal{A}$ and $\mathcal{B}$ that is Δ^0_α relative to $D(\mathcal{B})$.

For example, $(\mathbb{Z}; \leq)$ is Δ^0_2-categorical, as with the help of a Turing jump we can decide which pairs of elements are adjacent and build isomorphisms between different copies.

The theorem below gives us a syntactical characterization for the notion of relative Δ^0_α-categoricity. As a corollary we will get that the Scott rank of a structure is the least α such that the structure is relative Δ^0_α-categoricity on a cone (see Remark VII.23). This provides even more evidence for the robustness of the notion of Scott rank.

A *Scott family* for a structure $\mathcal{A}$ is a set S of formulas which define all the automorphism orbits in $\mathcal{A}$. That is, for each tuple in $\mathcal{A}$, there is a formula in S defining its orbit, and each formula in S defines some orbit in $\mathcal{A}$. The definition of Scott rank (Definition II.16) can be rephrased as saying that a structure has Scott rank α if and only if it has a Scott family of $\Sigma^{\mathtt{in}}_\alpha$ formulas over some tuple of parameters.

Theorem VII.21. [**AKMS89**] [**Chi90**] *Let $\mathcal{A}$ be a computable structure. $\mathcal{A}$ is relatively Δ^0_α-categorical if and only if it has a c.e. Scott family of $\Sigma^{\mathtt{c}}_\alpha$-formulas over a finite tuple of parameters.*

Proof. The $(\Leftarrow)$ direction is the easier one. Suppose $\mathcal{A}$ has a c.e. Scott family S of $\Sigma^{\mathtt{c}}_\alpha$-formulas over a tuple $\bar{c} \in A^{<\mathbb{N}}$. Given a copy $(\mathcal{B}, \bar{d})$ of $(\mathcal{A}, \bar{c})$, consider the set

$$\begin{aligned} I_{\mathcal{A},\mathcal{B}} &= \{\langle \bar{a}, \bar{b}\rangle \in A^{<\mathbb{N}} \times B^{<\mathbb{N}} : \\ &\qquad\qquad (\text{for some } \varphi \in S)\ \mathcal{A} \models \varphi(\bar{c}\bar{a})\ \&\ \mathcal{B} \models \varphi(\bar{d}\bar{b})\} \\ &= \{\langle \bar{a}, \bar{b}\rangle \in A^{<\mathbb{N}} \times B^{<\mathbb{N}} : (\mathcal{A}, \bar{c}\bar{a}) \cong (\mathcal{B}, \bar{d}\bar{b})\}, \end{aligned}$$

as in [**Part 1**, Observation III.14]. This set is Σ^0_α in $D(\mathcal{B})$ and has the back-and-forth property (Definition II.6). In [**Part 1**, Lemma III.15] we showed that there is then an isomorphism between $\mathcal{A}$ and $\mathcal{B}$ that is computable from an enumeration of $I_{\mathcal{A},\mathcal{B}}$. Since $I_{\mathcal{A},\mathcal{B}}$ is Σ^0_α in $D(\mathcal{B})$, there is an enumeration of $I_{\mathcal{A},\mathcal{B}}$ that is Δ^0_α in $D(\mathcal{B})$.

Let us now prove the $(\Rightarrow)$ direction. Let $\mathcal{K}$ be a computable injective ω-presentation of $\mathcal{A}$ with domain $\mathbb{N}$. $\mathcal{A}$ and $\mathcal{K}$ may be taken to be the same ω-presentation, but during the proof, it will be easier to think of $\mathcal{A}$ as the abstract structure for which we want to find a Scott family, and $\mathcal{K}$ as a fixed computable ω-presentation whose elements are named by natural numbers.

Let $\mathcal{G}$ be an $\mathcal{L}_{\mathtt{c},\omega}$-generic presentation of $\mathcal{A}$, and $g\colon \mathbb{N} \to A$ its corresponding generic enumeration. Let F_g be an isomorphism from $\mathcal{G}$ to $\mathcal{K}$ that is Δ^0_α relative to $\mathcal{G}$. $\mathcal{G}$ and $\mathcal{K}$ are injective ω-presentations with domain $\mathbb{N}$, so F_g is a bijection $\mathbb{N} \to \mathbb{N}$.

Notice that each generic g induces an isomorphism $F_g \circ g^{-1}$ from $\mathcal{A}$ to $\mathcal{K}$. Here is the general idea: To define the automorphism orbits, consider, for each tuple $\bar k \in \mathcal{K}$, the set of all the tuples $\bar a$ for which there is a generic enumeration g such that $F_g \circ g^{-1}(\bar a) = \bar k$, and give a τ-definition of this set using the forcing relation. Let us look at the details.

Let $\psi(x, y, \dot{\mathcal{G}})$ be an $\mathbb{N}$-Σ^0_α-formula such that

$$\forall n, k \in \mathbb{N} \ \ (F_g(n) = k \iff \psi(n, k, \mathcal{G})).$$

There is an $\mathbb{N}$-$\mathcal{L}_{\mathsf{c},\omega}$ formula that says that $\psi(x, y, \mathcal{G})$ is a bijection $\mathbb{N} \to \mathbb{N}$ and that it is an isomorphism between the ω-presentations of $\mathcal{G}$ and $\mathcal{K}$. This latter part uses $D(\mathcal{G})$ and $D(\mathcal{K})$, but since $D(\mathcal{K})$ is computable, the formula only uses $D(\mathcal{G})$. Let $\bar p$ force that ψ defines an isomorphism from $\mathcal{G}$ to $\mathcal{K}$. Then, for every $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration $\breve g$ extending $\bar p$, we have that for the associated generic presentation $\breve{\mathcal{G}}$, the set $\{\langle n, k\rangle \in \mathbb{N}^2 : \psi(n, k, \breve{\mathcal{G}})\}$ is the graph of an isomorphism from $\breve{\mathcal{G}}$ to $\mathcal{K}$. From now on, we use $F_{\breve g}$ to denote this isomorphism defined by $\psi(\cdot, \cdot, \breve{\mathcal{G}})$.

We want to define a c.e. Scott family of $\Sigma^{\mathsf{c}}_\alpha$-formulas over $\bar p$. That is, we want to find $\Sigma^{\mathsf{c}}_\alpha$ definitions for all automorphism orbits of $(\mathcal{A}, \bar p)$. Fix a tuple $\bar a \in A^{<\mathbb{N}}$. We want to write down a τ-$\Sigma^{\mathsf{c}}_\alpha$-formula that defines the automorphism orbit of $\bar a$ over $\bar p$.

Let $\bar k = F_g(g^{-1}(\bar p, \bar a)) \in K^{|\bar p\bar a|}$ so that $(\mathcal{A}, \bar p\bar a) \cong (\mathcal{K}, \bar k)$.[*] Let $O_{\bar k}$ be the set of all $\bar b \in A^{|\bar a|}$ for which there exists some $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration $\breve g$ extending $\bar p$ such that $F_{\breve g}(\breve g^{-1}(\bar p, \bar b)) = \bar k$. We claim that the following three statements are equivalent for every tuple $\bar b \in A^{<\mathbb{N}}$:

(1) $\bar b \in O_{\bar k}$.
(2) $\bar b$ is automorphic to $\bar a$ over $\bar p$.
(3) $\bar b$ satisfies the formula $\varphi_{\bar k}(\bar p, \bar x)$ given by

$$\exists \bar q \supseteq \bar p \ \bigvee_{\bar n \in |\bar q|^{|\bar p\bar a|}} \big(\bar p\bar x = \bar q(\bar n) \ \ \wedge \ \ \bar q \Vdash \bigwedge_{i<|\bar p\bar a|} \psi(n_i, k_i, \dot{\mathcal{G}})\big).$$

Let us recall that $\bar p\bar x = \bar q(\bar n)$ is shorthand for $\bigwedge_{i<|\bar n|} \bar q(n_i) = \bar p\bar x(n_i)$. Notice that the sub-formula $\bar q \Vdash \bigwedge_{i<|\bar k|} \psi(n_i, k_i, \dot{\mathcal{G}})$ implies that $\bar n = F_{\breve g}(\bar k)$ for every $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration $\breve g$ extending $\bar q$.

For the (1) ⇒ (2) implication, if $\bar b \in O_{\bar k}$, it can be shown that $\bar b$ must be automorphic to $\bar a$ over $\bar p$ by composing isomorphisms as follows:

$$\big(\mathcal{A}, \bar p\bar a\big) \xleftarrow[\cong]{g} \big(\mathcal{G}, g^{-1}(\bar p\bar a)\big) \xrightarrow[\cong]{F_g} \big(\mathcal{K}, \bar k\big) \xleftarrow[\cong]{F_{\breve g}} \big(\breve{\mathcal{G}}, \breve g^{-1}(\bar p\bar b)\big) \xrightarrow[\cong]{\breve g} \big(\mathcal{A}, \bar p\bar b\big).$$

The (2) ⇒ (3) implication follows from the fact that formulas are preserved under automorphisms, and that $\mathcal{A} \models \varphi_{\bar k}(\bar p, \bar a)$. That $\mathcal{A} \models$

[*] Recall that we use $\bar p\bar a$ to denote the tuple obtained by concatenating $\bar p$ and $\bar a$.

$\varphi_{\bar{k}}(\bar{p}, \bar{a})$ is witnessed by a long enough initial segment $\bar{q}$ of g and by $\bar{n} = g^{-1}(\bar{p}\bar{a}) = F_g^{-1}(\bar{k})$.

For the (3) ⇒ (1) implication, suppose that $\bar{b}$ satisfies $\varphi_{\bar{k}}(\bar{p}, \bar{x})$ as witnessed by $\bar{q}$ and $\bar{n}$. Let $\breve{g}$ be any $\mathcal{L}_{\mathsf{c},\omega}$-generic extension of $\bar{q}$. Then $\breve{g}(\bar{n}) = \bar{p}\bar{b}$ and, by the forcing-equals-truth theorem, we must have $\bar{n} = F_{\breve{g}}(\bar{k})$.

It follows that $O_{\bar{k}}$ is indeed the automorphism orbit of $\bar{a}$ over $\bar{p}$ and that it is definable by the $\Sigma^{\mathsf{c}}_{\alpha}$-formula $\varphi_{\bar{k}}(\bar{p}, \bar{x})$. We then have that

$$\{\varphi_{\bar{k}}(\bar{p}, \bar{x}) : \bar{k} \in K^{<\mathbb{N}}\}$$

is a c.e. Scott family of $\mathcal{A}$ of $\Sigma^{\mathsf{c}}_{\alpha}$-formulas over the parameters $\bar{p}$. □

Remark VII.22. In the theorem above, if we had that $\mathcal{A}$ was *unifomrly Δ^0_α-categorical*, meaning that there is a single Δ^0_α operator Γ such that, for all copies $\mathcal{G}$ of $\mathcal{A}$, $\Gamma^{D(\mathcal{G})}$ is an isomorphism between $\mathcal{G}$ and $\mathcal{A}$, then we would obtain a Scott family without parameters. The reason is that the same formula $\psi(x, y, \dot{\mathcal{G}})$ would define an isomorphism from $\mathcal{G}$ to $\mathcal{K}$ for all generic presentations $\mathcal{G}$, and hence this would be forced by the empty condition. So, we could take $\bar{p} = \langle\rangle$.

Remark VII.23. We say that a structure $\mathcal{A}$ is *relatively (uniformly) Δ^0_α-categorical on a cone* if there is an oracle X such that $\mathcal{A}$ is relatively (uniformly) Δ^0_α-categorical relative to X. It follows from the theorem above that a structure $\mathcal{A}$ is relatively Δ^0_α-categorical on a cone if and only if it has a Scott family of $\Sigma^{\mathtt{in}}_{\alpha}$ formulas over some tuple $\bar{p}$ of parameters. We thus get the following corollary.

Corollary VII.24. *The parametrized Scott rank of a structure is the least ordinal α such that the structure is relatively Δ^0_α-categorical on a cone.*

The parameterless Scott rank of a structure is the least ordinal α such that the structure is uniformly Δ^0_α-categorical on a cone.

VII.5. The Lopez-Escobar theorem

Recall from Section V.4.1 that Mod_τ is the set of all ω-presentations of τ-structures, and it inherits its topology from $2^{\mathbb{N}}$ by representing ω-presentations by their atomic diagrams. Recall also that $Mod(\varphi)$ is the set of ω-presentations of models of φ, and that if φ is a $\Pi^{\mathsf{c}}_{\alpha}$ sentence, then $Mod(\varphi)$ is a Π^0_α class.

We say that $\mathbb{K} \subseteq Mod_\tau$ is *closed under isomorphisms* if whenever $\mathcal{A} \cong \mathcal{B}$ and $\mathcal{A} \in \mathbb{K}$, we have $\mathcal{B} \in \mathbb{K}$. It is clear that, for any sentence φ, $Mod(\varphi)$ is closed under isomorphisms.

Theorem VII.25. *If a set of ω-presentations $\mathbb{K} \subseteq Mod_\tau$ is Π^0_α and closed under isomorphisms, then it is the set of models of some $\Pi^{\mathtt{c}}_\alpha$ sentence ψ.*

Lopez-Escobar [**LE65**] proved that every Borel class that is closed under isomorphisms is $\mathcal{L}_{\omega_1,\omega}$-axiomatizable. Vaught [**Vau75**] then improved this result to show that if the class is $\boldsymbol{\Pi}^0_\alpha$, then it is $\Pi^{\mathtt{in}}_\alpha$-axiomatizable using the method of Vaught's transforms. Vander-Boom [**VB07**] proved the lightface version we present here.

Proof. Let $\varphi(\dot{\mathcal{G}})$ be an $\mathbb{N}$-Π^0_α-sentence that defines $\mathbb{K}$ when viewed as a subset of $2^{\mathbb{N}}$. That is, $\varphi(\mathcal{K})$ holds if and only if $\mathcal{K} \in \mathbb{K}$. Consider the τ-$\Pi^{\mathtt{c}}_\alpha$-sentence $\mathit{Force}_\varphi(\langle\rangle)$, which says that the empty tuple forces that $\dot{\mathcal{G}}$ satisfies φ (see Definition VII.8). Recall that the sentence Force_φ is obtained uniformly from φ and is independent of the structure we are forcing with. That is, for all τ-structures $\mathcal{A}$,

$$\mathcal{A} \models \mathit{Force}_\varphi \iff \langle\rangle \Vdash_{\mathcal{A}} \varphi.$$

We claim that $\mathbb{K} = \mathit{Mod}(\mathit{Force}_\varphi)$: If $\mathcal{A} \in \mathbb{K}$, then all generic copies of $\mathcal{A}$ are in $\mathbb{K}$, and hence $\langle\rangle \Vdash_{\mathcal{A}} \varphi$ and $\mathcal{A} \in \mathit{Mod}(\mathit{Force}_\varphi)$. Similarly, if $\mathcal{A} \notin \mathbb{K}$, then all generic copies of $\mathcal{A}$ are outside $\mathbb{K}$, and hence $\langle\rangle \not\Vdash_{\mathcal{A}} \varphi$ and $\mathcal{A} \notin \mathit{Mod}(\mathit{Force}_\varphi)$. □

The theorem is also true for Σ^0_α classes and $\Sigma^{\mathtt{c}}_\alpha$ formulas by taking complements.

Exercise VII.26. Prove that if $\mathbb{K}$ is closed under isomorphisms and is $\Sigma^0_\alpha \wedge \Pi^0_\alpha$ (i.e., definable by an $\mathbb{N}$-formula of the form $\varphi_1 \wedge \varphi_2$, where φ_1 is Σ^0_α and φ_2 is Π^0_α), then $\mathbb{K} = \mathit{Mod}(\psi)$ for some τ-$(\Sigma^0_\alpha \wedge \Pi^0_\alpha)$ formula.

Be aware that the sets defined by formulas φ_1 and φ_2 individually may not be closed under isomorphisms.

VII.6. Lopez-Escobar's interpolation theorem

Lopez-Escobar proved a version for infinitary logic of the well-known Craig's interpolation theorem.

Theorem VII.27. *Consider two vocabularies, τ_1 and τ_2, and let $\tau = \tau_1 \cap \tau_2$. Let ψ_1 and ψ_2 be τ_1-$\mathcal{L}_{\mathtt{c},\omega}$- and τ_2-$\mathcal{L}_{\mathtt{c},\omega}$-sentences such that $\psi_1 \Rightarrow \psi_2$ (i.e., all models of ψ_1 satisfy ψ_2). Then, there is a τ-$\mathcal{L}_{\mathtt{c},\omega}$-sentence ψ such that $\psi_1 \Rightarrow \psi$ and $\psi \Rightarrow \psi_2$.*

PROOF. Let $\mathbb{K}$ be the set of ω-presentations of τ-structures which can be expanded to *some* τ_1-structure that is a model of ψ_1.[†] Let $\mathbb{S}$ be the set of ω-presentations of τ-structures *all* of whose expansions to τ_2-structures are models of ψ_2. We claim that, since $\psi_1 \Rightarrow \psi_2$, we get that $\mathbb{K} \subseteq \mathbb{S}$. To see this, take $\mathcal{A} \in \mathbb{K}$, and to prove that $\mathcal{A} \in \mathbb{S}$, consider a τ_2-expansion $\mathcal{C}$ of $\mathcal{A}$. Since $\mathcal{A}$ is in $\mathbb{K}$, there is some τ_1-expansion of $\mathcal{B}$ of $\mathcal{A}$ that satisfies ψ_1. Let $\mathcal{D}$ be the $\tau_1 \cup \tau_2$-expansion of $\mathcal{B}$ and $\mathcal{C}$. Since $\psi_1 \Rightarrow \psi_2$ and $\mathcal{D} \models \psi_1$, we must have $\mathcal{D} \models \psi_2$, and hence $\mathcal{C} \models \psi_2$. This proves that $\mathcal{A} \in \mathbb{S}$ and our claim that $\mathbb{K} \subseteq \mathbb{S}$. Notice also that $\mathbb{K}$ is Σ^1_1 and $\mathbb{S}$ is Π^1_1. By the Σ^1_1-separation theorem (Theorem IV.17), there is a Δ^1_1 set $\mathfrak{S} \subseteq Mod_\tau$ separating $\mathbb{K}$ and $\mathbb{S}$, that is, with $\mathbb{K} \subseteq \mathfrak{S} \subseteq \mathbb{S}$. We showed in Remark V.28 that all Δ^1_1 sets are $\mathbb{N}$-$\mathcal{L}_{\mathsf{c},\omega}$-definable. Let $\varphi(\dot{\mathcal{G}})$ be an $\mathbb{N}$-$\mathcal{L}_{\mathsf{c},\omega}$-formula defining $\mathfrak{S}$. Assume that φ is a Π-formula — if not, add a dummy $\bigwedge\!\!\!\bigwedge$ up front.

Let ψ be the $\mathcal{L}_{\mathsf{c},\omega}$-$\tau$-sentence $Force_\varphi(\langle\rangle)$, which says that the empty tuple forces $\dot{\mathcal{G}}$ to be in $\mathfrak{S}$. We claim that $\psi_1 \Rightarrow \psi$ and $\neg\psi_2 \Rightarrow \neg\psi$. If a τ_1-structure satisfies ψ_1, its τ-reduct $\mathcal{K}$ must be in $\mathbb{K}$. Then all copies $\mathcal{G}$ of $\mathcal{K}$ satisfy $\varphi(\mathcal{G})$, and hence $\langle\rangle \Vdash_{\mathcal{K}} \varphi(\dot{\mathcal{G}})$. It follows that $\psi_1 \Rightarrow \psi$. If a τ_2-structure satisfies $\neg\psi_2$, its τ-reduct $\mathcal{K}$ must be outside $\mathbb{S}$. Then all copies $\mathcal{G}$ of $\mathcal{K}$ satisfy $\neg\varphi(\mathcal{G})$, and hence $\langle\rangle \nVdash_{\mathcal{K}} \varphi(\dot{\mathcal{G}})$. It follows that $\psi \Rightarrow \psi_2$. □

VII.7. The boldface pairs-of-structures theorem

In this section, we give a new characterization of the back-and-forth relations in terms of how difficult it is to distinguish copies of structures.

DEFINITION VII.28. Let $\mathfrak{R}$ and $\mathfrak{S}$ be disjoint subsets of $2^{\mathbb{N}}$. We say that *distinguishing* elements of $\mathfrak{R}$ from ones of $\mathfrak{S}$ is $\boldsymbol{\Sigma}^0_\alpha$*-hard* if, for every $\boldsymbol{\Sigma}^0_\alpha$ subset $\mathfrak{K} \subseteq 2^{\mathbb{N}}$, there is a continuous function $\Gamma\colon 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ such that, for all $X \in 2^{\mathbb{N}}$,

$$\Gamma(X) \in \begin{cases} \mathfrak{R} & \text{if } X \in \mathfrak{K} \\ \mathfrak{S} & \text{if } X \notin \mathfrak{K}. \end{cases}$$

We call such a Γ a *Wadge reduction* from $\mathfrak{K}$ to $\mathfrak{R}, \mathfrak{S}$ (see Definition I.22). In the lightface case, i.e., to define Σ^0_α*-hardness*, we require Γ to be computable.

[†] A τ_1-structure $\mathcal{B}$ is an *expansion* of a τ-structure $\mathcal{A}$ if $\mathcal{A}$ and $\mathcal{B}$ have the same domain and coincide on the interpretations of all the τ-symbols. In this case, we also say that $\mathcal{A}$ is a *reduct* of $\mathcal{B}$.

If we are given τ-structures $\mathcal{A}$ and $\mathcal{B}$, we say that *distinguishing $\mathcal{A}$ from $\mathcal{B}$ is $\mathbf{\Sigma}^0_\xi$-hard* if distinguishing copies of $\mathcal{A}$ from copies of $\mathcal{B}$ is $\mathbf{\Sigma}^0_\xi$-hard. Recall that the copies of $\mathcal{A}$ and $\mathcal{B}$ live in the space Mod_τ which is essentially the same as $2^{\mathbb{N}}$ (Section V.4.1).

Observation VII.29. If distinguishing $\mathcal{A}$ from $\mathcal{B}$ is $\mathbf{\Sigma}^0_\xi$-hard, then $\mathcal{A} \leq_\xi \mathcal{B}$. To see this, suppose that $\mathcal{A} \not\leq_\xi \mathcal{B}$, as witnessed by some $\Pi^{\mathtt{in}}_\xi$ sentence φ that is true of $\mathcal{A}$, false of $\mathcal{B}$. To show that distinguishing $\mathcal{A}$ from $\mathcal{B}$ is not $\mathbf{\Sigma}^0_\xi$-hard, consider a Wadge reduction Γ from a set $\mathfrak{K} \subseteq 2^{\mathbb{N}}$ to copies of $\mathcal{A}$ and $\mathcal{B}$. Then, we would have a $\mathbf{\Pi}^0_\xi$ way of deciding membership in $\mathfrak{K}$, namely $X \in \mathfrak{K} \iff \Gamma(X) \models \varphi$. So, $\mathfrak{K}$ could not be $\mathbf{\Sigma}^0_\xi$-hard.[‡]

Theorem VII.30. *Let $\mathcal{A}$ and $\mathcal{B}$ be τ-structures. Then $\mathcal{A} \leq_\xi \mathcal{B}$ if and only if distinguishing $\mathcal{A}$ from $\mathcal{B}$ is (boldface) $\mathbf{\Sigma}^0_\xi$-hard.*

Proof. The ($\Leftarrow$) direction was proved in the observation above. Let us concentrate on the ($\Rightarrow$) direction. We will show that either $\mathcal{A} \not\leq_\xi \mathcal{B}$ or distinguishing $\mathcal{A}$ from $\mathcal{B}$ is $\mathbf{\Sigma}^0_\xi$-hard.

The proof uses Borel determinacy (Martin [**Mar75**]). For the reader not familiar with it, we will explain how it works along the way, but we will not prove it.

Let $\mathfrak{K}$ be a Σ^0_ξ-complete subset of $2^{\mathbb{N}}$. Consider a game played by two players, I and II, who alternatively play a binary bit for infinitely many turns as in the diagram below.

Player I	x_0		x_1		x_2		$\cdots$	$X \in 2^{\mathbb{N}}$
Player II		y_0		y_1		$\cdots$		$Y \in 2^{\mathbb{N}}$

Player I plays $x_i \in \{0,1\}$ on her ith move, and II plays $y_i \in \{0,1\}$. At the end of the game, we end up with two reals $X, Y \in 2^{\mathbb{N}}$. Player I *wins* the game if one of the following holds

- X is the atomic diagram of a copy of $\mathcal{A}$ and $Y \in \mathfrak{K}$,
- X is the atomic diagram of a copy of $\mathcal{B}$ and $Y \notin \mathfrak{K}$.

[‡] That a set $\mathfrak{R}$ that is $\mathbf{\Pi}^0_\xi$ cannot be Σ^0_ξ-hard can be proved by a standard diagonalization argument: Let $\mathfrak{S}$ be a lightface Π^0_ξ set and X_0 a real such that $\mathfrak{R} = \{X : \langle X, X_0\rangle \in \mathfrak{S}\}$. The set $\mathfrak{K} = \{\langle e, X\rangle : \langle \Gamma_e(e,X), X\rangle \notin \mathfrak{S}\}$ (where Γ_e is the eth Turing functional) is Σ^0_ξ and, if $\mathfrak{R}$ were Σ^0_ξ-hard, there would be some effective Wadge reduction Γ_{e_0} so that $\langle e, X\rangle \in \mathfrak{K} \iff \Gamma_{e_0}(e,X) \in \mathfrak{R}$. We obtain a contradiction as follows:

$$\langle e_0, X_0\rangle \in \mathfrak{K} \overset{\text{Def. }\Gamma_{e_0}}{\iff} \Gamma_{e_0}(e_0, X_0) \in \mathfrak{R} \overset{\text{Def. }\mathfrak{S}}{\iff} \langle \Gamma_{e_0}(e_0, X_0), X_0\rangle \in \mathfrak{S} \overset{\text{Def. }\mathfrak{K}}{\iff} \langle e_0, X_0\rangle \notin \mathfrak{K}.$$

Player II wins otherwise. In particular, if X is not the diagram of a copy of either $\mathcal{A}$ or $\mathcal{B}$, then II wins.

Notice that deciding if X is a copy of $\mathcal{A}$ or $\mathcal{B}$ is a Borel property, as all one has to do is check whether it satisfies the Scott sentence of either $\mathcal{A}$ or $\mathcal{B}$. Thus, deciding who wins the game is a Borel property of X and Y. A *strategy* for a player is a function $\sigma\colon 2^{<\mathbb{N}} \to 2$ that tells the player what to play next given the moves made so far by the opponent. For instance, if σ is a strategy for player II, then X, Y are obtained following σ if and only if $y_i = \sigma(x_0, ..., x_i)$ for all i. Thus, a strategy induces a continuous function $\sigma\colon 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ which maps X to Y, i.e, $Y = \sigma(X)$. Analogously, if π is a strategy for player I, we obtain a continuous function $\pi\colon 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ so that $X = \pi(Y)$. A *winning strategy* is one that, if the player follows it, they will win every time, for any sequence of moves played by the opponent. Martin's theorem of *Borel determinacy* tells us that one of the two players must have a winning strategy.

Suppose first that player I has a winning strategy. We then have a continuous function $\pi\colon 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ such that if $Y \in \mathfrak{K}$, then $\pi(Y)$ is a copy of $\mathcal{A}$, and if $Y \in 2^{\mathbb{N}} \smallsetminus \mathfrak{K}$, then $\pi(Y)$ is a copy of $\mathcal{B}$. We thus have a Wadge reduction from $\mathfrak{K}$ (which is Σ^0_ξ-complete) to copies of $\mathcal{A}$ and $\mathcal{B}$, showing that distinguishing of $\mathcal{A}$ from $\mathcal{B}$ is $\boldsymbol{\Sigma}^0_\xi$-hard.

Suppose now that player II has a winning strategy. We then have a continuous function $\sigma\colon 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ such that if X is the diagram of a copy of $\mathcal{A}$, then $\sigma(X) \notin \mathfrak{K}$, and if X is the diagram of a copy of $\mathcal{B}$, then $\sigma(X) \in \mathfrak{K}$. Let $\mathfrak{N} \subseteq 2^{\mathbb{N}}$ be the pre-image of $\mathfrak{K}$ under σ. Then $\mathfrak{N}$ is $\boldsymbol{\Sigma}^0_\xi$. Actually, $\mathfrak{N}$ is Σ^0_ξ relative to σ. Note that all copies of $\mathcal{B}$ are in $\mathfrak{N}$ and no copy of $\mathcal{A}$ is. Let $\varphi(D)$ be an $\mathbb{N}$-$\Pi^{\mathtt{c}\sigma}_\xi$ formula that defines $2^{\mathbb{N}} \smallsetminus \mathfrak{N}$. Consider forcing relative to σ.[§] Since all generic copies of $\mathcal{A}$ satisfy φ and no generic copy of $\mathcal{B}$ does, we have that $\mathit{Force}_\varphi(\langle\rangle)$ is a τ-$\Pi^{\mathtt{c}\sigma}_\xi$ sentence that is true of $\mathcal{A}$ and false of $\mathcal{B}$. It follows that $\mathcal{A} \not\leq_\xi \mathcal{B}$. □

VII.8. Computable functors and interpretability

The following application of forcing has to do with reducibilities between structures. One of the most commonly used reductions between structures is the Medvedev reduction. A *Medvedev reduction* from a structure $\mathcal{A}$ to a structure $\mathcal{B}$ is a computable operator Ψ which maps (diagrams of) copies of $\mathcal{A}$ into (diagrams of) copies of $\mathcal{B}$. This is a purely computability theoretic notion that, unfortunately, does not

[§] By forcing relative to σ we mean that generic enumerations now need to decide all infinitary formulas that are computable relative to σ. All the theorems we proved about $\mathcal{L}_{\mathtt{c},\omega}$-generics work the same way, but now relativized to σ.

have a structural counterpart. In [**Part 1**, Section VI.3], we considered a strengthening of this notion that we called a computable functor, where we require the Medvedev reduction to preserve isomorphisms in a computable and functorial way — see Definition VII.31 below. We then claimed that this notion is equivalent to that of effective interpretability, which is a purely syntactical notion of reduction, similar to the model theoretic notion of interpretability — see Definition VII.33 below. It was not hard to prove that effective interpretations induce computable functors. But we left the proof of the converse pending until now, as it needs the technique of product forcing.

In this chapter, we will consider the notions of Δ^0_α-functors and $\Delta^{\mathsf{c}}_\alpha$-interpretations. Our work in [**Part 1**, Section VI.3] will follow from the case $\alpha = 1$, which is already quite interesting. Knowledge of [**Part 1**, Section VI.3] is not necessary to read this section.

Definition VII.31 ([**MPSS18**] [**HTMM**]). Given structures $\mathcal{A}$ and $\mathcal{B}$, a *functor* from $\mathcal{B}$ to $\mathcal{A}$ consists of a pair of operators, $\Psi\colon 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ and $\Psi\colon 2^{\mathbb{N}} \times 2^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$,[¶] such that:

(1) For every copy $\hat{\mathcal{B}}$ of $\mathcal{B}$, $\Psi(\hat{\mathcal{B}}) = \hat{\mathcal{A}}$ for some copy $\hat{\mathcal{A}}$ of $\mathcal{A}$.[‖]

(2) For every isomorphism $f\colon \hat{\mathcal{B}} \to \widetilde{\mathcal{B}}$ between two copies of $\mathcal{B}$, $\Psi^{\hat{\mathcal{B}},\widetilde{\mathcal{B}}}(f)$ is an isomorphism from $\Psi(\hat{\mathcal{B}})$ to $\Psi(\widetilde{\mathcal{B}})$.[**]

We also require that the operator Ψ preserve the identity and composition of isomorphisms:

(3) $\Psi^{\hat{\mathcal{B}},\hat{\mathcal{B}}}(id) = id$ for every copy $\hat{\mathcal{B}}$ of $\mathcal{B}$.

(4) $\Psi^{\mathcal{B}_0,\mathcal{B}_2}(g \circ f) = \Psi^{\mathcal{B}_1,\mathcal{B}_2}(g) \circ \Psi^{\mathcal{B}_0,\mathcal{B}_1}(f)$ for copies $\mathcal{B}_0$, $\mathcal{B}_1$, and $\mathcal{B}_2$ of $\mathcal{B}$ and isomorphisms $f\colon \mathcal{B}_0 \to \mathcal{B}_1$ and $g\colon \mathcal{B}_1 \to \mathcal{B}_2$.

Ψ is a functor in the sense of category theory. It is a functor from the category of ω-presentations of $\mathcal{B}$, where morphisms are the isomorphisms between the copies of $\mathcal{B}$, to the category of ω-presentations of $\mathcal{A}$.

Example VII.32. Let $\mathcal{B}$ be a linear ordering, and let $\mathcal{A}$ be obtained by collapsing the elements of $\mathcal{B}$ which are finitely apart.[††] One can build $\Psi(\mathcal{B})$ by choosing the $<_{\mathbb{N}}$-least element of each finitely-apart equivalence class in the given ω-presentation $\mathcal{B}$. That is, let

[¶] Both operators have the same name, but since they have different domains, it will be clear which one we are applying when.

[‖] Here and throughout this section, we write $\Psi(\hat{\mathcal{B}}) = \hat{\mathcal{A}}$ as shorthand for $\Psi(D(\hat{\mathcal{B}})) = D(\hat{\mathcal{A}})$.

[**] Here $\Psi^{\hat{\mathcal{B}},\widetilde{\mathcal{B}}}(f)$ is shorthand for $\Psi(D(\hat{\mathcal{B}}), D(\widetilde{\mathcal{B}}), f)$.

[††] Two elements in a linear order are *finitely apart* if there are only finitely many elements between them.

$\Psi(B; \leq_B) = (A; \leq_B)$, where

$$A = \{b \in B : (\forall n <_{\mathbb{N}} b)\ n \not\sim_{\mathcal{B}} b\}.$$

Here, $n \sim_{\mathcal{B}} b$ if they are finitely apart in $\mathcal{B}$, and $<_{\mathbb{N}}$ represents the ordering of the natural numbers.

Then, if we have an isomorphism $f \colon \hat{\mathcal{B}} \to \widetilde{\mathcal{B}}$, and $a \in \Psi(\hat{\mathcal{B}})$, we let $\Psi^{\hat{\mathcal{B}},\widetilde{\mathcal{B}}}(f)(a)$ be the unique element of $\tilde{A}$ that is finitely apart from $f(a)$ in $\widetilde{\mathcal{B}}$.

Determining if two elements are finitely apart requires two Turing jumps, making this a Δ^0_3 functor.

We will prove that having a Δ^0_α functor is equivalent to having a $\Delta^{\mathtt{c}}_\alpha$-interpretation. Informally, a structure $\mathcal{A}$ is $\Delta^{\mathtt{c}}_\alpha$-*interpretable* in a structure $\mathcal{B}$ if there is an interpretation of $\mathcal{A}$ in $\mathcal{B}$ as in model theory, but where the domain of the interpretation is allowed to be a subset of $\mathbb{N} \times B^{<\mathbb{N}}$ instead of just B^n, and where all sets in the interpretation are required to be $\Delta^{\mathtt{c}}_\alpha$-definable instead of elementary first-order definable.

Definition VII.33. Let $\mathcal{A}$ be a τ-structure and $\mathcal{B}$ be any structure. Let us assume that τ is a relational vocabulary, $\tau = \{P_i : i \in I\}$, where P_i has arity $a(i)$. So $\mathcal{A} = (A; P_0^{\mathcal{A}}, P_1^{\mathcal{A}}, ...)$ and $P_i^{\mathcal{A}} \subseteq A^{a(i)}$.

We say that $\mathcal{A}$ is $\Delta^{\mathtt{c}}_\alpha$-*interpretable* in $\mathcal{B}$ if, in $\mathcal{B}$, there are $\Delta^{\mathtt{c}}_\alpha$-definable relations $A^{\mathcal{B}}$, $\sim^{\mathcal{B}}$, and $\{R_i^{\mathcal{B}} : i \in I\}$ such that

- $A^{\mathcal{B}} \subseteq \mathbb{N} \times B^{<\mathbb{N}}$ (the domain of the interpretation of $\mathcal{A}$ in $\mathcal{B}$),
- $\sim^{\mathcal{B}} \subseteq A^{\mathcal{B}} \times A^{\mathcal{B}}$ is an equivalence relation on $A^{\mathcal{B}}$ (interpreting equality), and
- each $R_i^{\mathcal{B}} \subseteq (A^{\mathcal{B}})^{a(i)}$ is closed under the equivalence $\sim^{\mathcal{B}}$ (interpreting the relations P_i);

and there is a function $\mathfrak{F} \colon A^{\mathcal{B}} \to A$ which induces an isomorphism:

$$(A^{\mathcal{B}}/\sim^{\mathcal{B}}; R_0^{\mathcal{B}}, R_1^{\mathcal{B}}, ...) \cong (A; P_0^{\mathcal{A}}, P_1^{\mathcal{A}}, ...).$$

Let us clarify this last line. The function $\mathfrak{F} \colon A^{\mathcal{B}} \to A$ must be an onto map such that $\mathfrak{F}(a) = \mathfrak{F}(b) \iff a \sim^{\mathcal{B}} b$ and $\mathfrak{F}(\bar{a}) \in P_i^{\mathcal{A}} \iff \bar{a} \in R_i^{\mathcal{B}}$. Notice that there is no restriction on the complexity or definability of $\mathfrak{F}$. We use $\mathcal{A}^{\mathcal{B}}$ to denote the structure $(A^{\mathcal{B}}/\sim^{\mathcal{B}}; R_0^{\mathcal{B}}, R_1^{\mathcal{B}}, ...)$.

Example VII.34. Let us consider Example VII.32, where a linear ordering $\mathcal{A}$ is obtained from a linear ordering $\mathcal{B}$ by collapsing elements that are finitely apart. This can be easily seen as a $\Delta^{\mathtt{c}}_3$-interpretation: Let $A^{\mathcal{B}} = B$, let $\sim^{\mathcal{B}}$ be the equivalence relation of being finitely apart in $\mathcal{B}$, and let $\leq^{\mathcal{A}^{\mathcal{B}}}$ be the ordering induced on the equivalence classes. Notice that both $\sim$ and $\leq^{\mathcal{A}^{\mathcal{B}}}$ are $\Sigma^{\mathtt{c}}_2$, and in particular $\Delta^{\mathtt{c}}_3$.

Lemma VII.35. *A $\Delta^{\mathtt{c}}_\alpha$-interpretation of $\mathcal{A}$ in $\mathcal{B}$ induces a Δ^0_α-functor from $\mathcal{B}$ to $\mathcal{A}$.*

Proof. Since $A^{\mathcal{B}}$, $\sim^{\mathcal{B}}$, and $\{R_i^{\mathcal{B}} : i \in I\}$ are $\Delta^{\mathtt{c}}_\alpha$-definable in $\mathcal{B}$, we have a Δ^0_α operator that produces those subsets of $\mathbb{N} \times \mathbb{N}^{<\mathbb{N}}$ within any copy $\hat{\mathcal{B}}$ of $\mathcal{B}$, using $D(\hat{\mathcal{B}})$ as an oracle. Thus, we have a Δ^0_α operator Φ that, given $\hat{\mathcal{B}} \cong \mathcal{B}$, outputs $D(\mathcal{A}^{\hat{\mathcal{B}}})$, the atomic diagram of the congruence ($\subseteq \mathbb{N} \times \mathbb{N}^{<\mathbb{N}}$)-presentation $\mathcal{A}^{\hat{\mathcal{B}}}$ of $\mathcal{A}$ with domain $A^{\hat{\mathcal{B}}} \subseteq \mathbb{N} \times \hat{B}^{<\mathbb{N}} = \mathbb{N} \times \mathbb{N}^{<\mathbb{N}}$. Φ acts on isomorphisms in a natural way: Every permutation g of $\mathbb{N}$ induces a permutation $\breve{g}$ of $\mathbb{N} \times \mathbb{N}^{<\mathbb{N}}$ given by $\breve{g}(\langle n, \langle k_0, ..., k_\ell\rangle\rangle) = \langle n, \langle g(k_0), ..., g(k_\ell)\rangle\rangle$. Then, if f is an isomorphism between $\hat{\mathcal{B}}$ and $\widetilde{\mathcal{B}}$, we let $\Phi^{\hat{\mathcal{B}},\widetilde{\mathcal{B}}}(f) = \breve{f} \restriction \mathcal{A}^{\hat{\mathcal{B}}}$. This operator Φ is a Δ^0_α functor, though the reader may complain that it does not output injective ω-presentations. Fixing a bijection between $\mathbb{N}$ and $\mathbb{N} \times \mathbb{N}^{<\mathbb{N}}$ and using Lemma [**Part 1**, 1 I.11], we get a computable operator Υ transforming congruence ($\subseteq \mathbb{N} \times \mathbb{N}^{<\mathbb{N}}$)-presentations into injective ω-presentations. It is not hard to see that Υ can be easily made into a Δ^0_α-functor. Composing these Δ^0_α functors we get the Δ^0_α functor $\Upsilon \circ \Phi$ we wanted. □

The following theorem shows the converse.

Theorem VII.36 (Harrison-Trainor, Miller, Montalbán [**HTMM**]). *Let $\mathcal{A}$ and $\mathcal{B}$ be countable structures. The following are equivalent:*

(1) *$\mathcal{A}$ is $\Delta^{\mathtt{c}}_\alpha$-interpretable in $\mathcal{B}$.*
(2) *There is a Δ^0_α-functor from $\mathcal{B}$ to $\mathcal{A}$.*

Furthermore, given a Δ^0_α-functor, the $\Delta^{\mathtt{c}}_\alpha$-interpretation we get in the proof of the theorem induces the original functor up to Δ^0_α-isomorphism of functors. See [**HTMM**] for more details on isomorphisms of functors.

We have already proved that (1) implies (2). The rest of this section is dedicated to prove the converse.

In [**HTMM**], they also consider the notions of invertible functors and bi-interpretability and prove an equivalence between these notions. A very interesting new example was found by Marker and R. Miller [**MM17**], who use Δ^0_2 functors to build effective bi-interpretations between graphs and the jumps of differentially closed fields of characteristic zero.

VII.8.1. Product forcing. The objective of product forcing is to build multiple generic enumerations of a structure so that they are generic relative to each other. Fix a structure $\mathcal{B}$. Given $\ell \in \omega$, we let our forcing conditions be tuples $\langle \bar{p}_1, \dots, \bar{p}_\ell \rangle$ where each $\bar{p}_i$ is in $B^\star$. We say that $\langle \bar{q}_1, \dots, \bar{q}_\ell \rangle$ *extends* $\langle \bar{p}_1, \dots, \bar{p}_\ell \rangle$ if $\bar{p}_i \subseteq \bar{q}_i$ for all $i \leq \ell$, and we say that $R \subseteq \mathcal{B}^{\star\ell}$ is *dense* if every tuple in $\mathcal{B}^{\star\ell}$ has an extension in R.

Definition VII.37. We say that ℓ injective enumerations $g_1, ..., g_\ell$ of $\mathcal{B}$ are *mutually $\mathcal{L}_{\mathsf{c},\omega}$-generic* if they meet every dense $\mathcal{L}_{\mathsf{c},\omega}$-definable subset of $\mathcal{B}^{\star\ell}$, that is, if for every dense $\mathcal{L}_{\mathsf{c},\omega}$-definable $R \subseteq \mathcal{B}^{\star\ell}$, there are initial segments $\bar{p}_1, \dots, \bar{p}_\ell$ of $g_1, ..., g_\ell$ with $\langle \bar{p}_1, \dots, \bar{p}_\ell \rangle \in R$.

For our forcing language, we consider formulas of arithmetic which contain second-order variables for unary relations $\dot{\mathcal{G}}_i$, for $i \leq \ell$, that represent the atomic diagrams of the generic presentations $\mathcal{G}_i = g_i^{-1}(\mathcal{B})$. We also add second-order variables for unary functions $\dot{h}_{i,j}$, for $i, j \leq \ell$, that represent the induced isomorphisms between the generic presentations, namely

$$h_{i,j} = g_j^{-1} \circ g_i \colon \mathcal{G}_i \to \mathcal{G}_j.$$

We will call these formulas $\mathbb{N}_\ell$-formulas.[‡‡]

The definition of the forcing relation is as expected. Let $\vec{\mathbf{p}} = \langle \bar{p}_1, \dots, \bar{p}_\ell \rangle \in B^{\star\ell}$. The cases for $\top$, $\bot$, $\bigvee\!\!\!\bigvee$ and $\bigwedge\!\!\!\bigwedge$ are exactly as in Definition VII.7. For the other cases:

- $\vec{\mathbf{p}} \Vdash_{\mathcal{B}^\ell} \dot{\mathcal{G}}_i(\mathbf{n}) \iff D_{\mathcal{B}}(\bar{p}_i)(n){\downarrow} = 1$.
- $\vec{\mathbf{p}} \Vdash_{\mathcal{B}^\ell} \neg\dot{\mathcal{G}}_i(\mathbf{n}) \iff D_{\mathcal{B}}(\bar{p}_i)(n){\downarrow} = 0$.
- $\vec{\mathbf{p}} \Vdash_{\mathcal{B}^\ell} \dot{h}_{i,j}(\mathbf{n}) = \mathbf{m} \iff \bar{p}_i(n){\downarrow} = \bar{p}_j(m){\downarrow}$.
- $\vec{\mathbf{p}} \Vdash_{\mathcal{B}^\ell} \dot{h}_{i,j}(\mathbf{n}) \neq \mathbf{m} \iff \bar{p}_i(n){\downarrow} \neq \bar{p}_j(m){\downarrow}$.

The rest of the development follows all the steps of section VII.2. There are two ways to continue. One is to go through all the definitions, lemmas, and theorems of section VII.2 and adapt them to this setting. The other is to observe that the product forcing $\mathcal{B}^{\star\ell}$ is equivalent to forcing with the structure $(\ell \times \mathcal{B}, \{R_i : i \leq \ell\}, \{h_{i,j} : i, j \leq \ell\})$, where $\ell \times \mathcal{B}$ represents the structure that consists of ℓ disjoint copies of the structure $\mathcal{B}$, R_i is a unary relation identifying the ith copy, and $h_{i,j}$ is the identity function between the ith and jth copies. Both ways are straightforward, so we leave the details to the reader. The first approach is spelled out in [**HTMM**].

Lemma VII.38. *For every $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration g_1, there exists an enumeration g_2 that is mutually $\mathcal{L}_{\mathsf{c},\omega}$-generic with g_1.*

[‡‡] Let us clarify that $\dot{\mathcal{G}}_i$ and $\dot{h}_{i,j}$ are just symbols (usually called *names*) that, only after we have our mutual generic enumerations, will be interpreted as $g_i^{-1}(\mathcal{B})$ and $g_j^{-1} \circ g_i$. Let us also emphasize that these are infinitary first-order formulas with relation and function symbols for $\dot{\mathcal{G}}_i$ and $\dot{h}_{i,j}$ — there is no second-order quantification.

Proof. Given $R \subseteq \mathcal{B}^{\star 2}$, let

$$\pi^1_{\bar{r}}(R) = \{\bar{p} \in \mathcal{B}^\star : (\exists \bar{q} \supseteq \bar{r})\ \langle \bar{p}, \bar{q}\rangle \in R\}$$

and

$$\pi^2_g(R) = \{\bar{q} \in \mathcal{B}^\star : (\exists \bar{p} \subset g)\ \langle \bar{p}, \bar{q}\rangle \in R\}.$$

Suppose that $R \subseteq \mathcal{B}^{\star 2}$ is dense and $\mathcal{L}_{\mathsf{c},\omega}$-definable. We claim that $\pi^2_{g_1}(R)$ is also dense: Fix $\bar{r} \in B^\star$. Observe that $\pi^1_{\bar{r}}(R)$ is dense for all tuples $\bar{r}$.[*] Since $\pi^1_{\bar{r}}(R)$ is $\mathcal{L}_{\mathsf{c},\omega}$ definable over $\bar{r}$, we have that g_1 meets $\pi^1_{\bar{r}}(R)$ at some $\bar{p} \subset g_1$. It follows that there is $\bar{q} \supseteq \bar{r}$, such that $\langle \bar{p}, \bar{q}\rangle \in R$. We then have $\bar{q} \in \pi^2_{g_1}(R)$, proving that $\pi^2_{g_1}(R)$ is dense.

As in the proof of the existence of $\mathcal{L}_{\mathsf{c},\omega}$-generics (Lemma VII.2), build g_2 so that it meets the sets $\pi^2_{g_1(R)}$ for all $R \subseteq \mathcal{B}^{\star 2}$ that are dense and $\mathcal{L}_{\mathsf{c},\omega}$-definable.

Notice that g_2 meets $\pi^2_{g_1}(R)$ if and only if $\langle g_1, g_2\rangle$ meets R. So, we have that g_1 and g_2 are mutually generic. □

Exercise VII.39. For every mutually $\mathcal{L}_{\mathsf{c},\omega}$-generic enumerations $g_1, ..., g_{\ell-1}$ and every $\bar{p} \in A^\star$, there exists an enumeration $g_\ell \supset \bar{p}$ that is mutually $\mathcal{L}_{\mathsf{c},\omega}$-generic with $g_1, ..., g_{\ell-1}$.

