Computable Structure Theory:
Beyond the arithmetic

Draft

Antonio Montalbán

Draft of Part 2 - Compiled on October 29, 2020

Saved: Oct, 2019
Compiled: October 29, 2020

Contents

Part 2

Beyond the arithmetic

Preface

Notation and Conventions

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 would 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 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 may choose to keep in mind whichever definition of computability feels intuitively more comfortable, 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 of them as using any kind of finitary object as inputs or outputs. This

[*]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 a finitely many steps using a finite amount of memory.

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

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 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 decide whether $\Phi_{e,s}(n)$ converges or not, computably: 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. The *Padding Lemma* states that every partial computable function has infinitely many indices — just add dummy instructions at the end of a program, getting essentially the same program, but with a different index.

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. We sometimes use Lachlan's notation: $W_e[s]$ instead of $W_{e,s}$. 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 during its own computation, and in particular is allowed to call and

---

[‡]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.

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.

[§]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 $\varphi_e(n) = \varphi_{g(e_0)} = f(\varphi_{e_0}(e_0), n) = f(g(e_0), n) = f(e, n)$.

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 $f : \mathbb{N} \to \mathbb{N}$ 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 $\leq_m$. 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 but requiring 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. That is why 1-equivalent sets are said to be *computably isomorphic.*

Turing reducibility. Given a function $f : \mathbb{N} \to \mathbb{N}$, we say that a partial function $g : \mathbb{N} \rightharpoonup \mathbb{N}$ 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 $\Phi_e^\sigma(n)$ 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; 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 $n < |\sigma|$.

Enumeration reducibility. Recall that an enumeration of a set A is just an onto function $f\colon \mathbb{N} \to A$. Given $A, B \subseteq \mathbb{N}$, we say that A is *enumeration reducible* (or *e-reducible*) to B, and write $A \leq_e B$, if every enumeration of B computes an enumeration of A. Selman [**Sel71**] showed that we can make this reduction uniformly: $A \leq_e B$ if and only if there is a Turing operator Φ such that, for every enumeration f of B, Φ^f is an enumeration of A. (See Theorem [**MonP1**, Theorem IV.12].) Another way of defining enumeration reducibility is via *enumeration operators*: An enumeration operator is a c.e. set Θ of pairs that acts as follows: For $B \subseteq \mathbb{N}$, we define

$$\Theta^B = \{n : (\exists D \subseteq_{fin} B)\ \langle \ulcorner D \urcorner, n \rangle \in \Theta\},$$

where $\subseteq_{fin}$ means 'finite subset of.' Selman also showed that $A \leq_e B$ if and only if there is an enumeration operator Θ such that $A = \Theta^B$.

The Turing degrees embed into the enumeration degrees via the map $\iota(A) = A \oplus A^c$. It is not hard to show that $A \leq_T B \iff \iota(A) \leq_e \iota(B)$.

Positive reducibility. We say that A *positively reduces* to B, and write $A \leq_p B$, if there is a computable function $f\colon \mathbb{N} \to (\mathbb{N}^{<\mathbb{N}})^{<\mathbb{N}}$ such that, for every $n \in \mathbb{N}$, $n \in A$ if and only if there is an $i < |f(n)|$ such that every entry of $f(n)(i)$ is in B [**Joc68**]. That is,

$$n \in A \iff \bigvee_{i<|f(n)|} \bigwedge_{j<|f(n)(i)|} f(n)(i)(j) \in B.$$

Notice that $\leq_p$ implies both Turing reducibility and enumeration reducibility, and is implied by many-one reducibility. In particular, the classes of computable sets and of c.e. sets are both closed downwards under $\leq_p$.

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. Furthermore, except for a few occasions, they will always be computable.

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*, 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.

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 $A \subseteq B$, $f_i^{\mathcal{A}} = f_i^{\mathcal{B}} \restriction A^{a_F(i)}$,

---

[¶] 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 $e \in A \iff \langle e, e\rangle \notin K \iff e \notin W_e \iff e \notin A$.

$R_j^{\mathcal{A}} = R_j^{\mathcal{B}} \restriction A^{a_R(i)}$ 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 $\tau_1 \setminus \tau_0$ 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, disjunctions, and implications. If we close the quantifier-free τ-formulas under existential quantification, we get the *existential τ-formulas*, of $\exists$-*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 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 ($x \leq y \ \&\ y \leq z \rightarrow x \leq z$), reflexive ($x \leq x$), and anti-symmetric ($x \leq y \ \&\ y \leq x \rightarrow x = y$). 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 $b_1 = b_2$

**By $\tau_0 \subseteq \tau_1$ we mean that every symbol in τ_0 is also in τ_1 and with the same arity 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 $\{x \in A : a <_{\mathcal{A}} x <_A b\}$. 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, and where 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 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; and 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}

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

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 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 $T \subseteq \mathbb{N}^{<\mathbb{N}}$: That is, if W is a c.e. operator such that $\psi(F) \iff 0 \notin W^F$, then $T = \{\sigma : 0 \in W^\sigma\}$ satisfies that $\psi(F) \iff F \in [T]$.

CHAPTER I

Ordinals

The ordinal numbers were introduced by Cantor in 1883 with the intention of extending the iteration of his derivative process beyond the finite steps of the iteration. 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$\underline{e}$tic* if it can be defined within ar$\underline{i}$thmetic, that is, within the structure $(\mathbb{N}; 0, 1, +, \times, \leq)$. The first step to go beyond *the* arithm$\underline{e}$tic is to extend ar$\underline{i}$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.