Corollary VII.40. *Let φ be an $\mathbb{N}_{\ell-1}$-Π-formula and consider $\langle \bar{p}_1, ..., \bar{p}_\ell\rangle \in B^{\star\ell}$. Then*

$$\langle \bar{p}_1, ..., \bar{p}_\ell\rangle \Vdash_{\mathcal{B}^\ell} \varphi \qquad \Longleftrightarrow \qquad \langle \bar{p}_1, ..., \bar{p}_{\ell-1}\rangle \Vdash_{\mathcal{B}^{\ell-1}} \varphi.$$

Proof. Recall that the forcing relation on Π-formulas is equivalent to semantical forcing. For the ($\Leftarrow$) direction, just notice that if $g_1, ..., g_\ell$ extending $\bar{p}_1, ..., \bar{p}_\ell$ are mutually generic, then so are $g_1, ..., g_{\ell-1}$, and hence they satisfy φ. For the ($\Rightarrow$) direction, note that if $\langle \bar{p}_1, ..., \bar{p}_{\ell-1}\rangle \nVdash_{\mathcal{B}^{\ell-1}} \varphi$, then there are mutually generic enumerations $g_1, ..., g_{\ell-1}$ which do not satisfy φ. Use the previous exercise to get $g_\ell \supset \bar{p}_\ell$ so that $g_1, ..., g_\ell$ are mutually generic. Since $g_1, ..., g_\ell$ do not satisfy φ either, $\langle \bar{p}_1, ..., \bar{p}_\ell\rangle \nVdash_{\mathcal{B}^\ell} \varphi$. □

VII.8.2. Building the interpretation. Consider a Δ^0_α-functor Ψ from $\mathcal{B}$ to $\mathcal{A}$. We will use it to define a $\Delta^{\mathsf{c}}_\alpha$-interpretation of $\mathcal{A}$ within $\mathcal{B}$, as needed for the proof of Theorem VII.36.

The functor Ψ acts on ω-presentations of $\mathcal{B}$, and we want to build an interpretation that is independent of presentations. The first idea is the following: For each generic presentation $\mathcal{G}$ of $\mathcal{B}$, we obtain a different ω-presentation $\Psi(\mathcal{G})$ of $\mathcal{A}$. If we have an isomorphism f between generic presentations $\mathcal{G}$ and $\check{\mathcal{G}}$ of $\mathcal{B}$, we obtain a isomorphism $\Psi^{\mathcal{G},\check{\mathcal{G}}}(f)$

[*] $\pi^1_{\bar{r}}(R)$ is dense because for every $\bar{s}$, since R is dense, there exists a pair $\langle \bar{p}, \bar{q}\rangle$ in R extending $\langle \bar{s}, \bar{r}\rangle$. We must then have that $\bar{p}$ is a extension of $\bar{s}$ in $\pi^1_{\bar{r}}(R)$.

between the ω-presentations $\Psi(\mathcal{G})$ and $\Psi(\check{\mathcal{G}})$. Thus, we can represent the elements of $\mathcal{A}$ as pairs $\langle g, i\rangle$, where g is a generic enumeration of $\mathcal{B}$ and $i \in \mathbb{N}$ indicates that we are looking at the ith element of $\Psi(\mathcal{G})$. We can then let $\langle g, i\rangle$ be equivalent to $\langle \check{g}, j\rangle$ if the isomorphism $\Psi^{\mathcal{G},\check{\mathcal{G}}}(f)$ maps i to j. The problem is, of course, that the objects $\langle g, i\rangle$ are not finitary. Instead, we can consider initial segments $\bar{b}$ of the generics and consider pairs $\langle \bar{b}, i\rangle$, where $\bar{b}$ forces i to be the same element in all ω-presentations $\Psi(\mathcal{G})$ among all generic enumerations g extending $\bar{b}$. That is, if we want to use $\langle \bar{b}, i\rangle$ as a name for an element of $\mathcal{A}$, we want to know that if we have two generic enumerations g_1 and g_2 extending $\bar{b}$, the number i represents the same element in the ω-presentations $\Psi(\mathcal{G}_1)$ and $\Psi(\mathcal{G}_2)$. The way to ensure that is to have $\bar{b}$ force that the isomorphism $\Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2})\colon \Psi(\mathcal{G}_1) \to \Psi(\mathcal{G}_2)$ leaves i fixed. For example, in the case when $\alpha = 1$ and Ψ is computable, we want to know that $\bar{b}$ is long enough so that $D_{\mathcal{B}}(\bar{b})$ provides enough of an initial segment of $D(\dot{\mathcal{G}}_1)$ and $D(\dot{\mathcal{G}}_2)$ so that $\Psi^{D_{\mathcal{B}}(\bar{b}),D_{\mathcal{B}}(\bar{b})}(id \restriction |\bar{b}|)$ converges, and that $\Psi^{D_{\mathcal{B}}(\bar{b}),D_{\mathcal{B}}(\bar{b})}(id \restriction |\bar{b}|)(i) = i$.

Definition VII.41. We define the domain of interpretation, $A^{\mathcal{B}}$, as a subset of $B^{\star} \times \mathbb{N}$ as follows: For $\langle \bar{b}, i\rangle \in B^{\star} \times \mathbb{N}$, let

$$\langle \bar{b}, i\rangle \in A^{\mathcal{B}} \iff \langle \bar{b}, \bar{b}\rangle \Vdash_{\mathcal{B}^2} \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2})(i) = i.$$

Next, we define a relation $\sim$ on $A^{\mathcal{B}}$, which we will later prove is an equivalence relation. For $\langle \bar{b}, i\rangle, \langle \bar{c}, j\rangle \in A^{\mathcal{B}}$, let

$$\langle \bar{b}, i\rangle \sim \langle \bar{c}, j\rangle \iff \langle \bar{b}, \bar{c}\rangle \Vdash_{\mathcal{B}^2} \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2})(i) = j.$$

Lastly, we need to interpret the relation symbols. For each relation symbol P_i of arity $a(i)$ in the vocabulary of $\mathcal{A}$, we define a relation R_i on $\mathcal{A}^{\mathcal{B}}$ as follows: For $\langle \bar{b}_1, k_1\rangle, \ldots, \langle \bar{b}_{a(i)}, k_{a(i)}\rangle \in A^{\mathcal{B}}$, let

$$\langle \langle \bar{b}_1, k_1\rangle, \ldots, \langle \bar{b}_{a(i)}, k_{a(i)}\rangle\rangle \in R_i \iff (\exists \bar{c} \in B^{\star}) \bigvee_{j_1,\ldots,j_{a(i)} < |\bar{c}|}$$

$$\left(\bigwedge_{s=1}^{a(i)} \langle \bar{b}_s, k_s\rangle \sim \langle \bar{c}, j_s\rangle\right) \;\&\; \left(\bar{c} \Vdash_{\mathcal{B}} \langle j_1, \ldots, j_{a(i)}\rangle \in P_i^{\Psi(\dot{\mathcal{G}})}\right).$$

Using the definability of forcing, and observing that the formulas being forced are all $\mathbb{N}_{\ell}$-Δ^0_{α}, we get that $A^{\mathcal{B}}$ and $\sim$ can both be defined within $\mathcal{B}$ by $\Sigma^{\mathsf{c}}_{\alpha}$ formulas and by $\Pi^{\mathsf{c}}_{\alpha}$ formulas. For R_i, we get a $\Sigma^{\mathsf{c}}_{\alpha}$ formula. To get it to be Δ_{α}, add a relation for the complement of P_i.

VII.8.3. Verification. We now want to show that this is indeed an interpretation of $\mathcal{A}$ in $\mathcal{B}$. The first thing to observe before starting the verification is that since Ψ is a functor that acts on all copies of $\mathcal{B}$, all of Ψ's Π-properties are forced by the empty conditions. For instance,

$$\langle\emptyset,\emptyset,\emptyset\rangle \Vdash_{\mathcal{B}^3} \Psi^{\dot{\mathcal{G}}_2,\dot{\mathcal{G}}_3}(\dot{h}_{2,3}) \circ \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2}) = \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_3}(\dot{h}_{1,3}).$$

Lemma VII.42. *$\sim$ is an equivalence relation on $\mathcal{A}^{\mathcal{B}}$.*

Proof. Reflexivity follows from the definition of $\mathcal{A}^{\mathcal{B}}$. Symmetry holds because $\langle\emptyset,\emptyset\rangle \Vdash_{\mathcal{B}^2} \Psi^{\dot{\mathcal{G}}_2,\dot{\mathcal{G}}_1}(\dot{h}_{2,1}) = \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2})^{-1}$. Transitivity follows from the fact that $\langle\emptyset,\emptyset,\emptyset\rangle \Vdash_{\mathcal{B}^3} \Psi^{\dot{\mathcal{G}}_2,\dot{\mathcal{G}}_3}(\dot{h}_{2,3}) \circ \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2}) = \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_3}(\dot{h}_{1,3})$. □

It is also easy to see from the definition of R_i that $\sim$ is a congruence relation.

The next objective is to define a map $\mathfrak{F}\colon A^{\mathcal{B}} \to A$ which gives an isomorphism between $\mathcal{A}^{\mathcal{B}}$ and $\mathcal{A}$. Let us fix an ω-presentation of $\mathcal{B}$, and let us assume we are working with the copy of $\mathcal{A}$ given by $\mathcal{A} = \Psi(\mathcal{B})$. Given $\langle\bar{b}, i\rangle \in A^{\mathcal{B}}$, we let

$$\mathfrak{F}(\langle\bar{b}, i\rangle) = \Psi^{\mathcal{G},\mathcal{B}}(g)(i),$$

where g is an $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration of $\mathcal{B}$ extending $\bar{b}$. Notice that $\mathcal{G}$ is the pull-back of $\mathcal{B}$ through g, and that $\Psi^{\mathcal{G},\mathcal{B}}(g)$ is an isomorphism from $\Psi(\mathcal{G})$ to $\mathcal{A}$. We need to verify that this definition is independent of g. Observe that, from the definition of $A^{\mathcal{B}}$, we get that if $\langle\bar{b}, i\rangle \in A^{\mathcal{B}}$, then for all $\mathcal{L}_{\mathsf{c},\omega}$-generic enumerations g_1 and g_2 extending $\bar{b}$, $\Psi^{\mathcal{G}_1,\mathcal{G}_2}(h_{1,2})(i) = i$. It then follows that

$$\Psi^{\mathcal{G}_1,\mathcal{B}}(g_1)(i) = \Psi^{\mathcal{G}_2,\mathcal{B}}(g_2) \circ \Psi^{\mathcal{G}_1,\mathcal{G}_2}(h_{1,2})(i) = \Psi^{\mathcal{G}_2,\mathcal{B}}(g_2)(i).$$

Second, we need to show that $\mathfrak{F}$ is $\sim$-invariant. Consider $\langle\bar{b}, i\rangle \sim \langle\bar{c}, j\rangle$. Then $\Psi^{\mathcal{G}_1,\mathcal{G}_2}(h_{1,2})(i) = j$ for all generic enumerations g_1, g_2 extending $\bar{b}, \bar{c}$. Therefore

$$\begin{aligned}\mathfrak{F}(\langle\bar{b}, i\rangle) &= \Psi^{\mathcal{G}_1,\mathcal{B}}(g_1)(i)\\ &= \Psi^{\mathcal{G}_2,\mathcal{B}}(g_2) \circ \Psi^{\mathcal{G}_1,\mathcal{G}_2}(h_{1,2})(i)\\ &= \Psi^{\mathcal{G}_2,\mathcal{B}}(g_2)(j)\\ &= \mathfrak{F}(\langle\bar{c}, j\rangle).\end{aligned}$$

Conversely, to show that $\mathfrak{F}$ is one-to-one on $\sim$-equivalence classes, suppose that $\mathfrak{F}(\langle\bar{b}, i\rangle) = \mathfrak{F}(\langle\bar{c}, j\rangle)$. Then we must have $\Psi^{\mathcal{G}_1,\mathcal{G}_2}(h_{1,2})(i) =$

j for all generic g_1, g_2 extending $\bar{b}, \bar{c}$. To see this, apply the inverse of $\Psi^{\mathcal{G}_2,\mathcal{B}}(g_2)$ to the third equality above. It follows that

$$\langle \bar{b}, \bar{c} \rangle \Vdash_{\mathcal{B}^2} \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2})(i) = j,$$

and hence $\langle \bar{b}, i \rangle \sim \langle \bar{c}, j \rangle$.

That $\mathfrak{F}\colon A^B \to A$ is onto follows from the lemma below.

Lemma VII.43. *For all $a \in A$ and all $\mathcal{L}_{\mathsf{c},\omega}$-generic enumerations g of $\mathcal{B}$, there exists $\bar{c} \subset g$ and $i \in \mathbb{N}$ such that $\mathfrak{F}(\langle \bar{c}, i \rangle) = a$.*

Proof. Let i be such that $\Psi^{\mathcal{G},\mathcal{B}}(g)(i) = a$. Let $g_1 = g$ and let g_2 be mutually generic with g_1 (as built in Lemma VII.38). Let $\mathcal{G}_1$ and $\mathcal{G}_2$ be the corresponding generic presentations. Let $j = \Psi^{\mathcal{G}_1,\mathcal{G}_2}(h_{1,2})(i)$. There exist initial segments $\bar{c} \subset g_1$ and $\bar{b} \subset g_2$ such that

$$\langle \bar{c}, \bar{b} \rangle \Vdash_{\mathcal{B}^2} \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2})(i) = j.$$

Since $\bar{c} \subset g$, we get that $\mathfrak{F}(\langle \bar{c}, i \rangle) = \Psi^{\mathcal{G},\mathcal{B}}(g)(i) = a$. We still need to show that $\langle \bar{c}, i \rangle \in A^{\mathcal{B}}$.

Notice that by flipping the direction, we have $\langle \bar{c}, \bar{b} \rangle \Vdash_{\mathcal{B}^2} \Psi^{\dot{\mathcal{G}}_2,\dot{\mathcal{G}}_1}(\dot{h}_{2,1})(j) = i$ too. It then follows that

$$\langle \bar{c}, \bar{b}, \bar{c} \rangle \Vdash_{\mathcal{B}^3} \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2})(i) = j \;\&\; \Psi^{\dot{\mathcal{G}}_2,\dot{\mathcal{G}}_3}(\dot{h}_{2,3})(j) = i,$$

and hence

$$\langle \bar{c}, \bar{b}, \bar{c} \rangle \Vdash_{\mathcal{B}^3} \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_3}(\dot{h}_{1,3})(i) = i.$$

Since g_2 does not appear in the formula above, by Corollary VII.40 we get

$$\langle \bar{c}, \bar{c} \rangle \Vdash_{\mathcal{B}^2} \Psi^{\dot{\mathcal{G}}_1,\dot{\mathcal{G}}_2}(\dot{h}_{1,2})(i) = i,$$

and hence that $\langle \bar{c}, i \rangle \in A^{\mathcal{B}}$. □

Finally, we need to show that $\mathfrak{F}$ preserves relations. Consider a relation symbol P_i of arity n in the vocabulary of $\mathcal{A}$. Let $\langle a_1, ..., a_n \rangle$ be a tuple from $\mathcal{A}$. Let g be an $\mathcal{L}_{\mathsf{c},\omega}$-generic enumeration of $\mathcal{B}$. Let j_i be such that $\Psi^{\mathcal{G},\mathcal{B}}(g)(j_i) = a_i$ for each $i \leq n$. From the previous lemma, we can obtain an initial segment $\bar{c} \subset g$ long enough such that $\mathfrak{F}(\langle \bar{c}, j_i \rangle) = a_i$ for all $i \leq n$. Furthermore, we can chose $\bar{c}$ even longer so that it decides the $\mathbb{N}$-formula $\langle j_1, \ldots, j_n \rangle \in P_i^{\Psi(\dot{\mathcal{G}})}$.

By the definition of R_i, we have that $\langle \langle \bar{c}, j_1 \rangle, \ldots, \langle \bar{c}, j_n \rangle \rangle \in R_i$ if and only if $\langle j_1, \ldots, j_n \rangle \in P_i^{\Psi(\mathcal{G})}$, which, going through $\Psi^{\mathcal{G},\mathcal{B}}(g)$, holds if and only if $\langle a_1, ..., a_n \rangle \in P_i^{\mathcal{A}}$. So $\mathfrak{F}$ maps R_i to $P_i^{\mathcal{A}}$.

CHAPTER VIII

The game metatheorem

More often than not in computability theory, one is posed with the task of building a certain computable object using non-computable information. Computability theorists have come up with all sorts of techniques to do such constructions, as for instance the priority arguments. Among these techniques, one that has been particularly useful for computable structure theory is the Ash and Knights's metatheorem, which was developed in the late 80's and 90's and is best explained in Ash and Knight's book [**AK00**]. In the next chapter, we will develop another such technique, the *iterated true-stage method*, which grew out of the Ash and Knights's metatheorem. One of the main applications that we will see of the iterated true-stage method is the *game metatheorem*, which was recently introduced by the author [**Mon**]. The game metatheorem provides a ready-to-apply method that is much simpler than the iterated true stages method and than Ash and Knights's metatheorem. The catch is that it is a little less flexible. However, it is still flexible enough that most of the applications of Ash and Knights's metatheorem we know of can be carried out using the game metatheorem instead. So we do not lose that much in terms of flexibility, while we gain a lot in terms of simplicity. We will see the statement of the game metatheorem in Theorem VIII.2, but we will leave its proof to the next chapter, once we develop the iterated true-stage method.

VIII.1. Game constructions

In this section, we introduce a metatheorem[*] stated in terms of a game. It exhibits the interplay between the η-back-and-forth relations and Δ^0_η-information in the clearest possible way.

Let η be a computable ω-presentation of an ordinal. The case $\eta = 2$ is already quite interesting and useful. Suppose we have a computable list of computable structures

$$\mathbb{A} = \{\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2, ...\},$$

[*] Ash and Knights used the word *metatheorem* to emphasize that their theorem provides a general framework which can be used to prove many theorems.

where *the back-and-forth relations are computable up to η*, as in Definition VIII.3 below.

We will now describe a type of construction that we will call an *η-$\mathbb{A}$-game*. This game involves three characters, the *engineer*, the *extender*, and the *oracle*. Together, when the game ends, they will have built an ω-presentation $\mathcal{L}$ which we call the *limit structure*. The goal of the engineer is for the limit structure $\mathcal{L}$ to satisfy a certain property. The extender is in charge of making $\mathcal{L}$ computable — he will not, in any way, coordinate his work with the engineer. The job of the oracle is to answer $\Delta^0_\eta(D(\mathcal{L}))$ questions posed by the engineer. The game is played through infinitely many stages. A *run of the game* is played as follows: At each stage $j \in \mathbb{N}$, first, the engineer plays a triple $\langle i_j, \bar{a}_j, e_j\rangle$ where $i_j, e_j \in \mathbb{N}$ and $\bar{a}_j \in A_{i_j}^{<\mathbb{N}}$, second, the extender plays a tuple $\bar{b}_j \in A_{i_j}^{<\mathbb{N}}$ extending $\bar{a}_j$, and third the oracle plays a pair $\langle n_j, \beta_j\rangle$, where n_j is a number that must be the answer to the e_jth $\Delta^0_\eta(D(\mathcal{L}))$ question and β_j is an ordinal below η.

engineer	$i_0, \bar{a}_0, e_0$		$i_1, \bar{a}_1, e_1$		$i_2, \bar{a}_2, e_2$		$\cdots$
extender		$\bar{b}_0$		$\bar{b}_1$		$\bar{b}_2$	$\cdots$
oracle		n_0, β_0		n_1, β_1		n_2, β_2	$\cdots$

At each stage $j > 0$, the tuple $\bar{a}_j$ played by the engineer must satisfy:

$$(\mathcal{A}_{i_{j-1}}, \bar{b}_{j-1}) \leq_{\beta_{j-1}} (\mathcal{A}_{i_j}, \bar{a}_j). \quad [†]$$

The tuple $\bar{b}_j$ played by the extender must be in the same structure just played by the engineer and must satisfy:

$$\bar{a}_j \subseteq \bar{b}_j.$$

After ω many moves, we get

$$D_{\mathcal{A}_{i_0}}(\bar{a}_0) \subseteq D_{\mathcal{A}_{i_0}}(\bar{b}_0) \subseteq D_{\mathcal{A}_{i_1}}(\bar{a}_1) \subseteq D_{\mathcal{A}_{i_1}}(\bar{b}_1) \subseteq D_{\mathcal{A}_{i_2}}(\bar{a}_2) \subseteq \cdots,$$

and hence we get a limit ω-presentation $\mathcal{L}$ whose atomic diagram is the union of the diagrams of the tuples played:

$$D(\mathcal{L}) = \bigcup_{j\in\mathbb{N}} D_{\mathcal{A}_{i_j}}(\bar{a}_j).$$

The numbers e_j represent $\Delta^0_\eta(\mathcal{L})$ questions, as defined below.[‡] The engineer is responsible for asking questions that converge. The numbers n_j played by the oracle must be the answers to these questions as in Definition VIII.1 below. Notice that we allow the engineer to ask

[†] Recall that $\bar{b}_{j-1} \leq_{\beta_{j-1}} \bar{a}_j$ means that $\bar{b}_{j-1} \leq_{\beta_{j-1}} \bar{a}_j \upharpoonright |\bar{b}_{j-1}|$ and allows $\bar{a}_j$ be longer than $\bar{b}_{j-1}$.

[‡] From now on, we will write $\Delta^0_\eta(\mathcal{L})$ instead of $\Delta^0_\eta(D(\mathcal{L}))$.

questions to be about the limit structure $\mathcal{L}$ even before $\mathcal{L}$ is built — the reader may be smelling the Recursion Theorem somewhere around here. The ordinals β_j must be below η. When η is a successor ordinal, we may assume $\beta_j = \eta - 1$ for all j without losing any generality, making the analysis a bit simpler. When η is a limit ordinal, we may assume β_j is a non-decreasing sequence converging up to η.

Definition VIII.1. Given $X \in 2^{\mathbb{N}}$, $n, e \in \mathbb{N}$, and a computable ordinal η, we say that n is *the answer to the eth $\Delta^0_\eta(X)$ question* if

$$n = \Phi_e^{\mathcal{S}^\eta_X}(0),$$

where $\mathcal{S}^\eta_X$ is some pre-fixed $\Delta^0_\eta(X)$-Turing-complete set and Φ_e is the eth Turing functional.[§]

A *strategy* for the engineer is a function that tells the engineer what to play next, given the previous moves by the extender and the oracle. We say that a strategy is *valid* if, on all possible plays by the extender and oracle, all of the $\Delta^0_\eta(\mathcal{L})$ questions e_j converge.

Theorem VIII.2. *Let $\mathbb{A}$ and η be as described above. For every computable valid strategy for the engineer in the η-$\mathbb{A}$-game, there is a run of the game where the engineer follows her strategy, the oracle answers correctly, and the limit ω-presentation $\mathcal{L}$ is computable.*

Furthermore, we will prove that there is a uniform effective procedure that, given the strategy for the engineer, produces the ω-presentation $\mathcal{L}$ given by the theorem. The proof is fully relativizable, so the result is also uniform in the oracle's answers: If the oracle responds to $\Delta^0_\eta(X)$ questions, then the resulting limit structure will be X-computable, also uniformly in X.

To be able to apply this theorem, one needs to describe a computable valid strategy for the engineer that, with the help of the oracle who is answering her Δ^0_η questions, will build an ω-presentation with the desired property independently of what the extender does. One can then cite the theorem to conclude that, even if the construction relies on the Δ^0_η information provided by the oracle, the resulting ω-presentation is computable.

[§] Let us remark that any finite number of questions of the form $\Phi_{e_0}^{\mathcal{S}^\eta_X}(k_0), \Phi_{e_1}^{\mathcal{S}^\eta_X}(k_1), ..., \Phi_{e_\ell}^{\mathcal{S}^\eta_X}(k_\ell)$ can be encoded into a single question using an index e such that $\Phi_e^{\mathcal{S}^\eta_X}(0)$ outputs a number encoding the tuple $\langle \Phi_{e_0}^{\mathcal{S}^\eta_X}(k_0), \Phi_{e_1}^{\mathcal{S}^\eta_X}(k_1), ..., \Phi_{e_\ell}^{\mathcal{S}^\eta_X}(k_\ell)\rangle$.

VIII.2. Computable back-and-forth relations

The game metatheorem requires us to be able to compute the back-and-forth relations on the structures involved.

Definition VIII.3. Given a list of ω-presentations $\{\mathcal{A}_i : i \in I\}$, we say that the *back-and-forth relations are computable up to* η if the set of quintuples

$$\{\langle \xi, i, \bar{a}, j, \bar{b}\rangle : \xi < \eta, i, j \in I, \bar{a} \in A_i^{<\mathbb{N}}, \bar{b} \in A_j^{<\mathbb{N}}, (\mathcal{A}_i, \bar{a}) \leq_\xi (\mathcal{A}_j, \bar{b})\}$$

is computable.

This is pretty much the same notion as that of η-friendliness used by Ash and Knight [**AK00**, Section 15.2].

Example VIII.4. Given a computable ordinal α, consider the collection of ordinals $\{\omega^\xi : \xi \leq \alpha\}$, where the ω-presentations are chosen as in Observation I.9. It follows from Exercise II.44 that the back-and-forth relations are computable up to 2α (see also [**Ash86a**, Lemma 7] or [**AK00**, Lemma 15.10]).

Example VIII.5. Given a computable ordinal α, consider the collection of linear orderings $\{\mathbb{Z}^\xi : \xi \leq \alpha\}$. Goncharov, Harizanov, Knight, McCoy, and R. Miller [**GHK$^+$05**] gave a complete analysis of the back-and-forth tuples within these structures. It follows from their work that the back-and-forth relations are computable up to 2α.

If the back-and-forth relations up to η are not computable, the game metatheorem can be applied relative to whichever oracle computes them. For instance, the (2η)-th jump of the list of structures is always enough, as in the exercise below.

Exercise VIII.6. Let $\mathbb{A}$ be a list of structures. Prove that the (2η)-th jump of the sequence of diagrams of the structures in $\mathbb{A}$ can compute the back-and-forth relations in $\mathbb{A}$ up to η.

VIII.3. Pairs of structures

The pair of structures theorem of Ash and Knight [**AK90**] is one of the most useful applications of Ash and Knight's metatheorem. It provides a lightface version of Theorem VII.30, which syntactically characterized when distinguishing between two structures $\mathcal{A}$ and $\mathcal{B}$ is (boldface) $\mathbf{\Sigma}^0_\xi$-hard: when $\mathcal{A} \leq_\xi \mathcal{B}$. The proof of Theorem VII.30 used determinacy to build ω-presentations that are far from computable. We now show that under some effectiveness conditions, we can get lightface Σ^0_α-hardness. Among the many applications of this theorem, two important ones are the construction of the αth jump inversion (Theorem X.5) and the construction of a structure whose degree spectrum is exactly the non-hyperarithmetic degrees [**GMS13**].

Theorem VIII.7. *Let η be a computable ordinal and $\mathcal{A}_0$ and $\mathcal{A}_1$ be ω-presentations whose back-and-forth relations are computable up to η. If $\mathcal{A}_0 \geq_\eta \mathcal{A}_1$, distinguishing copies of $\mathcal{A}_1$ from copies of $\mathcal{A}_0$ is Σ^0_η-hard.*

Recall from Definition VII.28 that distinguishing $\mathcal{A}_1$ from $\mathcal{A}_0$ is (lightface) Σ^0_η-*hard* if, for every Σ^0_η subset $\mathfrak{K} \subseteq 2^{\mathbb{N}}$, there is a computable operator $\Gamma\colon 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ such that, for all $X \in 2^{\mathbb{N}}$, Γ^X is the diagram of a copy of $\mathcal{A}_1$ if $X \in \mathfrak{K}$, and Γ^X is the diagram of a copy of $\mathcal{A}_0$ if $X \notin \mathfrak{K}$.

We already proved the cases $\eta = 1$ and $\eta = 2$ in Section II.6.2. Those proofs may give the reader some intuition for why the back-and-forth relations are necessary.

Proof. Let $\mathfrak{K}$ be a Σ^0_η subset of $2^{\mathbb{N}}$. Fix $X \in 2^{\mathbb{N}}$. We will define a computable strategy for the engineer to build a structure that is isomorphic to $\mathcal{A}_1$ if $X \in \mathfrak{K}$, and to $\mathcal{A}_0$ if $X \notin \mathfrak{K}$. Through relativization, we will let the oracle respond $\Delta^0_\eta(X)$ questions instead of Δ^0_η questions. Theorem VIII.2 will then guarantee that there exists a sequence of moves by the extender so that the limit structure $\mathcal{L}$ is uniformly computably in X. This will be the computable operator Γ needed to prove Σ^0_η-hardness.

Whether X is in $\mathfrak{K}$ or not is a Σ^0_η question, and not a $\Delta^0_\eta(X)$ question, so we cannot ask the oracle directly about it. Instead, we use a computable list of indices e_0, e_1,... for $\Delta^0_\eta(X)$ questions whose answers, n_0, n_1, ..., are either all zeros if $X \notin \mathfrak{K}$ or start with zeros and then change to all ones if $X \in \mathfrak{K}$. To see how to get these indices, let W be a c.e. operator such that $X \in \mathfrak{K} \iff 0 \in W^{\mathcal{S}^\eta_X}$, where $\mathcal{S}^\eta_X$ is a Δ^0_η-Turing-complete set. Then, let e_j be an index so that $\Phi^{\mathcal{S}^\eta_X}_{e_j}(0) = 1$ if 0 is enumerated in $W^{\mathcal{S}^\eta_X}$ in less than j steps and $\Phi^{\mathcal{S}^\eta_X}_{e_j}(0) = 0$ otherwise.

The idea behind the strategy for the engineer for the η-$\{\mathcal{A}_0, \mathcal{A}_1\}$-game is as follows: At stage j, ask the $\Delta^0_\eta(X)$ question e_j as in the previous paragraph. Play tuples in $\mathcal{A}_0$ while the oracle's answers are $n_j = 0$, and switch to playing tuples in $\mathcal{A}_1$ if the oracle ever switches to answering $n_j = 1$. The hypothesis that $\mathcal{A}_0 \geq_\eta \mathcal{A}_1$ is used to find $\bar{a}_{j+1} \in \mathcal{A}_1^{<\mathbb{N}}$ so that $(\mathcal{A}_0, \bar{b}_j) \leq_{\beta_j} (\mathcal{A}_1, \bar{a}_{j+1})$ when we do the switch. Let us describe this in more detail.

On the strategy's first move, play the empty tuple in $\mathcal{A}_0$, and ask about e_0 — i.e., play the triple $\langle 0, \langle\rangle, e_0\rangle$. On the $(j+1)$st move, play the triple $\langle i_j, \bar{a}_{j+1}, e_{j+1}\rangle$, which we define as follows: Let $i_{j+1} = n_j$, where n_j is the oracle's answer to the previous $\Delta^0_\eta(X)$-question, and let e_{j+1} be as defined a couple of paragraphs above. If $n_j = i_j$, stay in the same structure and play any tuple $\bar{a}_{j+1}$ in $\mathcal{A}_{n_j}$ extending $\bar{b}_j$. To ensure surjectivity, choose $\bar{a}_{j+1}$ so that it contains at least the first j elements of the ω-presentation of $\mathcal{A}_{n_j}$. If $n_j \neq i_j$, it must be because $i_j = 0$ and $n_j = 1$. In this case, play a tuple $\bar{a}_{j+1} \in \mathcal{A}_1^{<\mathbb{N}}$ such that $(\mathcal{A}_1, \bar{a}_{j+1}) \geq_{\beta_j} (\mathcal{A}_0, \bar{b}_j)$. The existence of such an $\bar{a}_{j+1}$ follows from the hypothesis that $\mathcal{A}_1 \leq_\eta \mathcal{A}_0$. We can find $\bar{a}_{j+1}$ computably because we are assuming that the back-and-forth relations are computable up to η.

If $X \notin \mathfrak{K}$, at the end of the game we get that $\{\bar{a}_j : j \in \mathbb{N}\}$ is an increasing sequence of tuples in $\mathcal{A}_0$, and hence the limit structure is isomorphic to $\mathcal{A}_0$. If $X \in \mathfrak{K}$ and s_0 is the first stage with $n_{s_0} = 1$, then $\{\bar{a}_j : j \in \mathbb{N}, j > s_0\}$ is an increasing sequence of tuples in $\mathcal{A}_1$, and hence the limit structure is isomorphic to $\mathcal{A}_1$. □

As we mentioned before, if the back-and-forth relations up to η are not computable, the game metatheorem can be applied relative to whichever oracle computes them. For instance, the (2η)-th jump of the given ω-presentations of $\mathcal{A}_0$ and $\mathcal{A}_1$ is always enough (see Exercise VIII.6). If we care about the complexity of the oracle relative to which the structures are Σ^0_η-hard, this new proof is much better than Theorem VII.30 because the (2η)-th jump is much lower than the oracle we get from Σ^0_η-determinacy, whose proof needs around η iterations of the power-set axiom of ZFC.

We can modify the proof of the theorem above in the situation when $\mathcal{A}_0 \equiv_\eta \mathcal{A}_1$ and get $\Delta^0_{\eta+1}$-hardness:

Theorem VIII.8. *Let η be a computable ordinal and $\mathcal{A}_0$ and $\mathcal{A}_1$ be ω-presentations whose back-and-forth relations are computable up to η. If $\mathcal{A}_0 \equiv_\eta \mathcal{A}_1$, distinguishing copies of $\mathcal{A}_1$ from copies of $\mathcal{A}_0$ is $\Delta^0_{\eta+1}$-hard.*

Proof. The proof is almost identical to the proof above. The only difference is that the answers n_0, n_1, ... to our Δ^0_η questions may flip finitely often between zeros and ones before they stabilize at either 0 or 1. The effect of this on the proof is that, when we have $n_j \neq i_j$, we could be going from 0 to 1 or from 1 to 0. If we are going from 0 to 1 (i.e., $i_j = 0$ and $n_j = 1$), we define $\bar{a}_{j+1} \in \mathcal{A}_1^{<\mathbb{N}}$ exactly as in the previous proof using $\mathcal{A}_1 \leq_\eta \mathcal{A}_0$. If we are going from 1 to 0 (i.e., $i_j = 1$ and $n_j = 0$), we just switch the roles of $\mathcal{A}_0$ and $\mathcal{A}_1$ and we define $\bar{a}_{j+1} \in \mathcal{A}_0^{<\mathbb{N}}$ using $\mathcal{A}_1 \geq_\eta \mathcal{A}_0$. □

Remark VIII.9. In the theorem above, the isomorphism between the limit structure $\mathcal{L}$ and whichever of $\mathcal{A}_0$ and $\mathcal{A}_1$ is supposed to be isomorphic to $\mathcal{L}$ is $\Delta^0_{\eta+1}$ (relative to X, of course). The isomorphism is actually Δ^0_η, but we need $\Delta^0_{\eta+1}$ if we want uniformity in X. This is because $\Delta^0_{\eta+1}$ can tell at which point the answers n_0, n_1, ... stabilize to either 0 or 1. After that point, we are just copying the elements of either $\mathcal{A}_0$ or $\mathcal{A}_1$, so we can easily produce the isomorphism from the run of the η-$\{\mathcal{A}_0, \mathcal{A}_1\}$-game. The complexity of the run of the game is given by the oracle's answers, which are Δ^0_η, and the answers by the extender, which can be taken to be Δ^0_η too. This follows from Remarks IX.27 and IX.31 after the proofs of the game metatheorem.

VIII.4. Linear ordering presentations

Here is another classical result that needs $0^{(\eta)}$ information.

Theorem VIII.10. *Let η be a computable ordinal and let $\mathcal{A}$ be a linear ordering with a first element. Then $\mathcal{A}$ has a $\Delta^0_{2\eta+1}$ copy if and only if $\omega^\eta \cdot \mathcal{A}$ has a computable copy.*

This theorem was proved by Watnick [**Wat84**] for the case $\eta = 1$ and then extended to all η by Ash, Jockusch, and Knight [**AJK90**] using workers and by Ash [**Ash91**] using 2η-systems.

Proof. The easier direction is the ($\Leftarrow$) direction. Suppose that $\mathcal{B}$ is a computable copy of $\omega^\eta \cdot \mathcal{A}$. Consider the equivalence relation on $\mathcal{B}$ given by $a \sim b$ if the interval $[a, b]_\mathcal{B}$ has order type less than ω^η. Recall from Lemma II.5 that this can be decided by a $\Sigma^{\mathtt{c}}_{2\eta}$ formula. Taking the quotient of $\mathcal{B}$ under $\sim$, we get a $\Delta^0_{2\eta+1}$ congruence ω-presentation of $\mathcal{A}$.

For the ($\Rightarrow$) direction, assume that $\mathcal{A}$ is itself a $\Delta^0_{2\eta+1}$ ω-presentation. Assume that the least element of $\mathcal{A}$ is the 0 of its ω-presentation.

The pool $\mathbb{A}$ of structures that we use for our game consists of all the linear orderings of the form $\omega^\eta \cdot \mathcal{F}$, where $\mathcal{F}$ is a finite linear ordering whose domain is an initial segment of $\mathbb{N}$. The back-and-forth relations between these structures are computable up to $2\eta+1$. Precise calculations of the back-and-forth relations among ordinals are done in Exercise II.44.

Another observation we need is that if $\mathcal{F}_0 \subseteq \mathcal{F}_1$ are linear orderings with the same first element 0, then $\omega^\eta \cdot \mathcal{F}_0$ is a $\Sigma^{\mathtt{in}}_{2\eta+1}$-elementary substructure of $\omega^\eta \cdot \mathcal{F}_1$. In other words, for every tuple $\bar{b} \in \omega^\eta \cdot \mathcal{F}_0$, we have that $(\omega^\eta \cdot \mathcal{F}_0, \bar{b}) \geq_{2\eta+1} (\omega^\eta \cdot \mathcal{F}_1, \bar{b})$. The reason is that if an interval (b_i, b_j) of $\omega^\eta \cdot \mathcal{F}_0$ changes when you view it in $\omega^\eta \cdot \mathcal{F}_1$, it is because we added a few intervals of the form ω^η in between b_i and b_j. That is, the interval changed from being isomorphic to $\omega^\eta \cdot k_0 + \beta$ in $\omega^\eta \cdot \mathcal{F}_0$ to being isomorphic to $\omega^\eta \cdot k_1 + \beta$ in $\omega^\eta \cdot \mathcal{F}_1$ for some $0 < k_0 < k_1 \in \mathbb{N}$ and

$\beta < \omega^\eta$. Recall from Lemma II.38 that $\omega^\eta \cdot k_0 \geq_{2\eta+1} \omega^\eta \cdot k_1$, getting the desired result from Lemma II.37.

We describe a computable strategy for the engineer in the $(2\eta+1)$-$\mathbb{A}$-game. At stage $j-1$, the engineer asks the oracle for a full description of $\mathcal{A} \restriction j$, i.e., for the ordering $\leq_{\mathcal{A}}$ on the first j natural numbers of the ω-presentation $\mathcal{A}$. At the following stage, stage j, she chooses the structure $\omega^\eta \cdot \mathcal{F}_j$ in $\mathbb{A}$, where $\mathcal{F}_j = \mathcal{A} \restriction j$. Note that $\mathcal{F}_j$ naturally extends $\mathcal{F}_{j-1}$. From our observation above, $\omega^\eta \cdot \mathcal{F}_{j-1}$ is a $\Sigma^{\mathtt{in}}_{2\eta+1}$-elementary substructure of $\omega^\eta \cdot \mathcal{F}_j$. So, we know that $(\omega^\eta \cdot \mathcal{F}_{j-1}, \bar{b}_{j-1}) \equiv_{2\eta} (\omega^\eta \cdot \mathcal{F}_j, \bar{b}_{j-1})$, and hence that the engineer can play any tuple $\bar{a}_j$ extending $\bar{b}_{j-1}$. All she needs to do is make sure that she ends up including all members of $\omega^\eta \cdot \mathcal{A}$ eventually. The limit structure will then be isomorphic to the limit of the structures $\omega^\eta \cdot \mathcal{F}_j$, namely $\omega^\eta \cdot \mathcal{A}$, as all the engineer ends up doing is enumerating longer and longer tuples from $\omega^\eta \cdot \mathcal{A}$. □

The theorem is still true if $\mathcal{A}$ has no least element (see [**Mon**]). The theorem is also true for $\mathbb{Z}^\eta \cdot \mathcal{A}$ instead of $\omega^\eta \cdot \mathcal{A}$ by essentially the same proof.

VIII.5. Δ^0_η-categoricity

A computable structure $\mathcal{A}$ is Δ^0_η-*categorical* if, for every computable copy $\mathcal{B}$ of $\mathcal{A}$, there is a Δ^0_η isomorphism between $\mathcal{A}$ and $\mathcal{B}$. Downey, Kach, Lempp, Lewis, Montalbán, and Turetsky [**DKL$^+$15**] proved that this property cannot be characterized structurally. However, a variant of it, namely the relative version, can be characterized in terms of structural properties, as we have shown in Section VII.4. The on-a-cone version has an even nicer characterization: A structure is Δ^0_η-categorical on a cone if and only if it has Scott rank less than or equal to η (Corollary VII.24).

Unfortunately, the three notions of plain, relative, and on-a-cone Δ^0_η-categoricity are not equivalent. Examples of this non-equivalence were built by Goncharov, Harizanov, Knight, McCoy, R. Miller, and Solomon [**GHK$^+$05**]. Other examples for the case $\eta = 1$ can be found in [**Part 1**, Section VIII.5]. However, they are equivalent for most natural structures one encounters. Ash proved that these notions are equivalent if we have enough structural information about $\mathcal{A}$. To understand his result, we need to use the notion of η-freeness.

Recall from Definition II.64 that a tuple $\bar{c}$ is η-*free* if and only if, for every $\bar{b} \supseteq \bar{c}$ and $\beta < \eta$, there exist tuples $\bar{c}' \subseteq \bar{b}'$ such that $\bar{b} \leq_\beta \bar{b}'$

but $\bar{c} \not\leq_\eta \bar{c}'$.

$$\begin{array}{ccc} \bar{c} & \not\leq_\eta & \bar{c}' \\ \rotatebox{90}{$\subseteq$} & & \rotatebox{90}{$\subseteq$} \\ \bar{b} & \leq_\beta & \bar{b}' \end{array}$$

We then proved in Lemma II.65 that $\bar{c}$ is η-free if and only if its $\Pi^{\mathtt{in}}_\eta$-type is not $\Sigma^{\mathtt{in}}_\eta$ supported. It follows from Theorem II.23 and Corollary VII.24 that, for a structure $\mathcal{A}$, the following are equivalent:

(1) $\mathcal{A}$ has Scott rank less than or equal to η.
(2) $\mathcal{A}$ is Δ^0_η-categorical on a cone.
(3) There is a tuple $\bar{p} \in \mathcal{A}^{<\mathbb{N}}$ such that no tuple $\bar{c} \in \mathcal{A}^{<\mathbb{N}}$ is η-free over $\bar{p}$.[¶]

In practice, when we have a good understanding the back-and-forth relations on a given structure, we can effectively decide which tuples are η-free and we can effectively find witnesses for the tuples that are not η-free. When that is the case, we say that *η-freeness is computable* in $\mathcal{A}$. It was under this assumption together with the computability of the back-and-forth relations, that Ash [**Ash87**] proved that Δ^0_η categoricity implies that all tuples are η-free over some tuples of parameters.

Theorem VIII.11. *Let $\mathcal{A}$ be a computable ω-presentation where both the back-and-forth relations up to η and η-freeness are computable. If $\mathcal{A}$ is Δ^0_η-categorical, it has Scott rank less than or equal to η.*

The proof we provide is essentially the η-$\mathcal{A}$-game version of Nurtazin's proof that computable categoricity for decidable copies is equivalent to effective atomicity over a finite set of parameters [**Part 1**, Theorem VIII.11] and of the proof that computable categoricity implies relative computable categoricity for 2-decidable structures [**Part 1**, Theorem VIII.13]. We recommend the reader study those proofs first, as many of those ideas are incorporated here.

Proof. Suppose that $\mathcal{A}$ has Scott rank greater than η and, thus, that over every tuple $\bar{p}$, there is a tuple that is η-free. We will show that $\mathcal{A}$ is not Δ^0_η-categorical.

We build a copy of $\mathcal{A}$ by defining a computable strategy for the engineer in an η-$\mathcal{A}$-game construction (with $\mathbb{A} = \{\mathcal{A}\}$). In the previous examples, the engineer always extended tuples played by the extender when she did not need to change structures. In this construction, the tuples played by the engineer will keep on jumping around the structure $\mathcal{A}$, of course always being β_j-greater than the ones played by the extender. We want to end up building a copy of $\mathcal{A}$, so we will make

[¶] By *η-free over $\bar{p}$*, we mean η-free within the structure $(\mathcal{A}, \bar{p})$.

sure that the tuples $\bar{a}_j$ stabilize in the limit, i.e., that for each $n \in \mathbb{N}$, $\lim_{j\to\infty} \bar{a}_j(n)$ exists — call this limit $g(n)$. We will then end up with a function

$$g\colon \omega \to \mathcal{A},$$

and the limit ω-presentation $\mathcal{L}$ will be the pull-back of $\mathcal{A}$ through g. The objective is to build $\mathcal{L}$ so that it is not Δ^0_η-isomorphic to $\mathcal{A}$. So that it witnesses that $\mathcal{A}$ is not Δ^0_η-categorical.

While the engineer is playing the game, she will be performing a finite-injury priority construction with infinitely many requirements R_e for $e \in \mathbb{N}$. That is, the run of the game itself will be a finite-injury priority construction where the engineer's moves may injure lower priority requirements she herself had sought to satisfy earlier. The only difference with a standard finite-injury priority construction is that after each stage, the extender will extend the tuple $\bar{a}_j$ to $\bar{b}_j$, and at the next stage, the engineer must play a tuple $\bar{a}_{j+1} \geq_{\beta_j} \bar{b}_j$. Our requirements take the following form:

> Requirement R_e: Ensure that $\Phi_e^{\mathcal{S}^\eta}$ is not an isomorphism from $\mathcal{L}$ to $\mathcal{A}$ as follows: Find a tuple $\bar{n} \in \mathbb{N}^{<\mathbb{N}}$ such that $\Phi_e^{\mathcal{S}^\eta}(\bar{n})$ and $g(\bar{n})$ are not automorphic in $\mathcal{A}$.[‖]

We order these requirements by order of priority: The smaller the e, the higher the priority. Note that R_e will ensure that $\Phi_e^{\mathcal{S}^\eta} \circ g^{-1}$ is not an automorphism of $\mathcal{A}$, and hence that $\Phi_e^{\mathcal{S}^\eta}$ is not an isomorphism from $\mathcal{L}$ to $\mathcal{A}$. To ensure its goal, R_e will choose a tuple $\bar{n}$ and wait for $\Phi_e^{\mathcal{S}^\eta}$ to converge on $\bar{n}$ to some tuple $\tilde{c}_e$. Then, if necessary, R_e will change the value of $g(\bar{n})$ so that it is not η-back-and-forth equivalent to $\tilde{c}_e$, and in particular, not automorphic to $\tilde{c}_e$. Since we do not know when or where $\Phi_e^{\mathcal{S}^\eta}$ converges, we cannot ask the Δ^0_η oracle directly about its values. All we can ask is, given a tuple $\bar{n}$ and a number s, whether $\Phi_e^{\mathcal{S}^\eta}$ converges on the numbers in the tuple $\bar{n}$ within s steps.

At each step j, we are given $\bar{b}_{j-1}$ and we, as the engineer, are supposed to define $\bar{a}_j$. The the beginning of step j, there is an initial segment R_0,...,$R_{k_{j-1}}$ of the list of requirements that are *active*. The value of k_j will increase and decrease throughout the construction but will eventually grow to infinity. Before her move, the engineer goes through the active requirements one at the time, checking any of them *requires attention* (defined below). Each requirement R_e will be given a tuple $\bar{p}_e \in A^{<\mathbb{N}}$ when initialized, and it will output a tuple $\bar{p}_{e+1}$ extending it. Lower priority requirements are not allowed to modify $\bar{p}_{e+1}$. The output tuple $\bar{p}_{e+1}$ may change when the requirement R_e acts, which

[‖] Recall that $\mathcal{S}^\eta$ is some fixed Δ^0_η-Turing-complete real.

would injure the work of lower priority requirements. If the input tuple $\bar{p}_e$ changes, the requirement R_e must be deactivated and re-initialized again. One should thus think of $\bar{p}_e$ as a tuple $\bar{p}_e[j]$ that depends on the stage j. All the inputs of the currently active requirements are initial segments of $\bar{b}_{j-1}$, i.e., $(\forall e \leq k_{j-1})\ \bar{p}[j-1] \subseteq \bar{b}_{j-1}$. All the outputs of the currently active requirements are going to be initial segments of $\bar{a}_j$, i.e., $(\forall e \leq k_j)\ \bar{p}[j] \subseteq \bar{a}_j$. When a higher priority requirement *acts*, the weaker requirements are deactivated, and their $\bar{p}_e$'s become undefined, to be re-defined later. We will see, however, that for each e, $\bar{p}_e[j]$ will stabilize as $j \to \infty$, and hence we will end up with a limit function $g\colon \omega \to \mathcal{A}$, where $g(n) = \lim_{j\to\infty} \bar{p}_{k_j}[j](n)$.