A linear ordering is well-ordered if and only 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.

When we talk about an *ordinal*, what we are referring to is 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 $(\subseteq\omega)$-presentation.

Let us 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 $W \setminus I$ 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*

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

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

$$g(a) = \Psi(a, g \upharpoonright W_{<a}) \quad \text{for every } a \in \mathrm{dom}(g). \tag{1}$$

First, we claim that if $f, g \in \mathcal{C}$, then f and g coincide on their common domain: If not, let $a \in \mathrm{dom}(f) \cap \mathrm{dom}(g)$ be a minimal element such that $f(a) \neq g(a)$. By the minimality of a, $f \upharpoonright W_{<a} = g \upharpoonright W_{<a}$, 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 $f(a) = \Psi(a, f \restriction W_{<a}) = \Psi(a, g \restriction W_{<a}) = g(a)$, 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 $f\colon \mathrm{dom}(g) \cup \{a\} \to X$ by copying g on $\mathrm{dom}(g)$, and letting $f(a) = \Psi(a, g)$. This new function clearly belongs to $\mathcal{C}$, but 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 $a, f(a), f(f(a)), ...$ would be an infinite descending sequence in α, which would contradict the well-orderness of α. To see this, we first note that $f(a) < a$ 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 $a \in \mathrm{dom}(g)$, 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.

The operations of addition and multiplication on ω_1 are just the addition and multiplication of linear orderings defined in page 13. 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: $1 + \omega \cong \omega \not\cong \omega + 1$ and $2\times\omega \cong \omega \not\cong \omega+\omega \cong \omega\times 2$. 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-other 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 order, 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 ordering-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, one can prove that $\mathcal{B}^\mathcal{A}$ has the same order type 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:

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

We leave the verification of these properties to the reader.

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

Exercise I.10. Prove that if $f\colon \omega_1 \to \omega_1$ is order preserving and continuous, it has uncountably many fixed points.

Exercise I.11. Prove that if $\mathcal{A}$ has no least element, $\mathcal{B}^{\mathcal{A}}$ is dense and has no endpoints

Exercise I.12. 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.13. 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.)

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 18.

Theorem I.14 (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.15 (Well-founded recursion). *Let $\mathcal{P} = (P; \leq_P)$ be a well-founded partial ordering, X any set, and Ψ an operator that, given*

[‡]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.

$a \in P$ 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

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

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.16. 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:

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

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.17. *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

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

[¶]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.

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.18. *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

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

In the case of trees we also get the converse.

Lemma I.19. *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 right-to-left 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 a $\subsetneq$-preserving map $f\colon T \to S$ defining $f(\tau)$ by recursion on the length $|\tau|$ 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.20. 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

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

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

THEOREM I.21. *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 28. Let us start with the reduction from trees to linear orderings.

DEFINITION I.22. 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.23. 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 $\text{KB}(T) = (T; \leq_{\text{KB}})$. Notice that

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

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

This gives us the reduction from trees to linear orderings we need for Theorem I.21:

THEOREM I.24. *A tree $T \subseteq \mathbb{N}^{<\mathbb{N}}$ is well-founded if and only if $(T; \leq_{\text{KB}})$ is well-ordered.*

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

Suppose now that $(T; \leq_{\text{KB}})$ is not well-ordered, and that $\sigma_0 \geq_{\text{KB}} \sigma_1 \geq_{\text{KB}} \sigma_2 \geq_{\text{KB}} \cdots$ is an infinite $\leq_{\text{KB}}$-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.25. (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 $\mathcal{P}$, 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.26. 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 $p \in \mathcal{P}$, 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.21. The class of well-founded trees effectively Wadge-reduces to the class of well-orderings via the Kleene-Brouwer ordering as in Theorem I.24. 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.21 holds the same way if, instead of considering ($\subseteq\omega$)-presentations, we consider indices for computable ($\subseteq\omega$)-presentations of well-orders, well-founded posets, and well-founded trees: the three

[‖]Repeat each branch infinitely often.

sets of indices are m-equivalent as sets of natural numbers. We will call this m-degree 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 is not to have to worry about whether a number is an index for a linear ordering or not. 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.27. *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 $\bigcup_s \mathcal{A}_{e,s}$. $\square$

DEFINITION I.28. 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. One can easily prove using Theorem I.21 that 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 believe $\mathcal{O}_{\text{wo}}$ is more natural, more direct, and closer to intuition. Kleene's original definition was quite different in format, but similar. Kleene created his own way of indexing the computable linear orderings 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.29. Show that $\mathcal{O}_{\text{wo}}$ is m-equivalent to the set of numbers e for which Φ_e is total and is the diagram of a $(\subseteq\omega)$-presentation of a well-ordering.

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 $\mathcal{L} \leq \omega_1^{CK}$. 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 $\mathcal{L} \cong \omega_1^{CK}$.

[**]This is an $(\subseteq\omega)$-presentation, but, since it is infinite, one can easily make it into an ω-presentation as in [**MonP1**, Observation I.3].

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,

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

THEOREM I.30. *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 undefinied.[††] 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 7 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, not necessarily computable. 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**].

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 simply 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 $\mathcal{L}_{\omega_1,\omega}$. We will mention $\mathcal{L}_{\infty,\omega}$ in some remarks here and there, as 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 $\bigvee\!\!\!\!\bigvee$. We reserve the symbol $\bigvee$ for infinitary conjunctions.