If none of the requirements R_e for $e \leq k_{j-1}$ requires attention, the engineer initializes the first inactive requirement, namely, $R_{k_{j-1}+1}$. Here is how the initialization works: Let $k_j = k_{j-1}+1$. For $e = k_j$, using the computability of η-freeness, the engineer looks for a tuple $\bar{c}_e$ that is η-free over $\bar{p}_e$ and adds it to the tuple played by the extender, say on position $\bar{n}_e \in \mathbb{N}^{<\mathbb{N}}$. (Recall that we will always have $\bar{p}_e[j-1] \subseteq \bar{b}_{j-1}$.) That is, she plays the tuple

$$\bar{a}_j = \bar{b}_{j-1}{}^\frown\bar{c}_e{}^\frown d,$$

where d is the least element in $\mathcal{A}$ not yet played, and $\bar{n}_e$ is the position of $\bar{c}_e$ within $\bar{a}_j$, namely $\langle |\bar{b}_{j-1}|, |\bar{b}_{j-1}|+1, ..., |\bar{b}_{j-1}{}^\frown\bar{c}_e|-1\rangle$. She then asks the oracle whether $\Phi_e^{\mathcal{S}^\eta}(\bar{n}_e)$ converges within j steps. She will keep on asking about this convergence at every later stage $j' > j$, using larger and larger time bounds, until she gets an answer. Actually, it is for all $e \leq k_j$ simultaneously that she asks whether $\Phi_e^{\mathcal{S}^\eta}(\bar{n}_e)$ converges within j steps, encapsulating all the questions into one question.[**] She defines $\bar{p}_{e+1} = \bar{a}_j$ and finishes this stage.

What do we mean by requiring attention, and what does the engineer do then? If we obtain an answer from the oracle that says that, for some $e \leq k_{j-1}$, $\Phi_e^{\mathcal{S}^\eta}(\bar{n}_e)$ converges within $j-1$ steps, we say that R_e *requires attention*. The engineer then picks the least such e and *acts* on it. Suppose that $\Phi_e^{\mathcal{S}^\eta}(\bar{n}_e) = \tilde{c}_e$. So, we have that $\bar{p}_{e+1}$ maps $\bar{n}_e$ to $\bar{c}_e$, while $\Phi_e^{\mathcal{S}^\eta}$ maps $\bar{n}_e$ to $\tilde{c}_e$, as pictured below.

Then, she checks if $\bar{c}_e \leq_\eta \tilde{c}_e$. **If not**, she does not need to do anything, as we would then know that $\bar{c}_e$ and $\tilde{c}_e$ are not non-automorphic. She plays $\bar{a}_j = \bar{b}_{j-1}{}^\frown d$, where d is the least element in $\mathcal{A}$ not yet played, declares R_e satisfied, leaves $\bar{p}_{e+1}$ unchanged (for now, and so long it is not re-initialized later), and lets $k_j = k_{j-1}$. Notice that if R_e is never

[**] To encapsulate many questions into one, do as in the footnote on page 135.

$$\begin{tikzcd}
\mathcal{A} & \mathcal{L} \arrow[l, "\cong", "g"'] \arrow[r, "\cong", "\Phi_e^{\mathcal{S}^\eta}"'] & \mathcal{A} \\
\bar{c}_e \arrow[u, dash, dotted] \arrow[d, phantom, "\not\geq_\eta" description] & \bar{n}_e \arrow[u, dash, dotted] \arrow[l, mapsto, dotted, "\bar{p}_{e+1}[j{-}1]"'] \arrow[r, mapsto, "\Phi^{\mathcal{S}^\eta}_{e,j-1}"'] \arrow[dl, mapsto, dotted, "\bar{p}_{e+1}[j]"] & \tilde{c}_e \arrow[u, dash, dotted] \\
\bar{c}'_e & &
\end{tikzcd}$$

re-initialized again, we will end up with $g \supseteq \bar{p}_{e+1}[j]$ and with

$$g(\bar{n}) = \bar{c}_e \not\leq_\eta \tilde{c}_e = \Phi_e^{\mathcal{S}^\eta}(\bar{n}_e).$$

If yes, that is, if $\bar{c}_e \leq_\eta \tilde{c}_e$, she will replace $\bar{c}_e$ by a tuple $\bar{c}'_e \not\geq_\eta \bar{c}_e$, getting that $\bar{c}'_e$ and $\tilde{c}_e$ are not $\equiv_\eta$-equivalent, and hence not automorphic. To find such a $\bar{c}'_e$, recall that she had chosen $\bar{c}_e$ so that it was η-free over $\bar{p}_e$. We can then apply η-freeness to the tuple $\bar{b}_{j-1} \supseteq \bar{p}_e\bar{c}_e$ and the ordinal β_{j-1} played by the oracle, and, using the computability of the back-and-forth relations, we get tuples $\bar{c}'_e$ and $\bar{b}'_j \supseteq \bar{p}_e\bar{c}'_e$ such that

$$\bar{b}_{j-1} \leq_{\beta_{j-1}} \bar{b}'_j, \quad \text{but} \quad \bar{p}_e\bar{c}_e \not\leq_\eta \bar{p}_e\bar{c}'_e.$$

The engineer now plays $\bar{a}_{j+1} = \bar{b}'_j{}^\frown d$, where d is the least element in $\mathcal{A}$ not yet played, declares R_e satisfied with $\bar{p}_{e+1} = \bar{a}_j$, deactivates lower priority requirements, and lets $k_j = e$. Notice that if R_e is never re-initialized again, we will end up with $g \supseteq \bar{p}_{e+1}[j]$ and with

$$g(\bar{n}) = \bar{c}'_e \not\equiv_\eta \tilde{c}_e = \Phi_e^{\mathcal{S}^\eta}(\bar{n}_e).$$

Each requirement R_e acts at most once after being initialized. One can then prove by induction on e that each requirement stops being deactivated by higher priority requirements from some point on. After the last time R_e is initialized, if it requires attention, it will eventually get it. We will end up with $\bar{p}_{e+1}[j] \subseteq \bar{a}_{j'}$ for all $j' \geq j$, and hence with $\bar{p}_{e+1}[j] \subseteq g$. We would have then satisfied R_e because: Either $\Phi_e^{\mathcal{S}^\eta}(\bar{n}_e)$ does not converge and R_e never requires attention after the last time it was initialized, or $\Phi_e^{\mathcal{S}^\eta}(\bar{n}_e)$ converges and, once R_e is given attention, the engineer ensures that g maps $\bar{n}_e$ to a tuple not η-back-and-forth equivalent to $\tilde{c}_e$. □

CHAPTER IX

Iterated True-Stage Arguments

In many constructions in computability theory, one builds a computable object using non-computable information. The main tool for such constructions is the priority method, which has become increasingly more involved and sophisticated since it was invented in the 1950s. On such a computable construction, one has to guess at whatever non-computable information is needed. Such guesses will be right sometimes and wrong some other times, and when they are wrong they will guide us in the wrong direction, maybe messing up other parts of the construction. By carefully organizing these guesses, one can find techniques to recover from the mistakes made when assuming wrong guesses. The method of iterated true stages provides a way to organize such guesses in a clean, combinatorial way. It is then up to the user to build the desired computable object using these guesses.

Priority arguments are classified in terms of how much non-computable information is needed throughout the construction. The most common priority constructions are the finite-injury ones [**Fri57, Muc56**] (see [**Part 1**, Chapter VII]). They are used when the information needed is $0'$-computable. Infinite-injury priority constructions [**Sho61, Sac63**] are used when $0''$-computable guesses are needed. There are various $0'''$-priority constructions in the literature [**Lac76**], but they are very complicated and far less common. Beyond that point, it becomes humanly impossible to keep track of the combinatorics. Well, that is unless the level-by-level combinatorics of the proof is uniform and one can describe the work done at all the levels simultaneously with a single procedure. There have been various proposals for general $0^{(n)}$-injury constructions: Harrington's workers method [**Har76**], Lempp and Lerman's trees of strategies [**LL95, Ler10**], Ash's [**Ash86b**] and Ash–Knight's [**AK00**] η-systems, Montalbán's iterated true stages [**Mon14**], and Montalbán's game metatheorem. Harrington's workers method refers to a particular way of visualizing a construction where we have a worker working at each level $0^{(m)}$, and each worker is monitoring the actions of the other workers, limit-guessing what the $0^{(m+1)}$ worker is doing, and knowing the jump of what the $0^{(m-1)}$ worker is doing. It was recently used in computable structure theory by Andrews and Knight [**AK18**] in work related to strongly minimal theories. Lempp and Lerman's method [**LL90**] provides a way to organize the requirements in a priority construction using trees of strategies, where we have a different tree at each level, all interacting with each other. It is great for constructions in degree theory and has been used, for instance, to prove that every finite jump upper semi-lattice with 0 can be embedded in the Turing degrees [**Ler10**]. Ash and Knight's η-systems are very different from anything done before. Their metatheorem says that if a certain combinatorial machinery can be put in place, one can then build the desired computable object using Δ^0_η-information.* It has many applications, all of them in computable structure theory, where the combinatorial features needed occur naturally.

The method we describe in this chapter is the iterated true stages method, which was inspired by Ash and Knight's η-systems and Lachlan's true stages. It is more hands-on and more flexible than Ash and Knight's metatheorem. We will give two applications: One is the proof of the game metatheorem. The other is the tree-of-structures theorem that generalizes the pair-of-structures theorem (Theorem IX.25). The tree-of-structures theorem cannot be proved using either the game metatheorem or the Ash and Knight's metatheorem, because one needs to pay attention to all the levels of the construction at every step, not just to the η-th level. Other applications can be found in Csima and Harrison-Trainor [**CHT17**], Greenberg and Turetsky [**GT**], and Day and Marks (in preparation).

We already developed the case $\eta = 1$ in [**Part 1**, Chapter VII]. Knowledge of [**Part 1**, Chapter VII] is not required to read this chapter, although it may help with some intuition and motivation.

IX.1. A global true-stage system

In [**Part 1**, Chapter VII], we developed the notion of true stages (which we now call 1-true stages) as a way to organize priority constructions that require guessing at Δ^0_2 information, and we saw how this combinatorial device is applied. The idea was that at each stage s, we have a finite string $\mathcal{T}_s$ which we think of as an approximation to some Δ^0_2-Turing-complete real $\mathcal{T} \in \mathbb{N}^\mathbb{N}$. These approximations are just guesses, and they are often wrong. The stages at which $\mathcal{T}_s$ is correct,

* It was first introduced by Ash in 1986 [**Ash86a, Ash86b**], and several slightly different versions were proposed later in the 90's by Ash and Knight [**Ash90, AK94b, AK94a, Kni95**]. The best and final formulation is due to Ash and Knight [**AK00**].

meaning that it is an initial segment of $\mathcal{T} \in \mathbb{N}^{\mathbb{N}}$, are said to be *true stages*. Of course, we want the sequence of strings $\mathcal{T}_s$ to be computable in s. Then, if we knew which stages were true, we could compute $\mathcal{T}$. So, the sequence of true stages will have the same Turing degree as $\mathcal{T}$. Here is a somewhat circular idea: We can use the sequence of true stages as our Δ^0_2-Turing-complete real $\mathcal{T}$, and let $\mathcal{T}_s$ be the string listing the stages $t \leq s$ that *appear* to be true at stage s. This notion of t *appearing to be true at* s, denoted $t \leq_1 s$, is the key notion of [**Part 1**, Chapter VII] and the launch pad for this chapter.

One of the main advantages of this technique is that it can be easily iterated through the arithmetic hierarchy and, with a bit more work, through the hyperarithmetic hierarchy. To be able to iterate the notion of t appearing to be true at s, we need to consider strings instead of numbers. We will define an ordering $\preccurlyeq$ on strings that will be the basic primitive notion of this whole chapter. All other orderings and all approximations to Δ^0_η-Turing-complete sequences will be built from it.

Definition IX.1. A *global 1-true-stage ordering* is a computable partial ordering $\preccurlyeq$ on $\mathbb{N}^{<\mathbb{N}}$ that satisfies the following properties:

(TS0) $\langle\rangle \preccurlyeq \tau$ for all τ.

(TS1) If $\sigma \preccurlyeq \tau$, then $\sigma \subseteq \tau$.

(TS2) For each $X \in \mathbb{N}^{\mathbb{N}}$, there is an infinite sequence of initial segments of X such that

$$\tau_0 \preccurlyeq \tau_1 \preccurlyeq \tau_2 \preccurlyeq \cdots \subset X.$$

(♣) For every $\tau \subset \sigma \subset \rho$, if $\tau \preccurlyeq \rho$, then $\tau \preccurlyeq \sigma$.

$$\begin{tikzcd}
\tau \arrow[r, phantom, "\subseteq"] \arrow[rr, bend left, dash, "\preccurlyeq"] \arrow[r, bend right, dotted, dash, "\preccurlyeq"'] & \sigma \arrow[r, phantom, "\subseteq"] & \rho
\end{tikzcd}$$

We say that $\tau \subset X$ is an *X-true substring* if there is an infinite sequence $\tau \preccurlyeq \tau_1 \preccurlyeq \tau_2 \preccurlyeq \cdots \subset X$, as in (TS2), starting with τ. If so, we write $\tau \preccurlyeq X$.

Notice that for $\tau \preccurlyeq X$, we have that $\sigma \preccurlyeq \tau$ implies $\sigma \preccurlyeq X$, as witnessed by the same $\preccurlyeq$-increasing sequence. Thus, when $\sigma \preccurlyeq \tau$, we say that σ is *τ-true* or that σ *looks true* to τ. Property (♣) can be read as follows: if τ looks true to ρ, then it looks true to any σ between τ and ρ. It is the key combinatorial property encapsulating how a "looking true" ordering should behave.

Lemma IX.2. *A string τ is an X-true substring if and only if $\tau \preccurlyeq \sigma$ for all σ with $\tau \subseteq \sigma \subset X$.*

Proof. For the ($\Rightarrow$) direction, consider a sequence $\tau \preccurlyeq \tau_1 \preccurlyeq \tau_2 \preccurlyeq \cdots \subset X$ witnessing that τ is an X-true substring. Now, given σ with $\tau \subseteq \sigma \subset X$, let k be such that $\sigma \subseteq \tau_k$. Apply ($\clubsuit$) on $\tau \subseteq \sigma \subseteq \tau_k$ to obtain that $\tau \preccurlyeq \sigma$.

For the ($\Leftarrow$) direction, consider a sequence $\tau_0 \preccurlyeq \tau_1 \preccurlyeq \tau_2 \preccurlyeq \cdots \subset X$ as in (TS2). Let k be such that $\tau \subseteq \tau_k$. By the assumption, we must have $\tau \preccurlyeq \tau_k$, and therefore τ must be an X-true substring too, as witnessed by $\tau \preccurlyeq \tau_k \preccurlyeq \tau_{k+1} \preccurlyeq \cdots \subset X$. □

Let $\mathfrak{T}_X \in \mathbb{N}^{\mathbb{N}}$ be the sequence of all X-true substrings listed in increasing order. That is,

$$\mathfrak{T}_X = \langle \tau \in \mathbb{N}^{<\mathbb{N}} : \tau \preccurlyeq X \rangle.$$

It follows from the lemma that if $\sigma \subseteq \tau$ are both in $\mathfrak{T}_X$, then $\sigma \preccurlyeq \tau$. So, $\mathfrak{T}_X$ is itself a $\preccurlyeq$-increasing sequence — a maximal one.

For $\rho \in \mathbb{N}^{<\mathbb{N}}$, we define

$$\mathfrak{T}_\rho = \langle \tau \in \mathbb{N}^{<\mathbb{N}} : \tau \preccurlyeq \rho \rangle$$

as our approximation to $\mathfrak{T}_X$ at ρ. We let the reader verify that ρ is an X-true substring if and only if $\mathfrak{T}_\rho$ is an initial segment of $\mathfrak{T}_X$, and that

$$\rho \preccurlyeq \tau \iff \mathfrak{T}_\rho \subseteq \mathfrak{T}_\tau.$$

The sequence $\mathfrak{T}_X$ is Π^0_1 in X, as one can see from the lemma above. However, the definition of global 1-true-stage ordering we gave above does not guarantee that it must have Turing degree X'.

Definition IX.3. We say that $\preccurlyeq$ is *complete* if, for every $X \in \mathbb{N}^{\mathbb{N}}$,

$$\mathfrak{T}_X \equiv_T X'$$

uniformly in X (i.e., there is a computable operator Γ such that $\Gamma^{\mathfrak{T}_X} = X'$ for all $X \in \mathbb{N}^{\mathbb{N}}$).

Recapitulating, given a complete global 1-true stage system, we can use $\mathfrak{T}_X$ as our $\Delta^0_2(X)$-Turing-complete oracle. In an X-computable construction, we can use the strings $\mathfrak{T}_\rho$ for $\rho \subseteq X$ as our guesses for initial segments of $\mathfrak{T}_X$. It is only at the X-true substrings that our guesses are going to be correct. Property ($\clubsuit$) will allow us to organize these guesses.

One can find examples of applications of $0'$-priority methods in [**Part 1**, Chapter VII]. In this chapter, we will use $\preccurlyeq$ as the building block for the systems of n- and η-true stages. Before that, we need to show that a complete global true-stage system exists.

Theorem IX.4. *There is a complete, global 1-true-stage system.*

Proof. To show that there is a complete, global 1-true-stage system $\preccurlyeq$, we use Lachlan's notion of *true stage* [**Lac73**]. Let K be a c.e. operator such that K^X is the Turing jump of X for all $X \in \mathbb{N}^{\mathbb{N}}$, i.e., $K^X = \{e : \Phi_e^X(e)\downarrow\}$. Assume that the operator K enumerates at most one element at each stage. For $\tau \in \mathbb{N}^{<\mathbb{N}}$, let K^τ be the finite set of elements enumerated by K using oracle τ in at most $|\tau|$ steps. So, if $\tau \subset X$, K^τ is an approximation to K^X. Not a good one, though. If we are trying to approximate a computation with oracle X', say Φ^{K^X}, then using Φ^{K^τ} may give us wrong answers for every τ. This is where Lachlan's true stages come in.

Let k_τ be the last number to get enumerated into K^τ. We then have that $K^\tau = \{k_\sigma : \sigma \subseteq \tau\}$. The numbers k_σ do not come in order though. Notice that the functions $\tau \mapsto k_\tau$ and $\tau \mapsto K^\tau$ are computable.

Lachlan's idea was to use $K^\tau \upharpoonright\!\!\upharpoonright k_\tau$, viewed as a binary string of length $k_\tau + 1$, as an approximation to X'.[†] The point is that this approximation must be correct infinitely often, as we will see below. Then, define

$$\tau \preccurlyeq \rho \qquad \iff \qquad \tau \subseteq \rho \ \ \& \ \ K^\tau \upharpoonright\!\!\upharpoonright k_\tau \subseteq K^\rho \upharpoonright\!\!\upharpoonright k_\rho,$$

where the inclusion is as strings. If K^τ is empty, let $k_\tau = -\infty$ and let $K^\tau \upharpoonright\!\!\upharpoonright k_\tau$ be the empty string. Notice that

$$\tau \preccurlyeq \rho \qquad \iff \qquad \forall \pi (\tau \subseteq \pi \subseteq \rho \Rightarrow k_\tau \leq k_\pi). \tag{5}$$

From Lemma IX.2, we have that τ is X-*true* if $\tau \subset X$ and $\tau \preccurlyeq \rho$ for all ρ with $\tau \subseteq \rho \subset X$. So, we have that $\tau \subset X$ is X-true if no element below k_τ is ever enumerated into K^X after stage $|\tau|$, or equivalently, if $K^\tau \upharpoonright\!\!\upharpoonright k_\tau$ is an initial segment of K^X (again, viewed as strings).

Let us now prove that $\preccurlyeq$ induces a complete, global true-stage system. Properties (TS0) and (TS1) are immediate from the definition. Let us show that $\preccurlyeq$ satisfies condition (TS2). We need to show that there are infinitely many X-true substrings. Fix $m \in \mathbb{N}$. We will find a substring τ of X of length larger than m such that $\tau \preccurlyeq \rho$ for all ρ with $\tau \subseteq \rho \subset X$ — this string τ will then be X-true. Let $\tau_0 = X \upharpoonright m$, and let k be the least element that is ever enumerated into K^X after stage m, that is, k is the least element of $X' \setminus K^{\tau_0}$, viewed as sets. Let τ be the smallest substring of X with $k \in K^\tau$ and $|\tau| > m$. We must then have $k_\tau = k$. Since no element below k_τ is ever enumerated into K^X after stage m, $K^\tau \upharpoonright\!\!\upharpoonright k_\tau$ is an initial segment of K^X, and τ is X-true.

To prove (♣), consider $\tau \subseteq \sigma \subseteq \rho$ such that $\tau \preccurlyeq \rho$. From (5), $\forall \pi (\tau \subseteq \pi \subseteq \rho \Rightarrow k_\tau \leq k_\pi)$. The same must then be true for any π between τ and σ. So, $\tau \preccurlyeq \sigma$.

[†] Recall that $\sigma \upharpoonright\!\!\upharpoonright k$ refers to the restriction of σ to $\{0, ..., k\}$.

Finally, to show that $\preccurlyeq$ is complete, just observe that

$$K^X = \bigcup \{K^\tau \upharpoonright\!\upharpoonright k_\tau : \tau \in \mathfrak{T}_X\},$$

where the union is the union of an increasing sequence of strings. It follows that X' is computable in $\mathfrak{T}_X$. □

IX.2. m-true-stage systems

The next step is to consider finite iterations of the previous construction. To simplify matters, let us go back to considering orderings on $\mathbb{N}$, as we did in [**Part 1**], instead of orderings in $\mathbb{N}^{<\mathbb{N}}$. The same ideas would work to produce orderings on $\mathbb{N}^{<\mathbb{N}}$, but the notation would get a bit more complicated.

An *m-true-stage system* is a computable family of partial orderings $\leq_n$ on $\mathbb{N}$, one for each $n \leq m$, that satisfies the following properties:

(TS0) $\leq_0$ is just the standard ordering on $\mathbb{N}$.

(TS1) The sequence of relations is *nested* (i.e., if $s \leq_{n+1} t$, then $s \leq_n t$).

(TS2) For every $n \leq m$, there exists an infinite $\leq_n$-increasing sequence

$$t_0 <_n t_1 <_n t_2 <_n \cdots .$$

(♣) For every $n < m$ and every $t < s < r$, if $t \leq_{n+1} r$ and $s \leq_n r$, then $t \leq_{n+1} s$.

$$\begin{tikzcd}
t \arrow[rr, dash, bend left, "<_{n+1}"] \arrow[r, dash, dotted, bend right, "<_{n+1}"'] & s \arrow[r, dash, bend right, "<_n"'] & r
\end{tikzcd}$$

Again, this last property (♣) is the key combinatorial property capturing how the approximations to the jumps behave. Notice that the case $n = 0$ of the (♣) property corresponds to the (♣) property from the previous section.

The idea behind the iteration is that once we have defined the ordering $\leq_n$, we can define the n-true stages, and then we define the ordering $\leq_{n+1}$ by considering the global ordering $\preccurlyeq$ along the n-true stages.

The following is a consequence of (♣) that will often be useful:

(◇) For every $k \leq m$, and every $t < s < r$, if $t \leq_k r$ and $s \leq_k r$, then $t \leq_k s$.

$$\begin{tikzcd}
t \arrow[rr, dash, bend left, "<_k"] \arrow[r, dash, dotted, bend right, "<_k"'] & s \arrow[r, dash, bend right, "<_k"'] & r
\end{tikzcd}$$

This follows from (♣) using $k = n+1$ and noticing that $s \leq_k r$ implies $s \leq_{k-1} r$.

Definition IX.5. A number t is said to be an *n-true stage* if it belongs to an infinite increasing $\leq_n$-chain. We use $\mathcal{T}^n$ to denote the sequence of n-true stages listed in increasing order.

Let us analyze the behavior of the n-true stages. First observe that by nestedness, if $n \leq k$, then the k-true stages are a sub-sequence of the n-true stages.

Lemma IX.6. *A number t is an n-true stage if and only if $t \leq_n s$ for all $(n-1)$-true stages $s > t$.*

For $n = 0$, all stages are 0-true stages. For $n = 1$, this follows from Lemma IX.2.

Proof. We prove the $(\Rightarrow)$ direction by induction on n. Consider an n-true stage t and an $(n-1)$-true stage $s > t$. Let $t <_n t_1 <_n t_2 <_n \cdots$ be a sequence witnessing that t is an n-true stage, and let k be such that $s < t_k$. By the induction hypothesis, since s is $(n-1)$-true and t_k is $(n-2)$-true, we get that $s \leq_{n-1} t_k$.[‡] Apply $(\clubsuit)$ on $t < s < t_k$ to obtain that $t \leq_n s$.

For the $(\Leftarrow)$ direction, consider a sequence $t_0 <_n t_1 <_n t_2 <_n \cdots$ as in (TS2). Notice that the t_k's are n-true stages, and in particular $(n-1)$-true stages. Let k be such that $t < t_k$. By the assumption, we must have $t \leq_n t_k$, and therefore $t <_n t_k <_n t_{k+1} <_n \cdots$, which witnesses that t is an n-true stage. □

Observation IX.7. If $s < t$ and t is an n-true stage, then s is an n-true stage if and only if $s \leq_n t$. To see this, suppose first that $s \leq_n t$. Since t is n-true, there is an increasing $\leq_n$-chain starting with t. We can then append s to the beginning of that chain to see that s is an n-true stage too. Conversely, suppose s is an n-true stage. Since n-true stages are also $(n-1)$-true, t is $(n-1)$-true, and hence, by the previous lemma, $s \leq_n t$.

Because of this observation, when $t \leq_n r$, we say that t *looks like an n-true stage at r* or that t is an *apparent n-true stage at r*. We thus define

$$\mathcal{T}^n_r = \langle t : t \leq_n r \rangle$$

as our approximation to $\mathcal{T}^n$ at r. We view $\mathcal{T}^n_r$ as a string listing the elements of the set $\{t : t \leq_n r\}$ in increasing order. When we compare different $\mathcal{T}^n_s$'s by inclusion, we will be referring to inclusion of strings. For instance, it is easy to see using $(\diamond)$ that

$$s \leq_n t \iff \mathcal{T}^n_s \subseteq \mathcal{T}^n_t,$$

[‡] In the case when $n = 1$, we get $s \leq_{n-1} t_k$ for free.

and that r is an n-true stage if and only if $\mathcal{T}^n_r \subset \mathcal{T}^n$. We let the reader verify these facts.

It also follows from the lemma above that $\mathcal{T}^n$ is Π^0_1 in $\mathcal{T}^{n-1}$. By induction, we then get that $\mathcal{T}^n$ is Π^0_n.

Definition IX.8. We say that an m-true-stage system is *complete* if, for each $n \leq m$, the set of n-true stages is Δ^0_{n+1}-Turing-complete.

Let us now build a complete m-true-stage system. The particular m-true-stage system one builds is not relevant for the applications of the system. All one needs to know in applications is that a complete m-true-stage system exists.

Theorem IX.9. *For every $m \in \mathbb{N}$, there exists a complete m-true-stage system.*

Proof. The construction is, of course, by recursion on m.

Let $\leq_0 = \leq_{\mathbb{N}}$ as in (TS0). Given $\leq_n$, we define

$$\mathcal{T}^n_r = \langle t : t \leq_n r \rangle \qquad \text{(Def. } \mathcal{T}^n_r\text{)}$$

and then let

$$s \leq_{n+1} t \iff \mathcal{T}^n_s \preccurlyeq \mathcal{T}^n_t, \qquad \text{(Def. } \leq_{n+1}\text{)}$$

where $\preccurlyeq$ is the ordering on strings defined in Theorem IX.4. To see nestedness, i.e. (TS1), notice that $\mathcal{T}^n_s \preccurlyeq \mathcal{T}^n_t$ implies $\mathcal{T}^n_s \subseteq \mathcal{T}^n_t$, which implies $s \leq_n t$.

The proof of (♣) for $\leq_n$ follows from (♣) for $\preccurlyeq$ as follows: Suppose that we have $t < s < r$ such that $s \leq_n r$ and $t \leq_{n+1} r$. We then have that $\mathcal{T}^n_s \subseteq \mathcal{T}^n_r$ and $\mathcal{T}^n_t \preccurlyeq \mathcal{T}^n_r$. Since both $\mathcal{T}^n_t$ and $\mathcal{T}^n_s$ are initial segments of $\mathcal{T}^n_r$ and $t < s$, we must have

$$\mathcal{T}^n_t \subseteq \mathcal{T}^n_s \subseteq \mathcal{T}^n_r.$$

Apply (♣) for $\preccurlyeq$ to get $\mathcal{T}^n_t \preccurlyeq \mathcal{T}^n_s$, and hence that $t \leq_{n+1} s$ as wanted.

Finally, we need to show that $\mathcal{T}^n$ is infinite and that $\mathcal{T}^n \equiv_T 0^{(n)}$. We prove this by induction. Since $\mathcal{T}^{n+1}$ is Π^0_{n+1}, we know that $\mathcal{T}^{n+1} \leq_T 0^{(n+1)}$. Let $X = \mathcal{T}^n$ and suppose we already know $\mathcal{T}^n \equiv_T 0^{(n)}$. Notice that for $t \in \mathcal{T}^n$, $X \upharpoonleft\upharpoonright t = \mathcal{T}^n_t$.[§] Therefore, for $t, s \in \mathcal{T}^n$,

$$t \leq_{n+1} s \text{ if and only if } X \upharpoonleft\upharpoonright t \preccurlyeq X \upharpoonleft\upharpoonright s.$$

We then have that $t \in \mathcal{T}^n$ is an $(n+1)$-true stage if and only if $X \upharpoonleft\upharpoonright t$ is X-true. In other words,

$$\boldsymbol{\mathcal{T}}_X = \langle \mathcal{T}^n_t : t \in \mathcal{T}^{n+1} \rangle.$$

[§] Here, by $X \upharpoonleft\upharpoonright t$ we mean $X \cap \{0, 1, ..., t\}$. If $t \in \mathcal{T}^n$, $\mathcal{T}^n \upharpoonleft\upharpoonright t$ is exactly $\{s : s \leq_n t\} = \mathcal{T}^n_t$.

Recall that, by Theorem IX.4, $\boldsymbol{\mathfrak{T}}_X \equiv_T X'$. We can use $\mathfrak{T}^{n+1}$ to compute X', which is $(\mathfrak{T}^n)'$, which, by the induction hypothesis, is Turing equivalent to $0^{(n+1)}$. □

IX.3. Pairs of structures

As we mentioned in the previous chapter, the pair of structures theorem of Ash and Knight [**AK90**] is one of the most useful applications of Ash and Knight's metatheorem. We already gave a proof of it using the game metatheorem (see Theorem VIII.7). In this section we give a more hands-on proof using the iterated true-stage method. The reason we give this second proof is that it is a good example to show the reader how the iterated true-stage method works. We only do the case for η finite, so that we do not have to introduce all the techniques at once. For infinite η, one need techniques that we will develop in Section IX.5. We already worked out the cases $\eta = 0$ and $\eta = 1$ in Section II.6.2. We recommend the reader to go through that section before reading this proof, and then to compare the proofs.

THEOREM IX.10. *Consider $n \in \mathbb{N}$ and let $\mathcal{A}_0$ and $\mathcal{A}_1$ be ω-presentations whose back-and-forth relations are computable up to n. If $\mathcal{A}_1 \leq_{n+1} \mathcal{A}_0$, then distinguishing the copies of $\mathcal{A}_1$ from the copies of $\mathcal{A}_0$ is Σ^0_{n+1}-hard.*

PROOF. Since the sequence of n-true stages $\mathfrak{T}^n$ is Δ^0_{n+1}-complete, there is a c.e. operator W such that the set $W^{\mathfrak{T}^n}$ is Σ^0_{n+1} complete. Uniformly computably in each $e \in \mathbb{N}$, we need to build a structure $\mathcal{B}$ that is isomorphic to $\mathcal{A}_1$ if $e \in W^{\mathfrak{T}^n}$ and isomorphic to $\mathcal{A}_0$ if $e \notin W^{\mathfrak{T}^n}$.¶ We fix such an e and, for each $s \in \mathbb{N}$, we use $\mathfrak{T}^n_s$ to define $X(s)$ as the stage-s approximation to that Σ^0_{n+1} complete question: That is,

$$X(s) = \begin{cases} 1 & \text{if } e \in W^{\mathfrak{T}^n_s} \\ 0 & \text{if } e \notin W^{\mathfrak{T}^n_s}. \end{cases} \quad \|$$

We also let $X(\infty) = 1$ if $e \in W^{\mathfrak{T}^n}$ and $X(\infty) = 0$ if $e \notin W^{\mathfrak{T}^n}$. So, we want $\mathcal{B}$ to be isomorphic to $\mathcal{A}_{X(\infty)}$.

Notice that, for $s \leq t$,

$$s \leq_n t \;\Rightarrow\; \mathfrak{T}^n_s \subseteq \mathfrak{T}^n_t \;\Rightarrow\; X(s) \leq X(t) \;\Rightarrow\; \mathcal{A}_{X(s)} \geq_{n+1} \mathcal{A}_{X(t)}. \quad (6)$$

Thus, if we restrict ourselves to the n-true stages, the value of $X(t)$ is either 0 for ever, or 1 from some point on. The problem is that

¶ If we want a reduction from a Σ^0_{n+1} subset of $2^{\mathbb{N}}$ instead of a subset of $\mathbb{N}$, we just relativize this proof.

‖ Recall the convention that if we have a finite oracle $\sigma \in 2^{<\mathbb{N}}$, when we consider W^σ, we only run it for $|\sigma|$ many stages.

we need to build $\mathcal{B}$ computably, so we do not know which stages are n-true: Whatever part of the diagram of $\mathcal{B}$ we build at a stage s we cannot change later, even if s is not n-true.

The construction consists of carefully choosing, at each stage s, a tuple $\bar{a}_s$ from $A_{X(s)}$, and at the end defining the diagram of $\mathcal{B}$ to be union of the diagrams of these tuples. We impose the following condition, which connects the apparent-m-true-stage relations and the back-and-forth relations. We call this condition (MC), as it is the *main condition* characteristic to all n-true stage arguments: For all $m \leq n$ and $s, t \in \mathbb{N}$,

$$\text{(MC)} \qquad s \leq_m t \qquad \Rightarrow \qquad (\mathcal{A}_{X(s)}, \bar{a}_s) \leq_m (\mathcal{A}_{X(t)}, \bar{a}_t).$$

If $s \leq_n t$ and $X(s) = X(t)$, we also require that $\bar{a}_s \subseteq \bar{a}_t$. One more thing: To make sure our enumeration is onto, we also require that the range of $\bar{a}_s$ includes the first s elements from $\mathcal{A}_{X(s)}$. We claim that that is all we need.

Verification: Suppose we manage to build such a sequence $\{\bar{a}_s : s \in \mathbb{N}\}$ computably. On one end, condition (MC) for the case $m = 0$ implies that

$$D_{\mathcal{A}_{X(0)}}(\bar{a}_0) \subseteq D_{\mathcal{A}_{X(1)}}(\bar{a}_1) \subseteq D_{\mathcal{A}_{X(2)}}(\bar{a}_2) \subseteq \cdots .$$

Hence, we get a computable limit ω-presentation $\mathcal{B}$ whose atomic diagram is the union of these diagrams:

$$D(\mathcal{B}) = \bigcup_{s \in \mathbb{N}} D_{\mathcal{A}_{X(s)}}(\bar{a}_s).$$

On the other end, condition (MC) at $m = n$ implies that, along the n-true stages, once $X(s)$ stabilizes to $X(\infty) \in \{0, 1\}$, $\mathcal{B}$ is built as the pull-back of an increasing sequence of tuples from $\mathcal{A}_{X(\infty)}$. Thus, $\mathcal{B}$ ends up being isomorphic to $\mathcal{A}_{X(\infty)}$. Let us explain this in more detail. Let t_0 be an n-true stage that is large enough so that $X(t_0) = X(\infty)$. Let $t_1 \leq_n t_2 \leq_n \cdots$ be the n-true stages after t_0. We then have that

$$\bar{a}_{t_0} \subseteq \bar{a}_{t_1} \subseteq \bar{a}_{t_2} \subseteq \cdots \subseteq \mathcal{A}_{X(\infty)},$$

and that $D(\mathcal{B}) = \bigcup_{j \in \mathbb{N}} D_{\mathcal{A}_{X(\infty)}}(\bar{a}_{t_j})$. Let $g \colon \mathbb{N} \to \mathcal{A}_{X(\infty)}$ be defined as $\bigcup_i \bar{a}_{t_i}$. So, $\mathcal{B}$ is the pull-back of $\mathcal{A}_{X(\infty)}$ through g, and hence it is isomorphic to $\mathcal{A}_{X(\infty)}$.

Notice that for the verifications, we only used condition (MC) for the cases $m = 0$ and $m = n$. The intermediate cases will be necessary to bridge the gap between those two during the construction.

Construction: We now need to show how to build a sequence of tuples satisfying (MC). At stage 0 let $\bar{a}_0$ be the empty tuple. Suppose we have already built $\bar{a}_0, ..., \bar{a}_{s-1}$ and we need to define $\bar{a}_s$. Each stage is divided in two steps:

(1) Fix the mistakes by previous stages.
(2) Incorporate the new information, namely the value $X(s)$, and change structures if necessary.

At each stage we act as if our current beliefs are correct (namely, as if $X(s) = X(\infty)$). Since we might have had different beliefs in the past, we might have acted under incorrect information and made some mistakes that we now need to fix. However, since we can never be sure our current belief is correct, we have to do it respecting at least some of the work done at previous stages. How much we respect the work done at a previous stage depends on how much we believe it — that is essentially what condition (MC) says. Just to provide some intuition, we can re-state (MC) as follows:

$$\mathfrak{T}^m_r \subseteq \mathfrak{T}^m_s \quad \text{implies} \quad \Pi^{\mathsf{c}}_m\text{-}tp_{\mathcal{A}_{X(r)}}(\bar{a}_r) \subseteq \Pi^{\mathsf{c}}_m\text{-}tp_{\mathcal{A}_{X(s)}}(\bar{a}_s),$$

which we can interpret as follows:

> If at a stage s we believe that our actions at a previous stage r used correct Π^0_m information (namely $\mathfrak{T}^m_r$), then at stage s, we should preserve all Π^{c}_m commitments made at stage r (namely $\Pi^{\mathsf{c}}_m\text{-}tp_{\mathcal{A}_{X(r)}}(\bar{a}_r)$).

Let us continue with the construction. For each s, we will also define a tuple $\bar{b}_s$ that belongs to $A_{X(t)}$ where t is the largest stage with $t <_n s$. The tuple $\bar{b}_s$ is used to fix the mistakes by previous stages. The tuple $\bar{b}_s$ will satisfy that, for every $m \leq n$ and every $r < s$,

$$r <_m s \quad \text{implies} \quad (\mathcal{A}_{X(r)}, \bar{a}_r) \leq_m (\mathcal{A}_{X(t)}, \bar{b}_s). \tag{7}$$

Thus, if we then define $\bar{a}_s$ satisfying

$$(\mathcal{A}_{X(t)}, \bar{b}_s) \leq_n (\mathcal{A}_{X(s)}, \bar{a}_s),$$

we will get property (MC).

For each $i \leq n$, let s_i be the greatest stage such that $s_i <_i s$. Notice that if $r <_i s$, then $r \leq_i s_i$ by ($\diamond$). So, to satisfy (7), we just need to ensure that

$$(\mathcal{A}_{X(s_m)}, \bar{a}_{s_m}) \leq_m (\mathcal{A}_{X(t)}, \bar{b}_s) \quad \text{for all } m \leq n. \tag{8}$$

It will be useful to note that the numbers s_j satisfy that $s_j \leq_j s_{j-1}$ for all $j \leq n$ as in the diagram below. To see that, just apply (♣) to

$s_j < s_{j-1} < s$.

$$\begin{tikzcd}
s_n \arrow[drrrrrrrr, dotted, dash, "\leq_n"'] & \leq_n & s_{n-1} \arrow[drrrrrr, dotted, dash, "\leq_{n1}"] & \leq_{n-1} & \cdots & \leq_2 & s_1 \arrow[drr, dotted, dash, "\leq_1"'] & \leq_1 & s_0 \arrow[d, dotted, dash, "\leq_0"] \\
 & & & & & & & & s
\end{tikzcd}$$

To find $\bar{b}_s$ such that $(\mathcal{A}_{X(s_i)}, \bar{a}_{s_i}) \leq_i (\mathcal{A}_{X(t)}, \bar{b}_s)$ for all $i \leq k$, we need the lemma below, which is often useful in constructions involving iterated true-stages systems. It was originally used by Ash and Knight, and it was key in all applications of their metatheorem.

Remark IX.11. Recall that the defining property of the back-and-forth relations is that, if $(\mathcal{A}, \bar{a}) \leq_{\beta+1} (\mathcal{B}, \bar{b})$, where $\bar{b}$ may be longer than $\bar{a}$, then there is a $\bar{c} \in A^{<\mathbb{N}}$ extending $\bar{a}$ with $(\mathcal{A}, \bar{c}) \geq_\beta (\mathcal{B}, \bar{b})$.

$$\begin{tikzcd}
(\mathcal{A}, \bar{a}) \arrow[d, dotted, dash, "\subseteq"'] \arrow[drr, dash, "\leq_{\beta+1}"] & & \\
(\mathcal{A}, \bar{c}) \arrow[rr, dotted, dash, "\geq_\beta"] & & (\mathcal{B}, \bar{b})
\end{tikzcd}$$

The following lemma is an iteration of this property.

Lemma IX.12. *Suppose we have a finite sequence of τ-structures $\mathcal{A}_0, ..., \mathcal{A}_n$, and tuples $\bar{a}_i \in A_i^{<\mathbb{N}}$ for $i \leq n$, such that*

$$(\mathcal{A}_n, \bar{a}_n) \ \leq_n \ (\mathcal{A}_{n-1}, \bar{a}_{n-1}) \ \leq_{n-1} \ \cdots \ \leq_2 \ (\mathcal{A}_1, \bar{a}_1) \ \leq_1 \ (\mathcal{A}_0, \bar{a}_0).$$

Then, there exists a tuple $\bar{c} \in A_n^{<\mathbb{N}}$ extending $\bar{a}_n$ such that $(\mathcal{A}_j, \bar{a}_j) \leq_j (\mathcal{A}_n, \bar{c})$ for all $j < n$.

$$\begin{tikzcd}
(\mathcal{A}_n, \bar{a}_n) \arrow[drr, dotted, dash, "\subseteq"'] & (\mathcal{A}_{n-1}, \bar{a}_{n-1}) \arrow[dr, dotted, dash, "\leq_{n-1}"] & \cdots & (\mathcal{A}_1, \bar{a}_1) \arrow[dl, dotted, dash, "\geq_1"'] & (\mathcal{A}_0, \bar{a}_0) \arrow[dll, dotted, dash, "\geq_0"] \\
 & & (\mathcal{A}_n, \bar{c}). & &
\end{tikzcd}$$

Proof. We will define a sequence of tuples $\bar{c}_j \in A_j^{<\mathbb{N}}$ extending $\bar{a}_j$ by induction on j as in the diagram below. Start with $\bar{c}_0 = \bar{a}_0$. Given $\bar{c}_j$, since $(\mathcal{A}_{j+1}, \bar{a}_{j+1}) \leq_{j+1} (\mathcal{A}_j, \bar{a}_j)$ and $\bar{a}_j \subseteq \bar{c}_j$, we have that $(\mathcal{A}_{j+1}, \bar{a}_{j+1}) \leq_{j+1} (\mathcal{A}_j, \bar{c}_j)$. Then, by the defining property of the back-and-forth relations mentioned above, we get that there exists a tuple

$\bar{c}_{j+1} \supseteq \bar{a}_{j+1} \in A_{j+1}^{<\mathbb{N}}$ such that $(\mathcal{A}_{j+1}, \bar{c}_{j+1}) \geq_j (\mathcal{A}_j, \bar{c}_j)$.

$$\begin{tikzcd}
(\mathcal{A}_n,\bar{a}_n) \arrow[r, phantom, "\leq_n"] \arrow[d, dash, dotted, "\subseteq"] \arrow[dr, dash, dotted, "\leq_n"] & (\mathcal{A}_{k-1},\bar{a}_{n-1}) \arrow[r, phantom, "\leq_{n-1}"] \arrow[d, dash, dotted, "\subseteq"] \arrow[dr, dash, dotted, "\leq_{n-1}"] & \cdots \arrow[r, phantom, "\leq_2"] \arrow[dr, dash, dotted, "\ddots\ \leq_2"] & (\mathcal{A}_1,\bar{a}_1) \arrow[r, phantom, "\leq_1"] \arrow[d, dash, dotted, "\subseteq"] \arrow[dr, dash, dotted, "\leq_1"] & (\mathcal{A}_0,\bar{a}_0) \arrow[d, dash, dotted, "="] \\
(\mathcal{A}_n,\bar{c}_n) \arrow[r, phantom, "\geq_{n-1}"] & (\mathcal{A}_{n-1},\bar{c}_{n-1}) \arrow[r, phantom, "\geq_{n-2}"] & \cdots \arrow[r, phantom, "\geq_1"] & (\mathcal{A}_1,\bar{c}_1) \arrow[r, phantom, "\geq_0"] & (\mathcal{A}_0,\bar{c}_0)
\end{tikzcd}$$

Using transitivity and nestedness of the back-and-forth relations, one can then easily prove that

$$(\mathcal{A}_i, \bar{c}_i) \geq_j (\mathcal{A}_j, \bar{a}_j) \quad \text{for all } i > j.$$

We end up with $\bar{c} = \bar{c}_n$ as needed. □

Let us continue on with the construction. Recall that the stages s_j satisfy $s_n \leq_n s_{n-1} \leq_{n-1} \cdots \leq_1 s_0$. Since we have been respecting (MC) so far throughout the construction, we know that

$$(\mathcal{A}_{X(s_n)}, \bar{a}_{s_n}) \leq_n (\mathcal{A}_{X(s_{n-1})}, \bar{a}_{s_{n-1}}) \leq_{n-1} \cdots \\ \cdots \leq_2 (\mathcal{A}_{X(s_1)}, \bar{a}_{s_1}) \leq_1 (\mathcal{A}_{X(s_0)}, \bar{a}_{s_0}).$$

These structures satisfy the assumptions needed to apply Lemma IX.12 above. We then obtain $\bar{b}_s$ satisfying (8) as the $\bar{c}$ from the lemma. We can find the tuple $\bar{b}_s$ computably because we are assuming that the back-and-forth relations are computable up to n. This finished the first step of the construction, namely *fixing the mistakes* of previous stages. The second step is to *incorporate new information*, namely $X(s)$.

Here is where the assumption that $\mathcal{A}_1 \geq_{n+1} \mathcal{A}_0$ comes into play. Since $t = s_n \leq_n s$, we know from (6) that $\mathcal{A}_{X(s_n)} \geq_{n+1} \mathcal{A}_{X(s)}$. One can then find a tuple $\bar{a}_s$ in $\mathcal{A}_{X(s)}$ so that $(\mathcal{A}_{X(s_n)}, \bar{b}_s) \leq_n (\mathcal{A}_{X(s)}, \bar{a}_s)$.

To ensure onto-ness, extend $\bar{a}_s$ to make sure that it contains the first s elements of the ω-presentation $\mathcal{A}_{X(s)}$.

$$\begin{tikzcd}
(\mathcal{A}_{X(s_n)},\bar{a}_{s_n}) \arrow[r, phantom, "\leq_n"] \arrow[drrr, dash, dotted, "\subseteq"] & (\mathcal{A}_{X(s_{n-1})},\bar{a}_{s_{n-1}}) \arrow[r, phantom, "\leq_{n-1}"] \arrow[drr, dash, dotted, "\leq_{n-1}"] & \cdots \arrow[r, phantom, "\leq_2"] & (\mathcal{A}_{X(s_1)},\bar{a}_{s_1}) \arrow[r, phantom, "\leq_1"] \arrow[d, dash, dotted, "\leq_1"] & (\mathcal{A}_{X(s_0)},\bar{a}_{s_0}) \arrow[dl, dash, dotted, "\leq_0"] \\
 & & & (\mathcal{A}_{X(s_n)},\bar{b}_s) \arrow[r, phantom, "\leq_n"] & (\mathcal{A}_{X(s)},\bar{a}_s)
\end{tikzcd}$$

We have thus defined $\bar{a}_s$ satisfying (MC) as needed. □

IX.4. Transfinite true-stage systems

We now want to iterate $\preccurlyeq$ through the transfinite. The definition will not be much more complicated, but the construction of a complete system will, as the limit case poses new difficulties that were not present at the finite case. Let η be a computable ω-presentation of an ordinal. An *η-true-stage system* is a computable family $\{\leq_\xi \colon \xi \leq \eta\}$ of partial orderings on $\mathbb{N}$ that satisfies the following properties:

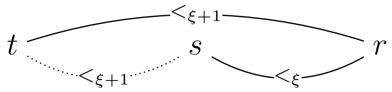
(TS0) $\leq_0$ is just the standard ordering on $\mathbb{N}$.

(TS1) The sequence of relations is *nested*, i.e., if $\gamma \leq \xi$ and $s \leq_\xi t$, then $s \leq_\gamma t$.

(TS2) For every ξ, there exists an infinite $\leq_\xi$-increasing sequence.

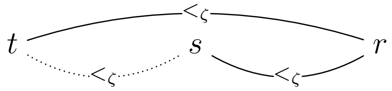
(TS3) The sequence of relations is *continuous*, i.e., if λ is a limit ordinal, then $s \leq_\lambda t \iff (\forall \xi < \lambda)\ s \leq_\xi t$.

(♣) For every $\xi < \eta$ and every $t < s < r$, if $t \leq_{\xi+1} r$ and $s \leq_\xi r$, then $t \leq_{\xi+1} s$.



The following is a consequence of (♣) that will be useful later:

(◇) For every $\zeta \leq \eta$ and every $t < s < r$, if $t \leq_\zeta r$ and $s \leq_\zeta r$, then $t \leq_\zeta s$.



The successor case follows from (♣) using $\zeta = \xi + 1$, and noticing that $s \leq_\zeta r$ implies $s \leq_{\zeta-1} r$. For the limit case, consider any $\xi < \zeta$ and use (♣) and that $\leq_\zeta$ implies both $\leq_\xi$ and $\leq_{\xi+1}$, to conclude that $t \leq_{\xi+1} s$. Then, by continuity, $t \leq_\zeta s$.

Definition IX.13. For each $\xi \leq \eta$, we say that t is a *ξ-true stage* if it belongs to an infinite $\leq_\xi$-increasing sequence. Let $\mathcal{T}^\xi \in \mathbb{N}^\mathbb{N}$ be the sequence of all ξ-true stages listed in increasing order.

Let us analyze the behavior of the ξ-true stages. First observe that by nestedness, if $\zeta \leq \xi$, then the ξ-true stages are a sub-sequence of the ζ-true stages.

Observation IX.14. If $s \leq_\xi t$ and t is a ξ-true stage, then s is a ξ-true stage too. To see this, just append s at the beginning of the increasing $\leq_\xi$-chain starting with t.

Lemma IX.15. *For all successor ordinals, $\xi + 1$, we have that*

$$t \text{ is a } \xi+1\text{-true stage} \iff t \leq_{\xi+1} s \text{ for all } \xi\text{-true stages } s \geq t.$$

For limit ordinals λ, t is a λ-true stage if and only if it is a ξ-true stage for all $\xi < \lambda$.

All stages are 0-true stages.

Proof. We use transfinite induction on ξ. The successor case works exactly as the proof of Lemma IX.6, so we won't repeat it here. Consider the limit case.

The ($\Rightarrow$) direction follows directly from the nesting condition. For the ($\Leftarrow$) direction, consider a stage t which is ξ-true for all $\xi < \lambda$. Let $s > t$ be a λ-true stage, and in particular a ξ-true stage for all $\xi < \lambda$. Then, $t \leq_\xi s$ for all $\xi < \lambda$ by the induction hypothesis. By continuity, we get that $t \leq_\lambda s$, and hence t is a λ-true stage too, by the observation above. □

Observation IX.16. If $s < t$ and t is a ξ-true stage, then s is a ξ-true stage if and only if $s \leq_\xi t$. We already noted the ($\Leftarrow$) direction above. For the other direction, suppose s is a ξ-true stage. By nestedness, t is a ζ-true stage for all $\zeta < \xi$, and hence by the previous lemma, $s \leq_{\zeta+1} t$. By either letting $\zeta = \xi - 1$ if ξ is a successor, or using continuity if ξ is a limit, we get $s \leq_\xi t$.

We say that s is an *apparent ξ-true stage at t* if $s \leq_\xi t$. Given ξ and t, we define the *stage-t approximation to $\mathcal{T}^\xi$*, denoted $\mathcal{T}^\xi_t$, as the tuple enumerating the apparent ξ-true stages at t:

$$\mathcal{T}^\xi_t = \langle s : s \leq_\xi t \rangle.$$

Note that using ($\diamond$), we get

$$s \leq_\xi t \iff \mathcal{T}^\xi_s \subseteq \mathcal{T}^\xi_t,$$

where the inclusion is as strings, and that

$$t \text{ is } \xi\text{-true} \iff \mathcal{T}^\xi_t \subseteq \mathcal{T}^\xi.$$

Using Lemma IX.15, one can show by transfinite induction that $\mathcal{T}^\xi$ is Π^0_ξ uniformly in ξ.

Definition IX.17. We say that an η-true-stage system is *complete* if $\mathcal{T}^\xi$ is $\Delta^0_{\xi+1}$-Turing-complete for all $\xi \leq \eta$, uniformly in ξ.

Theorem IX.18. *There exists a complete η-true-stage system.*

The proof is significantly more difficult than the proof of the finite iterates that we gave in previous sections. In any case, for applications of the η-true-stage method, it does not matter how the system is built. All that matters is that such a system exists. One does not need to know the proof to be able to apply it, so we leave it to the end of the chapter (Sections IX.8 and IX.9).

IX.5. The key lemmas

In this section we prove two lemmas that are key to most applications of the iterated true stages method.

Let us spend a few paragraphs motivating these lemmas. It may help the reader have the proof of the Pair of Structures theorem from Section IX.3 in mind to understand the motivation. The reader should then try to think of how to extend the proof of Theorem IX.10 to transfinite values of n.

When working with an η-system of true stages, there is always some version of the *main condition* that we are trying to satisfy.

$$\text{(MC)}\ (\forall \xi \leq \eta)(\forall s, t \in \mathbb{N})\quad s \leq_\xi t \quad \Rightarrow \quad (\mathcal{A}_{X(s)}, \bar{a}_s) \leq_\xi (\mathcal{A}_{X(t)}, \bar{a}_t).$$

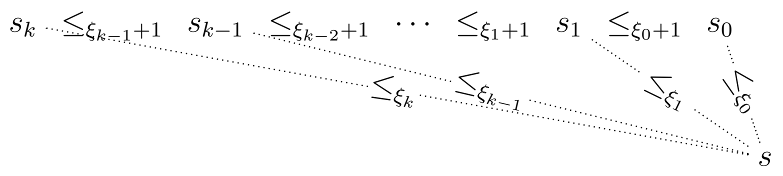
At each stage s, the first step is always to fix the mistakes made by previous stages. For that, we look for a tuple $\bar{b}_s \in A_{X(t)}^{<\mathbb{N}}$ that satisfies that, for every $\xi \leq \eta$ and every $r < s$,

$$r <_\xi s \quad \text{implies} \quad (\mathcal{A}_{X(r)}, \bar{a}_r) \leq_\xi (\mathcal{A}_{X(t)}, \bar{b}_s), \tag{9}$$

where t is the largest stage with $t <_\eta s$. In the case when η is finite as in Section IX.3, we did not need to worry about all r's in equation (9), but only about a few ones, namely $s_0, ..., s_n$ satisfying $s_n \leq_n s_{n-1} \leq_{n-1} \cdots \leq_1 s_0 < s$. When η is infinite, the situation is a bit more complicated. The first lemma will allow us to restrict property (9) to only a few values of r and a few ordinals ξ.

For each $\gamma \leq \eta$, let $s(\gamma)$ be the greatest stage such that $s(\gamma) <_\gamma s$. Notice that if $r <_\gamma s$, then $r \leq_\gamma s(\gamma)$ by ($\diamond$). So, to satisfy (9), we just need to ensure that, for all $\gamma \leq \eta$, $(\mathcal{A}_{X(s(\gamma))}, \bar{a}_{s(\gamma)}) \leq_\gamma (\mathcal{A}_{X(t)}, \bar{b}_s)$. We need to find these stages $s(\gamma)$ for all $\gamma \leq \eta$. Notice that if $\xi \geq \gamma$, then $s(\xi) \leq s(\gamma)$ by the nesting property. So, $\langle s(\xi) : \xi \leq \eta \rangle$ is a non-increasing sequence starting from $s(0) = s-1$ and ending with $s(\eta) = t$ as in Figure IX.1. There are only finitely many stages below s, so the values of $s(\xi)$ must repeat a lot.

Lemma IX.19. *For each $s \in \mathbb{N}$, there exist stages $s_k < s_{k-1} < \cdots < s_1 < s_0 = s - 1$ and ordinals $\eta = \xi_k > \xi_{k-1} > \cdots \xi_1 > \xi_0 = 0$ as in the diagram below satisfying the following condition: For all $i \leq k$ and all γ with $\xi_{i-1} < \gamma \leq \xi_i$,*[**] *we have that if $r <_\gamma s$ then $r \leq_\gamma s_i$.*



[**] taking $\xi_{-1} = -1$

Furthermore, all these objects can be found computably.

Proof. The set $\{s(\xi) : \xi \leq \eta\}$ is finite, as all its members are below s. Let $s_0, ..., s_k$ be the elements of this set listed in decreasing order. For each $i \leq k$, there is an interval of ξ's such that $s_i = s(\xi)$. Let ξ_i be the greatest ξ for which $s_i = s(\xi)$. Notice that ξ_i is the greatest such that $s_i \leq_{\xi_i} s$, and such a maximum exists by the continuity of the relations $\leq_\xi$. So, we have that $s(\gamma) = s_{i+1}$ for all γ with $\xi_i < \gamma \leq \xi_{i+1}$.

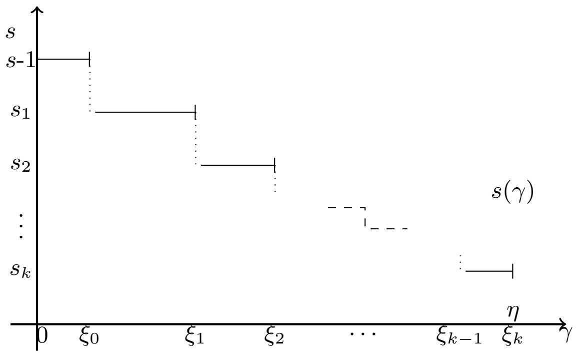


Figure IX.1. Illustration of the graph of the function $s\colon \{0, ..., \eta\} \to \{0, ..., s-1\}$.

Let us observe that s_i and ξ_i can be found computably by recursion: First, notice that s_0 is just $s - 1$. Then, given s_i, we can find ξ_i as the unique $\xi \leq \eta$ that satisfies $s_i \leq_\xi s$ and $s_i \not\leq_{\xi+1} s$. Then, we can find s_{i+1} as the greatest such that $s_{i+1} \leq_{\xi_i+1} s$, because s_{i+1} is the greatest number in $\{s(\xi) : \xi_i + 1 \leq \xi \leq \eta\}$.

The top line of the diagram in the statement of the lemma holds because, by (♣) applied to $s_{j+1} < s_j < s$, we get that $s_{j+1} \leq_{\xi_j+1} s_j$ for all $j < k$.

If $r <_\gamma s$, then we know $r \leq_\gamma s(\gamma)$ and $s(\gamma) = s_i$ for the least i with $\gamma \leq \xi_i$. □

This lemma is usually used together with the following transfinite version of Lemma IX.12.

Lemma IX.20. *Suppose we have a finite sequence of τ-structures $\mathcal{A}_0, ..., \mathcal{A}_\ell$, ordinals $\xi_{\ell-1} > \cdots > \xi_1 > \xi_0$, and tuples $\bar{a}_i \in A_i^{<\mathbb{N}}$ for $i \leq \ell$, such that*

$$(\mathcal{A}_\ell, \bar{a}_\ell) \leq_{\xi_{\ell-1}+1} (\mathcal{A}_{\ell-1}, \bar{a}_{\ell-1}) \leq_{\xi_{\ell-2}+1} \cdots$$
$$\cdots \leq_{\xi_1+1} (\mathcal{A}_1, \bar{a}_1) \leq_{\xi_0+1} (\mathcal{A}_0, \bar{a}_0).$$

Then, there exists a tuple $\bar{c} \in A_\ell^{<\mathbb{N}}$ extending $\bar{a}_\ell$ such that $(\mathcal{A}_j, \bar{a}_j) \leq_{\xi_j} (\mathcal{A}_\ell, \bar{c})$ for all $j < \ell$.

$$\begin{tikzcd}
(\mathcal{A}_\ell, \bar{a}_\ell) & (\mathcal{A}_{\ell-1}, \bar{a}_{\ell-1}) & \cdots & (\mathcal{A}_1, \bar{a}_1) & (\mathcal{A}_0, \bar{a}_0) \\
 & & (\mathcal{A}_\ell, \bar{c}). \arrow[ull, dotted, no head, "\subseteq"] \arrow[ul, dotted, no head, "\leq_{\xi_{\ell-1}}"'] \arrow[ur, dotted, no head, "\leq_{\xi_1}"] \arrow[urr, dotted, no head, "\leq_{\xi_0}"'] & &
\end{tikzcd}$$

Proof. The proof is essentially the same as that of Lemma IX.12. We repeat it, as we need to change the indices all over the place.

We will define a sequence of tuples $\bar{c}_j \in A_j^{<\mathbb{N}}$ extending $\bar{a}_j$ by induction on j. Start with $\bar{c}_0 = \bar{a}_0$. Given $\bar{c}_j$, since $(\mathcal{A}_{j+1}, \bar{a}_{j+1}) \leq_{\xi_j+1} (\mathcal{A}_j, \bar{a}_j)$ and $\bar{a}_j \subseteq \bar{c}_j$, we have that $(\mathcal{A}_{j+1}, \bar{a}_{j+1}) \leq_{\xi_j+1} (\mathcal{A}_j, \bar{c}_j)$. Then, by the defining property of the back-and-forth relations (namely Remark IX.11), we get that there exists a $\bar{c}_{j+1} \supseteq \bar{a}_{j+1} \in A_{j+1}^{<\mathbb{N}}$ such that $(\mathcal{A}_{j+1}, \bar{c}_{j+1}) \geq_{\xi_j} (\mathcal{A}_j, \bar{c}_j)$.

$$\begin{tikzcd}
(\mathcal{A}_\ell, \bar{a}_\ell) \arrow[dd, dotted, no head, "\subseteq"] \arrow[ddrr, dotted, no head, "\leq_{\xi_{\ell-1}+1}"] & \leq_{\xi_{\ell-1}+1} & (\mathcal{A}_{\ell-1}, \bar{a}_{\ell-1}) \arrow[dd, dotted, no head, "\subseteq"] \arrow[drr, dotted, no head, "\leq_{\xi_{\ell-2}+1}"] & \leq_{\xi_{\ell-2}+1} & \cdots & \leq_{\xi_1+1} & (\mathcal{A}_1, \bar{a}_1) \arrow[dd, dotted, no head, "\subseteq"] \arrow[ddrr, dotted, no head, "\leq_{\xi_0+1}"] & \leq_{\xi_0+1} & (\mathcal{A}_0, \bar{a}_0) \arrow[dd, dotted, no head, "="] \\
 & & & & \ddots \arrow[drr, dotted, no head, "\leq_{\xi_1+1}"] & & & & \\
(\mathcal{A}_\ell, \bar{c}_\ell) & \geq_{\xi_{\ell-1}} & (\mathcal{A}_{\ell-1}, \bar{c}_{\ell-1}) & \geq_{\xi_{\ell-2}} & \cdots & \geq_{\xi_1} & (\mathcal{A}_1, \bar{c}_1) & \geq_{\xi_0} & (\mathcal{A}_0, \bar{c}_0)
\end{tikzcd}$$

One can then easily prove that

$$(\mathcal{A}_i, \bar{c}_i) \geq_{\xi_j} (\mathcal{A}_j, \bar{a}_j) \quad \text{for all } i > j.$$

We end up with $\bar{c} = \bar{c}_\ell$ as needed. □

IX.6. The tree-of-structures theorem

In this section, we prove a version of the Ash–Knight's pair-of-structures theorem (Theorem VIII.7) where instead of having two structures to choose from, we have a whole tree of structures. This new version can be proved using the iterated true-stage method — as we will see below — but cannot be proved using either the game metatheorem or Ash-Knight's η-system metatheorem. The reason is that, during the construction, it requires guesses to oracles at all levels and not just at the highest level.

We include it in this book because it is another good example to show how iterated true-stage arguments work, and it is slightly more involved than the previous argument. The tree-of-structures theorem is the key lemma in [**Mon16**] to show that certain classes of structures are on top for effective reducibility (as defined in Section XI.4) and to give evidence that suggests that the classes of structures that are intermediate for effective reducibility are exactly the counterexamples to Vaught's conjecture (see Section XII.3).

Fix a computable ordinal η.

Definition IX.21. Let $2^{\circ\eta}$ be the set of all binary sequences $\sigma \in 2^\eta$ with only finitely many 1's.

Notice that $2^{\circ\eta}$ is countable and has a computable ω-presentation, as opposed to 2^η, which has size continuum for infinite η.

Definition IX.22. We say that a sequence $\sigma \in 2^{\circ\eta}$ is $\Sigma^0_{\xi\mapsto\xi+1}$ if deciding whether $\sigma(\xi) = 1$ is $\Sigma^0_{\xi+1}$ for all $\xi < \eta$, uniformly in ξ, or in other words, if there is a c.e. operator W_e, such that $\sigma(\xi) = 1 \iff 0 \in W_e^{\mathcal{T}^\xi}$, where $\mathcal{T}^\xi$ is a $\Delta^0_{\xi+1}$-Turing-complete real.

We call e a $\Sigma^0_{\xi\mapsto\xi+1}$-*index* for σ.

The reason these sequences were useful in [**Mon16**] is that, for each Σ^1_1-equivalence relation $\sim$ on ω, there exists a uniformly $\Sigma^0_{\xi\mapsto\xi+1}$ sequence $\{\sigma_n : n \in \omega\} \subseteq 2^{\circ\mathcal{H}}$ such that $n \sim m \iff \sigma_n{\restriction}\omega_1^{CK} = \sigma_m{\restriction}\omega_1^{CK}$. We will not delve deeper than this in this book. Such sequences were also used by Feiner [**Fei70**] and Thurber [**Thu94**] to build interesting Boolean algebras and by Hirschfeldt, Kach, and Montalbán [**HKM**] to study the notion of low for Δ-Feiner.

Definition IX.23. An *η-tree of structures* is a sequence of structures $\{\mathcal{A}_\sigma : \sigma \in 2^{\circ\eta}\}$ such that, for every $\sigma, \tau \in 2^{\circ\eta}$ and $\xi \leq \eta$, we have that

$$\sigma \restriction \xi = \tau \restriction \xi \quad \Rightarrow \quad \mathcal{A}_\sigma \equiv_{\xi+1} \mathcal{A}_\tau.$$

Example IX.24. Here is an example of an η-tree of structures on which one could apply the theorem below. Let $\mathcal{A}_\sigma$ be the linear ordering

$$\omega^{\alpha_1} \cdot \omega^* + \omega^{\alpha_2} \cdot \omega^* + \cdots + \omega^{\alpha_k} \cdot \omega^*$$

where $\alpha_1 < \cdots < \alpha_k$ are the ordinals α for which we have $\sigma(\alpha) = 1$. Using Corollary II.39, it is not hard to see that if $\sigma \restriction \xi = \tau \restriction \xi$, then $\mathcal{A}_\sigma \equiv_{\xi+1} \mathcal{A}_\tau$. The results from [**Mon16**] would then allow one to show that linear orderings are on top for effective reducibility as in Section XI.4.

Theorem IX.25 ([**Mon14**, Theorem 5.3]). *Let $\{\mathcal{A}_\sigma : \sigma \in 2^{\circ\eta}\}$ be a computable η-tree of structures where the back-and-forth relations are computable up to η. There is a computable procedure that, given a $\Sigma^0_{\xi\mapsto\xi+1}$-index for a sequence $\sigma \in 2^{\circ\eta}$, produces a computable structure $\mathcal{C}$ isomorphic to $\mathcal{A}_\sigma$.*

Proof. Let W be a c.e. operator such that $\sigma(\xi) = 1 \iff 0 \in W^{\mathcal{T}^\xi}$. For each s, we will define a sequence $\tau_s \in 2^{\circ\eta}$ as the stage-s approximation to σ. For each $\gamma < \eta$, let n_γ be the natural number that is in position γ in the given ω-presentation of η. We define τ_s computably as follows:

$$\tau_s(\xi) = 1 \iff 0 \in W^{\mathcal{T}^\xi_s} \quad \& \quad s > n_\xi.$$

Recall that, by our convention, when we consider a finite oracle ρ in W^ρ, we just run it for $|\rho|$ many stages. Since $n_\xi < s$ for only finitely many ξ's, we have that $\tau_s(\xi) = 1$ for only finitely many ξ's, so $\tau_s \in 2^{\circ\eta}$. We note that if $\mathcal{T}^\eta_t$ is correct and t is large enough so that, for all ξ with $\sigma(\xi) = 1$, we have $0 \in W^{\mathcal{T}^\xi_t}$ and $n_\xi < t$, then since all the $\mathcal{T}^\xi_t$ are correct, we must have $\tau_t = \sigma$. In other words, $\tau_t = \sigma$ for all large enough η-true stages t.

The construction consists of carefully choosing, at each stage s, a tuple $\bar{a}_s$ from A_{τ_s}. We impose the following condition, which we call (MC) and contains a small modification from the *main condition* of construction in the pair-of-structures theorem: For all $\xi \leq \eta$ and $r, s \in \mathbb{N}$,

$$\text{(MC)} \qquad r \leq_\xi s \quad \& \quad \tau_r \restriction \xi = \tau_s \restriction \xi \quad \Rightarrow \quad (\mathcal{A}_{\tau_r}, \bar{a}_r) \leq_\xi (\mathcal{A}_{\tau_s}, \bar{a}_s).$$

If $r \leq_\eta s$ and $\tau_r = \tau_s$, we also require that $\bar{a}_r \subseteq \bar{a}_s$. One more thing: To make sure our enumeration is onto, we also require that the range of $\bar{a}_s$ includes the first s elements from $\mathcal{A}_{\tau_s}$. We claim that this is all we need.

Verification: Suppose we manage to build such a computable sequence of tuples $\{\bar{a}_s : s \in \mathbb{N}\}$ satisfying (MC). On one end, condition (MC) for the case $\xi = 0$ implies that

$$D_{\mathcal{A}_{\tau_0}}(\bar{a}_0) \subseteq D_{\mathcal{A}_{\tau_1}}(\bar{a}_1) \subseteq D_{\mathcal{A}_{\tau_2}}(\bar{a}_2) \subseteq \dots.$$

Hence, we get a computable limit ω-presentation $\mathcal{B}$ whose atomic diagram is the union of these diagrams:

$$D(\mathcal{B}) = \bigcup_{s \in \mathbb{N}} D_{\mathcal{A}_{\tau_s}}(\bar{a}_s).$$

On the other end, condition (MC) at $\xi = \eta$ implies that, along the η-true stages, once τ_s stabilizes to $\sigma \in 2^{\circ\eta}$, $\mathcal{B}$ is built as the pull-back of an increasing sequence of tuples from $\mathcal{A}_\sigma$. Thus, $\mathcal{B}$ ends up being isomorphic to $\mathcal{A}_\sigma$. Let us explain this in more detail. Let t_0 be an

η-true stage that is large enough so that $\tau_{t_0} = \sigma$. Let $t_1 \leq_\eta t_2 \leq_\eta \cdots$ be the η-true stages after t_0. We then have that

$$\bar{a}_{t_0} \subseteq \bar{a}_{t_1} \subseteq \bar{a}_{t_2} \subseteq \cdots \subseteq A_\sigma$$

and that $D(\mathcal{B}) = \bigcup_{j\in\mathbb{N}} D_{\mathcal{A}_\sigma}(\bar{a}_{t_j})$. Let $g\colon \mathbb{N} \to A_\sigma$ be defined as $\bigcup_i \bar{a}_{t_i}$. So, $\mathcal{B}$ is the pull-back of $\mathcal{A}_\sigma$ through g, and hence it is isomorphic to $\mathcal{A}_\sigma$.

Notice that, so far, we have only used condition (MC) for the cases $\xi = 0$ and $\xi = \eta$. The intermediate cases will be necessary to bridge the gap between those two during the construction.

Construction: We now need to show how to build a sequence of tuples satisfying (MC). At stage 0, let $\bar{a}_0$ be the empty tuple. Suppose we have already built $\bar{a}_0, ..., \bar{a}_{s-1}$ and that we need to define $\bar{a}_s$.

Fix s. Let ζ be the largest ordinal for which there is a stage t satisfying

$$t <_\zeta s \quad \& \quad \tau_t \restriction \zeta = \tau_s \restriction \zeta. \tag{10}$$

Let t be the largest stage satisfying (10) for this ζ. Our first objective is to define a tuple $\bar{b}_s$ that belongs to A_{τ_t}. The tuple $\bar{b}_s$ will satisfy that, for every $\xi \leq \zeta$ and every $r < s$,

$$r <_\xi s \quad \& \quad \tau_r \restriction \xi = \tau_s \restriction \xi \quad \text{implies} \quad (\mathcal{A}_{\tau_r}, \bar{a}_r) \leq_\xi (\mathcal{A}_{\tau_t}, \bar{b}_s). \tag{11}$$

Thus, if at stage s we define $\bar{a}_s$ satisfying

$$(\mathcal{A}_{i_t}, \bar{b}_s) \leq_\zeta (\mathcal{A}_{i_s}, \bar{a}_s),$$

we will get property (MC). To define $\bar{b}_s$, we need to use Lemmas IX.19 and IX.20, as is often the case in constructions involving iterated true-stage systems. The first lemma is used to restrict property (11) to only a few values of r and a few ordinals ξ, and the second lemma will give us $\bar{b}_s$.

Apply Lemma IX.19 to get stages $s_k < s_{k-1} < \cdots < s_1 < s_0 = s-1$ and ordinals $\eta = \xi_k > \xi_{k-1} > \cdots > \xi_1 > \xi_0 = 0$.

Recall that we defined ζ and t as the maximum pair $\langle \zeta, t\rangle$ such that $t <_\zeta s$ and $\tau_t \restriction \zeta = \tau_s \restriction \zeta$. Let $\ell \leq k$ be the least with $\zeta \leq \xi_\ell$. So, we have that $t \leq_\zeta s_\ell$. This implies that, for all $\xi \leq \zeta$, $\mathcal{T}^\xi_t \subseteq \mathcal{T}^\xi_{s_\ell} \subseteq \mathcal{T}^\xi_s$ and hence that $\tau_t(\xi) \leq \tau_{s_\ell}(\xi) \leq \tau_s(\xi)$.[††] Since $\tau_t \restriction \zeta = \tau_s \restriction \zeta$, we also get that $\tau_{s_\ell} \restriction \zeta = \tau_s \restriction \zeta$. By the maximality of t, we get that $t = s_\ell$.

For $i < \ell$, we have $t = s_\ell \leq_{\xi_i} s_i \leq_{\xi_i} s$, and hence by the same argument we get that, for all $\xi \leq \xi_i$,

$$\tau_t(\xi) \leq \tau_{s_i}(\xi) \leq \tau_s(\xi) = \tau_t(\xi).$$

[††] Here we are just using the natural ordering on $\{0, 1\}$.

In particular, for all $i < \ell$, $\tau_{s_{i+1}} \restriction \xi_i + 1 = \tau_{s_i} \restriction \xi_i + 1$. We will use this below the claim to get that since we have been satisfying (MC) so far, we have $(\mathcal{A}_{\tau_{s_{i+1}}}, \bar{a}_{s_{i+1}}) \leq_{\xi_i+1} (\mathcal{A}_{\tau_{s_i}}, \bar{a}_{s_i})$.

Claim: To satisfy property (11), it is enough to get $\bar{b}_s$ so that

$$(\mathcal{A}_{\tau_{s_j}}, \bar{a}_{s_j}) \leq_\xi (\mathcal{A}_{\tau_t}, \bar{b}_s) \quad \text{for all } j \leq \ell. \tag{12}$$

Here is the proof of the claim: Notice that if $r <_\xi s$ and $\tau_r \restriction \xi = \tau_s \restriction \xi$, then, first we must have $\xi \leq \zeta$, and then $r \leq_\xi s_i$ for the least i with $\xi \leq \xi_i$. Since $\xi \leq \zeta$, we must have $i \leq \ell$. Since $\tau_{s_i} \Vert \xi_i = \tau_s \Vert \xi_i$, we must have $\tau_r \restriction \xi = \tau_{s_i} \restriction \xi$. Since we have been satisfying property (MC) so far, we have $(\mathcal{A}_{\tau_r}, \bar{a}_r) \leq_\xi (\mathcal{A}_{\tau_{s_i}}, \bar{a}_{s_i})$. Since $(\mathcal{A}_{\tau_{s_i}}, \bar{a}_{s_i}) \leq_{\xi_i} (\mathcal{A}_{\tau_t}, \bar{b}_s)$, we have $(\mathcal{A}_{\tau_r}, \bar{a}_r) \leq_\xi (\mathcal{A}_{\tau_t}, \bar{b}_s)$ as needed for property (11).

As we have been following (MC) so far up to this stage, we have

$$(\mathcal{A}_{\tau_{s_\ell}}, \bar{a}_{s_\ell}) \leq_{\xi_{\ell-1}+1} \cdots \leq_{\xi_1+1} (\mathcal{A}_{\tau_{s_1}}, \bar{a}_{s_1}) \leq_{\xi_0+1} (\mathcal{A}_{\tau_{s_0}}, \bar{a}_{s_0}).$$

Apply Lemma IX.20 to this sequence of structures to get $\bar{b}_s$ satisfying (12) as in the diagram below:

$$\begin{tikzcd}
(\mathcal{A}_{\tau_{s_t}}, \bar{a}_t) & \leq_{\xi_{\ell-1}+1} \cdots \leq_{\xi_1+1} & (\mathcal{A}_{\tau_{s_1}}, \bar{a}_{s_1}) & \leq_{\xi_0+1} & (\mathcal{A}_{\tau_{s_0}}, \bar{a}_{s_0}) \\
 & & (\mathcal{A}_{\tau_t}, \bar{b}_s) \arrow[ull, dotted, no head, "\subseteq"] \arrow[u, dotted, no head, "\leq_{\xi_1}"] \arrow[urr, dotted, no head, "\leq_{\xi_0}"'] & \leq_\zeta & (\mathcal{A}_{\tau_s}, \bar{a}_s)
\end{tikzcd}$$

We can find the tuple $\bar{b}_s$ computably because we are assuming that the back-and-forth relations are computable up to η. The last step is to define $\bar{a}_s \in A_{\tau_s}^{<\mathbb{N}}$ satisfying $(\mathcal{A}_{\tau_t}, \bar{b}_s) \leq_\zeta (\mathcal{A}_{\tau_s}, \bar{a}_s)$. Since $\{\mathcal{A}_\sigma : \sigma \in 2^{\circ\eta}\}$ is a computable η-tree and $\tau_t \restriction \zeta = \tau_s \restriction \zeta$, we have $\mathcal{A}_{\tau_t} \geq_{\zeta+1} \mathcal{A}_{\tau_s}$. We can then get $\bar{a}_s$ from the definition of the back-and-forth relation (see Remark IX.11). By extending $\bar{a}_s$ if necessary, make sure it contains the first s elements of the given ω-presentation of $\mathcal{A}_{\tau_s}$. This finishes the construction of the sequence of tuples $\langle \bar{a}_s : s \in \mathbb{N} \rangle$ satisfying (MC). □

IX.7. The proof of the game metatheorem

In this section, we use a complete η-true-stage system to prove the game metatheorem (Theorem VIII.2), which we already used in Chapter VIII in a variety of applications. Here is the statement of the game metatheorem again.

Theorem (Re-statement of Theorem VIII.2). *For every computable valid strategy for the engineer in the η-$\mathbb{A}$-game, there is a run of the game where the engineer follows her strategy, the oracle answers correctly, and the limit ω-presentation $\mathcal{L}$ is computable.*

We will first handle the successor case, where most of the ideas for the proof are already present but the setting is a bit simpler. Recall that in the $(\eta+1)$-$\mathbb{A}$-game, the ordinals β_j played by the oracle can be taken to be all equal to η. Throughout this section, we assume $\beta_j = \eta$ for all j, and we forget about them. We will consider a new version of the game that we call the *simplified version*. We will first show how to obtain Theorem VIII.2 from the simplified version of Theorem VIII.2 and then show the simplified version.

Definition IX.26. In the *simplified* $(\eta+1)$-$\mathbb{A}$-*game*, the engineer does not ask $\Delta^0_{\eta+1}$ questions. Instead, at stage j, the oracle always plays $\mathcal{T}^\eta(j+1)$ at stage j, namely the $(j+1)$st η-true stage.

Recall that $\mathcal{T}^\eta$ is $\Delta^0_{\eta+1}$-Turing complete, so if the engineer had a $\Delta^0_{\eta+1}$ question in mind, she would eventually be able to figure out the answer. Let us start by seeing how to transform the general version of the game into an instance of the simplified version.

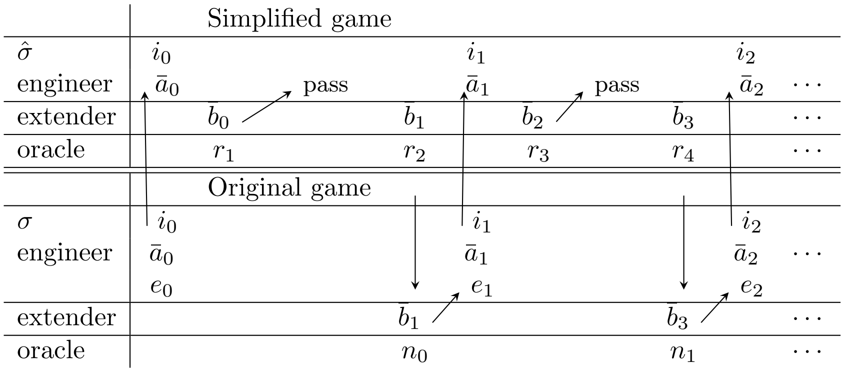
Proof of Theorem VIII.2. from the simplified version of Theorem VIII.2. Let σ be a computable, valid strategy for the engineer in the $(\eta+1)$-$\mathbb{A}$-game from Theorem VIII.2. We will build a computable ω-presentation for a limit structure obtained from a certain sequence by the extender where the engineer follows her strategy σ. The oracle will be answering $\Delta^0_{\eta+1}(\mathcal{L})$ questions along the way, but we do not build $\mathcal{L}$ until the end. The way around this is to use the recursion theorem, which will allow us to assume that, from the beginning, we have in hand an index ℓ for the computable limit structure. Here is how the recursion theorem is used: The construction will use a number ℓ as a parameter. So we are actually building a different ω-presentation $\mathcal{L}_\ell$ for each $\ell \in \mathbb{N}$. Let $g\colon \mathbb{N} \to \mathbb{N}$ be the computable function such that $g(\ell)$ is the index for the atomic diagram of $\mathcal{L}_\ell$, i.e., $\Phi_{g(\ell)} = D(\mathcal{L}_\ell)$. We then use the recursion theorem to find a computable index ℓ_0 such that

$$\Phi_{\ell_0} = \Phi_{g(\ell_0)} = D(\mathcal{L}_{\ell_0}).$$

We will then only consider the case $\ell = \ell_0$, so we may assume the parameter ℓ is an index for the very same computable diagram we are building. For this to work, we must produce a computable ω-presentation $\mathcal{L}_\ell$, even if the ℓth computable function Φ_ℓ is not total.

We will build a computable valid strategy $\hat{\sigma}$ for the engineer in the simplified $(\eta+1)$-$\mathbb{A}$-game, and we will do it uniformly in ℓ. We will do it in a way that, for every run of the simplified game following $\hat{\sigma}$, there is a run of the original game following σ that produces the same limit structure. The simplified version of Theorem VIII.2 (which we will prove below) will give a sequence of moves by the extender such that, when we follow $\hat{\sigma}$, it will produce a computable ω-presentation $\mathcal{L}_\ell$.

Here is how we define $\hat{\sigma}$: Let Γ be a computable operator such that $\Gamma^{\mathcal{T}^\eta}(e)$ is the answer to the eth $\Delta^0_{\eta+1}(\Phi_\ell)$ question. Let $\hat{\sigma}$'s first move be the same as σ's: That is, let $\hat{\sigma}(\langle\rangle) = \sigma(\langle\rangle)$. At each following stage, $\hat{\sigma}$ may either *pass* or *emulate* σ, depending on whether or not the oracle has given her enough information to answer the last $\Delta^0_{\eta+1}$-question she posed. At a stage $j+1$, whether she passes or emulates σ gets decided as follows: Suppose that, in the simplified game, the extender has just played $\bar{b}_j$, and the oracle so far has played the numbers $T_j = \langle r_1, ..., r_j\rangle$, which in the real run of the game will be equal to $\mathcal{T}^\eta \upharpoonright\!\!\upharpoonright j+1$. Let j_k be the last stage at which $\hat{\sigma}$ emulated σ — suppose it was the kth time where $\hat{\sigma}$ emulated σ. At that stage, σ asked a $\Delta^0_{\eta+1}(\Phi_\ell)$-question, say e_k. **If** $\Gamma^{T_j}(e_k)\uparrow$, let $\hat{\sigma}$ *pass*, that is, let it play $\langle i_{j+1}, \bar{a}_{j+1}\rangle$ where $i_{j+1} = i_j$, and $\bar{a}_{j+1}$ is any proper extension of $\bar{b}_j$ in $\mathcal{A}_{i_j}$. **If** $\Gamma^{T_j}(e_k)\downarrow$, let $\hat{\sigma}$ *emulate* σ and play the string that the strategy σ would play if the extender had played $\bar{b}_j$ and the oracle had played $n_k = \Gamma^{T_j}(e_k)$ in the original game. More concretely, let j_0,...,j_k be the previous stages at which $\hat{\sigma}$ emulated σ. Let e_r be the questions asked by σ at the r-th stage, and let $n_r = \Gamma^{T_j}(e_r)$. Then we let $\hat{\sigma}$ output $\sigma(\langle \bar{b}_{j_1-1}, n_0, \bar{b}_{j_2-1}, n_1, ..., \bar{b}_j, n_k\rangle)$ at stage $j+1$.



	Simplified game									
$\hat{\sigma}$	i_0				i_1				i_2	
engineer	$\bar{a}_0$		pass		$\bar{a}_1$		pass		$\bar{a}_2$	$\cdots$
extender		b_0		b_1		b_2		b_3		$\cdots$
oracle		r_1		r_2		r_3		r_4		$\cdots$
	Original game									
σ	i_0				i_1				i_2	
engineer	$\bar{a}_0$				$\bar{a}_1$				$\bar{a}_2$	$\cdots$
	e_0				e_1				e_2	
extender				b_1				b_3		$\cdots$
oracle				n_0				n_1		$\cdots$

FIGURE IX.2. This is an example of how $\hat{\sigma}$ is defined in the simplified game using σ for the original game. In this example, $\Gamma^{\langle r_1\rangle}(e_0)\uparrow$, $\Gamma^{\langle r_1,r_2\rangle}(e_0)\downarrow = n_0$, $\Gamma^{\langle r_1,r_2,r_3\rangle}(e_1)\uparrow$, and $\Gamma^{\langle r_1,r_2,r_3,r_4\rangle}(e_1)\downarrow = n_1$, and hence we have that $j_0 = 0$, $j_1 = 2$, and $j_2 = 4$.

If we apply the simplified version of Theorem VIII.2 to the strategy $\hat{\sigma}$, we end up building a computable ω-presentation $\mathcal{L}$. The diagram of $\mathcal{L}$ is total regardless of whether Φ_ℓ is total and of whether we ever get answers to the $\Delta^0_{\eta+1}(\Phi_\ell)$-questions e_k. This is because if $\hat{\sigma}$ ends up passing from some point j_k onwards, then the limit structure will end up isomorphic to $\mathcal{A}_{i_{j_k}}$. Thus, when ℓ_0 is given to us by the recursion theorem as above, we get that Φ_{ℓ_0} is total and is equal to the diagram of the limit structure $\mathcal{L}_{\ell_0}$ we just obtained. Since σ is a valid strategy, all the $\Delta^0_{\eta+1}(\Phi_{\ell_0})$-questions it asks converge, and hence, for all k, $\Gamma^{\mathcal{T}^\eta\restriction j+1}(e_k)$ converges for some large enough j. This means that there are infinitely many stages at which $\hat{\sigma}$ emulates σ, and we thus get that for every sequence of moves by the extender in the simplified game, there is a sequence of moves in the original game which gives us the same limit structure. □

Proof of Theorem VIII.2. for the simplified $(\eta+1)$-game. Here is where the η-true-stage system comes into play. Suppose we are given a strategy σ for the engineer. We want to show that there is a run of the game where the engineer follows σ, the oracle plays the η-true stages, and the limit structure is computable. To build this computable limit structure, we will build a computable sequence of pairs $\langle i_s, \bar{a}_s\rangle$ with $\bar{a}_s \in \mathcal{A}_{i_s}$ that satisfies that, for all $\xi \leq \eta$ and $r < s \in \mathbb{N}$,

$$r <_\xi s \quad \text{implies} \quad (\mathcal{A}_{i_r}, \bar{a}_r) \leq_\xi (\mathcal{A}_{i_s}, \bar{a}_s). \tag{MC}$$

In particular, we get that if $s \leq t$, then $(\mathcal{A}_{i_s}, \bar{a}_s) \leq_0 (\mathcal{A}_{i_t}, \bar{a}_t)$, which means that $D_{\mathcal{A}_{i_s}}(\bar{a}_s) \subseteq D_{\mathcal{A}_{i_t}}(\bar{a}_t)$. Hence, since the sequence $\langle\langle i_s, \bar{a}_s\rangle : s \in \mathbb{N}\rangle$ is computable, so is the structure $\mathcal{L}$ with diagram

$$D(\mathcal{L}) = \bigcup_s D_{\mathcal{A}_{i_s}}(\bar{a}_s).$$

To show that $\mathcal{L}$ is the limit structure under some run of the game following the engineer's strategy σ, we will show that if we restrict ourselves to the sequence of η-true stages $0 <_\eta t_1 <_\eta t_2 <_\eta \cdots$, then the sequence $\langle i_0, \bar{a}_0\rangle, \langle i_{t_1}, \bar{a}_{t_1}\rangle, \langle i_{t_2}, \bar{a}_{t_2}\rangle, \ldots$ can be seen as the sequence of moves by the engineer following σ in a run of the simplified $(\eta+1)$-$\mathbb{A}$-game. That is, that there exists some sequence of moves $\bar{b}_{t_1}, \bar{b}_{t_2}, \bar{b}_{t_3}, \ldots$ by the extender such that the following is a run of the game

engineer	$i_0, \bar{a}_0$		$i_{t_1}, \bar{a}_{t_1}$		$i_{t_2}, \bar{a}_{t_2}$		$\cdots$
extender		b_{t_1}		b_{t_2}		b_{t_3}	$\cdots$
oracle		t_1		t_2		t_3	$\cdots$

where, for each j, $\langle i_{t_{j+1}}, \bar{a}_{t_{j+1}}\rangle$ is played according to the strategy σ. That is:

$$\langle i_{t_{j+1}}, \bar{a}_{t_{j+1}}\rangle = \sigma(\bar{b}_{t_1}, t_1, \bar{b}_{t_2}, t_2, \ldots, \bar{b}_{t_j}, t_j, \bar{b}_{t_{j+1}}, t_{j+1}).$$

It follows that the limit structure of this run is the structure with diagram

$$\bigcup_{\substack{t\in\mathbb{N}\\ t\ \eta\text{-true}}} D_{\mathcal{A}_{i_t}}(\bar{a}_t) = \bigcup_{s\in\mathbb{N}} D_{\mathcal{A}_{i_s}}(\bar{a}_s)$$

which is computable as we mentioned above.

When we are at stage s, we believe we are at an η-true stage, so we will define $i_s, \bar{a}_s$ using the strategy σ as above. What we need to figure out is how to define $\bar{b}_s$.

For each s, we will define a tuple $\bar{b}_s$ that belongs to A_{i_t} for the largest t with $t <_\eta s$. (This is the tuple we will use as the move by the extender.) The tuple $\bar{b}_s$ will satisfy that, for every $\xi \leq \eta$ and every $r < s$,

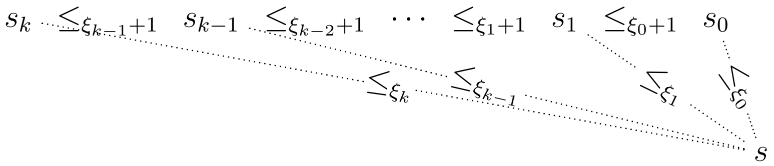
$$r <_\xi s \quad \text{implies} \quad (\mathcal{A}_{i_r}, \bar{a}_r) \leq_\xi (\mathcal{A}_{i_t}, \bar{b}_s). \tag{13}$$

Thus, if at stage s we define i_s and $\bar{a}_s$ satisfying

$$(\mathcal{A}_{i_t}, \bar{b}_s) \leq_\eta (\mathcal{A}_{i_s}, \bar{a}_s),$$

we will get property (MC). To define $\bar{b}_s$, we need to use Lemmas IX.19 and IX.20, as is often the case in constructions involving iterated true stages systems. The first lemma is used to restrict property (13) to only a few values of r and a few ordinals ξ, and the second lemma will give us $\bar{b}_s$.

Apply Lemma IX.19 to get stages $s_k < s_{k-1} < \cdots < s_1 < s_0 = s-1$, and ordinals $\eta = \xi_k > \xi_{k-1} > \cdots \xi_1 > \xi_0 = 0$ as in the diagram below and such that, for all $r < s$ and $\gamma \leq \eta$, if $r <_\gamma s$, then for the least i with $\gamma \leq \xi_i$, we have $r \leq_\gamma s_i$.



Now, to satisfy property (13), it is enough to get $\bar{b}_s$ so that

$$(\mathcal{A}_{i_{s_j}}, \bar{a}_{s_j}) \leq_{\xi_j} (\mathcal{A}_{i_t}, \bar{b}_s) \qquad \text{for all } j \leq k. \tag{14}$$

Since we have been respecting (MC) so far throughout the construction, we know that

$$(\mathcal{A}_{i_{s_k}}, \bar{a}_{s_k}) \leq_{\xi_{k-1}+1} (\mathcal{A}_{i_{s_{k-1}}}, \bar{a}_{s_{k-1}}) \leq_{\xi_{k-2}+1} \cdots$$
$$\cdots \leq_{\xi_1+1} (\mathcal{A}_{i_{s_1}}, \bar{a}_{s_1}) \leq_{\xi_0+1} (\mathcal{A}_{i_{s_0}}, \bar{a}_{s_0}).$$

These structures satisfy the assumptions needed to apply Lemma IX.20. We then obtain $\bar{b}_s$ satisfying (14) as the $\bar{c}$ from the lemma. We can find the tuple $\bar{b}_s$ computably because we are assuming that the back-and-forth relations are computable up to η. The last step is to define $\bar{a}_s$ using the strategy σ for the engineer in the simplified game.

$$\begin{tikzcd}
(\mathcal{A}_{i_{s_k}},\bar{a}_{s_k}) \arrow[drrrr, dotted, no head, "\subseteq"] & \leq_{\xi_{k-1}+1} & (\mathcal{A}_{i_{s_{k-1}}},\bar{a}_{s_{k-1}}) \arrow[drr, dotted, no head, "\leq_{\xi_{k-1}}"] & \leq_{\xi_{k-2}+1}\cdots\leq_{\xi_1+1} & (\mathcal{A}_{i_{s_1}},\bar{a}_{s_1}) \arrow[d, dotted, no head, "\geq_{\xi_1}"] & \leq_{\xi_0+1} & (\mathcal{A}_{i_{s_0}},\bar{a}_{s_0}) \arrow[dll, dotted, no head, "\geq_{\xi_0}"'] \\
 & & & & (\mathcal{A}_{i_{s_k}},\bar{b}_s) & \leq_\eta & (\mathcal{A}_{i_s},\bar{a}_s)
\end{tikzcd}$$

Let $0 = r_0 <_\eta r_1 <_\eta \cdots <_\eta r_j = s$ be the apparent η-true stages at s, i.e., $\langle r_0, ..., r_j\rangle = \mathcal{T}^\eta_s$. Notice also that, for each $\ell \leq j$, $\bar{b}_{r_\ell}$ was defined so that it belongs to $\mathcal{A}_{i_{t_\ell}}$, where t_ℓ is the largest with $t_\ell <_\eta r_\ell$, namely $t_\ell = r_{\ell-1}$. So, $\bar{b}_{r_\ell}$ belongs to $A_{i_{r_{\ell-1}}}$. We then let

$$\langle i_s, \bar{a}_s\rangle = \sigma(\bar{b}_{r_1}, r_1, \bar{b}_{r_2}, r_2, ..., \bar{b}_{r_j}, r_j).$$

That is, $\langle i_s, \bar{a}_s\rangle$ is what the engineer would play in her $(j+1)$st move if she was following σ and the previous moves by the extender were $\bar{b}_{r_1}, \bar{b}_{r_2}, ..., \bar{b}_{r_j}$ and the previous moves by the oracle were $r_1, r_2, ..., r_j$. This finishes the construction of the sequence of pairs $\langle i_s, \bar{a}_s\rangle$, and finishes the construction of a computable structure $\mathcal{L}$ with $D(\mathcal{L}) = \bigcup_{s\in\mathbb{N}} D_{\mathcal{A}_{i_s}}(\bar{a}_s)$.

Finally, to verify that it all works, we need to show $\mathcal{L}$ can also be produced by a run of the game following σ. Consider the sequence $0 < t_1 <_\eta t_2 <_\eta \cdots$ of η-true stages. These are the stages at which the oracle is playing the correct values. The following is a run of the simplified $(\eta+1)$-$\mathbb{A}$-game following strategy σ:

engineer	$i_0, \bar{a}_0$		$i_{t_1}, \bar{a}_{t_1}$		$i_{t_2}, \bar{a}_{t_2}$		$\cdots$
extender		$\bar{b}_{t_1}$		$\bar{b}_{t_2}$		$\bar{b}_{t_3}$	$\cdots$
oracle		t_1		t_2		t_3	$\cdots$

□

Remark IX.27. Note that in this proof, the sequence of moves played by the extender is computable in $\mathcal{T}^\eta$. Thus, $\mathcal{T}^\eta$ can reproduce the whole run of the game whose limit structure is the computable structure built in the previous proof.

IX.7.1. The limit case. Let us now consider the game metatheorem for the limit-ordinal case. Now, the ordinals β_j played by the oracle become relevant. Consider a limit ordinal λ, and suppose we have a complete λ-true stage system. Recall that $\mathcal{T}^\lambda$, the set of λ-true stages, is $\Delta^0_{\lambda+1}$-Turing complete, which is overkill for answering the

Δ^0_λ-questions posed by the engineer. We need a way to find guesses for a Δ^0_λ-Turing-complete set. We need a few definitions.