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 a 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 limit, 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 following $\Pi^{\mathtt{in}}_{2\cdot\gamma+1}$ formula which states that x has rank below $\omega \cdot \gamma + n$ for finite $n \geq 1$:

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

Finally, if $\beta = \gamma + 1$, then $\varphi_{\omega\beta}(x)$ is the formula $\bigvee\!\!\!\bigvee_{n\in\omega} \varphi_{\omega\gamma+n}(x)$ is a $\Sigma^{\mathtt{in}}_{2\cdot\gamma+2}$ formula stating that x has rank below $\omega \cdot \beta$. □

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 limit, 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 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 $\beta = \gamma+1$, $\varphi_{\omega^\beta}(x,y)$ is the formula $\bigvee\!\!\!\bigvee_{n\in\mathbb{N}} \varphi_{\omega^\gamma\cdot n}$, 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 [**MonP1**, 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. 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 $d \in B$, $\langle \bar{a}c, \bar{b}d\rangle \notin I$. 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,

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

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 [**MonP1**, 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:

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

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

where $D_\alpha(\bar{a})$ is the finite atomic diagram of the tuple $\bar{a}$ in $\mathcal{A}$ as defined in ?? and, if $D_{\mathcal{A}}(\bar{a}) = \sigma \in 2^{<\mathbb{N}}$, "$D(\bar{x}) = \sigma$" is the quantifier free formula stating that the atomic diagram of $\bar{x}$ is σ as in ??. 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 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 [**MonP1**, Chapter III] to study ∃-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 ∃-atomic if one adds enough relations to the vocabulary. This will allow us to use the whole artillery of results from [**MonP1**, Chapter III] to 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. $(\mathbb{Z}; \mathsf{Adj})$ is ∃-atomic,[‡] as the automorphism type of a tuple is determined the by order of its elements and the distance between the elements. $(\mathbb{Z}+\mathbb{Z}+\mathbb{Z}; \mathsf{Adj})$ is ∃-atomic over a finite set of parameters

[†] For the right-to-left direction, prove it for tuples within the structures and use transfinite induction in the rank of the formula.

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

(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 [**MonP1**, 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 is $\Delta^{\mathtt{in}}_\alpha$-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 $\varphi_{\omega^\beta}(x,z)$ 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 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 [**MonP1**, 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 representation 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, (Δ_α catgoricity ??), and (Lopez Escobar ??) 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 [**Mon15**], 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 my 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 [**MonP1**, 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 [**MonP1**, Chapter III], but we do not rely on them, so the reader who did not read [**MonP1**, Chapter III] can follow them without problem. Next, we will introduce the technique of Morleyization, which will allow us to lift the results from [**MonP1**, 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 [**Mon15**])). *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 $\neg\psi(\bar{\mathsf{c}}) \in S$.

Once we have S satisfying (A)-(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 C-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 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, i.e. the latter part of (E), 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 $\theta(\bar{x}) \in \Phi$ 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 $\bar{a}\bar{b}$. □

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

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 [**Mon15**]. 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 $\langle \bar{a}, \bar{b} \rangle \in P$. 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 [**MonP1**, 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 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. Suppose first 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 directly the definitions of the new relations, namely $\forall \bar{x}(\mathsf{R}_\psi(\bar{x}) \leftrightarrow \psi(\bar{x}))$ for $\psi \in \Psi$, has the 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

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

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

For our first application of Morelyization, let us consider [**MonP1**, 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 [**MonP1**, 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 $\Sigma^{\mathtt{in}}_\alpha$ are $\Sigma^{\mathtt{in}}_1$ over Ψ. Let $\breve{\mathcal{A}}$ be the Morley extension 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 [**MonP1**, 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 extension with respect to Ψ. Then $\breve{\mathcal{B}} \models M_\Psi$, and hence $\mathcal{B} \models \breve{\varphi}$, 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 $\breve{\varphi} \wedge M_\Psi$ is a $\Pi^{\mathtt{in}}_2$ Scott sentence for the Morley extension of $\mathcal{A}$. By [**MonP1**, Theorem III.34], every automorphism orbit in $\breve{\mathcal{A}}$ is definable by a $\exists$-$\breve{\tau}$-formula. Replacing R_ψ for ψ within

each of these definitions, we get equivalent formulas in $\breve{\mathcal{A}}$, 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 [**MonP1**, Chapter III]. For instance, we can lift the $\forall$-type-omitting theorem we proved in [**MonP1**, Lemma III.31] and make it a $\Pi^{\mathtt{in}}_{\alpha}$ type-omitting theorem. Let us first recall that [**MonP1**, 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 arithmetic, 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 $\breve{\tau}$. Then T is equivalent to a $\Pi^{\mathtt{in}}_{1}$ $\breve{\tau}$-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 [**MonP1**, Lemma III.31] to get a $\breve{\tau}$-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 $\Pi^{\mathtt{in}}_{2}$ $\breve{\tau}$-sentence $\breve{\varphi}$, each p_i is a $\Pi^{\mathtt{in}}_{1}$ $\breve{\tau}$-type (and in particular a $\forall$-type), and no p_i is $\exists$-supported over $\breve{\varphi} \wedge M_{\Psi}$, as otherwise they would be $\Sigma^{\mathtt{in}}_{\alpha}$-supported over φ. We can then apply [**MonP1**, Lemma III.31] to get a $\breve{\tau}$-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 [**Mon15**], 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 Morelyization on [**MonP1**, 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 these formulas are the simplest possible. That is when 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

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

For the base case, we let $(\mathcal{A},\bar{a}) \leq_0 (\mathcal{B},\bar{b})$ if $\bar{a}$ and $\bar{b} \restriction |\bar{a}|$ satisfy the same quantifier-free $\tau_{|\bar{a}|}$-formulas, or equivalently, if $D_{\mathcal{A}}(\bar{a}) \subseteq D_{\mathcal{B}}(\bar{b})$.¶

¶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. ?? Refer to "notation from Part I"-section

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.

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 <u>even</u>-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 <u>odd</u>-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 $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b})$.

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. We will make a parenthetic remark in the paragraph below about how one deals with this when we need it for the induction hypothesis.

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 $\bar{d}_i$ to any string $\bar{d}'_i$ that is long enough so that all the symbols in φ_i are in the finite approximation $\tau_{|\bar{d}'_i|}$ to the vocabulary τ. We would then get $\bar{c}'$ with $(\mathcal{A}, \bar{a}, \bar{c}') \geq_0 (\mathcal{B}, \bar{b}, \bar{d}'_i)$, 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 contrapositive. 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 $\bar{c} \in A^{<\mathbb{N}}$, 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

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

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

We will see later in Chapter **??** 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 $\mathcal{A}$ from $\mathcal{B}$ is $\mathbf{\Sigma}^0_\alpha$ hard.

There are other non-equivalent definitions of back-and-forth relations. 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 get messy after that. The most comprehensive analysis of back-and-forth scattered linear orders to date can be found in Alvir and Rossegger's paper [**AR**].

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 $n \leq |\mathcal{B}|$, if one choses 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*

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

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

Proof. The proof is a straightforward transfinite induction. □

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

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*

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

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 right-to-left 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 that this is also the case for the last intervals $(\langle\gamma_{k,\ell_k}, b_k\rangle, +\infty)_{\omega^\alpha\mathcal{B}}$ and

**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.

$(\langle \gamma_{k,\ell_k}, a_k\rangle, +\infty)_{\omega^\alpha \mathcal{A}}$. 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 left-to-right 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 element from $\mathcal{B}$, and consider the tuple

$$\langle 1, b_0\rangle, \langle 1, b_1\rangle, ..., \langle 1, b_{|A|}\rangle$$

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 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.[††]

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 best possible in the sense that there is no $\Pi^{\mathtt{in}}_{2\alpha}$ sentence expressing the same thing. See hint in footnote.[‡‡]

[††] 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.

[‡‡] 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.

(b) Write a $\Pi^{\mathtt{in}}_{2\alpha+1}$ sentence that is true exactly of the well-orderings less than or equal 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 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?

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

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 II.7 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.53 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.49. Alvir and Harrison-Trainor [**AGHTT**] showed that the Wadge degrees of the set of ω-presentations of a structure can only be $\mathbf{\Sigma}^0_{\alpha}$, $\mathbf{\Pi}^0_{\alpha}$, or d-$\mathbf{\Sigma}^0_{\alpha}$. This statement will make sense after Chapter ??, when we study the space of ω-presentations.

Alvir, Greenberg, Harrison-Trainor, and Turetsky [**AGHTT**] actually do a deep analysis of the landscape of Scott sentence complexities.

Definition II.50. 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.

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 $\Sigma^{\mathtt{in}}_2$ Scott sentence.

Finite structures have d-$\Sigma^{\mathtt{in}}_1$ Scott sentences, but we will not worry about them. Thus, the simplest Scott-sentence complexity of an infinite structure is $\Pi^{\mathtt{in}}_2$, which is the Scott-sentence complexity of $\exists$-atomic structures [**MonP1**, Theorem III.34]. We can also rule out $\Sigma^{\mathtt{in}}_{\alpha}$ and d-$\Sigma^{\mathtt{in}}_{\alpha}$ for limit ordinals α as possible Scott-sentence complexities: This is because if a structure satisfies a $\Sigma^{\mathtt{in}}_{\alpha}$ formula, it must satisfy one of its disjuncts which is $\Sigma^{\mathtt{in}}_{\beta}$ for some $\beta < \alpha$. Therefore, if a structure has a $\Sigma^{\mathtt{in}}_{\alpha}$ Scott sentence, it has a simpler one. Also, if a structure has a d-$\Sigma^{\mathtt{in}}_{\alpha}$ Scott sentence, the $\Sigma^{\mathtt{in}}_{\alpha}$-conjunct could be simplified to $\Sigma^{\mathtt{in}}_{\beta}$ for

Scott sentence	parametrized Scott rank	parameterless Scott rank	complexity of parameters
$\Sigma^{\mathtt{in}}_{\alpha+2}$	α	$\alpha+2$	$\Pi^{\mathtt{in}}_{\alpha+1}$
$d\text{-}\Sigma^{\mathtt{in}}_{\alpha+1}$	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 lines 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.

some $\beta < \alpha$, and hence the structure would have a $\Pi^{\mathtt{in}}_{\alpha}$ 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 $\Sigma^{\mathtt{in}}_{\alpha+2}$ Scott sentence and no $\Sigma^{\mathtt{in}}_{\beta+2}$ Scott sentence for any $\beta < \alpha$. This does not say anything about whether $\mathcal{A}$ has a $\Pi^{\mathtt{in}}_{\alpha+1}$ or $\Pi^{\mathtt{in}}_{\alpha+2}$ Scott sentence, which, as we will see, will depend on the complexity of the parameters over which $\mathcal{A}$ is $\Sigma^{\mathtt{in}}_{\alpha}$-atomic. Also, when α is a limit ordinal, this does not rule out $\mathcal{A}$ having a $\Sigma^{\mathtt{in}}_{\alpha+1}$ Scott sentence and still have Scott rank α.