Let

$$\langle \lambda[s] : s \in \mathbb{N} \rangle$$

be a computable, non-decreasing sequence of ordinals below λ, converging up to λ. We define a new partial ordering $\trianglelefteq_\lambda$ on $\mathbb{N}$:[‡‡]

$$s \trianglelefteq_\lambda t \quad \iff \quad s \leq_{\lambda[s]} t. \tag{Def. $\trianglelefteq_\lambda$}$$

We say that t is a $\trianglelefteq_\lambda$-true stage if there is an infinite sequence $t \trianglelefteq_\lambda t_0 \trianglelefteq_\lambda t_1 \trianglelefteq_\lambda t_2 \trianglelefteq_\lambda \cdots$, and let $\mathcal{S}^\lambda$ be the sequence of all $\trianglelefteq_\lambda$-true stages, listed in increasing order. Notice that $\leq_\lambda$ implies $\trianglelefteq_\lambda$, and hence that all λ-true stages are $\trianglelefteq_\lambda$-true stages.

Lemma IX.28. *A stage t is $\trianglelefteq_\lambda$-true if and only if t is $\lambda[t]$-true.*

Proof. If t is $\lambda[t]$-true and $r > t$ is λ-true, then $t \leq_{\lambda[t]} r$ and hence $t \trianglelefteq_\lambda r$. Thus t is $\trianglelefteq_\lambda$-true, as witnessed by the same sequence as r. Conversely, suppose that t is $\trianglelefteq_\lambda$-true and that there is an infinite sequence $t \trianglelefteq_\lambda s_0 \trianglelefteq_\lambda s_1 \trianglelefteq_\lambda \cdots$. Since $t \leq_{\lambda[t]} s_0 \leq_{\lambda[t]} s_1 \leq_{\lambda[t]} s_2 \leq_{\lambda[t]} \cdots$, we have that t is a $\lambda[t]$ true stage. □

Corollary IX.29. $\mathcal{S}^\lambda \equiv_T \bigoplus_{\xi<\lambda} \mathcal{T}^\xi$.

Proof. When $\xi < \lambda$, we get $\mathcal{T}^\xi \leq_T \mathcal{S}^\lambda$, because $s \in \mathcal{T}^\xi$ if and only if $s \leq_\xi t$ for the first $t \in \mathcal{S}^\lambda$ with $t > s$ and $\lambda[t] \geq \xi$. Conversely, $\mathcal{S}^\lambda \leq_T \bigoplus_{\xi<\lambda} \mathcal{T}^\xi$, because $s \in \mathcal{S}^\lambda$ if and only if $s \in \mathcal{T}^{\lambda[s]}$. □

It follows that $\mathcal{S}^\lambda$ is Δ^0_λ-Turing complete. We can define our finite guesses to $\mathcal{S}^\lambda$ using the same idea as before:

$$\mathcal{S}^\lambda_t = \langle s : s \trianglelefteq_\lambda t \rangle. \tag{Def. $\mathcal{S}^\lambda_t$}$$

We then have that t is a $\trianglelefteq_\lambda$-true stage if and only if $\mathcal{S}^\lambda_t$ is an initial segment of $\mathcal{S}^\lambda$. Also, $s \trianglelefteq_\lambda t$ if and only if $\mathcal{S}^\lambda_s \subseteq \mathcal{S}^\lambda_t$.

We are now ready to prove the game metatheorem in the limit case. Let us start by describing the simplified game. As in the simplified game of the previous section, the engineer does not ask questions to the oracle. Instead, at stage j, the oracle plays $\mathcal{S}^\lambda(j)$.

engineer	$i_0, \bar{a}_0$		$i_1, \bar{a}_1$		$i_2, \bar{a}_2$	$\cdots$
extender		b_1		b_2		$\cdots$
oracle		$\mathcal{S}^\lambda(1), \lambda[\mathcal{S}^\lambda(1)]$		$\mathcal{S}^\lambda(2), \lambda[\mathcal{S}^\lambda(2)]$		$\cdots$

[‡‡] To see that it is a partial ordering, notice that if $s \leq_{\lambda[s]} t \leq_{\lambda[t]} r$, since $\lambda[s] \leq \lambda[t]$, we have $s \leq_{\lambda[s]} r$.

As for the ordinals, the oracle plays $\beta_j = \lambda[\mathcal{S}^\lambda(j)]$. So, at each stage $j > 0$, the tuple $\bar{a}_j$ played by the engineer must satisfy:

$$(\mathcal{A}_{i_{j-1}}, \bar{b}_j) \leq_{\lambda[\mathcal{S}^\lambda(j)]} (\mathcal{A}_{i_j}, \bar{a}_j).$$

The proof of Theorem VIII.2 from the simplified version of Theorem VIII.2 goes exactly as the successor case on page 167, so we do not repeat it.

Proof of Theorem VIII.2 for the simplified λ-game. The proof follows the same format as the proof of Theorem VIII.2 for the simplified game on page 169. We will not write all the details again, and instead we just concentrate on the modifications.

Let σ be a computable strategy for the engineer. We build a computable sequence of pairs $\langle i_s, \bar{a}_s\rangle$ with $\bar{a}_s \in \mathcal{A}_{i_s}$ that satisfies that, for all $r < s \in \mathbb{N}$ and $\xi \leq \lambda[r]$,

$$r \leq_\xi s \quad \text{implies} \quad (\mathcal{A}_{i_r}, \bar{a}_r) \leq_\xi (\mathcal{A}_{i_s}, \bar{a}_s). \tag{MC}$$

Notice that a difference is that we only consider $\xi \leq \lambda[r]$. We also require that

$$r \trianglelefteq_\lambda s \ \ \& \ \nexists t (r \triangleleft_\lambda t \triangleleft_\lambda s) \quad \text{implies} \quad (\mathcal{A}_{i_r}, \bar{a}_r) \leq_{\lambda[s]} (\mathcal{A}_{i_s}, \bar{a}_s). \tag{MC+}$$

That is, if r is the last apparent $\triangleleft_\lambda$-true stage at s, then $(\mathcal{A}_{i_r}, \bar{a}_r) \leq_{\lambda[s]} (\mathcal{A}_{i_s}, \bar{a}_s)$.

As in the proof of Theorem VIII.2, we will then get that the limit sequence with diagram $\bigcup_s D_{\mathcal{A}_{i_s}}(\bar{a}_s)$ is computable. To show that this structure is the limit structure under some run of the game following the engineer's strategy σ, we will show that if we restrict ourselves to the sequence of $\trianglelefteq_\lambda$-true stages $t_0 \trianglelefteq_\lambda t_1 \trianglelefteq_\lambda t_2 \trianglelefteq_\lambda \cdots$, then the sequence $\langle i_{t_0}, \bar{a}_{t_0}\rangle, \langle i_{t_1}, \bar{a}_{t_1}\rangle, \langle i_{t_2}, \bar{a}_{t_2}\rangle, \ldots$ can be seen as the sequence of moves by the engineer following σ for some particular sequence of moves by the extender.

For each s, we will also define a tuple $\bar{b}_s$ that belongs to A_{i_t} extending $\bar{a}_t$, for the largest $t < s$ with $t \trianglelefteq_\lambda s$. The tuple $\bar{b}_s$ will satisfy that, for every $r < s$ and every $\xi \leq \lambda[r]$,

$$r <_\xi s \quad \text{implies} \quad (\mathcal{A}_{i_r}, \bar{a}_r) \leq_\xi (\mathcal{A}_{i_t}, \bar{b}_s). \tag{15}$$

Thus, if we then define i_s and $\bar{a}_s$ satisfying

$$(\mathcal{A}_{i_t}, \bar{b}_s) \leq_{\lambda[s]} (\mathcal{A}_{i_s}, \bar{a}_s), \tag{16}$$

we will satisfy properties (MC) and (MC+).* We will define $\bar{b}_s$ using Lemma IX.20 and the following new version of Lemma IX.19:

Lemma IX.30. *For each $s \in \mathbb{N}$, there exist stages $t < s_\ell < s_{\ell-1} < \cdots < s_1 < s_0 = s - 1$ and ordinals $\lambda > \xi_\ell > \xi_{\ell-1} > \cdots \xi_1 > \xi_0 = 0$ as in the diagram below satisfying the following condition: For all $i \leq \ell$ and all γ with $\xi_{i-1} < \gamma \leq \xi_i$, we have that if $r <_\gamma s$ then $r \leq_\gamma s_i$. Also, for γ with $\xi_\ell < \gamma \leq \lambda[r]$, we have that if $r <_\gamma s$, then $r \leq_\gamma t$.*

$$\begin{tikzcd}
t & \trianglelefteq_\lambda u \leq_{\xi_\ell+1} & s_\ell & \leq_{\xi_{\ell-1}+1} \cdots \leq_{\xi_1+1} & s_1 & \leq_{\xi_0+1} & s_0 \\
& & & & & & s
\arrow[from=1-1, to=2-7, dash, dotted, "\trianglelefteq_\lambda"']
\arrow[from=1-3, to=2-7, dash, dotted, "\leq_{\xi_\ell}"']
\arrow[from=1-5, to=2-7, dash, dotted, "\leq_{\xi_1}"']
\arrow[from=1-7, to=2-7, dash, dotted, "\leq_{\xi_0}"]
\end{tikzcd}$$

Furthermore, there is a stage u with $t \trianglelefteq_\lambda u \leq_{\xi_\ell+1} s_\ell$ and $\lambda[u] > \xi_\ell$ such that there is no v with $t < v \triangleleft_\lambda u$.

The existence of such u will be used later in our construction to deduce the following:

$$(\mathcal{A}_{i_t}, \bar{a}_t) \leq_{\xi_\ell+1} (\mathcal{A}_{i_{s_\ell}}, \bar{a}_{s_\ell}). \tag{17}$$

Proof. Consider stages $s_k < \cdots < s_0 < s$ and ordinals $\lambda = \xi_k > \cdots > \xi_1 > \xi_0$ as in Lemma IX.19. Let t be the largest number with $t \triangleleft_\lambda s$. Since $s_k \leq_\lambda s$, we have $s_k \trianglelefteq_\lambda s$, so $s_k \trianglelefteq_\lambda t$. We do not need to use the whole sequence. Let ℓ be such that

$$s_{\ell+1} \leq t < s_\ell.$$

Suppose now that we have $r <_\gamma s$ and $\gamma \leq \lambda[r]$. We know from Lemma IX.19 that $r \leq_\gamma s_i$ for the least i with $\gamma \leq \xi_i$. If $\gamma \leq \xi_\ell$, then $i \leq \ell$. Suppose instead that $\gamma > \xi_\ell$. Then we must have that $i > \ell$ and $r \leq s_i \leq s_{\ell+1} \leq t$. Since $r \leq t$, we must have $\lambda[r] \leq \lambda[t]$, and hence $\gamma \leq \lambda[t]$. Since $t \leq_{\lambda[t]} s$ and $r \leq_\gamma s$, we can use ($\diamond$) to get $r \leq_\gamma t$.

Let us now prove that a stage u, as in the last sentence of the lemma, exists. We consider a couple of cases.

Case 1: If $\lambda[t] > \xi_\ell$, then we have $t \leq_{\xi_\ell+1} s$. Then, by ($\clubsuit$) applied to $t < s_\ell < s$, we have $t \leq_{\xi_\ell+1} s_\ell$ and we can let $u = t$.

Case 2: If $\lambda[t] \leq \xi_\ell$, then ($\diamond$) applied to $t < s_\ell < s$ gives us $t \leq_{\lambda[t]} s_\ell$, and hence $t \trianglelefteq_\lambda s_\ell$. We split this case in two cases.

Case 2A: If there is no r with $t \triangleleft_\lambda r \triangleleft_\lambda s_\ell$, then we let $u = s_\ell$. The reason that $\lambda[s_\ell] > \xi_\ell$ is that, on the one hand we have $s_\ell \leq_{\xi_\ell} s$, and on the other, since $s_\ell \not\trianglelefteq_\lambda s$, we have $s_\ell \not\leq_{\lambda[s_\ell]} s$.

* To see why we get (MC+), notice that since t is the largest stage with $t \triangleleft_\lambda s$, we need to have that $(\mathcal{A}_{i_t}, \bar{a}_{i_t}) \leq_{\lambda[s]} (\mathcal{A}_{i_s}, \bar{a}_s)$. This follows from $(\mathcal{A}_{i_t}, \bar{b}_s) \leq_{\lambda[s]} (\mathcal{A}_{i_s}, \bar{a}_s)$ because $\bar{b}_s \supseteq \bar{a}_{i_t}$.

Case 2B: Otherwise, let u be the least with $t \lhd_\lambda u \lhd_\lambda s_\ell$. We cannot have $\lambda[u] \leq \xi_\ell$, as otherwise $u \leq_{\lambda[u]} s_\ell \leq_{\lambda[u]} s$, and hence $u \trianglelefteq_\lambda s$, contradicting that t was the largest with $t \trianglelefteq s$. So $\lambda[u] > \xi_\ell$. Then, since $u \lhd_\lambda s_\ell$, we have $u \leq_{\xi_\ell+1} s_\ell$. □

The reason we get (17) is that, by (MC+) and (MC), we get

$$(\mathcal{A}_{i_t}, \bar{a}_t) \leq_{\lambda[u]} (\mathcal{A}_{i_u}, \bar{a}_u) \leq_{\xi_\ell+1} (\mathcal{A}_{i_{s_\ell}}, \bar{a}_{s_\ell}).$$

We now apply Lemma IX.20 to the sequence

$$(\mathcal{A}_{i_t}, \bar{a}_t) \leq_{\xi_\ell+1} (\mathcal{A}_{i_{s_\ell}}, \bar{a}_{s_\ell}) \leq_{\xi_{\ell-1}+1} \cdots \leq_{\xi_1+1} (\mathcal{A}_{i_{s_1}}, \bar{a}_{s_1}) \leq_{\xi_0+1} (\mathcal{A}_{i_{s_0}}, \bar{a}_{s_0})$$

to get $\bar{b}_s \in A_t^{<\mathbb{N}}$ satisfying (15) as in the diagram below.

The last step is to define $\bar{a}_s$ using the strategy σ for the engineer in the simplified game.

$$\begin{tikzcd}
(\mathcal{A}_{i_t}, \bar{a}_t) \arrow[r, phantom, "\leq_{\xi_\ell+1}"] & (\mathcal{A}_{i_{s_\ell}}, \bar{a}_{s_\ell}) \arrow[r, phantom, "\leq_{\xi_{\ell-1}+1}"] & \cdots \arrow[r, phantom, "\leq_{\xi_1+1}"] & (\mathcal{A}_{i_{s_1}}, \bar{a}_{s_1}) \arrow[r, phantom, "\leq_{\xi_0+1}"] & (\mathcal{A}_{i_{s_0}}, \bar{a}_{s_0}) \\
 & & & (\mathcal{A}_{i_t}, \bar{b}_s) \arrow[ulll, dotted, no head, "\subseteq"] \arrow[ull, dotted, no head, "\leq_{\xi_\ell}"'] \arrow[u, dotted, no head, "\leq_{\xi_1}"] \arrow[ur, dotted, no head, "\leq_{\xi_0}"'] \arrow[r, phantom, "\leq_{\lambda[s]}"] & (\mathcal{A}_{i_s}, \bar{a}_s)
\end{tikzcd}$$

Let $0 = r_0, ..., r_j = s$ be the apparent $\trianglelefteq_\lambda$-true stages below s. Note that $\mathcal{S}^\lambda_{r_i} = \mathcal{S}^\lambda_s \upharpoonright\!\upharpoonright i$. We then let

$$\langle i_s, \bar{a}_s \rangle = \sigma(\bar{b}_{r_1}, r_1, \lambda[r_1], ..., \bar{b}_{r_j}, r_j, \lambda[r_j]).$$

That is, $\langle i_s, \bar{a}_s \rangle$ is what the engineer would play in her $(j+1)$st move if she was following σ and the previous moves by the extender were $\bar{b}_{r_1}, \bar{b}_{r_2}, ..., \bar{b}_{r_j}$ and the previous moves by the oracle were $r_1, ..., r_j$ and the ordinals $\beta_i = \lambda[r_i]$. Notice that $\beta_j = \lambda[s]$, as the last element of $\mathcal{S}^\lambda_s$ is $\mathcal{S}^\lambda_s(j) = r_j = s$. So, the move by the strategy σ satisfies $(\mathcal{A}_{i_t}, \bar{b}_s) \leq_{\lambda[s]} (\mathcal{A}_{i_s}, \bar{a}_s)$ as needed.

This finalized the construction of the computable sequence $\{\bar{a}_s : s \in \mathbb{N}\}$, and we get a computable structure $\mathcal{L}$ with $D(\mathcal{L}) = \bigcup_s D_{\mathcal{A}_{i_s}}(\bar{a}_s)$. To verify that it all works, we need to see how $\mathcal{L}$ can be obtained as a run of the game following σ. Consider the sequence $t_1 \trianglelefteq_\lambda t_2 \trianglelefteq_\lambda \cdots$ of $\trianglelefteq_\lambda$-true stages. We get that the following is a run of the simplified game following σ:

engineer	$i_0, \bar{a}_0$		$i_{t_1}, \bar{a}_{t_1}$		$i_{t_2}, \bar{a}_{t_2}$		$\cdots$
extender		b_{t_1}		b_{t_2}		$\cdots$	
oracle		$t_1, \lambda[t_1]$		$t_2, \lambda[t_2]$		$\cdots$	

It follows that the limit structure of this run of the game is the computable structure with diagram $\bigcup_s D_{\mathcal{A}_{i_s}}(\bar{a}_s)$. □

Remark IX.31. The sequence of moves played by the extender is computable in $\mathcal{S}^\lambda$. Thus, $\mathcal{S}^\lambda$ can reproduce the whole run of the game whose limit structure is the computable structure built in the previous proof.

Historical Remark IX.32. The first ones to carry out an iterated true-stage argument at a limit level where Csima and Harrison-Trainor in [**CHT17**]. The way we handled the limit case here is different than theirs.

IX.8. Complete ω-true-stage systems

We defined systems of n-true-stages for $n \in \mathbb{N}$ in Section IX.2 by iterating the global ordering $\preccurlyeq$ on the sequences of $(n-1)$-apparent true stages. We recommend the reader to review the proof of Theorem IX.9, as we will build up on those ideas. The limit case poses some extra difficulties. Let us start with the first limit ordinal, ω, and define $\leq_\omega$ so that we can see some of the issues that come up and how we solve them. We will consider larger limit ordinals in the next section.

The continuity condition (TS3) should tell us immediately how to define $\leq_\omega$: It is the intersection of the orderings $\leq_n$ for $n \in \mathbb{N}$. There are some problems, though. First, that intersection may turn out to be empty; second, even if non-empty, it might not be computable; and third, even if there are infinitely many ω-true stages, they may not be $\Delta^0_{\omega+1}$ complete. To solve these problems, we will need to modify the definitions of the relations $\leq_n$ from Section IX.2 a little bit.

At first glance, the intersection of the relations $\leq_n$ for $n \in \mathbb{N}$ is Π^0_1 and not necessarily computable. Let us start by defining the *diagonal intersection* $\trianglelefteq_\omega$ as follows:

$$s \trianglelefteq_\omega t \iff s \leq_s t.$$

This ordering is computable. Notice that this is a special case of the ordering $\trianglelefteq_\lambda$ defined in Section IX.7.1 taking $\lambda = \omega$ and $\omega[s] = s$.

We say that a stage t is $\trianglelefteq_\omega$*-true* if there is an infinite sequence $t \trianglelefteq_\omega t_1 \trianglelefteq_\omega t_2 \trianglelefteq_\omega \cdots$ starting with t. We let $\mathcal{S}^\omega$ denote the sequence of $\trianglelefteq_\omega$-true stages and define its approximations as

$$\mathcal{S}^\omega_t = \langle s : s \trianglelefteq_\omega t \rangle.$$

Notice that $\mathcal{S}^\omega_s \subseteq \mathcal{S}^\omega_t$ if and only if $s \trianglelefteq_\omega t$.

We will see in Lemma IX.37 below that after modifying the orderings $\leq_n$, the sequence of $\trianglelefteq_\omega$-true stages, $\mathcal{S}^\omega$, is non-empty and is Turing-equivalent to $0^{(\omega)}$. The ω-true stages should have Turing degree $0^{(\omega+1)}$ though.

Definition IX.33. We define

$$s \leq_\omega t \iff \mathcal{S}^\omega_s \preccurlyeq \mathcal{S}^\omega_t.$$

Now, using the results from Section IX.1, from the sequence of ω-true stages, $\mathcal{T}^\omega$, we will be able to compute $\boldsymbol{\mathcal{T}}_{\mathcal{S}^\omega}$, and hence $\mathcal{T}^\omega$ will have degree $0^{(\omega+1)}$ as needed. We now need to modify the definition of the relations $\leq_n$ for $n \in \mathbb{N}$ to ensure that there are infinitely many $\trianglelefteq_\omega$-true stages and that $\leq_\omega$, as defined above, coincides with the intersection of the relations $\leq_n$ for $n \in \mathbb{N}$.

Suppose we have defined $\leq_n$ and we want to define $\leq_{n+1}$. To each s, we associate two strings, $\mathring{\mathcal{T}}^n_s$ and $\mathcal{S}^{\omega\upharpoonright n}_s$, the first with numbers above n and the second with numbers below n. The former plays the same role as the string $\mathcal{T}^n_s$ we used in Section IX.2 but with a small modification (due to Greenberg and Turetsky [**GT**]) needed to ensure the existence of infinitely many $\trianglelefteq_\omega$-true stages.

$$\mathring{\mathcal{T}}^n_s = \langle t : n \leq t <_n s \rangle = \mathcal{T}^n_s \cap [n, s).$$

Notice that we still have that $s \leq_n t \Rightarrow \mathring{\mathcal{T}}^n_s \subseteq \mathring{\mathcal{T}}^n_t$.

We use $\mathcal{S}^{\omega\upharpoonright n}_s$ to approximate $\mathcal{S}^\omega_s$ and get continuity at level ω. The idea is that $\mathcal{S}^{\omega\upharpoonright n}_s$ is the longest initial segment of $\mathcal{S}^\omega_s$ we can calculate using only the relations $\leq_m$ for $m \leq n$. We define it as follows:

$$\mathcal{S}^{\omega\upharpoonright n}_s = \langle t \leq n : t \trianglelefteq_\omega s \rangle = \mathcal{S}^\omega_s \cap [0, n].$$

Notice that we have that $s \leq_n t \Rightarrow \mathcal{S}^{\omega\upharpoonright n}_s \subseteq \mathcal{S}^{\omega\upharpoonright n}_t$.

Definition IX.34. We then define

$$s \leq_{n+1} t \iff s \leq_n t \quad \& \quad \mathring{\mathcal{T}}^n_s \preccurlyeq \mathring{\mathcal{T}}^n_t \quad \& \quad \mathcal{S}^{\omega\upharpoonright n}_s \preccurlyeq \mathcal{S}^{\omega\upharpoonright n}_t.$$

Let us analyze this definition carefully. Suppose $s \leq_n t$. If $s \geq n$, then

$$\mathcal{S}^{\omega\upharpoonright n}_s = \langle r : r \leq n \;\&\; r \leq_r s \rangle = \langle r : r \leq n \;\&\; r \leq_r t \rangle = \mathcal{S}^{\omega\upharpoonright n}_t,$$

so we get $\mathcal{S}^{\omega\upharpoonright n}_s \preccurlyeq \mathcal{S}^{\omega\upharpoonright n}_t$ trivially. If $s \leq n$, then $\mathring{\mathcal{T}}^n_s = \langle\rangle$, so we get $\mathring{\mathcal{T}}^n_s \preccurlyeq \mathring{\mathcal{T}}^n_t$ trivially. Therefore, we can split the definition of $s \leq_{n+1} t$ into two cases:

$$s \leq_{n+1} t \iff s \leq_n t \;\&\; \begin{cases} \mathring{\mathcal{T}}^n_s \preccurlyeq \mathring{\mathcal{T}}^n_t & \text{if } s \geq n \\ \mathcal{S}^{\omega\upharpoonright n}_s \preccurlyeq \mathcal{S}^{\omega\upharpoonright n}_t & \text{if } s < n. \end{cases} \qquad (*)$$

Now that we are done with the definitions, we are ready to prove the main theorem of this section:

Theorem IX.35. *The orderings $\{\leq_\xi : \xi \leq \omega\}$ form a complete ω-true-stage system.*

Let us start by proving the continuity condition (TS3). It essentially follows from the fact that for all large enough n, namely $n \geq t$, we have $\mathcal{S}_t^{\omega\upharpoonright n} = \mathcal{S}_t^{\omega}$.

Lemma IX.36. *For all $s < t$, the following are equivalent:*

(1) $\mathcal{S}_s^{\omega} \preccurlyeq \mathcal{S}_t^{\omega}$.
(2) $s \leq_n t$ *for all* $n \in \mathbb{N}$.

Proof. For the (2)-to-(1) implication, suppose $s \leq_n t$ for all $n \in \mathbb{N}$. For all $n \geq t$, $\mathcal{S}_s^{\omega\upharpoonright n} = \mathcal{S}_s^{\omega}$ and $\mathcal{S}_t^{\omega\upharpoonright n} = \mathcal{S}_t^{\omega}$. Since $s \leq_{n+1} t$, we get that $\mathcal{S}_s^{\omega\upharpoonright n} \preccurlyeq \mathcal{S}_t^{\omega\upharpoonright n}$, and hence $\mathcal{S}_s^{\omega} \preccurlyeq \mathcal{S}_t^{\omega}$.

For the (1)-to-(2) implication, suppose that $\mathcal{S}_s^{\omega} \preccurlyeq \mathcal{S}_t^{\omega}$. This implies that $s \trianglelefteq_\omega t$, and hence that $s \leq_s t$. We will show by induction that for all $n \geq s$, $s \leq_n t$. By the observation above, when $s \leq n$, we have that $s \leq_{n+1} t \iff s \leq_n t \wedge \mathcal{S}_s^{\omega\upharpoonright n} \preccurlyeq \mathcal{S}_t^{\omega\upharpoonright n}$. Notice that

$$\mathcal{S}_s^{\omega\upharpoonright n} = \mathcal{S}_s^{\omega} \text{ and } \mathcal{S}_t^{\omega\upharpoonright n} \subseteq \mathcal{S}_t^{\omega}.$$

So, $\mathcal{S}_s^{\omega} \preccurlyeq \mathcal{S}_t^{\omega}$ implies $\mathcal{S}_s^{\omega\upharpoonright n} \preccurlyeq \mathcal{S}_t^{\omega\upharpoonright n}$ by (♣). It follows that $s \leq_{n+1} t \iff s \leq_n t$ for all $n \geq s$. By induction, one can then prove that $s \leq_n t$ for all n. □

Thus, defining $s \leq_\omega t$ by $\mathcal{S}_s^{\omega} \preccurlyeq \mathcal{S}_t^{\omega}$ as we did above, we satisfy the continuity condition (TS3).

Next we need to show that there are infinitely many $\trianglelefteq_\omega$-true stages.

Lemma IX.37. *Assume that for every n, there are infinitely many n-true stages.*

(1) *There are infinitely many stages s which are s-true.*
(2) *s is an $\trianglelefteq_\omega$-true stage if and only if it is s-true.*
(3) $\mathcal{S}^{\omega} \equiv_T \bigoplus_{n\in\mathbb{N}} \mathcal{T}^n$.

Proof. For the first part, pick a number n, and let s be the least n-true stage greater than or equal to n. Let r be an s-true stage greater than s. We will show that s is s-true by showing that $s \leq_s r$. Since r is also n-true, we know $s \leq_n r$. We use induction to show that $s \leq_m r$ for every m with $n \leq m \leq s$. Recall that for $m \leq s$, $s \leq_m r$ if and only if $s \leq_{m-1} r$ and $\mathring{\mathcal{T}}_s^m \preccurlyeq \mathring{\mathcal{T}}_r^m$. Since there are no n-true stages between n and s, for no t with $n \leq t < s$ do we have $t \leq_n s$. It follows that $\mathring{\mathcal{T}}_s^n = \langle\rangle$. Actually, $\mathring{\mathcal{T}}_s^m = \langle\rangle$, and hence $\mathring{\mathcal{T}}_s^m \preccurlyeq \mathring{\mathcal{T}}_r^m$ for all $m \geq n$. It follows that, for all m with $n \leq m < s$, we have $s \leq_{m+1} r \iff s \leq_m r$. That gives us the induction step needed to show that $s \leq_s r$.

For the second part, note that the stages s which are s-true form an infinite $\trianglelefteq_\omega$-increasing sequence. To see this, observe that if we have $s < t$, with s s-true and t t-true, then $s \leq_s t$ and hence $s \trianglelefteq_\omega t$. So, they are all $\trianglelefteq_\omega$-true stages. Conversely, suppose that s is a $\trianglelefteq_\omega$-true stage. There is an infinite sequence $s \trianglelefteq_\omega t_1 \trianglelefteq_\omega t_2 \trianglelefteq_\omega \cdots$. In particular, we have $s \leq_s t_1 \leq_s t_2 \leq_s \cdots$, and hence s is an s-true stage.

For the third part, it is clear from (2) that $\mathcal{S}^\omega \leq_T \bigoplus_{n\in\mathbb{N}} \mathcal{T}^n$ as $s \in \mathcal{S}^\omega$ if and only if $s \in \mathcal{T}^s$. For the other reduction, given n and s, we can tell if $s \in \mathcal{T}^n$ as follows: Look for $r \in \mathcal{S}^\omega$ greater than n and s. Then, since r is r-true, and in particular n-true, s is n-true if and only if $s \leq_n r$. □

We have shown that $\leq_\omega$ behaves the way it should. We still need to verify that the new relations $\leq_n$ form an ω-true stage system. Conditions (TS0) and (TS1) hold trivially. We already verified (TS3). For (TS2), we already verified there are infinitely many ω-true stages, assuming that there are infinitely many n-true stages for every n, which we have not verified yet.

Lemma IX.38. *There are infinitely many n-true stages for every $n \in \mathbb{N}$.*

Proof. All stages are 0-true. Suppose we already know that there are infinitely many n-true stages. Let X be the increasing sequence of all n-true stages that are greater than or equal to n, i.e., $X = \mathring{\mathcal{T}}^n$. From Section IX.1, we get an infinite sequence of finite increasing substrings of X which are X-true substrings:

$$\langle\rangle \preccurlyeq \sigma_0 \preccurlyeq \sigma_1 \preccurlyeq \cdots \preccurlyeq X.$$

In other words, $\boldsymbol{\mathcal{T}}_X = \langle\langle\rangle, \sigma_0, \sigma_1, \cdots\rangle$. Let $s_i = \max(\sigma_i)$. Notice that $s_i \leq_n s_{i+1}$ for all i just because they belong to $\mathring{\mathcal{T}}^n$. Furthermore,

$$\mathring{\mathcal{T}}^n_{s_{i+1}} = \sigma_i.$$

Recall that for $s \geq n$ and $s \leq_n t$, $s \leq_{n+1} t \iff \mathring{\mathcal{T}}^n_s \preccurlyeq \mathring{\mathcal{T}}^n_t$. Since $\mathring{\mathcal{T}}^n_{s_i} \preccurlyeq \mathring{\mathcal{T}}^n_{s_{i+1}}$ and $n \leq s_i$, it follows that $s_i \leq_{n+1} s_{i+1}$ for all i, and hence these are all $(n+1)$-true stages. □

This finishes the proof of (TS2).

Lemma IX.39. $\mathcal{T}^\omega \equiv_T 0^{(\omega+1)}$.

Proof. Continuing with the ideas from the proof of the previous lemma, in particular, we get

$$\boldsymbol{\mathcal{T}}_X = \{\mathring{\mathcal{T}}^n_s : s \in \mathring{\mathcal{T}}^{n+1}\},$$

and hence $(\mathring{\mathcal{T}}^n)' \equiv_T \boldsymbol{\mathcal{T}}_X \equiv_T \mathring{\mathcal{T}}^{n+1}$. So, we get by induction that each $\mathring{\mathcal{T}}^n$ is Turing equivalent to $0^{(n)}$. Since $\mathcal{T}^n$ and $\mathring{\mathcal{T}}^n$ differ only on finitely many elements, $\mathcal{T}^n$ is also Turing equivalent to $0^{(n)}$. Furthermore, this

Turing equivalence is uniform in n, as $s \in \mathcal{T}^n$ if and only if $s \leq_n t$ for the first $t \in \mathring{\mathcal{T}}^n$ with $s \leq t$. This then implies that $\mathcal{S}^\omega \equiv_T 0^{(\omega)}$, and by our comment right after Definition IX.33, $\mathcal{T}^\omega \equiv_T 0^{(\omega+1)}$. □

Lemma IX.40. *The relations $\leq_n$ satisfy (♣).*

Proof. We use induction on n. Consider $t < s < r$, satisfying $t \leq_{n+1} r$ and $s \leq_n r$. By the induction hypothesis, since $t \leq_n r$ and $s \leq_{n-1} r$, we get that $t \leq_n s$. In the case $n = 0$, this is immediate. Since $t \leq_{n+1} r$, we have that

$$\mathring{\mathcal{T}}^n_t \preccurlyeq \mathring{\mathcal{T}}^n_r \;\wedge\; \mathcal{S}^{\omega\upharpoonright n}_t \preccurlyeq \mathcal{S}^{\omega\upharpoonright n}_r.$$

Since $s \leq_n r$, we have that

$$\mathring{\mathcal{T}}^n_s \subseteq \mathring{\mathcal{T}}^n_r \;\wedge\; \mathcal{S}^{\omega\upharpoonright n}_s \subseteq \mathcal{S}^{\omega\upharpoonright n}_r.$$

Using (♣) for $\preccurlyeq$, we then get

$$\mathring{\mathcal{T}}^n_t \preccurlyeq \mathring{\mathcal{T}}^n_s \;\wedge\; \mathcal{S}^{\omega\upharpoonright n}_t \preccurlyeq \mathcal{S}^{\omega\upharpoonright n}_s,$$

as needed to show that $t \leq_{n+1} s$. □

This finishes the proof of Theorem IX.35.

IX.9. The full construction

Fix an ω-presentation of a large computable well-ordering $\mathcal{L}$. The objective of this section is to prove Theorem IX.18, that is, to define relations $\leq_\xi$ for $\xi \in \mathcal{L}$ that form a complete $\mathcal{L}$-true-stage system. Many of the ideas of the construction were introduced in the previous section, except that now, the chosen ω-presentation of the ordinals becomes relevant, we have lots of limit ordinals to worry about, and the proof becomes more complicated. The construction of this section is self-contained, but it is quite technical and working through the previous sections will help the reader have a better intuition.

This theorem was first proved in [**Mon14**, Lemma 7.8], using ideas from Marcone and Montalbán [**MM11**]. A different construction was later given by Greenberg and Turetsky in [**GT**] which simplified it greatly. The proof we give here is new. This new construction incorporates ideas from all those papers [**MM11, Mon14, GT**], as well as a new way of dealing with the limit levels.

The domain of $\mathcal{L}$ is the set of natural numbers, so its elements are numbers. We will sometimes think of the elements of $\mathcal{L}$ as ordinals and sometimes as natural numbers. To emphasize this, we will use Greek letters $\zeta, \xi, \gamma, \lambda, \kappa$ when we think of elements of $\mathcal{L}$ as ordinals, and we will write n_γ for the natural number corresponding to the ordinal γ in this fixed representation of $\mathcal{L}$. Then, for instance, if we write $\gamma < \lambda$, we are comparing them as ordinals in the ordering of $\mathcal{L}$, while if we write $n_\gamma < n_\lambda$, we are comparing them as natural numbers.

To each non-zero ordinal $\lambda \in \mathcal{L}$, we assign a *fundamental sequence*: For $s \in \mathbb{N}$, let

$$\lambda[s] = \max\{\xi : \xi < \lambda \ \ \& \ \ n_\xi \leq s\}.$$

It is not hard to see that $\langle \lambda[s] : s \in \mathbb{N}\rangle$ is a non-decreasing sequence and that, when λ is a limit ordinal, converges up to λ:

$$\sup_{s\in\mathbb{N}} \lambda[s] = \lambda.$$

To see this, just notice that, for each $\xi < \lambda$, $\lambda[n_\xi] \geq \xi$. When λ is a successor ordinal, we get a non-decreasing sequence that eventually stabilizes at $\lambda[s] = \lambda - 1$, namely from $s = n_{\lambda-1}$ onwards.

Lemma IX.41. *Let $s, t \in \mathbb{N}$ and $\gamma, \lambda \in \mathcal{L} \smallsetminus \{0\}$.*

1. *If $s \leq t$ and $\gamma \leq \lambda$, then $\gamma[s] \leq \lambda[t]$.*
2. *If $\lambda[t] \leq \gamma \leq \lambda$, then $(\forall s \leq t)\ \gamma[s] = \lambda[s]$.*

Proof. For the first implication, just notice that you are taking a maximum of a larger set. For the second one, it follows from the definition of $\lambda[t]$ that, for every $\xi < \lambda$ with $n_\xi \leq t$, we have $\xi \leq \lambda[t]$, and hence $\xi \leq \gamma$. So the sets $\{\xi : \xi < \lambda \ \ \& \ \ n_\xi \leq s\}$ and $\{\xi : \xi < \gamma \ \ \& \ \ n_\xi \leq s\}$ are the same. □

IX.9.1. The main characters. Let us start by informally describing the main characters in the construction. The full, formal definition will be in the next section, where we will define various objects by simultaneous recursion.

For each ordinal ξ, we will define two orderings $\leq_\xi$ and $\trianglelefteq_\xi$ on $\mathbb{N}$. As we will see below, $\trianglelefteq_\xi$ will be defined using diagonal intersections of the orderings $\leq_\zeta$ for $\zeta < \xi$:

$$s \trianglelefteq_\xi t \qquad \Longleftrightarrow \qquad s \leq_{\xi[s]} t.$$

We use $\mathcal{T}^\xi$ to denote the sequence of $\leq_\xi$-true stages and $\mathcal{S}^\xi$ to denote the sequence of $\trianglelefteq_\xi$-true stages. We will show that

$$\mathcal{S}^\lambda \equiv_T \bigoplus_{\xi<\lambda} \mathcal{T}^\xi \qquad \text{and } \mathcal{T}^\lambda \equiv_T (\mathcal{S}^\lambda)'$$

uniformly in λ. This will then imply that $\mathcal{S}^\lambda$ is Δ^0_λ-Turing complete, and $\mathcal{T}^\lambda$ is $\Delta^0_{\lambda+1}$-Turing complete.

To define $\leq_\xi$ from $\trianglelefteq_\xi$, the first idea would be to let $s \leq_\xi t$ if and only if $\mathcal{S}^\xi_s \preccurlyeq \mathcal{S}^\xi_t$, where $\mathcal{S}^\xi_s$ is the stage-s finite approximation to $\mathcal{S}^\xi$. As we already saw in the case for ω, this will not work well. We need two modifications. First, instead of using $\mathcal{S}^\xi_s$ we will consider the strings $\mathring{\mathcal{S}}^\xi_s$, where we only consider numbers above n_ξ and we do not include the top element s. That is

$$\mathring{\mathcal{S}}^\xi_s = \mathcal{S}^\xi_s \cap [n_\xi, s).$$

Second, when defining $\leq_\xi$, we will also look at segments of $\mathring{\mathcal{S}}^\kappa_s$ for $\kappa > \xi$. We will only consider the initial segment of $\mathring{\mathcal{S}}^\kappa_s$ that can be computed from $\mathcal{S}^\xi_s$, a segment that we will call $\mathring{\mathcal{S}}^{\kappa\restriction\xi}_s$.

Let us now make all these definitions in more detail.

IX.9.2. The formal definition. We will define the family of partial orderings $\leq_\xi$ and families of stings $\mathring{\mathcal{S}}^{\kappa\restriction\xi}_t$ by simultaneous effective transfinite recursion on ξ. We will need to develop the diagonal orderings $\trianglelefteq_\xi$ before we motivate the definition of $\mathring{\mathcal{S}}^{\kappa\restriction\xi}_t$. So, for now, let us just say that, to each $t \in \mathbb{N}$, $\xi, \kappa \in \mathcal{L}$, we associate a finite string $\mathring{\mathcal{S}}^{\kappa\restriction\xi}_t \in \mathbb{N}^{<\mathbb{N}}$ and that the definition of $\mathring{\mathcal{S}}^{\kappa\restriction\xi}_t$ uses only the relations $\leq_\zeta$ for $\zeta < \xi$. The format of the definition of $\leq_\xi$ should be thought as follows: given ξ, we associate to each $t \in \mathbb{N}$ a family of finite strings $\{\mathring{\mathcal{S}}^{\kappa\restriction\zeta}_t : \kappa \in \mathcal{L}, \zeta \leq \xi\}$, and we let $s \leq_\xi t$ if the strings associated to s are $\preccurlyeq$-below the strings associated to t.

For $\xi = 0$, we have $\mathring{\mathcal{S}}^{\kappa\restriction\xi}_t = \langle\rangle$. Also, in the lemma below, we will use that if $\mathring{\mathcal{S}}^{\kappa\restriction\xi}_t$ is a proper initial segment of $\mathring{\mathcal{S}}^{\kappa\restriction\xi}_s$, then $t < s$.

Definition IX.42. Given $s \leq t \in \mathbb{N}$ and $\xi \in \mathcal{L}$, define

$$s \leq_\xi t \iff (\forall \kappa \in \mathcal{L})(\forall \zeta \leq \xi) \;\; \mathring{\mathcal{S}}^{\kappa\restriction\zeta}_s \preccurlyeq \mathring{\mathcal{S}}^{\kappa\restriction\zeta}_t. \qquad \text{(Def. } \leq_\xi)$$

Without trying to make sense yet of what the strings $\mathring{\mathcal{S}}^{\kappa\restriction\xi}_t$ are, just by knowing that their definition depends only on $\leq_\zeta$ for $\zeta < \xi$, we know that the transfinite recursion in the definition of $\leq_\xi$ works. We will see how to make this definition effective later, by noting one only needs to consider finitely many comparisons of the form $\mathring{\mathcal{S}}^{\kappa\restriction\zeta}_s \preccurlyeq \mathring{\mathcal{S}}^{\kappa\restriction\zeta}_t$ in the definition above.

It is clear from the definition that the relations $\leq_\xi$ are nested, that is, that if $\xi \leq \gamma$ and $s \leq_\gamma t$, then $s \leq_\xi t$. Observe also that $s \leq_0 t \iff s \leq t$, as $\mathring{\mathcal{S}}^{\kappa\restriction 0}_t = \langle\rangle$ for any κ.

Lemma IX.43. *For each $\xi \in \mathcal{L}$, $\leq_\xi$ is a partial ordering which satisfies the following property:*

$$\forall s, t, r \in \mathbb{N} \left(s \leq t \leq_\xi r \Rightarrow \big(s \leq_\xi t \iff s \leq_\xi r \big) \right). \qquad (\diamond)$$

Proof. The reflexivity property of partial orderings is obvious from the definition. The anti-symmetric property of partial orderings follows from the fact that $s \leq_\xi t$ implies $s \leq t$. Transitivity follows from transitivity of $\preccurlyeq$. For property ($\diamond$), the ($\Rightarrow$) direction follows from transitivity. The ($\Leftarrow$) direction is straightforward from ($\clubsuit$) applied to the strings $\mathring{\mathcal{S}}_s^{\kappa\restriction\zeta} \subseteq \mathring{\mathcal{S}}_t^{\kappa\restriction\zeta} \subseteq \mathring{\mathcal{S}}_r^{\kappa\restriction\zeta}$ for each κ and $\zeta \leq \xi$. □

IX.9.3. The diagonal orderings. We define another family of partial orderings, $\trianglelefteq_\xi$ for $\xi \in \mathcal{L}$, which are the *diagonal intersections* of the $\leq_\gamma$'s.

$$s \trianglelefteq_\xi t \iff s \leq_{\xi[s]} t. \qquad \text{(Def. } \trianglelefteq_\xi)$$

Notice that if ξ is a successor ordinal, then $\trianglelefteq_\xi$ coincides with $\leq_{\xi-1}$ from some point on, namely from $s = n_{\xi-1}$ onwards. The orderings $\trianglelefteq_\xi$ become handy when ξ is a limit ordinal, and they will help us deal with the continuity condition (TS3).

Lemma IX.44. (1) *Each relation $\trianglelefteq_\xi$ is a partial ordering on* $\mathbb{N}$.

(2) *The relations $\trianglelefteq_\xi$ are* nested*, that is, if $\xi \leq \gamma$ and $s \trianglelefteq_\gamma t$, then $s \trianglelefteq_\xi t$.*

(3) *The relations $\trianglelefteq_\xi$ satisfy the continuity condition. That is, if λ is a limit ordinal, then*

$$s \trianglelefteq_\lambda t \iff (\forall \xi < \lambda)\ s \trianglelefteq_\xi t.$$

(4) *The relations $\trianglelefteq_\xi$ satisfy the following property:*

$$(\forall s < t < r)\ s \trianglelefteq_\xi r \ \&\ t \trianglelefteq_\xi r \Rightarrow s \trianglelefteq_\xi t. \qquad (\diamond)$$

Proof. For (1), the anti-symmetric property of partial orderings follows from the fact that $s \trianglelefteq_\xi t$ implies $s \leq_0 t$. The reflexivity property is obvious. For transitivity, if we have $s \trianglelefteq_\xi t \trianglelefteq_\xi r$, then $s \leq_{\xi[s]} t \leq_{\xi[t]} r$, and since $\xi[s] \leq \xi[t]$, we get $s \leq_{\xi[s]} r$ by Lemma IX.41(1).

For nestedness, recall from Lemma IX.41(1) that $\xi \leq \gamma$ implies $\xi[s] \leq \gamma[s]$, and hence $s \leq_{\gamma[s]} t$ implies $s \leq_{\xi[s]} t$.

For (3), the ($\Rightarrow$) direction follows from nestedness. For the ($\Leftarrow$) direction, recall that if ξ is strictly in between $\lambda[s]$ and λ, then $\xi[s] = \lambda[s]$ by Lemma IX.41(2), and hence $s \trianglelefteq_\lambda t \iff s \trianglelefteq_\xi t$.

Part (4) follows from ($\diamond$) for $\leq_{\xi[s]}$. □

Definition IX.45. We say that t is a $\trianglelefteq_\xi$*-true stage* if there is an infinite sequence $t \triangleleft_\xi t_1 \triangleleft_\xi t_2 \triangleleft_\xi t_3 \cdots$. We let $\mathcal{S}^\xi$ be the sequence of all $\trianglelefteq_\xi$-true stages listed in increasing order.

The existence of $\trianglelefteq_\xi$-true stages is not obvious, and we will prove it later. We will also show that $\mathcal{S}^\xi$ is Δ^0_ξ-Turing complete. But let us not get ahead of ourselves, and let us consider now the finite approximations to $\mathcal{S}^\xi$.

$$\mathcal{S}^\xi_t = \langle s : s \trianglelefteq_\xi t \rangle. \qquad \text{(Def. } \mathcal{S}^\xi_t)$$

Observe that $t \trianglelefteq_\xi r$ if and only if $\mathcal{S}^\xi_t \subseteq \mathcal{S}^\xi_r$. The symbol '$\subseteq$' here refers to inclusion of strings.

For limit ordinals λ, one can show that $\mathcal{S}^\lambda_t$ is the limit of $\mathcal{S}^\xi_t$ for $\xi < \lambda$. Furthermore, $\mathcal{S}^\lambda_t$ and $\mathcal{S}^\xi_t$ coincide on longer and longer initial segments as ξ converges up to λ. They will coincide on all entries s which satisfy $\lambda[s] = \xi[s]$, as we would then have $s \trianglelefteq_\lambda t \iff s \trianglelefteq_\xi t$. It will be useful to give a name to the first s where that does not happen.

Definition IX.46. For $\xi, \lambda \in \mathcal{L}$, define

$$m_{[\xi,\lambda)} = \min\{n_\zeta : \xi \le \zeta < \lambda\}.$$

If $\xi \ge \lambda$, we let $m_{[\xi,\lambda)} = +\infty$.