Let $\bar{p} \in A^{<\mathbb{N}}$ be such that $(\mathcal{A}, \bar{p})$ is $\Sigma^{\mathtt{in}}_{\alpha}$-atomic. The first observation is that the orbit of these parameters must be $\Pi^{\mathtt{in}}_{\alpha+1}$-definable: We know that $(\mathcal{A}, \bar{p})$ has a $\Pi^{\mathtt{in}}_{\alpha+1}$ Scott sentence $\varphi(\bar{p})$ (Proposition II.26), and hence $\varphi(\bar{x})$ is a $\Pi^{\mathtt{in}}_{\alpha+1}$ 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 $\Sigma^{\mathtt{in}}_{\alpha+1}$ definable.* In this case we know from Proposition II.26 that $\mathcal{A}$ does not have a $\Pi^{\mathtt{in}}_{\alpha+2}$ Scott sentence, and hence its Scott-sentence complexity must be $\Sigma^{\mathtt{in}}_{\alpha+2}$. Here are a couple of examples.

Exercise II.51. (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 have a d-$\Sigma_{\alpha+1}^{\mathtt{in}}$ Scott sentence. We know from Proposition II.26 that $\mathcal{A}$ does not have a $\Pi_{\alpha+1}^{\mathtt{in}}$ Scott sentence. This implies that the Scott complexity must be either d-$\Sigma_{\alpha+1}^{\mathtt{in}}$ or $\Sigma_{\alpha+1}^{\mathtt{in}}$. 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-$\Sigma_{\alpha+1}^{\mathtt{in}}$.

Exercise II.52. Show that $\omega^\alpha+\omega^\alpha$ has Scott sentence complexity d-$\Sigma_{2\alpha+1}^{\mathtt{in}}$. See hint in footnote.[‡]

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

Theorem II.53 (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 $\Sigma_{\alpha+2}^{\mathtt{in}}$ Scott sentence and a $\Pi_{\alpha+2}^{\mathtt{in}}$ Scott sentence.*
2. *$\mathcal{A}$ is $\Sigma_\alpha^{\mathtt{in}}$-atomic over a tuple of parameters whose automorphism orbit is $\Sigma_{\alpha+1}^{\mathtt{in}}$-definable.*
3. *$\mathcal{A}$ has a d-$\Sigma_{\alpha+1}^{\mathtt{in}}$ Scott sentence.*

Proof. (Alvir [**AKM**]) Start by assuming (1) and let us prove (2). Since $\mathcal{A}$ has a $\Sigma_{\alpha+2}^{\mathtt{in}}$ Scott sentence, $\mathcal{A}$ must be $\Sigma_\alpha^{\mathtt{in}}$-atomic over some tuple of parameters. Since $\mathcal{A}$ also has a $\Pi_{\alpha+2}^{\mathtt{in}}$ Scott sentence, the automorphism orbit of any other tuple is definable by a $\Sigma_{\alpha+1}^{\mathtt{in}}$ 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 $\Sigma_{\alpha+1}^{\mathtt{in}}$ formula defining the automorphism orbit of $\bar{p}$. The following formula is a $\Sigma_{\alpha+1}^{\mathtt{in}}\wedge\Pi_{\alpha+1}^{\mathtt{in}}$ 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) $\Rightarrow$ (1) is straightforward. □

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

[‡] Use Exercise II.42 on α-limits.

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.54. *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.*

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 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 $\bar{b}$ preserving $\bar{p}$. 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 $\mathcal{A} \models \gamma(\bar{p}', \bar{b}') \wedge \delta(\bar{p}', \bar{b}')$. 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.55. *The original proof of Theorem II.53 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**] analyses which Scott-sentence complexity are possible by studying the Borel complexity of the sets of ω-presentations, as we will see in Chapter ??. 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.53 given above is quite recent and due to Rachel Alvir [**AKM**]. They also prove a computability infinitary 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.56 (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* $\psi(\bar{x}) \in \Psi$ *and* $\bar{a} \in A^{|\bar{x}|}$, $\mathcal{A} \models \psi(\bar{a}) \iff \mathcal{B} \models \psi(\bar{a})$.

Proof. The proof is by induction on formulas the same way as in the standard proof of Vaught's criterion, the only difference being 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.57 (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 Lemma II.63 and in Theorem (Δ_α catgoricity??) 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 non-equivalente definitions. The closest definition to ours is form Ash and Knight [**AK00**, Section 6.7], who use the same back-and-forth relations we use, but a slightly different definition. 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.58. *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 $\bar{b} \in A^{|\bar{a}|}$,

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

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 it 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 $\{\bar{b} \in A^{|\bar{a}|} : \bar{b} \geq_\alpha \bar{a}\}$ 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 $\{\bar{b} \in A^{|\bar{a}|} : \bar{b} \geq_\alpha \bar{a}\}$, 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 $\leq_b$ coincides with the automorphism relation on $\mathcal{A}$, and if $\beta < \mathrm{SR}_{\text{p-less}}(\mathcal{A})$ then $\leq_b$ 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.59. 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 α-free tuple.

Definition II.60. (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

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

Lemma II.61. *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 left-to-right 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 that 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 $\bar{a} \leq_\alpha \bar{a}'$. ∎

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

Theorem II.62 (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.

Lemma II.63. *For each vocabulary τ and ordinal α, there is a computably infinitary 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

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

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 $\bigvee_{\sigma \in 2^{\ell_{|\bar{x}|}}} D(\bar{x}) = \sigma \wedge D(\bar{y}) = \sigma$. □

CHAPTER III

Computably Infinitary Languages

To study the computational properties of structures syntactically the appropriate language is the computably infinitary language, as first noticed by Chris Ash in [**Ash86b**]. It is 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 computably infinitary language in [**MonP1**]. The main result connecting these formulas with computational complexity is the Ash-Knight-Manasse-Slaman–Chisholm Theorem [**MonP1**, 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 **??** 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 $\bigvee\!\!\!\bigvee$, $\bigwedge\!\!\!\bigwedge$, $\forall \mathtt{x}$, 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 $\sigma \in$ is not a leaf, then $\ell(\sigma)$ can be one of: $\bigvee\!\!\!\bigvee$, $\bigwedge\!\!\!\bigwedge$, $\forall \mathtt{x}$, 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.

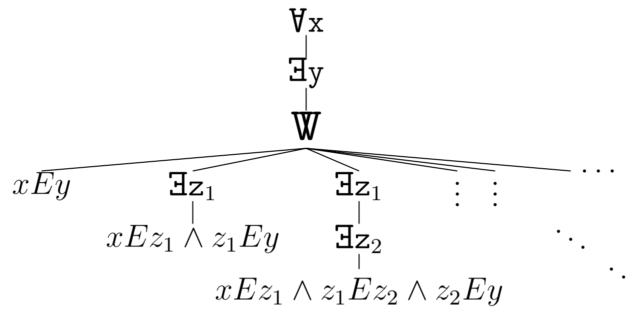

FIGURE III.1. Tree for infinitary sentence that says that a graph is connected.

(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 of 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 $b \in A$, $v(\tau, p \cup \{x \mapsto b\}) = \mathtt{True}$.
- 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 $b \in A$, $v(\tau, p \cup \{x \mapsto b\}) = \mathtt{True}$.

- 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. Finally, 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.30). Let us look, for instance, at the formula $\psi_\alpha(x)$ from Section II.1.3 that expreses that the well-founded rank of x in a partial ordering is at most α. Recall that we defined

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

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 α in β, gives us an index for a computable tree-representation of the formula $\psi_\alpha(x)$. This is a direct application of effective transfinite recursion (Theorem I.30): 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 $\forall y < x \bigvee_{\gamma<\alpha} \psi_\gamma(y)$.

The sentences from Lemma II.63 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 computably 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 *computably infinitary* Σ_α *formulas*, which we denote by $\Sigma^{\mathtt{c}}_\alpha$, are the computably infinitary formulas for which we can computably witness that they are $\Sigma^{\mathtt{in}}_\alpha$. That is, given a computable ordinal α, a computably infinitary formula is $\Sigma^{\mathtt{c}}_\alpha$ if there is a computable *ranking function* that assigns to each node in the tree representation a symbol of the form $\Sigma^{\mathtt{c}}_\beta$ or $\Pi^{\mathtt{c}}_\beta$ for $\beta \leq \alpha$ following the obvious rules: Formulas that start with $\forall$ and $\bigdoublewedge$ are assigned $\Pi^{\mathtt{c}}$'s, and formulas that start with $\exists$ and $\bigdoublevee$ are assigned $\Sigma^{\mathtt{c}}$'s; every time a node switches with respect to its parent node from either $\forall$ or $\bigdoublewedge$ to either $\exists$ and $\bigdoublevee$ 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 computably 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 $\bigdoublevee$ on one side, or $\forall$ or $\bigdoublewedge$ on the other. 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 computably 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 [**MonP1**, 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 $\varphi^{\Pi^{\mathtt{c}}_0}_{i,j}(\bar{x}) = \varphi^{\Sigma^{\mathtt{c}}_0}_{i,j}(\bar{x}) = \varphi^{\mathrm{qf}}_{i,j}(\bar{x})$.

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

Given a computable ordinal α, we define $\varphi_{e,j}^{\Sigma^{\mathsf{c}}_\alpha}(x_1,...,x_j)$, the e-th $\Sigma^{\mathsf{c}}_\alpha$ formula with j free variables as follows:

$$\varphi_{e,j}^{\Sigma^{\mathsf{c}}_\alpha}(x_1,...,x_j) \quad \text{is} \quad \bigvee\!\!\!\!\bigvee_{\substack{\langle i,k,\beta\rangle\in W_e \\ \beta\in\alpha}} \exists y_1,...,y_k \;\; \varphi_{i,j+k}^{\Pi^{\mathsf{c}}_\beta}(\bar{x},\bar{y}),$$

and define

$$\varphi_{e,j}^{\Pi^{\mathsf{c}}_\alpha}(x_1,...,x_j) \quad \text{as} \quad \bigwedge\!\!\!\!\bigwedge_{\substack{\langle i,k,\beta\rangle\in W_e \\ \beta\in\alpha}} \forall y_1,...,y_k \;\; \varphi_{i,j+k}^{\Pi^{\mathsf{c}}_\beta}(\bar{x},\bar{y}).$$ †

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 $\varphi_{e,j}^{\Sigma^{\mathsf{c}}_\beta}$ and $\varphi_{e,j}^{\Pi^{\mathsf{c}}_\beta}$, and computable ranking functions. Conversely, again by effective transfinite recursion, given a tree representation with a computable ranking function for a $\Sigma^{\mathsf{c}}_\alpha$ formula, we can effectively find an index for it.