Lemma IX.47. *For $\xi < \lambda$,*

$$m_{[\xi,\lambda)} = \min\{r \in \mathbb{N} : \lambda[r] \ge \xi\} = \min\{r \in \mathbb{N} : \lambda[r] \ne \xi[r]\}.$$

Proof. Let ζ be such that $m_{[\xi,\lambda)} = n_\zeta$. So, $\xi \le \zeta < \lambda$, and the value of n_ζ is least among all such ζ's. On the one hand, for $r = n_\zeta$, we have $\lambda[r] \ge \zeta \ge \xi > \xi[r]$. On the other hand, for $r < n_\zeta$, the sets $\{\gamma < \lambda : n_\gamma \le r\}$ and $\{\gamma < \xi : n_\gamma \le r\}$ are equal, because by our choice of ζ, if $n_\gamma < n_\zeta$, then either $\gamma < \xi$ or $\gamma \ge \lambda$. So, $\lambda[r] = \xi[r] < \xi$. So, $r = n_\zeta$ is the least number satisfying $\lambda[r] \ge \xi$, and also the least number satisfying $\lambda[r] \ne \xi[r]$. □

Lemma IX.48. *For $\xi < \lambda$ and $t \in \mathbb{N}$,*

$$\mathcal{S}^\lambda_t \cap [0, m_{[\xi,\lambda)}) = \mathcal{S}^\xi_t \cap [0, m_{[\xi,\lambda)}).$$

Proof. For $s \in [0, m_{[\xi,\lambda)})$, we have $\xi[s] = \lambda[s]$, and hence $s \trianglelefteq_\xi t \iff s \trianglelefteq_\lambda t$. □

This lemma motivates the following definition: Let

$$\mathcal{S}^{\lambda\restriction\xi}_t = \mathcal{S}^\lambda_t \cap [0, m_{[\xi,\lambda)}). \qquad \text{(Def. } \mathcal{S}^{\lambda\restriction\xi}_t\text{)}$$

By the lemma above, $\mathcal{S}^{\lambda\restriction\xi}_t = \mathcal{S}^\xi_t \cap [0, m_{[\xi,\lambda)})$. The idea behind $\mathcal{S}^{\lambda\restriction\xi}_t$ is that it is the longest initial segment of $\mathcal{S}^\lambda_t$ that we can define using only $\mathcal{S}^\xi_t$. The strings $\mathcal{S}^{\lambda\restriction\xi}_t$ are similar to the stings we used in Definition IX.42. We still need to modify them a little bit though to ensure that we actually get infinitely many $\trianglelefteq_\kappa$-true stages at limit levels. To do this, we use the Greenberg–Turetsky trick we used in the previous section of considering only a segment of $\mathcal{S}^\kappa_t$. Define

$$\mathring{\mathcal{S}}^\kappa_t = \langle r : n_\kappa \le r \triangleleft_\kappa t\rangle, \qquad \text{(Def. } \mathring{\mathcal{S}}^\kappa_t\text{)}$$

or, equivalently

$$\mathring{\mathcal{S}}^\kappa_t = \mathcal{S}^\kappa_t \cap [n_\kappa, t).$$

As in (Def. $\mathcal{S}_t^{\lambda\restriction\xi}$), we can define $\mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ to be the longest initial segment of $\mathring{\mathcal{S}}_t^{\kappa}$ that we can define using only the relations $\leq_\zeta$ for $\zeta < \xi$. Let

$$\mathring{\mathcal{S}}_t^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_t^{\kappa} \cap [0, m_{[\xi,\kappa)}) = \mathcal{S}_t^{\kappa} \cap [n_\kappa, \min(t, m_{[\xi,\kappa)})).$$

Recall that, to ensure that the transfinite recursion in the definition of $\leq_\xi$ works, we promised right before Definition IX.42 that $\mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ could be defined in terms of relations $\leq_\zeta$ for $\zeta < \xi$. To make this obvious, we can rewrite its definition as follows:

$$\mathring{\mathcal{S}}_t^{\kappa\restriction\xi} = \langle r \in \mathbb{N} : \ \kappa[r] < \xi \quad \& \quad n_\kappa \leq r <_{\kappa[r]} t \rangle. \qquad (\text{Def. } \mathring{\mathcal{S}}_t^{\kappa\restriction\xi})$$

It follows from Lemma IX.48 that, when $\xi < \kappa$,

$$\mathring{\mathcal{S}}_t^{\kappa\restriction\xi} = \mathcal{S}_t^{\xi} \cap [n_\kappa, \min(t, m_{[\xi,\kappa)})).$$

When $\xi \geq \kappa$, $\mathring{\mathcal{S}}_t^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_t^{\kappa}$.

Lemma IX.49. *Consider $t \leq r \in \mathbb{N}$ and $\xi \in \mathcal{L}$.*

1. *If $t \trianglelefteq_\xi r$, then $\mathring{\mathcal{S}}_t^{\kappa\restriction\xi} \subseteq \mathring{\mathcal{S}}_r^{\kappa\restriction\xi}$.*
2. *If $t \trianglelefteq_\xi r$ and $m_{[\xi,\kappa)} \leq t$, then $\mathring{\mathcal{S}}_t^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_r^{\kappa\restriction\xi}$.*
3. *If $\zeta \leq \xi$ and $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$, then $\mathring{\mathcal{S}}_s^{\kappa\restriction\zeta} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\zeta}$.*

Proof. For (1), first notice that $t \trianglelefteq_\xi r$ implies $\mathcal{S}_t^\xi \subseteq \mathcal{S}_r^\xi$. If $\kappa > \xi$, then

$$\mathring{\mathcal{S}}_t^{\kappa\restriction\xi} = \mathcal{S}_t^{\xi} \cap [n_\kappa, \min(t, m_{[\xi,\kappa)})) \subseteq \mathcal{S}_r^{\xi} \cap [n_\kappa, \min(r, m_{[\xi,\kappa)})) = \mathring{\mathcal{S}}_r^{\kappa\restriction\xi}.$$

If $\kappa \leq \xi$, we have $t \trianglelefteq_\kappa r$, and $\mathring{\mathcal{S}}_t^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_t^{\kappa} \subseteq \mathring{\mathcal{S}}_r^{\kappa} = \mathring{\mathcal{S}}_r^{\kappa\restriction\xi}$.

For (2), let us first observe that since $m_{[\xi,\kappa)} \neq +\infty$, $\xi < \kappa$. Recall that

$$\begin{aligned} \mathring{\mathcal{S}}_t^{\kappa\restriction\xi} &= \langle s \in \mathbb{N} : s \trianglelefteq_\xi t, \ n_\kappa \leq s < \min(t, m_{[\xi,\kappa)}) \rangle, \text{ and} \\ \mathring{\mathcal{S}}_r^{\kappa\restriction\xi} &= \langle s \in \mathbb{N} : s \trianglelefteq_\xi r, \ n_\kappa \leq s < \min(r, m_{[\xi,\kappa)}) \rangle. \end{aligned}$$

Since $t \trianglelefteq_\xi r$, we have from ($\diamond$) in Lemma IX.43 that $s \trianglelefteq_\xi t \iff s \trianglelefteq_\xi r$. Since $m_{[\xi,\kappa)} \leq t$, we have $\min(t, m_{[\xi,\kappa)}) = m_{[\xi,\kappa)} = \min(r, m_{[\xi,\kappa)})$. So, $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$.

For the third part, first we observe that $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ implies $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \subseteq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$, which implies $\mathring{\mathcal{S}}_s^{\kappa\restriction\zeta} \subseteq \mathring{\mathcal{S}}_t^{\kappa\restriction\zeta}$.

There are two cases to consider. If κ is such that $m_{[\zeta,\kappa)} \leq s$, then $\mathring{\mathcal{S}}_s^{\kappa\restriction\zeta}$ and $\mathring{\mathcal{S}}_t^{\kappa\restriction\zeta}$ are both equal to $\mathcal{S}_s^{\kappa} \cap [n_\kappa, m_{[\zeta,\kappa)})$, and then $\mathring{\mathcal{S}}_s^{\kappa\restriction\zeta} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\zeta}$ holds trivially.

If κ is such that $s \leq m_{[\zeta,\kappa)} \leq m_{[\xi,\kappa)}$, we have $\mathring{\mathcal{S}}_s^{\kappa} = \mathring{\mathcal{S}}_s^{\kappa\restriction\zeta} = \mathring{\mathcal{S}}_s^{\kappa\restriction\xi}$. Since $\mathring{\mathcal{S}}_t^{\kappa\restriction\zeta} \subseteq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$, we can apply ($\clubsuit$) to the triple of strings $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_s^{\kappa\restriction\zeta} \subseteq \mathring{\mathcal{S}}_t^{\kappa\restriction\zeta} \subseteq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ to get $\mathring{\mathcal{S}}_s^{\kappa\restriction\zeta} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\zeta}$. □

Using the third part of the lemma, we can improve Definition IX.42 slightly and write:

$$s \leq_\xi t \quad \iff \quad s \leq t \quad \& \quad (\forall \kappa \in \mathcal{L}) \ \mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}. \qquad \text{(Def. } \leq_\xi)$$

To see that the definition is effective, notice that if $n_\kappa \geq s$, then $\mathring{\mathcal{S}}_s^\kappa = \langle\rangle$ and $\mathring{\mathcal{S}}_s^\kappa \preccurlyeq \mathring{\mathcal{S}}_t^\kappa$ holds trivially. So, in (Def. $\leq_\xi$), we only care about the finitely many κ's with $n_\kappa < s$, making that definition effective. The whole definition can then be carried out by effective transfinite recursion as in Section I.4.1, and thus we get that the $\mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ and $s \leq_\xi t$ are uniformly computable in κ, ξ, s, and t.

The following lemma provides an equivalent way of defining the $\leq_\xi$ relations that will be useful in the verifications.

Lemma IX.50. *If $s \geq n_\xi$, then*

$$s \leq_\xi t \iff s \trianglelefteq_\xi t \ \wedge \ \mathring{\mathcal{S}}_s^\xi \preccurlyeq \mathring{\mathcal{S}}_t^\xi,$$

and if $s \leq n_\xi$, then

$$s \leq_\xi t \iff s \trianglelefteq_\xi t \ \wedge \ (\forall \kappa > \xi) \ \mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}.$$

Proof. By nestedness, $s \leq_\xi t$ implies $s \leq_{\xi[s]} t$, which implies $s \trianglelefteq_\xi t$. The ($\Rightarrow$) direction is then clear in both cases. Let us prove the other direction. Assume the right-hand side and fix $\kappa \in \mathcal{L}$ — we want to show that $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$.

For $\kappa \leq \xi[s]$, we have $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_s^\kappa = \mathring{\mathcal{S}}_s^{\kappa\restriction\xi[s]}$ and $\mathring{\mathcal{S}}_t^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_t^\kappa = \mathring{\mathcal{S}}_t^{\kappa\restriction\xi[s]}$. Since $s \trianglelefteq_\xi t$, we have $s \leq_{\xi[s]} t$, and hence $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi[s]} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi[s]}$, and hence $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$.

For κ with $\xi[s] < \kappa < \xi$, we must have $s < n_\kappa$ (as otherwise we would have $\xi[s] \geq \kappa$ by the definition of n_κ), so $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} = \langle\rangle$ (by the observation after (Def. $\leq_\xi$)) and $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ holds trivially.

For $\kappa = \xi$, $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_s^\xi$. So, the top case gives us $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ for free, and in the bottom case, since $s \leq n_\xi = n_\kappa$, $\mathring{\mathcal{S}}_s^\kappa = \langle\rangle$ and $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ holds trivially.

For $\kappa > \xi$, the bottom case gives us $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ for free, so suppose we are dealing with the top case and $s \geq n_\xi$. Noting that $n_\xi \geq m_{[\xi,\kappa)}$, part (2) of Lemma IX.49 tells us that $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$. So, $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ holds trivially. □

IX.9.4. Verifications. So far, we have shown that $\{\leq_\xi : \xi \in \mathcal{L}\}$ is a computable nested family of partial orderings. We are still missing property (♣), continuity (TS3), the existence of true stages (TS2), and completeness (i.e., that $\mathcal{T}^\xi$ is $\Delta^0_{\xi+1}$-Turing complete).

LEMMA IX.51. *The relations $\leq_\xi$ satisfy ($\clubsuit$). That is,*

$$(\forall s < t < r)\Big((s \leq_{\xi+1} r \;\&\; t \leq_\xi r) \;\Rightarrow\; s \leq_{\xi+1} t\Big).$$

PROOF. This is just a corollary of ($\clubsuit$) for $\preccurlyeq$. Here are the details. Suppose we have s, t, and r satisfying $s \leq_{\xi+1} r$ & $t \leq_\xi r$. Then, for every κ, we have $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi+1} \preccurlyeq \mathring{\mathcal{S}}_r^{\kappa\restriction\xi+1}$. Also, since $t \leq_\xi r$ implies $t \trianglelefteq_{\xi+1} r$, we have $\mathring{\mathcal{S}}_t^{\kappa\restriction\xi+1} \subseteq \mathring{\mathcal{S}}_r^{\kappa\restriction\xi+1}$, by Lemma IX.49(1). By ($\diamond$) of Lemma IX.43, we also have $s \leq_\xi t$, and by the same argument $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi+1} \subseteq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi+1}$.

$$\begin{tikzcd}
\mathring{\mathcal{S}}_s^{\kappa\restriction\xi+1} \arrow[r, phantom, "\subseteq"] \arrow[rr, bend left, dash, "\preccurlyeq"] \arrow[r, bend right, dotted, dash, "\preccurlyeq"'] & \mathring{\mathcal{S}}_t^{\kappa\restriction\xi+1} \arrow[r, phantom, "\subseteq"] & \mathring{\mathcal{S}}_r^{\kappa\restriction\xi+1}
\end{tikzcd}$$

By ($\clubsuit$) for $\preccurlyeq$, we get $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi+1} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi+1}$. □

LEMMA IX.52. *The relations $\leq_\xi$ satisfy the continuity condition. That is, for every limit ordinal $\lambda \in \mathcal{L}$,*

$$s \leq_\lambda t \qquad \Longleftrightarrow \qquad (\forall \xi < \lambda)\; s \leq_\xi t.$$

PROOF. The ($\Rightarrow$) direction follows from nestedness.

For the ($\Leftarrow$) direction, consider any ξ strictly in between $\lambda[t]$ and λ. Then $m_{[\xi,\lambda)} > t$, as otherwise we would have $\lambda[t] \geq \lambda[m_{[\xi,\lambda)}] \geq \xi$. Pick any $\kappa \in \mathcal{L}$. Let us consider two cases. If $t < m_{[\xi,\kappa)}$, then we also have $t < m_{[\lambda,\kappa)}$, because $m_{[\xi,\kappa)} \leq m_{[\lambda,\kappa)}$.[†] If $t \geq m_{[\xi,\kappa)}$, then $m_{[\xi,\kappa)} = m_{[\lambda,\kappa)}$, because $m_{[\xi,\kappa)} = n_\gamma$ for some γ that cannot be in $[\xi,\lambda)$, because $\xi > \lambda[t] \geq \lambda[n_\gamma] \geq \gamma$. In either case, $\min(t, m_{[\lambda,\kappa)}) = \min(t, m_{[\xi,\kappa)})$. It follows that, for all κ, $\mathring{\mathcal{S}}_t^{\kappa\restriction\lambda} = \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$. The same holds for s. So, $s \leq_\xi t$ implies $s \leq_\lambda t$. □

To show that the relations $\leq_\xi$ form an $\mathcal{L}$-system of true stages, we need to show that there are infinitely many ξ-true stages for all ξ. We start by showing that there are infinitely many $\trianglelefteq_\xi$-true stages for all $\xi \in \mathcal{L}$. The proof is by transfinite induction. For $\xi = 0$, all stages are 0-true.

LEMMA IX.53. *Suppose that there are infinitely many ξ-true stages for all $\xi < \lambda$. Then there are infinitely many $\trianglelefteq_\lambda$-true stages.*

Furthermore, $\mathcal{S}^\lambda \equiv_T \bigoplus_{\xi<\lambda} \mathcal{T}^\xi$ uniformly in λ.

PROOF. When λ is a successor ordinal, this holds because $\trianglelefteq_\lambda$ coincides with $\leq_{\lambda-1}$ for all $s > n_{\lambda-1}$.

Let λ be a limit ordinal in $\mathcal{L}$. We make the following two claims:

[†] For $\xi < \lambda$ we have $m_{[\xi,\kappa)} \leq m_{[\lambda,\kappa)}$ because $m_{[\xi,\kappa)}$ is a minimum taken over a larger set than $m_{[\lambda,\kappa)}$.

(1) There are infinitely many stages s which are $\lambda[s]$-true.
(2) Those are exactly the $\trianglelefteq_\lambda$-true stages.

Let γ be an ordinal in the fundamental sequence of λ, meaning that γ is of the form $\lambda[s_0]$ for some s_0. If we chose the least such s_0, we get $s_0 = n_\gamma$. Suppose also that γ is such that $n_\gamma > n_\lambda$. Let s be the first γ-true stage that is greater than or equal to n_γ — we claim that s is $\lambda[s]$-true. Since we are starting above any γ with $n_\gamma > n_\lambda$ in the fundamental sequence of λ, this will imply (1).

Let $t > s$ be $\lambda[s]$-true. Since both s and t are γ-true, we know that $s \leq_\gamma t$ — we need to show that $s \leq_{\lambda[s]} t$. We will prove that $s \leq_\xi t$ by induction on ξ with $\gamma \leq \xi \leq \lambda[s]$.

Pick ξ with $\gamma < \xi \leq \lambda[s]$ — we want to show that $s \leq_\xi t$ assuming $s \leq_\delta t$ for all $\delta < \xi$. By the induction hypothesis, we know that $s \trianglelefteq_\xi t$. For $\kappa < \xi$, we get $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ just because $s <_\kappa t$. Consider κ with $\xi \leq \kappa < \lambda$. Then $n_\kappa > s_0$, as otherwise we would have $\kappa \leq \lambda[s_0] < \xi$. Since s is the first γ-true stage after n_γ, there are no stages r with $n_\gamma \leq r <_\gamma s$. So, we have no r's with $n_\gamma \leq r \trianglelefteq_\kappa s$ (notice that $\kappa[r] \geq \gamma$). Therefore, $\mathring{\mathcal{S}}_s^\kappa = \langle\rangle$. So, we get $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ trivially.

Suppose now that $\kappa \geq \lambda$. Then $m_{[\xi,\kappa)} \leq n_\lambda \leq n_\gamma \leq s$. Since $s \trianglelefteq_\xi t$, we get $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} = \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ from Lemma IX.49 (2). So, again, we get $\mathring{\mathcal{S}}_s^{\kappa\restriction\xi} \preccurlyeq \mathring{\mathcal{S}}_t^{\kappa\restriction\xi}$ trivially.

This finishes the proof of (1). For (2), notice that if s is $\lambda[s]$-true and $t > s$ is $\lambda[t]$-true, then $s \leq_{\lambda[s]} t$, and hence $s \trianglelefteq_\lambda t$. So, the infinitely many stages s which are $\lambda[s]$-true form an infinite $\trianglelefteq_\lambda$-increasing sequence.

Conversely, suppose that r is $\trianglelefteq_\lambda$-true and that there is an infinite sequence $r \trianglelefteq_\lambda s_0 \trianglelefteq_\lambda s_1 \trianglelefteq_\lambda \cdots$. Since $r \leq_{\lambda[r]} s_0 \leq_{\lambda[r]} s_1 \leq_{\lambda[r]} s_2 \leq_{\lambda[r]} \cdots$, we have that r is a $\lambda[r]$ true stage.

Let us now prove that $\mathcal{S}^\lambda \equiv_T \bigoplus_{\xi<\lambda} \mathcal{T}^\xi$. We have $\mathcal{T}^\xi \leq_T \mathcal{S}^\lambda$ because $s \in \mathcal{T}^\xi$ if and only if $s \leq_\xi t$ for the first $t \in \mathcal{S}^\lambda$ with $t > s$ and $\lambda[t] \geq \xi$. Conversely, $\mathcal{S}^\lambda \leq_T \bigoplus_{\xi<\lambda} \mathcal{T}^\xi$ because $s \in \mathcal{S}^\lambda$ if and only if $s \in \mathcal{T}^{\lambda[s]}$, as in Claim (2). □

LEMMA IX.54. *Suppose that there are infinitely many $\trianglelefteq_\lambda$-true stages. Then there are infinitely many λ-true stages.*

Furthermore, $\mathcal{T}^\lambda \equiv_T (\mathcal{S}^\lambda)'$ uniformly in λ.

PROOF. Recall that $\mathring{\mathcal{S}}^\lambda = \langle s \in \mathcal{S}^\lambda : s \geq n_\lambda \rangle$. First, let us notice that $\mathcal{S}^\lambda \equiv_T \mathring{\mathcal{S}}^\lambda$ uniformly in λ because they coincide above n_λ and, for both sets, once you know an element, you know all the elements below it.

Let

$$t_0 \trianglelefteq_\lambda t_1 \trianglelefteq_\lambda t_2 \trianglelefteq_\lambda \cdots$$

be the sequence of all $\trianglelefteq_\lambda$-true stages that are greater than or equal to n_λ. So, $\mathring{\mathcal{S}}^\lambda = \langle t_0, t_1, t_2, \cdots \rangle$. Notice that $\mathring{\mathcal{S}}^\lambda_{t_i} = \mathring{\mathcal{S}}^\lambda \cap [0, t_i)$. By Lemma IX.50, among these stages we have

$$t_i \leq_\lambda t_j \qquad \Longleftrightarrow \qquad \mathring{\mathcal{S}}^\lambda_{t_i} \preccurlyeq \mathring{\mathcal{S}}^\lambda_{t_j}.$$

Let $X = \mathring{\mathcal{S}}^\lambda$, and let $\sigma_0 \preccurlyeq \sigma_1 \preccurlyeq \cdots \subset X$ be the X-true sub-strings of X. Let $s_i = \max(\sigma_i)$. We then have that $\sigma_i = \mathring{\mathcal{S}}^\lambda_{s_{i+1}}$, and hence that $s_0 \leq_\lambda s_1 \leq_\lambda \cdots$. So, the s_i's are the λ-true stages.

To show that $\mathcal{T}^\lambda \equiv_T (\mathring{\mathcal{S}}^\lambda)'$, the reader might have guessed that the reason is that

$$\mathcal{T}^\lambda \equiv_T \boldsymbol{\mathcal{T}}_{\mathring{\mathcal{S}}^\lambda}.$$

To see this, we have that $t \in \mathcal{T}^\lambda$ if, for the first $\sigma \in \boldsymbol{\mathcal{T}}_{\mathring{\mathcal{S}}^\lambda}$ with $t \leq \max(\sigma)$, we have $t \leq_\lambda \max(\sigma)$. Conversely, $\sigma \in \boldsymbol{\mathcal{T}}_{\mathring{\mathcal{S}}^\lambda}$ if, for the first $s \in \mathcal{T}^\lambda$ with $s > \max(\sigma)$ and $s \geq n_\lambda$, we have $\sigma = \mathring{\mathcal{S}}^\lambda_s$. □

Lemma IX.55. *For every $\lambda \in \mathcal{L}$,*

- *$\mathcal{S}^\lambda$ is Δ^0_λ-Turing complete.*
- *$\mathcal{T}^\lambda$ is $\Delta^0_{\lambda+1}$-Turing complete.*

Proof. From the previous two lemmas, we get that, for all $\lambda \in \mathcal{L}$,

$$\mathcal{S}^\lambda \equiv_T \big(\bigoplus_{\xi<\lambda} \mathcal{S}^{\xi\prime}\big).$$

All these Turing equivalences are uniform in λ. We can prove by transfinite recursion that $\mathcal{S}^\lambda$ is Δ^0_λ-Turing complete and $\mathcal{T}^\lambda$ is $\Delta^0_{\lambda+1}$-Turing complete uniformly in λ. □

CHAPTER X

Iterating the jump of a structure

We introduced the notions of jump of a relation and jump of a structure in [**Part 1**, Section II.2.1] and [**Part 1**, Chapter IX]. With all the tools we have developed so far, we can now easily iterate these notions through the computable ordinals and prove the basic results about them.

The Kleene's complete r.i.c.e. relation $\vec{\mathsf{K}}^{\mathcal{A}}$ was defined by putting together all $\Sigma^{\mathtt{c}}_1$-definable relations ([**Part 1**, Definition II.30]).

Definition X.1. We now define the *complete r.i.-*$\Sigma^{\mathtt{c}}_\alpha$ *relation* $\vec{\mathsf{K}}^{\mathcal{A}}_\alpha$ on a structure $\mathcal{A}$ by putting together all $\Sigma^{\mathtt{c}}_\alpha$-definable relations:

$$\vec{\mathsf{K}}^{\mathcal{A}}_\alpha = \{\langle i, \bar{b}\rangle : \mathcal{A} \models \varphi^{\Sigma^{\mathtt{c}}_\alpha}_{i,|\bar{b}|}(\bar{b})\} \subseteq \mathbb{N} \times A^{<\mathbb{N}},$$

where $\varphi^{\Sigma^{\mathtt{c}}_\alpha}_{i,j}(\bar{x})$ is the ith τ-$\Sigma^{\mathtt{c}}_\alpha$-formula with j free variables as in Section III.2. It will also be useful to consider the *complete r.i.-*$\Sigma^{\mathtt{c}}_{<\alpha}$ *relation* $\vec{\mathsf{K}}^{\mathcal{A}}_{<\alpha}$ by putting together all $\Sigma^{\mathtt{c}}_{<\alpha}$-definable relations:

$$\vec{\mathsf{K}}^{\mathcal{A}}_{<\alpha} = \{\langle i, \bar{b}\rangle : \mathcal{A} \models \varphi^{\Sigma^{\mathtt{c}}_{<\alpha}}_{i,|\bar{b}|}(\bar{b})\} \subseteq \mathbb{N} \times A^{<\mathbb{N}},$$

where $\varphi^{\Sigma^{\mathtt{c}}_{<\alpha}}_{i,j}(\bar{x})$ is the ith τ-$\Sigma^{\mathtt{c}}_{<\alpha}$-formula with j free variables.

Notice that if $\alpha = \beta + 1$, then $\vec{\mathsf{K}}^{\mathcal{A}}_{<\alpha}$ is essentially the same as $\vec{\mathsf{K}}^{\mathcal{A}}_\beta$, up to some computable permutation of the columns.

The jump of a structure was defined by adding $\vec{\mathsf{K}}^{\mathcal{A}}$ to it. By iterating this operation, we could define the nth jump of a structure for finite natural numbers n. For transfinite ordinals, we have the following definition.

Definition X.2. Given a τ-structure $\mathcal{A}$ and an infinite computable ordinal α, we define the *α-jump* of $\mathcal{A}$ to be the new structure obtained by adding the complete r.i.-$\Sigma^{\mathtt{c}}_{<\alpha}$ relation to it. That is, we let

$$\mathcal{A}^{(\alpha)} = (\mathcal{A}, \vec{\mathsf{K}}^{\mathcal{A}}_{<\alpha}).$$

$\mathcal{A}^{(\alpha)}$ has the same domain as $\mathcal{A}$ but a larger vocabulary. It is a τ'-structure, where τ' consists of τ together with infinitely many new relation symbols naming the relations $K_{i,j}^{<\alpha} = \{\bar{b} \in A^j : \mathcal{A} \models \varphi_{i,j}^{\Sigma^{\mathtt{c}}_{<\alpha}}(\bar{b})\}$ for $i, j \in \mathbb{N}$.

Recall that $0^{(\beta+1)}$ is $\Sigma^0_{1+\beta}$-complete and that the difference between $\beta+1$ and $1+\beta$ is not a typo (Theorem V.16). The ordinals $\beta+1$ and $1+\beta$ are the same when β is finite but different when it is infinite. If β is infinite, $1+\beta = \beta$. That is an unfortunate historical mismatch of notations between the $0^{(\delta)}$ and Σ^0_γ hierarchies. The same mismatch carries over to the β jumps of structures. Thus, if α is an infinite successor ordinal, say $\alpha = \beta+1$, then $\mathcal{A}^{(\alpha)}$ is defined by adding the Σ^0_β-complete relation, which is equivalent to the $\Sigma^0_{1+\beta}$-complete relation.* If α is a limit ordinal, then $\mathcal{A}^{(\alpha)}$ is defined by adding the Σ^0_β-complete relations for all $\beta < \alpha$ in a uniform way.

Notice that the definition of the α-th jump of a structure is independent of the presentation of $\mathcal{A}$. The isomorphism type of $\mathcal{A}^{(\alpha)}$ depends only on the isomorphism type of $\mathcal{A}$. We should mention that the isomorphism type of $\mathcal{A}^{(\alpha)}$ also depends — in an totally unessential way — on the Gödel numbering of the τ-$\Sigma^{\mathtt{c}}_{<\alpha}$-formulas, in the same unessential way that the Turing jump of a real depends on the Gödel numbering of the partial computable functions.

Remark X.3. Let us remark that the αth jump preserves effective bi-interpretability (see Definition VII.33 for the case $\Delta^{\mathtt{c}}_\alpha = \Delta^{\mathtt{c}}_1$). That is, if $\mathcal{A}$ and $\mathcal{B}$ are effectively bi-interpretable, then so are $\mathcal{A}^{(\alpha)}$ and $\mathcal{B}^{(\alpha)}$. The interpretation maps are the same. All one has to observe is that the relation $\vec{\mathsf{K}}^{\mathcal{A}^{\mathcal{B}}}_{<\alpha}$ within the copy $\mathcal{A}^{\mathcal{B}}$ is r.i.-$\Sigma^{\mathtt{c}}_{<\alpha}$ in $\mathcal{B}$ and therefore r.i. computable in $\mathcal{B}^{(\alpha)}$.

X.1. The α-jump-inversion theorems

Friedberg's jump-inversion theorem [**Part 1**, Theorem IV.6] says that every Turing degree above $0'$ is the jump of some degree. Friedberg's theorem can be generalized to transfinite iterates of the Turing jump as follows:

Theorem X.4 (Iterated-jump inversion theorem for reals. MacIntyre [**Mac77**]). *For every computable ordinal α and every real $Z \geq_T 0^{(\alpha)}$, there exists an X such that*

$$X^{(\alpha)} \equiv_T X \oplus 0^{(\alpha)} \equiv_T Z.$$

* That equivalence is assuming we have a computable isomorphism between β and $1+\beta$.

The proof of this theorem is essentially the same as that of Friedberg's theorem we gave in [**Part 1**, Theorem IV.6], but using α-Cohen generic reals instead of 1-Cohen generic reals. We will introduce α-generic ω-presentations and prove this theorem in Section X.2 below.

There are two different ways one could generalize this theorem to the α-jump of structures. We call them the first and second jump-inversion theorems. The first jump-inversion theorem is a generalization to the semi-lattice of structures ordered by effective interpretability.

Theorem X.5 (First iterated-jump inversion theorem). *For every computable ordinal α and every structure $\mathcal{A}$ which computably codes $0^{(\alpha+1)}$, there is a structure $\mathcal{C}$ whose $(\alpha+1)$th-jump is effectively bi-interpretable with $\mathcal{A}$.*

Recall that *effective bi-interpretation* is one of the strongest notions of equivalence among structures we have in computable structure theory. For a computable structure theorist, structures that are effectively bi-interpretable are *essentially* the same structure. For more background, see [**Part 1**, Section VI.3.1].

Recall that a structure $\mathcal{A}$ *computably codes* a real X if X is computable in every copy of $\mathcal{A}$ (see [**Part 1**, Section II.1.4]). Even if a structure $\mathcal{B}$ does not computably code $0^{(\alpha+1)}$, we can still apply the theorem above to get a structure $\mathcal{C}$ whose $(\alpha+1)$th-jump is effectively bi-interpretable with $\mathcal{B} \oplus 0^{(\alpha+1)}$, where $\mathcal{B} \oplus 0^{(\alpha+1)}$ is built by adding zero-ary relations to $\mathcal{A}$ coding $0^{(\alpha+1)}$. To be precise, $\mathcal{A} = (\mathcal{B}, R)$ where $R = 0^{(\alpha+1)} \times \{\langle\rangle\} \subseteq \mathbb{N} \times B^{<\mathbb{N}}$.

We prove this theorem in Section X.3.

The second α-jump-inversion theorem is not a generalization of the usual jump-inversion theorem to a more general class of degrees but a generalization in the sense that, given $Z \in 2^{\mathbb{N}}$, it yields $X \in 2^{\mathbb{N}}$ with $X^{(\alpha)} \equiv_T Z$ and some extra properties.

Theorem X.6 (Second iterated-jump inversion theorem). *If $Z \in 2^{\mathbb{N}}$ computes a copy of $\mathcal{B}^{(\alpha)}$, then there is an $X \in 2^{\mathbb{N}}$ satisfying $X^{(\alpha)} \equiv_T Z$ that computes a copy $\mathcal{G}$ of $\mathcal{B}$.*

We will prove this theorem in page 196 below.

X.2. $\Sigma^{\mathtt{c}}_{\alpha}$-generics

In this section, we prove the iterated version of Friedberg's jump inversion theorem for reals, Theorem X.4, and we prove the second iterated-jump inversion theorem for structures, Theorem X.6. To prove these theorems, we need a new tool: $\Sigma^{\mathsf{c}}_{<\alpha}$-generics.

Definition X.7. An injective enumeration g of a structure $\mathcal{A}$ is $\Sigma^{\mathsf{c}}_{\alpha}$*-generic* if for every $\Sigma^{\mathsf{c}}_{\alpha}$-definable relation $R \subseteq A^{\star}$, g either forces in or forces out of R. That is, either there is an initial segment of g in R (forces in) or there is an initial segment of g with no extensions in R (forces out). We say that g is $\Sigma^{\mathsf{c}}_{<\alpha}$*-generic* if, for every $\Sigma^{\mathsf{c}}_{<\alpha}$-definable relation $R \subseteq A^{\star}$, g forces either in or out of R.

If $\varphi(\dot{\mathcal{G}})$ is an $\mathbb{N}$-$\Sigma^{\mathsf{c}}_{<\alpha}$-sentence, then the set $R \subset A^{\star}$ of $\bar{p}$ forcing φ is $\Sigma^{\mathsf{c}}_{<\alpha}$-definable by the formula *Force*$_{\varphi}$. Thus, if g is a $\Sigma^{\mathsf{c}}_{<\alpha}$-generic enumeration, then for every $\mathbb{N}$-$\Sigma^{\mathsf{c}}_{<\alpha}$-sentence $\varphi(\dot{\mathcal{G}})$, g has an initial segment $\bar{p}$ that either forces φ or has no extension that forces φ, and hence forces $\neg\varphi$. We thus get the following version of Corollary VII.11.

Lemma X.8. *If g is $\Sigma^{\mathsf{c}}_{<\alpha}$-generic, for every $\mathbb{N}$-$\Sigma^{\mathsf{c}}_{<\alpha}$-sentence φ, there is a $\bar{p} \subset g$ that decides φ.*

One can then adapt Lemma VII.12 and the forcing-equals-truth theorem, Theorem VII.13, as follows:

Theorem X.9 (Forcing-equals-truth for $\Sigma^{\mathsf{c}}_{<\alpha}$-generics). *If g is an $\Sigma^{\mathsf{c}}_{<\alpha}$–generic enumeration of $\mathcal{A}$, $\mathcal{G} = g^{-1}(\mathcal{A})$, and $\varphi(\dot{\mathcal{G}})$ is an $\mathbb{N}$-$\Sigma^{\mathsf{c}}_{\alpha}$-sentence, then*

$$\varphi(\mathcal{G}) \quad \iff \quad (\exists \bar{p} \subset g)\ \bar{p} \Vdash \varphi.$$

Proof. The proof is identical to that of Theorem VII.13 using Lemma X.8 above instead of Corollary VII.11, and using transfinite induction only up level $\Sigma^{\mathsf{c}}_{<\alpha}$ in the proof of Lemma VII.12.

The reason this theorem works for $\Sigma^{\mathsf{c}}_{\alpha}$-sentences, and not just $\Sigma^{\mathsf{c}}_{<\alpha}$-sentences, is that the Σ-case of the transfinite induction does not need to use the genericity of $\mathcal{G}$. That is, if φ is of the form $\bigvee\!\!\!\bigvee_i \psi_i$, then $\varphi(\mathcal{G})$ holds if and only if $\psi_i(\mathcal{G})$ holds for some i, and $\bar{p} \Vdash \varphi$ if and only if $\bar{p} \Vdash \psi_i$ for some i. □

The advantage of $\Sigma^{\mathsf{c}}_{<\alpha}$-generics over $\mathcal{L}_{\mathsf{c},\omega}$-generics is that $\Sigma^{\mathsf{c}}_{<\alpha}$-generics are easier to compute:

Lemma X.10. *Let α be an infinite computable ordinal. Every ω-presentation $\mathcal{A}$ has a $\Sigma^{\mathsf{c}}_{<\alpha}$-generic enumeration computable in $D(\mathcal{A}^{(\alpha)})$.*

Proof. We build g as the union of an increasing sequence $\{\bar{p}_s : s \in \mathbb{N}\}$ with $\bar{p}_s \in A^{\star}$. At stage $s+1 = 2e$, we define $\bar{p}_{s+1}$ to decide the e-th $\Sigma^{\mathsf{c}}_{<\alpha}$-definable relation $R_e \subseteq A^{\star}$ as follows: If there is a $\bar{q} \supseteq \bar{p}_s$ with $\bar{q} \in R_e$, we let $\bar{p}_{s+1} = \bar{q}$. Otherwise, we let $\bar{p}_{s+1} = \bar{p}_s$. At stage

$s+1 = 2e+1$, let $\bar{p}_{s+1} = \bar{p}_s{}^\frown a$, where a is the $\leq_{\mathbb{N}}$-least element of the ω-presentation $\mathcal{A}$ which has not been included in $\bar{p}_s$. Finally, we let $g = \bigcup_s \bar{p}_s \in A^{\mathbb{N}}$. It is not hard to check that g is one-to-one, onto, and $\Sigma^{\mathsf{c}}_{<\alpha}$-generic.

To carry out this construction, we need to check at each stage $s+1$ whether there exists a $\bar{q} \supseteq \bar{p}_s$ with $\bar{q} \in R_e$ or not. The set of $\bar{p}$'s such that $\exists \bar{q} \supseteq \bar{p}$ $(\bar{q} \in R_e)$, namely the downward closure of R_e, is $\Sigma^{\mathsf{c}}_{<\alpha}$-definable, and its index can be obtained uniformly from e. Hence, $\vec{\mathsf{K}}^{\mathcal{A}}_{<\alpha}$ can decide whether $\bar{p}_s$ belongs to the downward closure of R_e or not. The whole construction is thus computable in $\vec{\mathsf{K}}^{\mathcal{A}}_{<\alpha}$. □

Lemma X.11. *Let α be an infinite computable ordinal. If $\mathcal{G}$ is a $\Sigma^{\mathsf{c}}_{<\alpha}$-generic ω-presentation, then*

$$D(\mathcal{G}^{(\alpha)}) \equiv_T D(\mathcal{G})^{(\alpha)}.$$

Proof. That $\vec{\mathsf{K}}^{\mathcal{G}}_{<\alpha} \leq_T D(\mathcal{G})^{(\alpha)}$ follows immediately from Lemma V.6 and Theorem V.16.

For the other direction, recall that, for each m, there is an $\mathbb{N}$-Σ^0_β sentences $\varphi_m(\dot{\mathcal{G}})$ for some $\beta < \alpha$ that holds of $D(\mathcal{G})$ if and only if $m \in D(\mathcal{G})^{(\alpha)}$. We can find these sentences computably in m (Lemma V.15). We then have that

$$\begin{aligned} m \in D(\mathcal{G})^{(\alpha)} \quad &\iff \quad \bigvee\!\!\!\bigvee_{n\in\mathbb{N}} \Big(g \restriction n \Vdash \varphi_m\Big) \\ &\iff \quad \bigvee\!\!\!\bigvee_{n\in\mathbb{N}} \Big(\mathcal{G} \models \mathit{Force}_{\varphi_m}(\langle 0, ..., n\rangle)\Big) \\ &\iff \quad \mathcal{G} \models \bigvee\!\!\!\bigvee_{n\in\mathbb{N}} \mathit{Force}_{\varphi_m}(\langle 0, ..., n\rangle), \end{aligned}$$

where the first equivalence follows from the forcing-equals-truth theorem, the second from the definition of $\mathit{Force}_{\varphi_m}$, and the third by the definition of forcing a Σ-formula. The sentence $\bigvee\!\!\!\bigvee_{n\in\mathbb{N}} \mathit{Force}_{\varphi_m}(\langle 0, ..., n\rangle)$ is $\Sigma^{\mathsf{c}}_\beta$, and hence we can decide whether it holds or not using $\vec{\mathsf{K}}^{\mathcal{G}}_{<\alpha}$. □

The proof of the iterated-jump inversion theorem for reals uses $\Sigma^{\mathsf{c}}_{<\alpha}$-Cohen-generic reals, the same way the proof of the Friedberg jump inversion theorem [**Part 1**, Theorem IV.6] used 1-generic reals. We could redevelop the whole theory of $\Sigma^{\mathsf{c}}_{<\alpha}$-generics for the case of Cohen forcing, or we could just note that Cohen generic reals are essentially the same as the generic enumerations of the structure $\mathcal{C} = (C; P)$, which has only one unary relation symbol, P, and which has infinitely many elements in P and infinitely many outside of P. The enumerations g of

$\mathcal{C}$ are in one-to-one correspondence with the infinite binary sequences $G \in 2^{\mathbb{N}}$, where $G(n) = 1 \iff g(n) \in P$. We say that $G \in 2^{\mathbb{N}}$ is $\Sigma^{\mathsf{c}}_{<\alpha}$-Cohen-generic if it can be obtained this way from a $\Sigma^{\mathsf{c}}_{<\alpha}$-generic enumeration g of $\mathcal{C}$.

Recall that the iterated-jump inversion theorem for reals, Theorem X.4, states that, for every $Z \geq_T 0^{(\alpha)}$, there exists a $G \in 2^{\mathbb{N}}$ such that

$$G^{(\alpha)} \equiv_T G \oplus 0^{(\alpha)} \equiv_T Z.$$

Proof of Theorem X.4. The iterated-jump inversion theorem for reals. Notice that since $\mathcal{C}$ is computable, $D(\mathcal{C}^{(\alpha)}) \equiv_T 0^{(\alpha)}$. Let g be a $\Sigma^{\mathsf{c}}_{<\alpha}$-generic enumeration of $\mathcal{C}$ that is computable in Z and is built as follows: Carry out the construction of Lemma X.10 step by step except that, at stages $s+1 = 2e+1$, define $\bar{p}_{s+1} = \bar{p}_s{}^\frown a$, where a is the $\leq_{\mathbb{N}}$-least element of $P^{\mathcal{C}} \smallsetminus \bar{p}_s$ if $e \in Z$, and the $\leq_{\mathbb{N}}$-least element of $(C \smallsetminus P^{\mathcal{C}}) \smallsetminus \bar{p}_s$ if $e \notin Z$.[†] Let $\mathcal{G}$ be the associated $\Sigma^{\mathsf{c}}_{<\alpha}$-generic ω-presentation and G be the associated $\Sigma^{\mathsf{c}}_{<\alpha}$-Cohen-generic.

Clearly $G^{(\alpha)} \geq_T G \oplus 0^{(\alpha)}$. From $G \oplus 0^{(\alpha)}$, we can compute Z because $G \oplus 0^{(\alpha)}$ can reconstruct the sequence $\langle \bar{p}_s : s \in \mathbb{N} \rangle$ in the construction of g: Using $0^{(\alpha)}$, we can run the even stages $s+1 = 2e$ of the construction of g, and using G, we can figure out if, at the odd stages $s+1 = 2e+1$ we picked an element a from P or from $C \smallsetminus P$ .We can thus figure out whether $e \in Z$ or not. Finally, to see that Z can compute $G^{(\alpha)}$, we first note that $G \equiv_T D(\mathcal{G})$ and then that

$$G^{(\alpha)} \equiv_T D(\mathcal{G})^{(\alpha)} \equiv_T D(\mathcal{G}^{(\alpha)}) \equiv_T \vec{\mathsf{K}}^{\mathcal{G}}_{<\alpha} \leq_T \vec{\mathsf{K}}^{\mathcal{C}}_{<\alpha} \oplus g \leq_T Z,$$

where the second Turing equivalence uses Lemma X.11 and that $\mathcal{G}$ is $\Sigma^{\mathsf{c}}_{<\alpha}$-generic, the first Turing inequality uses that $\vec{\mathsf{K}}^{\mathcal{G}}_{<\alpha} = g^{-1}(\vec{\mathsf{K}}^{\mathcal{C}}_{<\alpha})$, and the last Turing inequality uses that $\vec{\mathsf{K}}^{\mathcal{C}}_{<\alpha} \equiv_T 0^{(\alpha)} \leq_T Z$. □

We can now prove the second iterated-jump inversion theorem for structures, Theorem X.6. Recall that it states that if Z computes $\mathcal{B}^{(\alpha)}$, then there is a real X with $X^{(\alpha)} \equiv_T Z$ which computes a copy $\mathcal{G}$ of $\mathcal{B}$.

Proof of Theorem X.6. . The second iterated-jump inversion theorem. Consider a $\Sigma^{\mathsf{c}}_{<\alpha}$-generic enumeration g of $\mathcal{B}$ computable in $D(\mathcal{B}^{(\alpha)})$, and hence in Z. Let $\mathcal{G} = g^{-1}(\mathcal{B})$ and

$$Y = D(\mathcal{G}).$$

Since $\vec{\mathsf{K}}^{\mathcal{G}}_{<\alpha} = g^{-1}(\vec{\mathsf{K}}^{\mathcal{B}}_{<\alpha})$, we have that

$$D(\mathcal{G}^{(\alpha)}) \equiv_T \vec{\mathsf{K}}^{\mathcal{G}}_{<\alpha} \leq_T \vec{\mathsf{K}}^{\mathcal{B}}_{<\alpha} \oplus g \leq_T Z.$$

[†] To see why g is onto notice that, since Z is non-computable, it is infinite and co-infinite.

Since $\mathcal{G}$ is $\Sigma^{\mathtt{c}}_{<\alpha}$-generic,

$$D(\mathcal{G}^{(\alpha)}) \equiv_T D(\mathcal{G})^{(\alpha)} = Y^{(\alpha)},$$

as proved in Lemma X.11. Thus, $Y^{(\alpha)} \leq_T Z$. By the α-jump inversion theorem for reals (Theorem X.4) relativized to Y, there is a real $X \geq_T Y$ with $X^{(\alpha)} \equiv_T Z$. This X computes $\mathcal{G}$, a copy of $\mathcal{B}$. □

As a corollary, we get that the degree spectrum of the α-jump of a structure is what it should be: the set of α-jumps of the degrees in the spectrum of the original structure.[‡]

Corollary X.12. *For every structure $\mathcal{B}$,*

$$DgSp(\mathcal{B}^{(\alpha)}) = \{Z \in 2^{\mathbb{N}} : Z \geq_T X^{(\alpha)} \text{ for some } X \in DgSp(\mathcal{B})\}.$$

Proof. For the $\supseteq$-inclusion, it is clear that if $Z \geq_T X^{(\alpha)}$ for some $X \in DgSp(\mathcal{B})$, then Z computes a copy of $\mathcal{B}^{(\alpha)}$. For the $\subseteq$-inclusion, if Z computes a copy of $\mathcal{B}^{(\alpha)}$, then by the theorem, there is an X such that $Z \geq_T X^{(\alpha)}$ and $X \in DgSp(\mathcal{B})$. □

X.3. The first iterated-jump inversion theorem

Recall that the first iterated-jump inversion theorem states that if a structure $\mathcal{A}$ computably codes $0^{(\alpha+1)}$, it is effectively bi-interpretable with the $(\alpha+1)$st jump of another structure $\mathcal{C}$. The main ideas in the proof of this theorem are due to Goncharov, Harizanov, Knight, McCoy, R. Miller, and Solomon [**GHK⁺05**]. The notions of jump of a structure or effectively bi-interpretable did not exist back then, so they did not really prove this same theorem, but the construction of the structure $\mathcal{C}$ below is theirs.

Proof of Theorem X.5. The first iterated-jump inversion theorem. If α is finite, the theorem follows by iterating the first single-jump-inversion theorem [**Part 1**, Theorem IV.6] α times. Suppose α is infinite, and hence that the $(\alpha+1)$st jump of a structure is built by adding to it the complete r.i.-$\Sigma^{\mathtt{c}}_{\alpha}$ relation $\vec{\mathsf{K}}^{\mathcal{A}}_{\alpha}$.

Every structure is effectively bi-interpretable with a graph [**Part 1**, Theorem VI.27]. Therefore, we may assume $\mathcal{A}$ is a graph $(A; E)$ with domain A and edge relation E. The key idea behind this proof is the following: Pick two structures such that distinguishing between their copies is $\Delta^0_{\alpha+1}$-complete, and attach to each pair of vertices of A one of the two structures, depending on whether or not there is an edge between them. Let us look at the details.

[‡] Recall from [**Part 1**, Definition V.1] that the *degree spectrum* of a structure $\mathcal{M}$ is defined as $DgSp(\mathcal{M}) = \{X \in 2^{\mathbb{N}} : X \text{ computes a copy of } \mathcal{M}\}$.

Consider two computable, rigid,[§] uniformly $\Delta^0_{\alpha+1}$-categorical[¶] structures, $\mathcal{K}$ and $\mathcal{L}$, which are α-back-and-forth equivalent but $(\alpha+1)$-back-and-forth incomparable. So, we have that $\mathcal{K} \equiv_\alpha \mathcal{L}$, $\mathcal{K} \not\leq_{\alpha+1} \mathcal{L}$ and $\mathcal{L} \not\leq_{\alpha+1} \mathcal{K}$. Assume also that their back-and-forth relations are computable up to α and that there are computable $\Pi^{\mathtt{c}}_{\alpha+1}$ formulas witnessing that $\mathcal{K}$ and $\mathcal{L}$ are $\leq_{\alpha+1}$-incomparable, i.e., $\Pi^{\mathtt{c}}_{\alpha+1}$ formulas φ and ψ such that $\mathcal{K} \models \varphi \wedge \neg\psi$ and $\mathcal{L} \models \neg\varphi \wedge \psi$. Examples of such structures will be built in Lemma X.14 below. Just to simplify the notation, let us assume these structures are linear orderings, as are the ones we will build in Lemma X.14. Let us use $\trianglelefteq$ to denote the ordering relation on these structures, so we have $\mathcal{K} = (K; \trianglelefteq^{\mathcal{K}})$ and $\mathcal{L} = (L; \trianglelefteq^{\mathcal{L}})$. If we are given a computable ω-presentation that we know is isomorphic to either $\mathcal{K}$ or $\mathcal{L}$, we can use the $\Pi^{\mathtt{c}}_{\alpha+1}$ formulas φ and ψ to tell whether we have a copy of $\mathcal{K}$ or of $\mathcal{L}$ in a $\Delta^0_{\alpha+1}$ way. Conversely, from the pair-of-structures theorem (Theorem VIII.7), we know that distinguishing between computable ω-presentations of $\mathcal{K}$ and $\mathcal{L}$ is $\Delta^0_{\alpha+1}$-hard. So, distinguishing between computable ω-presentations of $\mathcal{K}$ and $\mathcal{L}$ is $\Delta^0_{\alpha+1}$-complete.