† When we write Π^{c}_β 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 15). 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 left-to-right direction is straightforward. For the right-to-left direction, prove the contrapositive as follows: If $\forall n\ \exists F_n \neg\theta(F_n, n)$, then $F = \bigoplus_n F_n$ witnesses that $\exists F\ \forall n\ \neg\theta(F^{[n]}, n)$.

Recall from Definition I.28 that *Kleene's* $\mathcal{O}_{wo}$ is the set of indices for computable well-orderings, using the indexing from Lemma I.27 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 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 $\mathbb{WO}$ 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

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

which is equivalent to

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

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 15 or [**MonP1**, Definition V.17].)

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*

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

---

*See Definition I.20.

†Recall from page 14 that a *bounded formula* of arithmetic is one where all quantifiers are of the form $\forall x < a$ or $\exists y < b$.

(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

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

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 Π^1_1-complete.*

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 $f(n) \in \mathcal{O}_{\text{wo}}$. □

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 branch 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 could climb it straight up in a step-by-step way without ever getting stuck. This would produce an S-computable path. □

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.27).

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.1. 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 $e \in \mathcal{O} \setminus B$. 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 $\mathcal{L} \in \mathfrak{A}$. □

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 $e \in \mathcal{O} \setminus B$. 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 sets 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 of them are. 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.19. 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.19, there is a $\subsetneq$-preserving map $f \colon S \to T$. Define $g \colon S \to S*T$ by $g(\sigma) = \sigma * (f(\sigma) \upharpoonright |\sigma|)$ 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 in page 28. 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.26 that if $\mathcal{L}$ is well-ordered, then $\text{rk}(T_{\mathcal{L}}) \cong \mathcal{L}$, and if $\mathcal{L}$ is not well-ordered, then $\text{rk}(T_{\mathcal{L}}) = \infty$. If $e \in A$, then $\text{rk}(T_{L_{f(e)}}) = \infty$, and hence $\text{rk}(S_e) = \text{rk}(T_{\mathcal{L}_e}) \cong \mathcal{L}_e$. Recall from Exercise I.25 that $\text{rk}(T) < (T; \leq_{\text{KB}})$ 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 $\mathcal{L}_e \preccurlyeq (S_e; \leq_{\text{KB}})$. 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 $\mathcal{L} \in \mathfrak{A} \iff \exists X \in \mathbb{N}^{\mathbb{N}}\ \mathcal{L} \oplus X \in [S]$ for all $\mathcal{L} \in 2^{\mathbb{N}^2}$. 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 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

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

$\mathcal{L} \in \mathfrak{A}$. Thus $(\mathcal{L} \upharpoonright n) \oplus (Y \upharpoonright n) \in S$ 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.26). 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 $\mathcal{O}_{\text{wo}\leq\alpha} = \{e : \mathcal{L}_e \preccurlyeq \alpha\}$, 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 $A \subseteq \mathcal{O}_{\text{wo}}$ is Σ^1_1, then $A \subseteq \mathcal{O}_{\text{wo}\leq\alpha}$ for some $\alpha < \omega_1^{CK}$. Notice that the sets $\mathcal{O}_{\text{wo}\leq\alpha}$ are Δ^1_1 (that is, both Π^1_1 and Σ^1_1): The definition we gave is Σ^1_1, and also $e \in \mathcal{O}_{\text{wo}\leq\alpha} \iff e \in \mathcal{O}_{\text{wo}}\ \&\ \alpha + 1 \not\preccurlyeq \mathcal{L}_e$, 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 $\mathcal{O}_{\text{wo}}$. By Σ^1_1 bounding, since $f(A)$ is a Σ^1_1 subset of $\mathcal{O}_{\text{wo}}$, there is an $\alpha < \mathcal{O}_{\text{wo}}$ such that $\mathcal{L}_{f(e)} \preccurlyeq \alpha$ for all $e \in A$. Let $C = \{e \in \mathbb{N} : \mathcal{L}_{f(e)} \preccurlyeq \alpha\} = f^{-1}(\mathcal{O}_{\text{wo}\leq\alpha})$. It is clear that $A \subseteq C \subseteq B^c$. Since $\mathcal{O}_{\text{wo}\leq\alpha}$ is Δ^1_1, so is C. ☐

Corollary IV.17. *A set $C \subseteq \mathbb{N}$ is Δ^1_1 if and only if $C \leq_m \mathcal{O}_{\text{wo}\leq\alpha}$ for some $\alpha < \omega_1^{CK}$.*

Proof. The right-to-left direction follows from the observation above that the sets $\mathcal{O}_{\text{wo}\leq\alpha}$ are Δ^1_1. For the left-to-right 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.18 (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 $\mathcal{L}_e$ has a computable presentation. ☐

Corollary IV.19. *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. ☐

[§]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.

IV.2. Gandy basis theorem

This is another extremely useful theorem. It is often the case that we want to find reals with a certain property and we know that such reals exist, but we want to find such reals which are not too complex. For instance, if the property is Π^0_1, by the low basis theorem of Jockusch and Soare [**JS72**], 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 Σ^1_1 definition.