We can now define $\mathcal{C}$ by removing the edge relation E and attaching to each pair of vertices of A one of these two structures, depending on whether there is an edge between the two vertices or not. We define $\mathcal{C}$ as $(C; A, R)$, where A is a unary relation and R a 4-ary relation. The domain C of $\mathcal{C}$ consists of the disjoint union of the domain A of $\mathcal{A}$ and another set B. We use the unary relation A to identify the elements of A. Partition B into infinitely many set $B_{a,b}$ indexed by $\langle a,b\rangle \in A^2$. On $B_{a,b}$, define a binary relation $R_{a,b}$ such that $(B_{a,b}; R_{a,b})$ is a structure isomorphic to either $\mathcal{K}$ or $\mathcal{L}$, and it is isomorphic to $\mathcal{K}$ if and only if $\langle a,b\rangle \in E$. Finally, we define the 4-ary relation

$$R \subseteq A \times A \times B \times B$$

by putting together the relations $R_{a,b}$. That is $R = \{\langle a,b,c,d\rangle : \langle c,d\rangle \in R_{a,b}\}$.

[§] A structure is rigid if it has no non-trivial automorphisms.

[¶] A computable structure $\mathcal{A}$ is *uniformly Δ^0_β-categorical* if there is a Δ^0_β operator Γ such that, for all copies $\mathcal{G}$ of $\mathcal{A}$, $\Gamma^{D(\mathcal{G})}$ is an isomorphism between $\mathcal{G}$ and $\mathcal{A}$.

$\mathcal{C}$ can be easily effectively interpreted in $\mathcal{A}$ as follows. Let $B = \mathbb{N} \times A^2$, and let $C^{\mathcal{A}} = A \cup B$. Then define R as follows:

$$\begin{aligned} R^{\mathcal{C}^{\mathcal{A}}} \;=\; & \{\langle a, b, \langle n, a, b\rangle, \langle m, a, b\rangle\rangle \in A^2 \times B^2 : \\ & \qquad\qquad \text{for } \langle a, b\rangle \in E \;\;\&\;\; \langle n, m\rangle \in \trianglelefteq^{\mathcal{K}}\} \\ \cup \;\; & \{\langle a, b, \langle n, a, b\rangle, \langle m, a, b\rangle\rangle \in A^2 \times B^2 : \\ & \qquad\qquad \text{for } \langle a, b\rangle \in A^2 \smallsetminus E \;\;\&\;\; \langle n, m\rangle \in \trianglelefteq^{\mathcal{L}}\}. \end{aligned}$$

To show that this is actually an effective interpretation of $\mathcal{C}^{(\alpha+1)}$, and not just of $\mathcal{C}$, we need to show that $\vec{\mathsf{K}}_\alpha^{\mathcal{C}^{\mathcal{A}}}$ (viewed as a relation in $\mathbb{N} \times A^{<\mathbb{N}}$) is r.i. computable in $\mathcal{A}$. To see this, fix an ω-presentation of $\mathcal{A}$. The construction above then gives us an ω-presentation $\mathcal{C}^{\mathcal{A}}$ of $\mathcal{C}$. Use the α-jump-inversion theorem for reals (Theorem X.4) to get an oracle $X \in 2^{\mathbb{N}}$ such that $X^{(\alpha+1)} \equiv_T D(\mathcal{A})$ (we can do this because $\mathcal{A}$ computably codes $0^{(\alpha+1)}$ by assumption). We will now construct $\widetilde{\mathcal{C}}$, a second copy of $\mathcal{C}$ that is computable in X. For each $\langle a, b\rangle \in A^2$, $X^{(\alpha+1)}$ knows whether or not $\langle a, b\rangle \in E$, and hence computably in X, we can uniformly build structures $\widetilde{\mathcal{B}}_{a,b}$ for each $\langle a, b\rangle \in A^2$ such that

$$\widetilde{\mathcal{B}}_{a,b} \cong \begin{cases} \mathcal{K} & \text{if } \langle a, b\rangle \in E, \\ \mathcal{L} & \text{if } \langle a, b\rangle \notin E. \end{cases}$$

To do this, we use the pair-of-structures theorem (Theorem VIII.8).

We then define $\widetilde{\mathcal{C}}$ by putting the set A together with disjoint copies of all the $\widetilde{\mathcal{B}}_{a,b}$ for $\langle a, b\rangle \in A^2$ and defining

$$\widetilde{R}(a, b, n, m) \iff \langle n, m\rangle \in \trianglelefteq^{\widetilde{\mathcal{B}}_{a,b}}.$$

An important point is that $D(\mathcal{A})$ can compute an isomorphism between $\widetilde{\mathcal{C}}$ and $\mathcal{C}^{\mathcal{A}}$. This is because $X^{(\alpha+1)}$ can compute isomorphisms between $\widetilde{\mathcal{B}}_{a,b}$ and $\mathcal{B}_{a,b}$ for all $\langle a, b\rangle \in A^2$, as noted in Remark VIII.9. Since $D(\widetilde{\mathcal{C}}) \leq_T X$, we have that $\vec{\mathsf{K}}_\alpha^{\widetilde{\mathcal{C}}}$ is computable in $X^{(\alpha+1)}$, and hence in $D(\mathcal{A})$. Going through the isomorphism between $\widetilde{\mathcal{C}}$ and $\mathcal{C}^{\mathcal{A}}$, we get that $\vec{\mathsf{K}}_\alpha^{\mathcal{C}^{\mathcal{A}}}$ is also computable in $D(\mathcal{A})$. Since this worked for every ω-presentation of $\mathcal{A}$, we have that $\vec{\mathsf{K}}_\alpha^{\mathcal{C}^{\mathcal{A}}}$ is r.i. computable in $\mathcal{A}$. This proves that we have an effective interpretation of $\mathcal{C}^{(\alpha+1)}$ in $\mathcal{A}$.

The effective interpretation of $\mathcal{A}$ within $\mathcal{C}^{(\alpha+1)}$ is more direct. The domain of the interpretation is, of course, A itself, as identified by the relation A within $\mathcal{C}$. Notice that E is $\Delta^{\mathsf{c}}_{\alpha+1}$ in $\mathcal{C}$. This is because, to decide if $\langle a, b\rangle \in A^2$, we need to decide whether $\mathcal{B}_{a,b} \cong \mathcal{K}$ or $\mathcal{B}_{a,b} \cong \mathcal{L}$, which we can do by checking which of the $\Pi^{\mathsf{c}}_{\alpha+1}$ sentences φ and ψ holds on the structure $\mathcal{B}_{a,b}$.

The last step is to check that these two effective interpretations form an effective *bi*-interpretation, i.e., that the composition of the isomorphisms are r.i. computable in the respective structures. First, notice that the interpretation of $\mathcal{A}$ inside $\mathcal{C}$ inside $\mathcal{A}$ is the identity, and hence obviously r.i. computable in $\mathcal{A}$. Second, for the interpretation of $\mathcal{C}$ inside $\mathcal{A}$ inside $\mathcal{C}$, the A-part stays the same. The copies of $\mathcal{B}_{a,b}$ are not the same, but since they are isomorphic to either $\mathcal{K}$ or $\mathcal{L}$, and $\mathcal{K}$ and $\mathcal{L}$ are rigid and uniformly relatively $\Delta^0_{\alpha+1}$-categorical, the unique isomorphism between them can be computed in $\mathcal{C}^{(\alpha+1)}$. Let us see why this is the case. Since $\mathcal{K}$ and $\mathcal{L}$ are uniformly $\Delta^0_{\alpha+1}$-categorical, they have c.e. Scott families of τ-$\Sigma^{\mathtt{c}}_{\alpha+1}$ formulas (Remark VII.22). These formulas are $\Sigma^{\mathtt{c}}_1$ over the vocabulary of $\mathcal{C}^{(\alpha+1)}$, which contains symbols for all $\Sigma^{\mathtt{c}}_{\alpha}$-relations. So, $\mathcal{K}$ and $\mathcal{L}$ are computably categorical relative to $D(\mathcal{C}^{(\alpha+1)})$. We thus have that the unique isomorphism from $\mathcal{C}$ to the copy of $\mathcal{C}$ inside $\mathcal{A}$ inside $\mathcal{C}$ is r.i.-computable in $\mathcal{C}^{(\alpha+1)}$. □

Remark X.13. The structure we built in the proof above is sometimes called a *strong* $(\alpha+1)$st jump inversion. The reason is that it satisfies the following stronger property:

> For every real X, if $X^{(\alpha+1)}$ computes a copy of $\mathcal{A}$,
> then X computes a copy of $\mathcal{C}$.

We showed that this was the case when we built $\widetilde{\mathcal{C}}$ in the proof above.

Lemma X.14. *For every computable ordinal α, there exist computable, rigid, uniformly $\Delta^0_{\alpha+1}$-categorical linear orderings which are α-back-and-forth equivalent and $(\alpha+1)$-back-and-forth incomparable. Furthermore, the α back-and-forth relations are computable up to α, and the $(\alpha+1)$ back-and-forth incomparability is witnessed by two computable $\Pi^{\mathtt{c}}_{\alpha+1}$ formulas.*

Proof. Let us start with an intermediate step. Let us first show that there exist computable ordinals $\mathcal{A}$ and $\mathcal{B}$ with $\mathcal{A} <_{\alpha+1} \mathcal{B}$ which are rigid, uniformly relatively $\Delta^0_{\alpha+1}$-categorical, and have the back-and-forth relations computable up to $\alpha+1$. We consider two cases depending on whether α is even or odd.

If $\alpha+1 = 2\beta+1$, consider the linear orderings

$$\mathcal{A} = \omega^\beta + \omega^\beta \quad \text{and} \quad \mathcal{B} = \omega^\beta.$$

It follows from Lemma II.38 that $\omega^\beta \cdot 2 <_{2\beta+1} \omega^\beta$. Their parametrized Scott rank is 2β (Corollary II.40). To get this Scott rank, $\mathcal{B}$ needs no parameters, while $\mathcal{A}$ needs one parameter, namely the first element of the second copy of ω^β. This parameter is $\Pi^0_{2\beta}$, as it is the only point which is a β-limit (see Exercise II.20). Then, as in Case 2 on page 44, we get that these structures have parameterless Scott rank $2\beta+1$. This means that every element has a $\Sigma^{\mathtt{in}}_{2\beta+1}$ definition. These definitions are actually $\Sigma^{\mathtt{c}}_{2\beta+1}$ — we leave it to the reader to verify this.[‖] So, the structures are uniformly $\Delta^0_{\alpha+1}$-categorical. It is also easy to see that there is a computable $\Pi^{\mathtt{c}}_{2\beta+1}$ sentence true in $\mathcal{B}$, false in $\mathcal{A}$, saying that there is no β-limit.

If $\alpha+1=2\beta+2$, consider the linear orderings

$$\mathcal{A}=\omega^{\beta+1}+\omega^{\beta}, \quad \text{and} \quad \mathcal{B}=\omega^{\beta+1}.$$

It follows from Exercise II.44 that $\omega^{\beta+1}+\omega^{\beta} <_{2\beta+1} \omega^{\beta+1}$.[**] Their parametrized Scott rank is $2\beta+2$ (Corollary II.40). $\mathcal{B}$ does not need parameters, while $\mathcal{A}$ needs one parameter, namely the first element of the rightmost copy of ω^{β}. This parameter is $\Pi^0_{2\beta+1}$, as it is the rightmost β-limit (see Exercise II.20). Then, as in Case 3 on page 46, we get that these structures have parameterless Scott rank $2\beta+2$. This means that every element has a $\Sigma^{\mathtt{in}}_{2\beta+2}$ definition. These definitions are actually $\Sigma^{\mathtt{c}}_{2\beta+2}$ — again, we leave it to the reader to verify this. So, the structures are uniformly $\Delta^0_{\alpha+1}$-categorical. It is also easy to see that there is a computable $\Pi^{\mathtt{c}}_{2\beta+2}$ sentence true in $\mathcal{B}$, false in $\mathcal{A}$, saying that there is no rightmost β-limit.

All well-orders are rigid. The back-and-forth relations are computable up to $\alpha+1$, as they can be calculated using Exercise II.44. So, $\mathcal{A}$ and $\mathcal{B}$ satisfy all the properties we wanted them to. Finally, let

$$\mathcal{K}=\mathcal{A}+1+\mathcal{B}^* \quad \text{and} \quad \mathcal{L}=\mathcal{B}+1+\mathcal{A}^*.$$

Here $\mathcal{B}^*$ is the reverse linear ordering, that is, $(B;\trianglelefteq)^*=(B;\trianglerighteq)$. It is not hard to see that, given infinite well-orderings $\mathcal{C}$, $\mathcal{D}$, $\mathcal{E}$, and $\mathcal{F}$, $\mathcal{C}+1+\mathcal{D}^* \leq_\alpha \mathcal{E}+1+\mathcal{F}^*$ if and only if $\mathcal{C}\leq_\alpha\mathcal{E}$ and $\mathcal{D}\leq_\alpha\mathcal{F}$.[††] Since the structures $\mathcal{A}$ and $\mathcal{B}$ that we defined above satisfy $\mathcal{A}\equiv_\alpha\mathcal{B}$ and $\mathcal{B}\not\leq_{\alpha+1}\mathcal{A}$, we get $\mathcal{K}\equiv_\alpha\mathcal{L}$, $\mathcal{K}\not\leq_{\alpha+1}\mathcal{L}$ and $\mathcal{L}\not\leq_{\alpha+1}\mathcal{K}$. Rigidity, uniform $\Delta^0_{\alpha+1}$ categoricity,[‡‡] and the computability of the back-and-forth relations remain true in $\mathcal{K}$ and $\mathcal{L}$. □

[‖] These definitions say that the interval to the left of the point or between the middle element and the point has a certain order-type.

[**] When we apply Exercise II.44, we are in the situation where $\alpha=\beta$, $\delta=0$, $\beta_1=\omega+1$, $\gamma_1=\omega$, $m=1$, and $n=0$.

[††] That $\mathcal{C}\leq_\alpha\mathcal{E}$ and $\mathcal{D}\leq_\alpha\mathcal{F}$ imply $\mathcal{C}+1+\mathcal{D}^*\leq_\alpha\mathcal{E}+1+\mathcal{F}^*$ follows from Lemma II.37. That $\mathcal{C}+1+\mathcal{D}^*\leq_\alpha\mathcal{E}+1+\mathcal{F}^*$ implies $\mathcal{C}\leq_\alpha\mathcal{E}$ and $\mathcal{D}\leq_\alpha\mathcal{F}$ follows from the observation that the 1 in the middle is the only point that is a left- and right-limit in both linear orderings, and hence has a $\Pi^{\mathtt{c}}_2$ definition and hence must be matched.

[‡‡] $\Delta^0_{\alpha+1}$ categoricity is uniform because, since the middle 1 is $\Pi^{\mathtt{c}}_2$-definable, we easily pick it up first using a $\Delta^0_{\alpha+1}$ oracle.

Goncharov, Harizanov, Knight, McCoy, R. Miller, and Solomon [**GHK$^+$05**] introduced this construction mainly to prove the following result:

Corollary X.15. *There is a structure that is $\Delta^0_{\alpha+1}$-categorical but not relatively so.*

Proof. Assume α is infinite so that $0^{(\alpha+1)}$ is $\Delta^0_{\alpha+1}$-Turing complete. For finite n, the proof is the same, as long as we keep in mind that it is $0^{(n)}$ who is Δ^0_{n+1} Turing complete, instead of $0^{(n+1)}$.

Using ideas of Nurtazin [**Nur74**], Goncharov showed that there exists a computably categorical structure that is not relatively computably categorical [**Part 1**, Theorem VIII.14]. Relativizing [**Part 1**, Theorem VIII.14] to $0^{(\alpha+1)}$, we get a $0^{(\alpha+1)}$-computable structure $\mathcal{A}$ that is $0^{(\alpha+1)}$-computably categorical, but not $0^{(\alpha+1)}$-relatively computably categorical. Let $\mathcal{C}$ be the structure built from $\mathcal{A}$ in the proof of the first $(\alpha+1)$-jump-inversion theorem. From Remark X.13, we get that $\mathcal{C}$ has a computable presentation. Thus, we may assume that $\mathcal{C}$ is that computable ω-presentation and that $\mathcal{A}$ is the ω-presentation obtained from the effective bi-interpretation with $\mathcal{C}^{(\alpha+1)}$. We claim that $\mathcal{C}$ is $\Delta^0_{\alpha+1}$-categorical but not relatively so.

To prove that $\mathcal{C}$ is $\Delta^0_{\alpha+1}$ categorical, let $\hat{\mathcal{C}}$ be a computable copy of $\mathcal{C}$. Then, $\hat{\mathcal{C}}$ is associated via the effective bi-interpretation with a copy $\hat{\mathcal{A}}$ of $\mathcal{A}$. Notice that the ω-presentation $\hat{\mathcal{A}}$ is computable in $0^{(\alpha+1)}$. Since $\mathcal{A}$ is $0^{(\alpha+1)}$-computably categorical, $0^{(\alpha+1)}$ can compute an isomorphism between $\mathcal{A}$ and $\hat{\mathcal{A}}$. Using the effective bi-interpretations, $0^{(\alpha+1)}$ can then compute an isomorphism from $\mathcal{C}$ to $\hat{\mathcal{C}}$.

Let us now prove that $\mathcal{C}$ is not relatively Δ^0_α-categorical. Since $\mathcal{A}$ is not $0^{(\alpha+1)}$-relatively computably categorical, there is a copy $\hat{\mathcal{A}}$ of $\mathcal{A}$ computable in some oracle $Y \geq_T 0^{(\alpha+1)}$ that is not Y-computably isomorphic to $\mathcal{A}$. Let $\hat{\mathcal{C}}$ be the copy of $\mathcal{C}$ associated via the effective bi-interpretation with $\hat{\mathcal{A}}$. Use the α-jump-inversion theorem for reals (Theorem X.4) to get $X \in 2^{<\mathbb{N}}$, with $X^{(\alpha+1)} \equiv_T Y$. The oracle X might not compute the ω-presentation $\hat{\mathcal{C}}$, but as in the proof of the theorem, it computes a copy $\widetilde{\mathcal{C}}$ of $\hat{\mathcal{C}}$ that is $X^{(\alpha+1)}$-computably isomorphic to $\hat{\mathcal{C}}$. We claim that there is no $X^{(\alpha+1)}$-computable isomorphism between $\mathcal{C}$ and $\widetilde{\mathcal{C}}$. That would prove that $\mathcal{C}$ is not relatively $\Delta^0_{\alpha+1}$-categorical. As for the claim, if there was an $X^{(\alpha+1)}$-computable isomorphism between $\mathcal{C}$ and $\widetilde{\mathcal{C}}$, there would be one between $\mathcal{C}$ and $\hat{\mathcal{C}}$, and using the effective bi-interpretations, we would get a $X^{(\alpha+1)}$-computable isomorphism between $\mathcal{A}$ and $\hat{\mathcal{A}}$, which we assumed does not exist. □

Theorem X.5 works only for successor ordinals $\alpha + 1$. Ivan Soskov [**Sos13**] proved that it is not possible to invert the λ-jump for limit ordinals λ, even up to Muchnik equivalence. He showed that there exists a structure $\mathcal{A}$ which computably codes $0^{(\omega)}$, but such that there is no structure $\mathcal{C}$ whose ω-jump is Muchnik equivalent to $\mathcal{A}$. (Recall that two structures are Muchnik equivalent if they have the same degree spectra [**Part 1**, Section VI.1].)

Corollary X.15, which was proved in [**GHK$^+$05**], is stated only for successor ordinals because the proof we give does not work for limit ordinals. The limit case was proved a few years later by Chisholm, Fokina, Goncharov, Harizanov, Knight, and Quinn [**CFG$^+$09**].

CHAPTER XI

The isomorphism problem

So far we have been looking at complexity questions about single structures. Let us concentrate now on classes of structures. We consider classes $\mathbb{K} \subseteq Mod_\tau$ that are *closed under isomorphisms*, that is, such that if $\mathcal{A} \cong \mathcal{B}$ and $\mathcal{A} \in \mathbb{K}$, then $\mathcal{B} \in \mathbb{K}$ too. Recall that Mod_τ is the class of all ω-presentations of τ-structures. The first way to measure the complexity of a class of structures is by how hard it is to recognize that a structure belongs to it. We already showed that a class $\mathbb{K}$ that is closed under isomorphisms is $\mathbb{N}$-Σ^0_α (as a subset of $2^\mathbb{N}$) if and only if it is axiomatizable by a τ-$\Sigma^{\mathtt{c}}_\alpha$ sentence (Theorem VII.25). A second measure of complexity is by how hard it is to tell when two structures within the class are isomorphic to each other. We call this the *isomorphism problem*:

Definition XI.1. Given a class of ω-presentations $\mathbb{K} \subseteq Mod_\tau$, we let

$$\mathbb{I}\mathrm{so}(\mathbb{K}) = \{\langle \mathcal{A}, \mathcal{B}\rangle \in \mathbb{K}^2 : \mathcal{A} \cong \mathcal{B}\}.$$

$\mathbb{I}\mathrm{so}(\mathbb{K})$ is an equivalence relation on $\mathbb{K}$. Viewing $\mathbb{K}$ as a subset of $2^\mathbb{N}$, $\mathbb{I}\mathrm{so}(\mathbb{K})$ is Σ^1_1, as that is what it takes to say that there exists a function which is an isomorphism. The study of Σ^1_1 equivalence relations on reals is a deep and active topic of investigation in descriptive set theory.

XI.1. Complexity as set of pairs

The first way to study the complexity of the isomorphism problem is by looking at $\mathbb{I}\mathrm{so}(\mathbb{K})$ as a subset of $(2^\mathbb{N})^2$, which is itself homeomorphic to $2^\mathbb{N}$, and look at its complexity as a set. We already mentioned that it is Σ^1_1. For some classes of structures, this set of pairs is $\boldsymbol{\Sigma}^1_1$ complete, and for others it is not. An example of a class where this set of pairs is $\boldsymbol{\Sigma}^1_1$ complete is the class of linear orderings. Recall that $\mathbb{LO}$ denotes the class of ω-presentations of linear orderings.

Lemma XI.2. *$\mathbb{I}\mathrm{so}(\mathbb{LO})$ is Σ^1_1 complete.*

Proof. Let $\mathfrak{R} \subseteq 2^\mathbb{N}$ be a Σ^1_1 set. We will define a computable map that, given $X \in 2^\mathbb{N}$, produces a pair of structures $\langle \mathcal{L}^X, \mathcal{H}^X\rangle$ which are isomorphic to each other if and only if $X \in \mathfrak{R}$. The right structure in the pair is the Harrison linear ordering relative to X (see Lemma VI.11). We will make $\mathcal{L}^X$ isomorphic to $\mathcal{H}^X$ if $X \in \mathfrak{R}$ (the Σ^1_1 case) and $\mathcal{L}^X$ well-ordered if $X \notin \mathfrak{R}$ (the Π^1_1 case).

Let T be a computable tree such that, for $X \in 2^{\mathbb{N}}$, $X \in \mathfrak{R}$ if and only if T^X is ill-founded (see the proof of Corollary IV.7). Let S be a computable operator that outputs the Harrison tree relative to X, that is, the tree of descending sequences of $\mathcal{H}^X$.[*] So, S^X is an ill-founded tree that has no paths hyperarithmetic in X. Recall the product operation on trees (Definition IV.14): Given two trees T and S, it produces a new tree $T * S$ such that any path through $T * S$ is essentially of the form $X \oplus Y$, where X is a path through T and Y is a path through S. Recall also that $\leq_{\text{KB}}$ denotes the Kleene–Brouwer linear ordering on $\mathbb{N}^{<\mathbb{N}}$ from Definition I.24. Define

$$\mathcal{A}^X = (T^X * S^X; \leq_{\text{KB}}) \quad \text{and} \quad \mathcal{L}^X = \mathcal{A}^X \cdot \omega.$$

Since S^X has paths for all X, we have that $\mathcal{A}^X$ has a path if and only if T^X does. Equivalently, $\mathcal{L}^X$ is well-ordered if and only if $X \notin \mathfrak{R}$. Furthermore, any path through $T^X * S^X$ must compute a path through S^X, so it cannot be hyperarithmetic in X. So, the Kleene–Brouwer ordering $(T^X * S^X; \leq_{\text{KB}})$ has no descending sequence hyperarithmetic in X.[†] So, $\mathcal{A}^X$ is isomorphic to an initial segment of $\omega_1^X(1+\mathbb{Q})$ (Theorem VI.7). Then, if $\mathcal{A}^X$ is ill-founded, it must have order type $\omega_1^X + \omega_1^X \cdot \mathbb{Q} + \beta$ for some $\beta < \omega_1^X$. Using that $\beta + \omega_1^X \cong \omega_1^X$, we get that, in that case,

$$\begin{aligned}\mathcal{L}^X = \mathcal{A}^X \cdot \omega &\cong (\omega_1^X + \omega_1^X \cdot \mathbb{Q} + \beta) \cdot \omega \\ &\cong \omega_1^X + \omega_1^X \cdot \mathbb{Q} + (\beta + \omega_1^X + \omega_1^X \cdot \mathbb{Q}) \cdot \omega \\ &\cong \omega_1^X \cdot (1+\mathbb{Q}) \cdot (1+\omega) \cong \omega_1^X \cdot (1+\mathbb{Q}).\end{aligned}$$

So, if $\mathcal{A}^X$ is ill-founded, $\mathcal{L}^X$ is isomorphic to $\mathcal{H}^X$. We then have that $\mathcal{L}^X \cong \mathcal{H}^X$ if and only if $X \in \mathfrak{R}$. The map $X \mapsto \langle \mathcal{L}^X, \mathcal{H}^X \rangle$ reduces $\mathfrak{R}$ to $\mathbb{I}\text{so}(\mathbb{LO})$. □

The lemma above can be used to show that $\mathbb{I}\text{so}(\mathbb{K})$ is Σ^1_1-complete for a whole lot of other classes by reducing $\mathbb{LO}$ to $\mathbb{K}$, but only worrying about the well-orderings and the Harrison linear orderings within $\mathbb{LO}$ and ignoring the rest.

[*] Recall that, given a linear ordering $\mathcal{P}$, the tree of descending sequences of $\mathcal{P}$ is $T_{\mathcal{P}} = \{\sigma \in P^{<\mathbb{N}} : \sigma(0) >_P \sigma(1) >_P \cdots >_P \sigma(|\sigma|-1)\}$. Its infinite paths are exactly the infinite descending sequences of $\mathcal{P}$. See page 12.

[†] This is because, if we look into the proof of Theorem I.26, we can see that if f is a descending sequence in the Kleene–Brouwer ordering of a tree, its jump, f', can compute a path through the tree (as it can be obtained using a limit).

If a $\mathbf{\Sigma}^1_1$-subset of $2^{\mathbb{N}}$ is not $\mathbf{\Sigma}^1_1$-complete, it must be Borel: This follows from a theorem in descriptive set theory called Wadge's theorem, which uses $\mathbf{\Sigma}^1_1 \wedge \mathbf{\Pi}^1_1$-determinacy.[‡] The following theorem characterizes the classes for which $\mathbb{I}\mathrm{so}(\mathbb{K})$ is not Σ^1_1-complete.

Theorem XI.3. *Let $\mathbb{K}$ be closed under isomorphisms. The following are equivalent:*

(1) The isomorphism problem for $\mathbb{K}$ is Borel.

(2) $\mathbb{K}$ has bounded Scott rank. *That is, there is an $\alpha < \omega_1$ such that all structures in $\mathbb{K}$ have Scott rank less than α.*

Proof. For the implication (2)⇒(1), notice that all structures in $\mathbb{K}$ have $\Sigma^{\mathtt{in}}_{\alpha+2}$ Scott sentences (Proposition II.26). Hence, if two structures in $\mathbb{K}$ are $\alpha+2$-back-and-forth equivalent, they are isomorphic. The relation $\equiv_{\alpha+2}$ is $\Pi^0_{2\alpha+4}$ (just count quantifiers in Definition II.32) and in particular Borel.

For the implication (1)⇒(2), let $\alpha < \omega_1$ be such that $\mathbb{I}\mathrm{so}(\mathbb{K})$ is $\mathbf{\Sigma}^0_\alpha$. For each structure $\mathcal{A} \in \mathbb{K}$, the class of ω-presentations $\mathcal{B}$ which are isomorphic to $\mathcal{A}$ is $\mathbf{\Sigma}^0_\alpha$ with parameter $D(\mathcal{A})$. By the Lopez-Escobar–Vaught theorem, Theorem VII.25, this class must be axiomatizable by a $\Sigma^{\mathtt{in}}_\alpha$ sentence, meaning that $\mathcal{A}$ has a $\Sigma^{\mathtt{in}}_\alpha$-Scott sentence. Therefore, it must have Scott rank at below α (Proposition II.26). □

XI.2. Complexity as equivalence relations on the reals

Reducibilities between classes allow us to classify their complexity by comparing them to other classes. With this in mind, Friedman and Stanley [**FS89**] defined the notion of Borel reducibility. Since then, the study of Borel reducibility on arbitrary Borel and analytic equivalence relations has been extremely active in descriptive set theory.

Definition XI.4. (H. Friedman and L. Stanley [**FS89**]) A class of structures $\mathbb{K}$ is *Borel reducible* to a class $\mathbb{S}$, written $\mathbb{K} \leq_B \mathbb{S}$, if there is a Borel function $f : \mathbb{K} \to \mathbb{S}$ that preserves isomorphism. That is, f maps ω-presentations in $\mathbb{K}$ to ω-presentations in $\mathbb{S}$, and for all $\mathcal{A}, \tilde{\mathcal{A}} \in \mathbb{K}$,

$$\mathcal{A} \cong \tilde{\mathcal{A}} \iff f(\mathcal{A}) \cong f(\tilde{\mathcal{A}}).$$

[‡] Under AD, Wadge's theorem [**Wad83**] states that for every two sets $\mathfrak{R}, \mathfrak{S} \subseteq 2^{\mathbb{N}}$, either there is a continuous function $F\colon 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ such that $\mathfrak{R} = F^{-1}(\mathfrak{S})$, or there is a continuous function $G\colon 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ such that $\mathfrak{S} = G^{-1}(\mathfrak{R}^c)$. When $\mathfrak{R} = F^{-1}(\mathfrak{S})$, we say that $\mathfrak{R}$ *continuously reduces* to $\mathfrak{S}$. If $\mathfrak{S}$ is not $\mathbf{\Sigma}^1_1$-complete and $\mathfrak{R}$ is, then $\mathfrak{R}$ cannot continuously reduce to $\mathfrak{S}$. So, by Wadge's theorem, $\mathfrak{S}$ must continuously reduce to the complement of $\mathfrak{R}$, which is $\mathbf{\Pi}^1_1$, and hence $\mathfrak{S}$ is Π^1_1 itself too, and in particular Borel. Wadge's theorem for $\mathbf{\Sigma}^1_1$ sets uses $\mathbf{\Sigma}^1_1 \wedge \mathbf{\Pi}^1_1$-determinacy.

A class $\mathbb{K}$ is *on top for Borel reducibility* if, for every vocabulary τ, the class of τ-structures is Borel-reducible to $\mathbb{K}$.[§]

Friedman and Stanley first observed that it is enough to use the vocabulary with only one binary relation (i.e., directed graphs) in the definition above. Then they built Borel reductions to show that the classes of graphs [**Part 1**, Theorem VI.27], trees, linear orderings [**Part 1**, Lemma VI.18], 2-step nilpotent groups, and fields are all on top for Borel reducibility. Camerlo and Gao [**CG01**] added Boolean algebras to that list. Paolini and Shelah [**PS**] have recently circulated a preprint that proved that torsion-free abelian groups are also on top, a question that had been open since Friedman and Stanley's '89 paper.

If a class $\mathbb{K}$ is on top under Borel reducibility, its isomorphism problem must be $\mathbf{\Sigma}^1_1$-complete as a set of pairs. This is because if $\mathbb{K}$ is on top under Borel reducibility, $\mathbb{LO}$ must reduce to $\mathbb{K}$, and we can use the Borel reduction and the Σ^1_1-completeness of $\mathbb{I}\text{so}(\mathbb{LO})$ to show that $\mathbb{I}\text{so}(\mathbb{K})$ must be Σ^1_1-complete too. Therefore, if $\mathbb{K}$ is on top under Borel reducibility, it must have unbounded Scott rank below ω_1. Classes like finitely branching trees, p-groups of finite rank, $\mathbb{Q}$-vector spaces, algebraically closed fields, equivalence structures, etc., all have bounded Scott rank, and hence are not on top under Borel reducibility. An example that stands out is *torsion abelian groups*. Their isomorphism problem is Σ^1_1-complete as sets of pairs, but they are not on top for Borel reducibility [**FS89**, Theorem 5]. Briefly, the reason why torsion abelian groups are not on top for Borel reducibility is that if a $\mathcal{G}$ is a torsion abelian group with $\omega_1^{D(\mathcal{G})} = \omega_1^{CK}$, then the isomorphism type of $\mathcal{G}$ is determined by its $\mathcal{L}_{\mathsf{c},\omega}$ theory (using the Ulm invariants of its p-sub-groups), while there are non-hyperarithmetic structures $\mathcal{A}$ with $\omega_1^{D(\mathcal{A})} = \omega_1^{CK}$ for which you need all their $\mathcal{L}_{\mathsf{c},\omega}^{D(\mathcal{A})}$ theories to determine their isomorphism type.

XI.3. Turing-computable reducibility

If the reduction f in the definition of Borel reducibility (Definition XI.4) is continuous, we say that $\mathbb{K}$ is *continuously reducible* to $\mathbb{S}$. If the reduction f is computable, we say that $\mathbb{K}$ is *Turing-computable reducible* to $\mathbb{S}$. The notion of Turing-computable reducibility between classes of structures was first studied by Calvert, Cummins, Knight,

[§] In the literature, these classes are sometimes called *Borel complete*. We want to avoid that notation here. The reason is that when we say that $\mathbb{K}$ is $\mathbf{\Sigma}^1_1$-complete, we mean that there is a continuous reduction from any $\mathbf{\Sigma}^1_1$ subset of 2^ω to the isomorphism problem of $\mathbb{K}$ as a set and not as an equivalence relation. Reductions that preserve equivalence relations are quite different.

and S. Miller [**CCKM04**]. A class $\mathbb{K}$ is then *on top for Turing-computable (continuous) reducibility* if, for every computable vocabulary τ, the class of τ-structures Turing-computably (continuously) reduces to $\mathbb{K}$.

Notice that $\mathbb{K}$ is continuously reducible to $\mathbb{S}$ if and only if it is Turing-computable reducible to $\mathbb{S}$ relative to some oracle X. The following theorem connects Borel reducibility with Turing-computable reducibility. We use $\mathbb{K}^{(\alpha)}$ to denote the class of α-jumps of the structures in $\mathbb{K}$ (see Chapter X).

Theorem XI.5. *A class $\mathbb{K}$ Borel reduces to a class $\mathbb{S}$ if and only if there is an oracle X and an X-computable ordinal α such that $\mathbb{K}^{(\alpha)}$ X-Turing computably reduces to $\mathbb{S}$.*

Proof. The ($\Leftarrow$) implication is straightforward, as we would then have a $\Delta^0_\alpha(X)$ reduction from $\mathbb{K}$ to $\mathbb{S}$.

For the ($\Rightarrow$) implication, let Φ be a Borel reduction from $\mathbb{K}$ to $\mathbb{S}$. Let X and α be such that Φ is $\Delta^0_\alpha(X)$. There is an X-computable operator Ψ such that $\Phi(\mathcal{D}(\mathcal{B})) = \Psi(\mathcal{D}(\mathcal{B})^{(\alpha)})$ for every $\mathcal{B} \in \mathbb{K}$. Notice that we want an operator that acts on $\mathcal{D}(\mathcal{B}^{(\alpha)})$, while Ψ acts on $\mathcal{D}(\mathcal{B})^{(\alpha)}$. So, a bit more work is needed.

Let us define an X-Turing computable reduction Γ from $\mathbb{K}^{(\alpha)}$ to $\mathbb{S}$. Suppose we are given $D(\mathcal{A}^{(\alpha)})$, and we want to define $\Gamma(D(\mathcal{A}^{(\alpha)}))$. We can uniformly produce a $\Sigma^{\mathsf{c}}_{<\alpha}$-generic enumeration g of $\mathcal{A}$ computable in $\mathcal{D}(\mathcal{A}^{(\alpha)})$ (see Lemma X.10). Let $\mathcal{G}$ be the associated $\Sigma^{\mathsf{c}}_{<\alpha}$-generic ω-presentation. We then get that

$$D(\mathcal{G})^{(\alpha)} \equiv_T D(\mathcal{G}^{(\alpha)}) \leq_T g \oplus \mathcal{D}(\mathcal{A}^{(\alpha)}) \leq_T \mathcal{D}(\mathcal{A}^{(\alpha)})$$

(see Lemma X.11). Furthermore, we can observe from the proofs of Lemmas X.10 and X.11 that $D(\mathcal{G})^{(\alpha)} \leq_T \mathcal{D}(\mathcal{A}^{(\alpha)})$ uniformly in $\mathcal{A}$. Define $\Gamma(D(\mathcal{A}^{(\alpha)}))$ as the output of the X-computable operator Ψ when applied to $D(\mathcal{G})^{(\alpha)}$. We get that $\Gamma(D(\mathcal{A}^{(\alpha)}))$ is a structure isomorphic to $\Phi(\mathcal{G})$, which is isomorphic to $\Phi(\mathcal{A})$. □

All the reducibilities produced in [**FS89**] are not only Borel but also effective, showing that trees, linear orderings, nilpotent groups, and fields are actually on top for Turing-computable reducibility. This happens for a reason:

Corollary XI.6. *If $\mathbb{K}$ is on top for Borel reducibility, it is on top for continuous reducibility.*

Proof. Let $\mathbb{G}$ be the class of graphs, which we know is on top for continuous reducibility ([**Part 1**, Theorem VI.27]). Since $\mathbb{K}$ is on top for Borel reducibility, $\mathbb{G}$ Borel embeds in $\mathbb{K}$, and hence $\mathbb{G}^{(\alpha)}$ continuously embeds in $\mathbb{K}$ for some ordinal α. Let $\mathbb{G}^{(-\alpha)}$ be the class of α-jump inversions of all graphs, as in the first iterated-jump inversion theorem, Theorem X.5.[¶] The class $\mathbb{G}^{(-\alpha)}$ reduces to $\mathbb{G}$ via effective bi-interpretability (again by [**Part 1**, Theorem VI.27]). So $\mathbb{G}$, which is effectively bi-interpretable with $\mathbb{G}^{(-\alpha)(\alpha)}$, continuously embeds in $\mathbb{G}^{(\alpha)}$, which continuously embeds in $\mathbb{K}$. □

The fact that Turing-computable reducibility is finer than Borel reducibility allows us to get finer comparability results. For instance, any two classes of structures with countably-infinite many models are Borel-equivalent. However, this is not the case for Turing-computable reducibility, and an interesting structure can be found among the classes with only countably many models (see [**KMVB07**]). There are even classes with finitely many structures that are not trivial under Turing-computable reducibility. One of the most interesting facts about Turing-computable reducibility is that it preserves the back-and-forth structure:

Theorem XI.7 (Pull Back theorem). *(Knight, S. Miller and Vanden Boom* [**KMVB07**]*) Let Φ be a Turing computable reduction from $\mathbb{K}$ to $\mathbb{S}$. Then, for every $\Pi^{\mathtt{c}}_\alpha$-sentence φ, there is a $\Pi^{\mathtt{c}}_\alpha$ sentence φ^* such that, for all $\mathcal{A}\in\mathbb{K}$,*

$$\mathcal{A}\models\varphi^* \quad\iff\quad \Phi(\mathcal{A})\models\varphi.$$

Proof. Consider the forcing that produces generic copies of $\mathcal{A}$ for $\mathcal{A}$ in $\mathbb{K}$. Let ψ be the $\mathbb{N}$-Π^0_α-formula that says that the structure $\Phi(\dot{\mathcal{G}})$ satisfies φ. Let φ^* be the sentence $\mathit{Force}_\psi(\langle\rangle)$, which says that the empty tuple forces $\Phi(\dot{\mathcal{G}})$ to satisfy φ. Notice that the sentence $\mathit{Force}_\psi(\langle\rangle)$ does not depend on the structure $\mathcal{A}$.

If $\mathcal{A}\models\varphi^*$, then $\langle\rangle\Vdash\psi$, and hence for every generic copy $\mathcal{G}$ of $\mathcal{A}$, $\Phi(\mathcal{G})$ satisfies φ. Since Φ preserves isomorphisms, $\Phi(\mathcal{A})\models\varphi$ too. Conversely, if $\Phi(\mathcal{A})\models\varphi$, then since Φ preserves isomorphisms, $\Phi(\mathcal{G})\models\varphi$ for all generic copies $\mathcal{G}$ of $\mathcal{A}$. It follows that $\langle\rangle\Vdash\psi$, and hence that $\mathcal{A}\models\varphi^*$. □

Corollary XI.8. *Let Φ be a Turing computable reduction from $\mathbb{K}$ to $\mathbb{S}$ and $\mathcal{A}$ a structure in $\mathbb{K}$. Then $\Pi^{\mathtt{c}}_\alpha$-$Th(\Phi(\mathcal{A}))\leq_m\Pi^{\mathtt{c}}_\alpha$-$Th(\mathcal{A})$.*

[¶] Note that the construction in the proof of Theorem X.5 works even if the graph does not computably code $0^{(\alpha)}$. The α-th jump of the inversion will then be equivalent to the original graph joined with $0^{(\alpha)}$.

Proof. To see this, one has to notice that the map from φ to φ^* of the previous theorem is computable in φ. This map provides the desired many-one reduction. □

Corollary XI.9. *Let Φ be a Turing computable reduction from $\mathbb{K}$ to $\mathbb{S}$, let $\mathcal{A}$ and $\widetilde{\mathcal{A}}$ be structures in $\mathbb{K}$, and let α be any ordinal. Then*

$$\mathcal{A} \leq_\alpha \widetilde{\mathcal{A}} \quad \Rightarrow \quad \Phi(\mathcal{A}) \leq_\alpha \Phi(\widetilde{\mathcal{A}}).$$

Proof. Assume $\mathcal{A} \leq_\alpha \widetilde{\mathcal{A}}$. We want to show that every $\Pi^{\mathtt{in}}_\alpha$ sentence true about $\Phi(\mathcal{A})$ is also true about $\Phi(\widetilde{\mathcal{A}})$. By relativization, the lemma above works for all infinitary formulas as well, not just the computable ones. So, if a $\Pi^{\mathtt{in}}_\alpha$ sentence φ is true about $\Phi(\mathcal{A})$, then the $\Pi^{\mathtt{in}}_\alpha$ sentence φ^* is true about $\mathcal{A}$, and hence about $\widetilde{\mathcal{A}}$, and hence φ is true about $\Phi(\widetilde{\mathcal{A}})$. □

This theorem allowed Knight, S. Miller and Vanden Boom [**KMVB07**] to characterize the classes $\mathbb{K}$ such that $\mathbb{K}$ Turing-computably reduces to $\mathbb{S}$ for certain fixed classes $\mathbb{S}$. An interesting example is $\mathbb{Q}$-vector spaces:

Corollary XI.10. *If a class $\mathbb{K}$ Turing-computably embeds in the class $\mathbb{VS}$ of $\mathbb{Q}$-vector spaces, then there exists a computable sequence of $\Pi^{\mathtt{c}}_2$ sentences $\langle \psi_n : n \in \mathbb{N}\rangle$ such that:*

- *φ_n implies φ_{n+1} for all n, and*
- *if two structures satisfy the same sentences from this sequence, they are isomorphic.*

Proof. Let φ_n be the $\Pi^{\mathtt{c}}_2$ sentence that says that a $\mathbb{Q}$-vector space does not have $n+1$ linearly independent vectors, or, in other words, that it has dimension less than or equal to n. Let ψ_n be the formula φ^*_n given by the theorem above. It is not hard to verify that the formulas ψ_n are as needed. □

The converse of this corollary is also true [**KMVB07**]. To prove this, one first has to computably build, from a structure $\mathcal{A} \in \mathbb{K}$, a $D(\mathcal{A})$-computable function $f\colon \mathbb{N} \to \mathbb{N}$ whose lim-inf is the greatest n such that $\mathcal{A} \models \psi_n$, and then use this function to computably build a vector space whose dimension is the lim-inf of f, using similar ideas to those in [**Part 1**, Lemma VII.13]. We omit the details as they are not relevant to the material in this chapter.

XI.4. The isomorphism problem on indices

Another way of studying the complexity of the isomorphism problem is by looking at it as an equivalence relation on numbers, namely on the indices of the computable structures in the class. Let

$$\mathrm{iso}(\mathbb{K}) = \{\langle n, m\rangle \in \omega^2 : \ n \text{ and } m \text{ being indices} \\ \text{for isomorphic computable structures in } \mathbb{K}\}.$$

At first, it might seem that there should not be any meaningful difference between the study of iso($\mathbb{K}$) and $\mathbb{I}$so($\mathbb{K}$) on natural classes of structures. Surprisingly enough, recent work has shown that there are interesting qualitative differences [**FF09, FFH$^+$12, CHM12, Mon16**].

Definition XI.11. We say that a class of structures $\mathbb{K}$ is *effectively reducible* to a class $\mathbb{S}$ if there is a computable function $f : \omega \to \omega$ which maps indices of computable structures in $\mathbb{K}$ to indices of computable structures in $\mathbb{S}$ such that, for indices m and n of structures in $\mathbb{K}$,

$$\langle m, n\rangle \in \mathrm{iso}(\mathbb{K}) \quad \iff \quad \langle f(m), f(n)\rangle \in \mathrm{iso}(\mathbb{S}).$$

A class of structures $\mathbb{K}$ is said to be *on top for effective reducibility* if, for any computable vocabulary τ, the class of τ-structures effectively reduces to it.

E. Fokina, S. Friedman, V. Harizanov, J. Knight, C. McCoy and A. Montalbán [**FFH$^+$12**] gave proofs that linear orderings, trees, fields, p-groups, and torsion-free abelian groups are all on top for effective reducibility. Montalbán [**Mon16**] then provided a general method for proving that a class is on top by showing that if one can build an η-tree as in Section IX.6 for a non-standard η where paths that are not equivalent below ω_1^{CK} have non-isomorphic structures, then one can reduce any Σ^1_1-equivalence relation on $\mathbb{N}$ to iso($\mathbb{K}$).

It is not hard to see that Turing-computable reducibility implies effective reducibility. This implication does not reverse. Effective reducibility does not even imply Borel-reducibility. The main example is p-groups, which is on top for effective reducibility but not for Borel-reducibility.

It is not hard to see that if a class is on top for effective reducibility, its isomorphism-index-set, iso($\mathbb{K}$), must be Σ^1_1-complete. Thus, $\mathbb{Q}$-vector spaces, equivalence structures, torsion-free abelian groups of finite rank, etc. cannot be on top because they have arithmetic isomorphism problems. So far, this is the only way we know to produce examples of classes which are not on top.

Definition XI.12. A class $\mathbb{K}$ is *intermediate for effective reducibility* if it is not on top for effective reducibility, and its isomorphism-index-set is not hyperarithmetic.

No specific example of an intermediate class is known. Becker [**Bec13**], and independently Knight and Montalbán [unpublished], showed that such a class of structures exists under the assumption that Vaught's conjecture fails (relative to some oracle). We will give a proof of this in the next chapter.

CHAPTER XII

Vaught's Conjecture

Vaught's Conjecture states that the number of countable models of an $\mathcal{L}_{\omega_1,\omega}$-sentence is either countable or continuum, but never in between [**Vau61**]. Of course, we are counting isomorphism types of models, not ω-presentations. One has to think of Vaught's Conjecture in the context where the continuum hypothesis (CH) is false, as otherwise Vaught's Conjecture is trivially true. The conjecture is still open.

One of the most important partial results towards Vaught's conjecture is Morley's theorem [**Mor70**]. Morley showed that the number of countable models of an $\mathcal{L}_{\omega_1,\omega}$-sentence has to be either countable, $\aleph_1$, or continuum, ruling out all other options. Some years later, Burgess showed that this is part of a more general behavior: The number of equivalence classes of a $\mathbf{\Sigma}^1_1$ equivalence relation E on $2^{\mathbb{N}}$ has to be either countable, $\aleph_1$, or continuum [**Bur78**]. This applies to Vaught's conjecture, as if $\mathbb{K} \subseteq Mod_\tau$ is the class of ω-presentations of some $\mathcal{L}_{\omega_1,\omega}$-sentence Θ, then $\mathbb{I}\mathrm{so}(\mathbb{K})$ is Σ^1_1, and the number of countable models of Θ is the number of equivalence classes of $\mathbb{I}\mathrm{so}(\mathbb{K})$. Burgess's result is actually a bit stronger. It says that if there are more than $\aleph_1$ equivalence classes, there must be *perfectly many classes*, meaning that there is a perfect subset of $2^{\mathbb{N}}$ all of whose members are E-inequivalent. Each perfect subset of $2^{\mathbb{N}}$ is the set of paths $[T]$ of some tree $T \subseteq 2^{<\mathbb{N}}$ without dead ends and without isolated paths. Such trees are isomorphic to $2^{<\mathbb{N}}$, and their sets of paths are homeomorphic to $2^{\mathbb{N}}$. Perfect sets always contain continuum many elements. So, we can re-state Vaught's conjecture as stating that the set of models of an $\mathcal{L}_{\omega_1,\omega}$-sentence is either countable or contains a perfect set of non-isomorphic ω-presentations. This formulation is now meaningful independent of whether CH holds or not.

The original statement of Vaught's Conjecture was for finitary first-order theories instead of $\mathcal{L}_{\omega_1,\omega}$-sentences, and, as far as we know, the $\mathcal{L}_{\omega_1,\omega}$ formulation we use here may be stronger than the other. Since most techniques used to study Vaught's conjecture from the viewpoint of computability or descriptive set theory work the same for both situations, and $\mathcal{L}_{\omega_1,\omega}$ is better suited for dealing with complexity considerations, it is common for computability or descriptive set theorists to use the $\mathcal{L}_{\omega_1,\omega}$ formulation.

XII.1. The back-and-forth structure

Morley's theorem can be proved using Silver's theorem [**Sil80**]. Silver's theorem states that every Borel equivalence relation on $2^{\mathbb{N}}$ has either countably many or perfectly many equivalence classes.

Proof of Morley's theorem. Let $\mathbb{K} \subseteq \mathit{Mod}_\tau$ be the set of ω-presentations of an $\mathcal{L}_{\omega_1,\omega}$-sentence. Assume that $\mathbb{K}$ has less than continuum many models. We will show it has at most $\aleph_1$.

For each $\alpha < \omega_1$, the α-back-and-forth equivalence relation $\equiv_\alpha$ is a Borel equivalence relation on Mod_τ.* Thus, the number of $\equiv_\alpha$-equivalence classes within $\mathbb{K}$ is either countable or continuum. Therefore, if the number of non-isomorphic structures in $\mathbb{K}$ is less than $2^{\aleph_0}$, the number of $\equiv_\alpha$-equivalence classes must be countable for each countable ordinal α.

We claim that it follows that, for each $\alpha < \omega_1$, the number of structures in $\mathbb{K}$ of Scott rank α is at most countable, implying that the total number must be at most $\aleph_1$. To see this, just recall that all structures of Scott rank α have $\Sigma^{\mathtt{in}}_{\alpha+2}$ Scott sentences (Proposition II.26). Therefore, if two structures of Scott rank α are $(\alpha+2)$-back-and-forth equivalent, they must be isomorphic. Since there are only countably many $\equiv_{\alpha+2}$-equivalence classes, there are only countably many isomorphism types among the structures of Scott rank α. □

We say that an $\mathcal{L}_{\omega_1,\omega}$-sentence Θ is *scattered* if it has countably many $\equiv_\alpha$-equivalence classes for all $\alpha < \omega_1$. By the argument in the proof above, if a sentence Θ is scattered and the Scott ranks of the models of Θ are bounded by some $\alpha < \omega_1$, then Θ must have countably many models. The proof above also tells us that if Θ has less than continuum many models, it must be scattered. Conversely, if a sentence Θ is scattered, it cannot have perfectly many non-isomorphic models: This is clear under $\neg\mathsf{CH}$. The proof that it is true under CH uses techniques from set theory that are beyond the scope of this book.

Definition XII.1. We say that an $\mathcal{L}_{\omega_1,\omega}$-sentence Θ is *unbounded* if it has models of arbitrarily high Scott rank below ω_1.

* Just by counting quantifiers in the definition of $\equiv_\alpha$ (Definition II.32), one can see that it is $\Pi^0_{2\alpha}$.

When we refer to a *counterexample to Vaught's conjecture* we mean a scattered, unbounded $\mathcal{L}_{\omega_1,\omega}$-sentence.

Definition XII.2. A class of structures $\mathbb{K}$ is *$\Sigma^{\mathtt{in}}_{\alpha}$-small* if there are only countably many different $\Sigma^{\mathtt{in}}_{\alpha}$-types realized among all the tuples on all the structures in $\mathbb{K}$.

If two structures are $\equiv_{\alpha+2}$-equivalent, they must realize the same $\Sigma^{\mathtt{in}}_{\alpha}$-types, because every tuple in one structure is $\equiv_{\alpha}$-equivalent to a tuple in the other. So, a class is scattered if and only if it is $\Sigma^{\mathtt{in}}_{\alpha}$-small for all $\alpha < \omega_1$. $\Sigma^{\mathtt{in}}_{\alpha}$-small classes are very nice from a complexity point of view. For instance, if we have a $\Sigma^{\mathtt{in}}_{\alpha}$-small class, we can define canonical structural α-jumps, and we can translate many of the results on Σ-small classes from [**Part 1**, Chapter X]. We saw many examples of $\Sigma^{\mathtt{in}}_{1}$-small classes in [**Part 1**, Chapter X]. The class of linear orderings is $\Sigma^{\mathtt{in}}_{2}$-small but not $\Sigma^{\mathtt{in}}_{3}$ small ([**Kni86**], see [**Part 1**, Chapter X]). The class of Boolean algebras is $\Sigma^{\mathtt{in}}_{n}$-small for all $n \in \mathbb{N}$, but it is not $\Sigma^{\mathtt{in}}_{\omega}$-small ([**JS94**], see [**HM12**]).

Definition XII.3. Given a class of structures $\mathbb{K}$, we define its *back-and-forth ordinal* as the least ordinal α such $\mathbb{K}$ is not Σ_{α}-small.

If Θ has countably many models, we let its back-and-forth ordinal be ∞. If Θ is a counterexample to Vaught's conjecture, its back-and-forth ordinal is ω_1.[†] If Θ has perfectly many models, it cannot be scattered, so its back-and-forth ordinal must be below ω_1.

One can build examples of $\mathcal{L}_{\omega_1,\omega}$ sentences with any given back-and-forth ordinal if one is allowed to choose the axiomatizing sentence to be of any complexity. But, if one is only allowed to use, say $\Pi^{\mathtt{in}}_{2}$ sentences, the highest back-and-forth ordinal we know so far that is not ∞ is that of Boolean algebras: ω. One is not really losing much generality by restricting oneself to $\Pi^{\mathtt{in}}_{2}$ sentences, as one can transform any $\mathcal{L}_{\omega_1,\omega}$-axiomatizable class into a $\Pi^{\mathtt{in}}_{2}$-axiomatizable class using Morleyization as in Section II.5. A possible proof of Vaught's conjecture may come by proving that no unbounded $\Pi^{\mathtt{in}}_{2}$ sentence has back-and-forth ordinal above ω. This last line is related, but not equivalent, to Martin's strengthening of Vaught's conjecture (see [**Gao01**] for more information on the model-theoretic Martin's conjecture).

XII.2. Minimal theories

Recall that a sentence Θ is unbounded if it has models of arbitrarily high Scott rank below ω_1.

[†] Since counterexample to Vaught's conjecture have uncountably many countable models, they realize uncountably many $\mathcal{L}_{\omega_1,\omega}$-types by Lemma II.7.

Definition XII.4. We say that an $\mathcal{L}_{\omega_1,\omega}$-sentence Θ is *minimally unbounded* if it is unbounded, but for every $\mathcal{L}_{\omega_1,\omega}$-sentence φ, one of $\Theta \wedge \varphi$ or $\Theta \wedge \neg\varphi$ is bounded.

Harnik and Makkai showed that if there is a counterexample to Vaught's conjecture, there is one that is minimally unbounded ([**HM77**], see also [**Ste78**, Theorem 1.5.11]). Minimally unbounded $\mathcal{L}_{\omega_1,\omega}$-sentences have interesting properties, as we will see in the next section. This section is dedicated to proving Steel's theorem. We need two lemmas.

Lemma XII.5. *For every structure $\mathcal{A}$, there is a sentence $\psi_{\mathcal{A},\alpha}$ such that, for any other structure $\mathcal{B}$,*

$$\mathcal{B} \models \psi_{\mathcal{A},\alpha} \quad\iff\quad \mathcal{B} \equiv_\alpha \mathcal{A}.$$

Proof. In Lemma VI.14, we defined formulas $\varphi_{\bar{a},\beta}$ and $\psi_{\bar{a},\beta}$ for $\bar{a} \in A^{<\mathbb{N}}$ such that, for every structure $\mathcal{B}$ and tuple $\bar{b} \in B^{|\bar{a}|}$,

$$\mathcal{B} \models \varphi_{\bar{a},\beta}(\bar{b}) \quad\iff\quad (\mathcal{A},\bar{a}) \leq_\beta (\mathcal{B},\bar{b}),$$

and

$$\mathcal{B} \models \psi_{\bar{a},\beta}(\bar{b}) \quad\iff\quad (\mathcal{A},\bar{a}) \geq_\beta (\mathcal{B},\bar{b}).$$

The sentence $\psi_{\mathcal{A},\alpha}$ is then defined as $\varphi_{\langle\rangle,\alpha} \wedge \psi_{\langle\rangle,\alpha}$. □

Lemma XII.6. *Let $\{\mathcal{A}_i : i \in \mathbb{N}\}$ be a sequence of structures and $\langle \alpha_i : i \in \mathbb{N}\rangle$ an increasing sequence of countable ordinals such that*

$$\mathcal{A}_0 \equiv_{\alpha_0+3} \mathcal{A}_1 \equiv_{\alpha_1+3} \mathcal{A}_2 \equiv_{\alpha_2+3} \mathcal{A}_3 \equiv_{\alpha_3+3} \cdots$$

There is a structure $\mathcal{A}_\infty$ with $\mathcal{A}_\infty \equiv_{\alpha_i} \mathcal{A}_i$ for all $i \in \mathbb{N}$.

Proof. We will build a sequence $\{\bar{a}_i : i \in \mathbb{N}\}$ of tuples with $\bar{a}_i \in A_i^{<\mathbb{N}}$ such that, for each $i \in \mathbb{N}$,

(1) $(\mathcal{A}_i,\bar{a}_i) \equiv_{\alpha_i+1} (\mathcal{A}_{i+1},\bar{a}_{i+1})$, and
(2) for each $\bar{b} \in A_i^{<\mathbb{N}}$, there is a $k > i$ and a tuple $\bar{c}$ of elements of $\bar{a}_k$ such that $(\mathcal{A}_i,\bar{a}_i,\bar{b}) \leq_{\alpha_i} (\mathcal{A}_k,\bar{a}_k \restriction |\bar{a}_i|,\bar{c})$.

Since $(\mathcal{A}_i,\bar{a}_i) \equiv_{\alpha_i+1} (\mathcal{A}_{i+1},\bar{a}_{i+1})$, we have that $D_{\mathcal{A}_i}(\bar{a}_i) \subseteq D_{\mathcal{A}_{i+1}}(\bar{a}_i)$, and hence we can define $\mathcal{A}_\infty$ to be the ω-presentation with diagram

$$D(\mathcal{A}_\infty) = \bigcup_n D_{\mathcal{A}_i}(\bar{a}_i).$$

Before proving that the limit structure $\mathcal{A}_\infty$ is as needed, let us prove that such a sequence of tuples exists.

Start with $\bar{a}_0 = \langle\rangle$ as usual. Suppose $\bar{a}_0, \bar{a}_1, ..., \bar{a}_s$ have been defined already. We define $\bar{a}_{s+1}$ in two steps. To take care of (1), using that $\mathcal{A}_s \geq_{\alpha_s+3} \mathcal{A}_{s+1}$, find $\bar{d}_0 \in A_{s+1}^{<\mathbb{N}}$ such that $(\mathcal{A}_s,\bar{a}_s) \leq_{\alpha_s+2} (\mathcal{A}_{s+1},\bar{d}_0)$. To take care of (2), we consider only one $i < s$ and one tuple $\bar{b} \in A_i^{<\mathbb{N}}$

at a time: Let $i, j < s$ be such that $\ulcorner\langle i,j\rangle\urcorner = s$,[‡] and let $\bar{b}$ be the tuple of the first j elements in the ω-presentation of $\mathcal{A}_i$. Using that $(\mathcal{A}_i, \bar{a}_i) \equiv_{\alpha_i+1} (\mathcal{A}_s, \bar{a}_s \restriction |\bar{a}_i|) \leq_{\alpha_s+1} (\mathcal{A}_{s+1}, \bar{d}_0 \restriction |\bar{a}_i|)$, and in particular that $(\mathcal{A}_i, \bar{a}_i) \geq_{\alpha_i+1} (\mathcal{A}_{s+1}, \bar{d}_0 \restriction |\bar{a}_i|)$, find $\bar{c} \in A_{s+1}^{<\mathbb{N}}$ such that $(\mathcal{A}_i, \bar{a}_i, \bar{b}_i) \leq_{\alpha_i} (\mathcal{A}_{s+1}, \bar{d}_0 \restriction |\bar{a}_i|, \bar{c})$, and let $\bar{a}_{s+1} = \bar{d}_0{}^\frown\bar{c}$.[§]

Now that we have defined the tuples $\bar{a}_s \in \mathcal{A}_s^{<\mathbb{N}}$, we want to prove that, for each i, $(\mathcal{A}_i, \bar{a}_i) \equiv_{\alpha_i} (\mathcal{A}_\infty, \langle 0, ..., |\bar{a}_i| - 1\rangle)$. We prove by transfinite induction that, for all ordinals γ, we have that

- for all i with $\gamma \leq \alpha_i$, $(\mathcal{A}_i, \bar{a}_i) \geq_\gamma (\mathcal{A}_\infty, \langle 0, ..., |\bar{a}_i| - 1\rangle)$, and
- for all i with $\gamma \leq \alpha_i$, $(\mathcal{A}_i, \bar{a}_i) \leq_\gamma (\mathcal{A}_\infty, \langle 0, ..., |\bar{a}_i| - 1\rangle)$.

For the first part, take $\delta < \gamma$ and $\bar{b}$ in $\mathcal{A}_i$ disjoint from $\bar{a}_i$. Let k and $\bar{c}$ be as in (2). By the induction hypothesis, $(\mathcal{A}_k, \bar{a}_k) \leq_\delta (\mathcal{A}_\infty, \langle 0, ..., |\bar{a}_k| - 1\rangle)$, and hence $(\mathcal{A}_i, \bar{a}_i, \bar{b}) \leq_\delta (\mathcal{A}_\infty, \langle 0, ..., |\bar{a}_i| - 1\rangle, \bar{n})$, where $\bar{n}$ is the list of indices of the elements of $\bar{c}$ within $\bar{a}_k$. This shows that $(\mathcal{A}_i, \bar{a}_i) \geq_\gamma (\mathcal{A}_\infty, \langle 0, ..., |\bar{a}_i| - 1\rangle)$.

For the second part, consider $\delta < \gamma$ and $\bar{c}$ in $\mathcal{A}_\infty$ disjoint from $\langle 0, ..., |\bar{a}_i| - 1\rangle$. Let k be large enough so that $\bar{c}$ is included in $\langle 0, ..., |\bar{a}_k| - 1\rangle$ within $\mathcal{A}_\infty$. By the induction hypothesis, we know that $(\mathcal{A}_k, \bar{a}_k) \geq_\delta (\mathcal{A}_\infty, \langle 0, ..., |\bar{a}_k| - 1\rangle)$. Since $(\mathcal{A}_i, \bar{a}_i) \equiv_{\alpha_i+1} (\mathcal{A}_{k+1}, \bar{a}_{k+1})$, there is a tuple $\bar{b}$ in $\mathcal{A}_i$ such that $(\mathcal{A}_i, \bar{a}_i, \bar{b}) \geq_{\alpha_i} (\mathcal{A}_k, \bar{a}_k \restriction |\bar{a}_i|, \bar{c})$. In particular, we have that $(\mathcal{A}_i, \bar{a}_i, \bar{b}) \geq_\delta (\mathcal{A}_k, \bar{a}_k \restriction |\bar{a}_i|, \bar{c}) \geq_\delta (\mathcal{A}_\infty, \langle 0, ..., |\bar{a}_i| - 1\rangle, \bar{c})$, as needed to show $(\mathcal{A}_i, \bar{a}_i) \leq_\gamma (\mathcal{A}_\infty, \langle 0, ..., |\bar{a}_i| - 1\rangle)$. □

With a bit more work, one can prove a sharper formulation of this lemma where the assumption is just that $\mathcal{A}_i \geq_{\alpha_i} \mathcal{A}_{i+1}$ for all i, the conclusion is that $\mathcal{A}_i \geq_{\alpha_i} \mathcal{A}_\infty$, and the sequence of α_i's is only assumed to be non-decreasing and may even be constant. We do not need that formulation here.

THEOREM XII.7. *If Θ is a counterexample to Vaught's conjecture, there is an $\mathcal{L}_{\omega_1,\omega}$-sentence φ such that $\Theta \wedge \varphi$ is a minimally unbounded counterexample to Vaught's conjecture.*

PROOF. Suppose, working toward a contradiction, that there is no such formula φ. Let α be such that Θ is $\Pi^{\mathtt{in}}_\alpha$.

The first step is to build a tree of structures $\{\mathcal{A}_\sigma : \sigma \in 2^{<\mathbb{N}}\}$ and an increasing sequence of countable ordinals $\langle \alpha_i : i \in \mathbb{N}\rangle$ with $\alpha_0 = \alpha$ such that, for all $i \in \mathbb{N}$, $\sigma, \tau \in 2^{<\mathbb{N}}$,

$$\sigma \restriction i = \tau \restriction i \quad \iff \quad \mathcal{A}_\sigma \equiv_{\alpha_i} \mathcal{A}_\tau.$$

[‡] Here, $\ulcorner\langle i,j\rangle\urcorner$ is the number coding the pair $\langle i,j\rangle$ in some standard effective bijection $\mathbb{N}^2 \to \mathbb{N}$.

[§] If $\bar{c}$ is not disjoint from $\bar{d}_0$, define $\bar{a}_{s+1}$ by adding only the elements of $\bar{c}$ that are not in $\bar{d}_0$.

We will then use this tree to build perfectly many models of Θ and reach a contradiction. The structures $\mathcal{A}_\sigma$ are defined by recursion on the length of σ. Let $\mathcal{A}_{\langle\rangle}$ be a model of Θ for which $\psi_{\mathcal{A},\alpha}$ is unbounded, where $\psi_{\mathcal{A},\alpha}$ is as in Lemma XII.5 above. In other words, $\mathcal{A}_{\langle\rangle}$ satisfies that there are structures $\equiv_\alpha$-equivalent to it of arbitrarily high Scott rank below ω_1. To see why such an $\mathcal{A}_{\langle\rangle}$ exists, notice that, since there exist only countably many $\equiv_\alpha$-equivalence classes among the models of Θ, if they were all bounded below ω_1, Θ would be bounded too. We remark that since Θ is $\Pi^{\mathtt{in}}_\alpha$ and $\mathcal{A}$ is a model of Θ, $\psi_{\mathcal{A},\alpha}$ implies Θ.

Suppose we have already defined α_i and $\mathcal{A}_\sigma$ for all $\sigma \in 2^i$ in such a way that $\psi_{\mathcal{A}_\sigma,\alpha_i}$ is unbounded. For each $\sigma \in 2^i$, since we are assuming that no sentence $\Theta \wedge \varphi$ is minimally unbounded, there must exist a sentence φ_σ such that both $\psi_{\mathcal{A}_\sigma,\alpha_i} \wedge \varphi_\sigma$ and $\psi_{\mathcal{A}_\sigma,\alpha_i} \wedge \neg\varphi_\sigma$ are unbounded. Let α_{i+1} be such that all those formulas are $\Pi^{\mathtt{in}}_{\alpha_{i+1}}$ for all $\sigma \in 2^i$. For each $\sigma \in 2^i$, let $\mathcal{A}_{\sigma^\frown 0}$ be a structure satisfying $\psi_{\mathcal{A}_\sigma,\alpha_i} \wedge \varphi_\sigma$ and such that $\psi_{\mathcal{A}_{\sigma^\frown 0},\alpha_{i+1}}$ is unbounded. To see why such a structure exists, notice that, since there exist only countably many $\equiv_{\alpha_{i+1}}$-equivalence classes among the models of $\psi_{\mathcal{A}_\sigma,\alpha_i} \wedge \varphi_\sigma$, one must be unbounded. Analogously, define $\mathcal{A}_{\sigma^\frown 1}$ satisfying $\psi_{\mathcal{A}_\sigma,\alpha_i} \wedge \neg\varphi_\sigma$ and such that $\psi_{\mathcal{A}_{\sigma^\frown 1},\alpha_{i+1}}$ is unbounded. Notice that both $\mathcal{A}_{\sigma^\frown 0}$ and $\mathcal{A}_{\sigma^\frown 1}$ are $\equiv_{\alpha_i}$-equivalent to $\mathcal{A}_\sigma$, but $\mathcal{A}_{\sigma^\frown 0} \not\equiv_{\alpha_{i+1}} \mathcal{A}_{\sigma^\frown 1}$. This finishes the construction of the tree.

Finally, for the contradiction, we build a perfect set of non-isomorphic models of Θ. For each $X \in 2^{\mathbb{N}}$, let $\mathcal{A}_X$ be a structure such that $\mathcal{A}_X \equiv_{\alpha_{i-3}} \mathcal{A}_{X\restriction i}$ for all $i \geq 2$, as given by Lemma XII.6 above. To see that these models are all non-isomorphic, consider X and $Y \in 2^{\mathbb{N}}$. Let i be such that $X \restriction i \neq Y \restriction i$. We then have

$$\mathcal{A}_X \equiv_{\alpha_i} \mathcal{A}_{X\restriction i+3} \not\equiv_{\alpha_i} \mathcal{A}_{Y\restriction i+3} \equiv_{\alpha_i} \mathcal{A}_Y.$$

Notice that these are all models of Θ, since, as we mentioned above, Θ is implied by $\psi_{\mathcal{A}_{\langle\rangle},\alpha_0}$. This contradiction with Θ being a counterexample to Vaught's conjecture came from the assumption that no sentence of the form $\Theta \wedge \varphi$ is minimally unbounded. □

The following lemma gives a characterization of minimally unbounded theories.

Lemma XII.8. *If Θ is minimally unbounded, there is a closed and unbounded set of countable ordinals $\mathbb{C}$ such that, for all $\alpha \in \mathbb{C}$, there is only one α-back-and-forth equivalence class among the models of Θ of Scott rank greater than or equal to α.*

Proof. We will define $\mathbb{C}$ as the set of fixed points of a monotone, continuous function $f\colon \omega_1 \to \omega_1$, which we define as follows. For each ordinal β, there must exist a unique β-back-and-forth equivalence class of models of Θ that is unbounded: There must be at least one because there are only countably many $\equiv_\beta$-equivalence classes and they cannot all be bounded. There is at most one because if $\mathcal{A}_\beta$ is a model of Θ in that $\equiv_\beta$-equivalence classes, then $\Theta \wedge \psi_{\mathcal{A}_\beta,\beta}$ is unbounded and, by the minimality of Θ, $\Theta \wedge \neg\psi_{\mathcal{A}_\beta,\beta}$ is bounded. Fix such a structure $\mathcal{A}_\beta$ for each $\beta < \omega_1$. Notice that if $\gamma > \beta$, then $\mathcal{A}_\gamma \equiv_\beta \mathcal{A}_\beta$. Define

$$\mathbb{K}_\beta = \{\mathcal{B} \in \mathit{Mod}(\Theta), \mathcal{B} \not\equiv_\beta \mathcal{A}_\beta\}$$

and

$$f(\beta) = \sup(\{\mathrm{SR}(\mathcal{B}) + 1 : \mathcal{B} \in \mathbb{K}_\beta\} \cup \{\beta\}).$$

Observe that, for an ordinal α, $f(\alpha) = \alpha$ if and only if all models in $\mathbb{K}_\alpha$ have Scott rank less than α. Equivalently, $f(\alpha) = \alpha$ if and only if $\mathcal{A}_\alpha$ is α-back-and-forth equivalent to all models of Θ of Scott rank greater than or equal to α. We define $\mathbb{C}$ as the set of fixed points of f.

If $\beta \leq \gamma$, then $\mathbb{K}_\beta \subseteq \mathbb{K}_\gamma$, and hence $f(\beta) \leq f(\gamma)$. If λ is a limit ordinal, then $\mathbb{K}_\lambda = \bigcup_{\gamma<\lambda} \mathbb{K}_\gamma$ by the continuity of the back-and-forth relations. It follows that $f(\lambda) = \sup_{\gamma<\lambda} f(\gamma)$. The function f is thus continuous and monotone.

$\mathbb{C}$ is unbounded because for every $\beta_0 \in \omega_1$, $\lim_{n\in\mathbb{N}} \overbrace{f \circ f \circ \cdots \circ f}^{n \text{ times}}(\beta_0)$ is a fixed point of f greater than or equal to β_0. It is closed because if $\gamma_0, \gamma_1, ...$ is an increasing sequence of members of $\mathbb{C}$ with limit λ, then $f(\lambda) = \sup_{i\in\mathbb{N}} f(\gamma_i) = \sup_{i\in\mathbb{N}} \gamma_i = \lambda$, and hence $\lambda \in \mathbb{C}$ too. □

REMARK XII.9. For Θ as in the previous lemma, we also get that relative to all oracles on some cone, all models of Θ of Scott rank greater than or equal to ω_1^{CK} are ω_1^{CK}-back-and-forth equivalent to each other. This follows from a descriptive set theoretic consequence of Turing determinacy: For every closed and unbounded set $\mathbb{C} \subseteq \omega_1$, there is an oracle C such that all C-admissible ordinals belong to $\mathbb{C}$, where the C-admissible ordinals are those of the form ω_1^X for some $X \geq_T C$.

XII.3. Connections with computability theory

Minimal counterexamples to Vaught's conjecture, if they exist, have very interesting computability theoretic properties. Studying their properties could either help us build one or lead us to a contradiction and a proof that they do not exist.

Suppose that an $\mathcal{L}_{\omega_1,\omega}$ sentence Θ is a minimal counterexample to Vaught's conjecture. The following properties hold on a cone:[¶]

(1) Every model $\mathcal{A}$ of Θ with $\omega_1^{\mathcal{A}} = \omega_1^{CK}$ has a computable copy. Furthermore, there is a computable list containing all those models and where their back-and-forth relations are computable up to ω_1^{CK}.[‖]

(2) Every model $\mathcal{A}$ of Θ has degree spectrum $\{X \in 2^{\mathbb{N}} : \omega_1^X \geq \omega_1^{\mathcal{A}}\}$.

(3) There is only one computable model of Θ of high Scott rank, and it has Scott rank $\omega_1^{CK} + 1$.

(4) Θ is intermediate for effective reducibility.

We will not prove these results, as the techniques fall outside the scope of this book.

Property (1) was proved in [**Mon13**] using the existence of $0^\sharp$. The first line, namely that all models with $\omega_1^{\mathcal{A}} = \omega_1^{CK}$ have computable copies, follows from a much more general result from [**Mon15a**]: If a $\mathbf{\Sigma}_1^1$-equivalence relation E on $2^{\mathbb{N}}$ does not have perfectly many equivalence classes relative to all oracles on some cone, we have that, every real that is low for ω_1 is E-equivalent to a computable real. The second line, namely that one can list all those models in a way that the back-and-forth structure is computable, is done in [**Mon13**]. To show that the oracles that can do this are co-final in the Turing degrees, one has to use overspill to show that, given X and a list of X-computable models of Θ, there is a $Y \geq_T X$ with $\omega_1^X = \omega_1^Y$ that computes the back-and-forth relations on that list up to some X-computable Harrison linear ordering. Once we know that those oracles are co-final, we can use Turing determinacy to get that there is a cone of them. It is not hard to see that if Θ satisfies (1) on a cone, then it cannot have perfectly many models. It was proved in [**Mon13**] that an $\mathcal{L}_{\omega_1,\omega}$-sentence Θ is a counterexample to Vaught's conjecture if and only if it is unbounded and satisfies that, for every oracle on some cone, every hyperarithmetic model has a computable copy. This is a computability theoretic statement equivalent to Vaught's conjecture.

Item (2) follows from (1) by relativization.

For (3), let C be an oracle such that all ordinals of the form ω_1^X for $X \geq_T C$ belong to the set $\mathbb{C}$ as in Remark XII.9. Furthermore, assume

[¶] When we say that a property holds *on a cone* we mean that there exist a $C \in 2^{\mathbb{N}}$ such that, for every $X \geq_T C$, we have that the property holds relative to X.

[‖] We mean that the back-and-forth relations are computable up to ω_1^{CK}, but not including ω_1^{CK}, of course.

that relative to all oracles on that cone, there is a list of all the computable models of Θ where the back-and-forth structure is computable as in (1). First, there must exist at least one model of high-Scott rank because of Corollary VI.26. Since there is only one $\equiv_{\omega_1^{CK}}$-equivalence class that contains all the models of Θ of Scott Rank beyond ω_1^{CK}, we get that all computable models of Θ of high-Scott rank must be $\equiv_{\omega_1^{CK}}$-equivalent to each other. They must then be isomorphic by Lemma VI.17. By Turing determinacy, either this unique model has Scott rank $\omega_1^{CK}+1$ on a cone or it has Scott rank ω_1^{CK} on a cone. The latter situation is ruled out by a result of Sacks [**Sac83**] that states that, on a counterexample to Vaught's conjecture, there must exist models with $SR(\mathcal{A}) = \omega_1^{\mathcal{A}} + 1$.

We know of no examples of $\mathcal{L}_{\mathsf{c},\omega}$-sentences with a unique computable model of high-Scott rank, except for the counterexamples to Vaught's conjecture — of which we know none. It is not known whether Θ having exactly one model of high Scott rank relative to every oracle on some cone implies that Θ is a counterexample to Vaught's conjecture.

Property (4) refers to the notion of effective reducibility introduced in Section XI.4. Use the same cone as in the previous paragraphs. The equivalence relation iso(Mod_Θ) on ω would then have only one non-hyperarithmetic equivalence class, namely the class of indices of the unique model of high Scott rank. All the other ones have Scott rank below ω_1^{CK}, and hence have computable Scott sentences (Theorem VI.15), which makes their equivalence class hyperarithmetic. There are plenty of Σ_1^1 equivalence relations on $\mathbb{N}$ that have more than one equivalence class that is Σ_1^1 complete. These relations could not effectively reduce to iso(Mod_Θ). It follows that iso(Mod_Θ) is not on top for effective reducibility. Since we have a list of the computable models of Θ where we can compute the back-and-forth relations, we get from the pair-of-structures theorem (Theorem VIII.7) that the isomorphism relation cannot be hyperarithmetic. So, iso(Mod_Θ) is intermediate for effective reducibility.[**] It is not know whether this is the only way to obtain a $\mathcal{L}_{\mathsf{c},\omega}$-sentence that is intermediate for effective reducibility. Results of [**Mon16**] get close, but short of proving that Vaught's conjecture is equivalent to the statement that there are no $\mathcal{L}_{\omega_1,\omega}$-sentences that are intermediate for effective reducibility relative to all oracles on some cone.

Example XII.10. Here is an example of a class that satisfies all the computability theoretic properties listed above, except that it is

[**] The proof we present here is due to Knight and Montalbán [unpublished]. This was also proved independently by Becker [**Bec13**].

not $\mathcal{L}_{\omega_1,\omega}$-axiomatizable. A class of structures is said to be *pseudo-elementary* if it can be axiomatized by a sentence of the form $\exists R\, \varphi(R)$, where φ is a $\tau \cup \{R\}$-formula, and R is a second-order variable for a relation. Such classes are $\mathbf{\Sigma}^1_1$ but not necessarily Borel. Kunen found the following pseudo-elementary class that has uncountably but not perfectly many models: the class of linear orderings on which every two elements are automorphic. It can be shown these are exactly the linear orderings of the form $\mathbb{Z}^{\mathcal{L}}$ for some linear ordering $\mathcal{L}$ (see Exercise I.14), which as we saw in Observation I.10, are the linear orderings of form $\mathbb{Z}^\alpha$ or $\mathbb{Z}^\alpha \cdot \mathbb{Q}$ for some ordinal α.

Bibliography

[AGHTT] Rachel Alvir, Noam Grenberg, Matthew Harrison-Trainor, and Dan Turetsky. Scott complexity of countable structures. submitted for publication.

[AJK90] C. J. Ash, C. G. Jockusch, Jr., and J. F. Knight. Jumps of orderings. *Trans. Amer. Math. Soc.*, 319(2):573–599, 1990.

[AK90] C. J. Ash and J. F. Knight. Pairs of recursive structures. *Ann. Pure Appl. Logic*, 46(3):211–234, 1990.

[AK94a] C. J. Ash and J. F. Knight. Mixed systems. *J. Symbolic Logic*, 59(4):1383–1399, 1994.

[AK94b] C. J. Ash and J. F. Knight. Ramified systems. *Ann. Pure Appl. Logic*, 70(3):205–221, 1994.

[AK00] C.J. Ash and J. Knight. *Computable Structures and the Hyperarithmetical Hierarchy.* Elsevier Science, 2000.

[AK18] Uri Andrews and Julia F. Knight. Strongly minimal theories with recursive models. *J. Eur. Math. Soc. (JEMS)*, 20(7):1561–1594, 2018.

[AKM20] Rachael Alvir, Julia F. Knight, and Charles McCoy. Complexity of Scott sentences. *Fund. Math.*, 251(2):109–129, 2020.

[AKMS89] Chris Ash, Julia Knight, Mark Manasse, and Theodore Slaman. Generic copies of countable structures. *Ann. Pure Appl. Logic*, 42(3):195–205, 1989.

[Alv] Rachel Alvir. Finitely α-generated structures. In preparation.

[AR] Rachael Alvir and Dino Rossegger. The complexity of scott sentences of scattered linear orderings. Submitted for publication.

[AR20] Rachael Alvir and Dino Rossegger. The complexity of Scott sentences of scattered linear orders. *J. Symb. Log.*, 85(3):1079–1101, 2020.

[Ash86a] C. J. Ash. Recursive labelling systems and stability of recursive structures in hyperarithmetical degrees. *Trans. Amer. Math. Soc.*, 298(2):497–514, 1986.

[Ash86b] C. J. Ash. Stability of recursive structures in arithmetical degrees. *Ann. Pure Appl. Logic*, 32(2):113–135, 1986.

[Ash87] C. J. Ash. Categoricity in hyperarithmetical degrees. *Ann. Pure Appl. Logic*, 34(1):1–14, 1987.

[Ash90] C. J. Ash. Labelling systems and r.e. structures. *Ann. Pure Appl. Logic*, 47(2):99–119, 1990.

[Ash91] C. J. Ash. A construction for recursive linear orderings. *J. Symbolic Logic*, 56(2):673–683, 1991.

[Bar67] Kenneth Jon Barwise. *Infinitary logic and admissible sets.* PhD thesis, Stanford University, 1967.

[Bar69] Jon Barwise. Infinitary logic and admissible sets. *J. Symbolic Logic*, 34(2):226–252, 1969.

[Bar75] Jon Barwise. *Admissible sets and structures.* Springer-Verlag, Berlin, 1975. An approach to definability theory, Perspectives in Mathematical Logic.

[Bec13] Howard Becker. Isomorphism of computable structures and Vaught's conjecture. *J. Symbolic Logic*, 78(4):1328–1344, 2013.

[Bur78] John P. Burgess. Equivalences generated by families of Borel sets. *Proc. Amer. Math. Soc.*, 69(2):323–326, 1978.

[CCKM04] W. Calvert, D. Cummins, J. F. Knight, and S. Miller. Comparison of classes of finite structures. *Algebra Logika*, 43(6):666–701, 759, 2004.

[CFG$^+$09] John Chisholm, Ekaterina B. Fokina, Sergey S. Goncharov, Valentina S. Harizanov, Julia F. Knight, and Sara Quinn. Intrinsic bounds on complexity and definability at limit levels. *J. Symbolic Logic*, 74(3):1047–1060, 2009.

[CG01] Riccardo Camerlo and Su Gao. The completeness of the isomorphism relation for countable Boolean algebras. *Trans. Amer. Math. Soc.*, 353(2):491–518, 2001.

[Chi90] John Chisholm. Effective model theory vs. recursive model theory. *J. Symbolic Logic*, 55(3):1168–1191, 1990.

[CHM12] Samuel Coskey, Joel David Hamkins, and Russell Miller. The hierarchy of equivalence relations on the natural numbers under computable reducibility. *Computability*, 1(1):15–38, 2012.

[CHT17] Barbara F. Csima and Matthew Harrison-Trainor. Degrees of categoricity on a cone via η-systems. *J. Symb. Log.*, 82(1):325–346, 2017.

[CKM06] Wesley Calvert, Julia F. Knight, and Jessica Millar. Computable trees of Scott rank ω_1^{CK}, and computable approximation. *J. Symbolic Logic*, 71(1):283–298, 2006.

[Coo04] S. Barry Cooper. *Computability theory.* Chapman & Hall/CRC, Boca Raton, FL, 2004.

[Cut80] Nigel Cutland. *Computability.* Cambridge University Press, Cambridge-New York, 1980. An introduction to recursive function theory.

[DKL$^+$15] Rodney G. Downey, Asher M. Kach, Steffen Lempp, Andrew E. M. Lewis-Pye, Antonio Montalbán, and Daniel D. Turetsky. The complexity of computable categoricity. *Advances in Mathematics*, 268:423–466, 2015.

[End11] Herbert B. Enderton. *Computability theory.* Elsevier/Academic Press, Amsterdam, 2011. An introduction to recursion theory.

[Fei70] Lawrence Feiner. Hierarchies of Boolean algebras. *J. Symbolic Logic*, 35:365–374, 1970.

[FF09] Ekaterina B. Fokina and Sy-David Friedman. Equivalence relations on classes of computable structures. In *Mathematical theory and computational practice*, volume 5635 of *Lecture Notes in Comput. Sci.*, pages 198–207. Springer, Berlin, 2009.

[FFH$^+$12] E. B. Fokina, S. Friedman, V. Harizanov, J. F. Knight, C. McCoy, and A. Montalbán. Isomorphism relations on computable structures. *Journal of Symbolic Logic*, 77(1):122–132, 2012.

[Fri57] Richard M. Friedberg. Two recursively enumerable sets of incomparable degrees of unsolvability (solution of Post's problem, 1944). *Proc. Nat. Acad. Sci. U.S.A.*, 43:236–238, 1957.

[FS89] Harvey Friedman and Lee Stanley. A Borel reducibility theory for classes of countable structures. *J. Symbolic Logic*, 54(3):894–914, 1989.

[Gan60] R. O. Gandy. Proof of Mostowski's conjecture. *Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys.*, 8:571–575, 1960.

[Gao01] Su Gao. Some dichotomy theorems for isomorphism relations of countable models. *J. Symbolic Logic*, 66(2):902–922, 2001.

[GHK^{+}05] Sergey Goncharov, Valentina Harizanov, Julia Knight, Charles McCoy, Russell Miller, and Reed Solomon. Enumerations in computable structure theory. *Ann. Pure Appl. Logic*, 136(3):219–246, 2005.

[GMS13] N. Greenberg, A. Montalbán, and T. A. Slaman. Relative to any nonhyperarithmetic set. *Journal of Mathematical Logic*, 13(1), 2013.

[GT] Noam Greenberg and Daniel Turetsky. Completeness of the hyperaritmetic isomorphism equivalence relation. Submitted for publication.

[Har68] J. Harrison. Recursive pseudo-well-orderings. *Transactions of the American Mathematical Society*, 131:526–543, 1968.

[Har76] L. Harrington. Mclaughlin's conjecture. Handrwitten notes, 11 pages, September 76.

[HKM] Denis Hirschfeldt, Asher M. Kach, and Antonio Montalbán. A Feiner look at the intermediate degrees. Unpublished notes.

[HM77] V. Harnik and M. Makkai. A tree argument in infinitary model theory. *Proc. Amer. Math. Soc.*, 67(2):309–314, 1977.

[HM12] Kenneth Harris and Antonio Montalbán. On the n-back-and-forth types of Boolean algebras. *Trans. Amer. Math. Soc.*, 364(2):827–866, 2012.

[HT18] Matthew Harrison-Trainor. Scott ranks of models of a theory. *Adv. Math.*, 330:109–147, 2018.

[HTMM] M. Harrison-Trainor, R. Miller, and A. Montalbán. Generic functors and infinitary interpretations. In preparation.

[Joc68] Carl G. Jockusch, Jr. Semirecursive sets and positive reducibility. *Trans. Amer. Math. Soc.*, 131:420–436, 1968.

[JS72] Carl G. Jockusch, Jr. and Robert I. Soare. Degrees of members of Π^0_1 classes. *Pacific J. Math.*, 40:605–616, 1972.

[JS94] Carl G. Jockusch, Jr. and Robert I. Soare. Boolean algebras, Stone spaces, and the iterated Turing jump. *J. Symbolic Logic*, 59(4):1121–1138, 1994.

[Kar65] Carol R. Karp. Finite-quantifier equivalence. In *Theory of Models (Proc. 1963 Internat. Sympos. Berkeley)*, pages 407–412. North-Holland, Amsterdam, 1965.

[Kei71] H. Jerome Keisler. *Model theory for infinitary logic. Logic with countable conjunctions and finite quantifiers.* North-Holland Publishing Co., Amsterdam, 1971. Studies in Logic and the Foundations of Mathematics, Vol. 62.

[KM10] J. F. Knight and J. Millar. Computable structures of rank ω_1^{CK}. *J. Math. Log.*, 10(1-2):31–43, 2010.

[KMVB07] Julia F. Knight, Sara Miller, and M. Vanden Boom. Turing computable embeddings. *J. Symbolic Logic*, 72(3):901–918, 2007.

[Kni86] Julia F. Knight. Degrees coded in jumps of orderings. *J. Symbolic Logic*, 51(4):1034–1042, 1986.

[Kni95] Julia F. Knight. Requirement systems. *J. Symbolic Logic*, 60(1):222–245, 1995.

[Kre61] G. Kreisel. Set theoretic problems suggested by the notion of potential totality. In *Infinitistic Methods (Proc. Sympos. Foundations of Math., Warsaw, 1959)*, pages 103–140. Pergamon, Oxford; Państwowe Wydawnictwo Naukowe, Warsaw, 1961.

[Kun80] K. Kunen. *Set Theory. An Introduction to Independence Proofs.* North Holland, 1980.

[Lac73] A. H. Lachlan. The priority method for the construction of recursively enumerable sets. In *Cambridge Summer School in Mathematical Logic (Cambridge, 1971)*, pages 299–310. Lecture Notes in Math., Vol. 337. Springer, Berlin, 1973.

[Lac76] Alistair H. Lachlan. A recursively enumerable degree which will not split over all lesser ones. *Ann. Math. Logic*, 9(4):307–365, 1976.

[LE65] E. G. K. Lopez-Escobar. An interpolation theorem for denumerably long formulas. *Fund. Math.*, 57:253–272, 1965.

[LE66] E. G. K. Lopez-Escobar. On defining well-orderings. *Fund. Math.*, 59:13–21, 1966.

[Ler10] Manuel Lerman. *A framework for priority arguments*, volume 34 of *Lecture Notes in Logic.* Association for Symbolic Logic, La Jolla, CA, 2010.

[LL90] Steffen Lempp and Manuel Lerman. Priority arguments using iterated trees of strategies. In *Recursion theory week (Oberwolfach, 1989)*, volume 1432 of *Lecture Notes in Math.*, pages 277–296. Springer, Berlin, 1990.

[LL95] Steffen Lempp and Manuel Lerman. A general framework for priority arguments. *Bull. Symbolic Logic*, 1(2):189–201, 1995.

[Mac77] John M. Macintyre. Transfinite extensions of Friedberg's completeness criterion. *J. Symbolic Logic*, 42(1):1–10, 1977.

[Mak81] M. Makkai. An example concerning Scott heights. *J. Symbolic Logic*, 46(2):301–318, 1981.

[Mar75] Donald A. Martin. Borel determinacy. *Ann. of Math. (2)*, 102(2):363–371, 1975.

[Mar16] David Marker. *Lectures on infinitary model theory*, volume 46 of *Lecture Notes in Logic.* Association for Symbolic Logic, Chicago, IL; Cambridge University Press, Cambridge, 2016.

[Mil78] Douglas E. Miller. The invariant Π^0_α separation principle. *Trans. Amer. Math. Soc.*, 242:185–204, 1978.

[Mil83] Arnold W. Miller. On the Borel classification of the isomorphism class of a countable model. *Notre Dame J. Formal Logic*, 24(1):22–34, 1983.

[MM11] Alberto Marcone and A. Montalbán. The Veblen functions for computability theorists. *Journal of Symbolic Logic*, 76(2):575–602, 2011.

[MM17] David Marker and Russell Miller. Turing degree spectra of differentially closed fields. *J. Symb. Log.*, 82(1):1–25, 2017.

[Mon] Antonio Montalbán. A new game metatheorem for ash-knight style priority constructions. Submitted for publication.

[Mon10] Antonio Montalbán. Counting the back-and-forth types. *Journal of Logic and Computability*, page doi: 10.1093/logcom/exq048, 2010.

[Mon13] Antonio Montalbán. A computability theoretic equivalent to Vaught's conjecture. *Adv. Math.*, 235:56–73, 2013.

[Mon14] Antonio Montalbán. Priority arguments via true stages. *Journal of Symbolic Logic*, 79(4):1315–1335, 2014.

[Mon15a] Antonio Montalbán. Analytic equivalence relations satisfying hyperarithmetic-is-recursive. *Forum Math. Sigma*, 3:e8, 11, 2015.

[Mon15b] Antonio Montalbán. A robuster Scott rank. *Proc. Amer. Math. Soc.*, 143(12):5427–5436, 2015.

[Mon16] Antonio Montalbán. Classes of structures with no intermediate isomorphism problems. *J. Symb. Log.*, 81(1):127–150, 2016.

[Mor] Michael Morley. The hanf number for ω-logic. (Abstract).

[Mor65] Michael Morley. Omitting classes of elements. In *Theory of Models (Proc. 1963 Internat. Sympos. Berkeley)*, pages 265–273. North-Holland, Amsterdam, 1965.

[Mor70] Michael Morley. The number of countable models. *J. Symbolic Logic*, 35:14–18, 1970.

[MPSS18] Russell Miller, Bjorn Poonen, Hans Schoutens, and Alexandra Shlapentokh. A computable functor from graphs to fields. *J. Symb. Log.*, 83(1):326–348, 2018.

[Muc56] A. A. Muchnik. On the unsolvability of the problem of reducibility in the theory of algorithms. *Dokl. Akad. Nauk SSSR, N.S.*, 108:194–197, 1956.

[Nad74] Mark Nadel. Scott sentences and admissible sets. *Ann. Math. Logic*, 7:267–294, 1974.

[Nur74] A. T. Nurtazin. Strong and weak constructivizations, and enumerable families. *Algebra i Logika*, 13:311–323, 364, 1974.

[PS] Gianluca Paolini and Saharon Shelah. Torsion-free abelian groups are borel complete. Submitted for publication.

[Res73] J.-P. Ressayre. Boolean models and infinitary first order languages. *Ann. Math. Logic*, 6:41–92, 1973.

[Res77] J. P. Ressayre. Models with compactness properties relative to an admissible language. *Ann. Math. Logic*, 11(1):31–55, 1977.

[Sac63] Gerald E. Sacks. *Degrees of unsolvability.* Princeton University Press, Princeton, N.J., 1963.

[Sac83] Gerald E. Sacks. On the number of countable models. In *Southeast Asian conference on logic (Singapore, 1981)*, volume 111 of *Stud. Logic Found. Math.*, pages 185–195. North-Holland, Amsterdam, 1983.

[Sco65] Dana Scott. Logic with denumerably long formulas and finite strings of quantifiers. In *Theory of Models (Proc. 1963 Internat. Sympos. Berkeley)*, pages 329–341. North-Holland, Amsterdam, 1965.

[Sho61] J. R. Shoenfield. Undecidable and creative theories. *Fund. Math.*, 49:171–179, 1960/61.

[Sil80] Jack H. Silver. Counting the number of equivalence classes of Borel and coanalytic equivalence relations. *Ann. Math. Logic*, 18(1):1–28, 1980.

[Soa16] Robert I. Soare. *Turing computability.* Theory and Applications of Computability. Springer-Verlag, Berlin, 2016. Theory and applications.

[Sos13] Ivan N. Soskov. A note on ω-jump inversion of degree spectra of structures. In *The nature of computation*, volume 7921 of *Lecture Notes in Comput. Sci.*, pages 365–370. Springer, Heidelberg, 2013.

[Spe55] Clifford Spector. Recursive well-orderings. *Journal Symbolic Logic*, 20:151–163, 1955.

[Spe60] C. Spector. Hyperarithmetical quantifiers. *Fund. Math.*, 48:313–320, 1959/1960.

[Ste75] John Steel. Descending sequences of degrees. *J. Symbolic Logic*, 40(1):59–61, 1975.

[Ste78] John R. Steel. On Vaught's conjecture. In *Cabal Seminar 76–77 (Proc. Caltech-UCLA Logic Sem., 1976–77)*, volume 689 of *Lecture Notes in Math.*, pages 193–208. Springer, Berlin, 1978.

[Thu94] John J. Thurber. Recursive and r.e. quotient Boolean algebras. *Arch. Math. Logic*, 33(2):121–129, 1994.

[Vau61] R. L. Vaught. Denumerable models of complete theories. In *Infinitistic Methods (Proc. Sympos. Foundations of Math., Warsaw, 1959)*, pages 303–321. Pergamon, Oxford, 1961.

[Vau75] Robert Vaught. Invariant sets in topology and logic. *Fund. Math.*, 82:269–294, 1974/75.

[VB07] M. Vanden Boom. The effective Borel hierarchy. *Fund. Math.*, 195(3):269–289, 2007.

[Wad83] William Wilfred Wadge. *Reducibility and Determinateness on the Baire Space*. ProQuest LLC, Ann Arbor, MI, 1983. Thesis (Ph.D.)–University of California, Berkeley.

[Wat84] Richard Watnick. A generalization of Tennenbaum's theorem on effectively finite recursive linear orderings. *J. Symbolic Logic*, 49(2):563–569, 1984.