Lemma IV.20. *The class* $\mathfrak{D} = \{Y \subseteq \mathbb{N} : Y \textit{ is } \Delta^1_1\}$ *is* Π^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 $S_n \not\preccurlyeq T_n$ if $n \in Y$, and $T_n \not\preccurlyeq S_n$ if $n \notin Y$, where $T \preccurlyeq S$ means that there is a $\subsetneq$-preserving embedding 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 embeddings is Σ^1_1. Thus, it will follow from the claim that $\mathfrak{D}$ is Δ^1_1.

To show the claim, let us first recall that there is a $\subsetneq$-preserving embedding from T to S if and only if $\mathrm{rk}(T) \leq \mathrm{rk}(S)$ by Lemma I.19. So if Y satisfies the right-hand side, then for every $n \in \mathbb{N}$, $n \in Y$ if and only if $S_n \not\preccurlyeq T_n$, which happens if and only if $T_n \preccurlyeq S_n$. 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 $n \in \mathbb{N}$,

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

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

Proof. Let $\mathfrak{S} \subseteq \mathbb{N}^{\mathbb{N}}$ 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 $X \in \mathbb{N}^{\mathbb{N}}$,

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

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

Not only does $\mathcal{O}_{\text{wo}}$ 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.22. 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.23. A set $X \subseteq \mathbb{N}$ is *low for* ω_1 if $\omega_1^X = \omega_1^{CK}$.

Lemma IV.24. *For every $X \subseteq \mathbb{N}$, $\omega_1^X > \omega_1^{CK}$ if and only if $\mathcal{O}_{\text{wo}}$ is Δ^1_1 relative to X.*

Proof. For the left-to-right 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 right-to-left 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 30, is isomorphic to ω_1^{CK}. We claim that $\mathcal{L}$ is Σ^1_1 in X. The domain is $\{\langle e, n\rangle : e \in \mathcal{O}_{\text{wo}}, n \in L_e\}$, and 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 $n_0 <_{\mathcal{L}_e} n_1$. Notice that the domain of $\mathcal{L}$ is Σ^1_1 in X, and the ordering is computable. We thus have a Σ^1_1-in-X ω-presentation of $\mathcal{L}$. By Spector's theorem (Theorem IV.18), $\sum_{e \in \mathcal{O}_{\text{wo}}} \mathcal{L}_e$ 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.25 (Gandy basis theorem). *Every non-empty Σ^1_1 set $\mathfrak{S}$ of reals has a member that is computable in $\mathcal{O}_{\text{wo}}$ 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.20, 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.21). 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.26. 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.3. An application of the Gandy basis theorem

Let us build some interesting-looking structures.

Theorem IV.27. *If a Π^1_1 set of indices of computably 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 $\psi(X, x)$ 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 computably infinitary sentences, then the set of ω-presentations $\mathcal{A}$ such that $\forall e\ (e \in S \rightarrow \mathcal{A} \models \varphi_e)$ is Σ^1_1. It is also non-empty, as we are assuming that this set of sentences has a model. By the Gandy basis theorem (Theorem IV.25), 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.28. *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.63. This is a Π^1_1 set of computably 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 confortable with this assumption may take a fragment of ZFC instead.

Corollary IV.29. *(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 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.57), 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 all computable well-orderings. 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}. □

This model is often quite useful in proofs because the fact that $\mathrm{ON}^{\mathcal{M}}$ is ill-founded in reality but well-ordered according to $\mathcal{M}$, which is a model of ZFC satisfying all true Π_1^1 sentences, results in some interesting consequences.

CHAPTER V

Hyperarithmetic Sets

The hyperarithmetic hierarchy extends the arithmetical hierarchy through the computable ordinals, given 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 computably infinitary formula, that is, if there is a computably 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 computably 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 computably 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 contains no variables other than w, neither free nor quantified. For the last step, for

[*]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.

each $n \in \mathbb{N}$, consider the 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 $\bot$ depending on whether the equality is true or false.

We are now left with computably infinitary formulas φ_n whose only symbols are $\bigvee\!\!\!\bigvee$, $\bigwedge\!\!\!\bigwedge$, $\bot$, 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 $\bigvee\!\!\!\bigvee$, $\bigwedge\!\!\!\bigwedge$, $\bot$, 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 $\Sigma^{\mathtt{c}}_\alpha$ formula of arithmetic if and only if there is a computable sequence of $\Sigma^{\mathtt{c}}_\alpha$ 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 $\Sigma^{\mathtt{c}}_\alpha$ 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 14.

Lemma V.6. *Let $\mathcal{A}$ be a computable ω-presentation of a τ-structure and $\varphi(\bar{x})$ a $\Sigma^{\mathtt{c}}_\alpha$ τ-formula. 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 $\Sigma^{\mathtt{c}}_1$ 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 $\Sigma^{\mathtt{c}}_1$ $\mathcal{N}$-formula about $(\mathbb{N}; +, \times, 0, 1, <)$ and a $\Pi^{\mathtt{c}}_1$ $\mathcal{N}$-formula about $(\mathbb{N}; +, \times, 0, 1, <)$. If we replace each atomic τ-formula in ψ^τ_i by its equivalent $\Sigma^{\mathtt{c}}_1$ $\mathcal{N}$-formula, we get that ψ^τ_i is equivalent to a $\Sigma^{\mathtt{c}}_1$ $\mathcal{N}$-formula $\psi^{\mathcal{N}}_i$. If we replace each atomic τ-formula in θ^τ_i by its equivalent $\Pi^{\mathtt{c}}_1$ $\mathcal{N}$-formula, we get that θ^τ_i is equivalent to a $\Sigma^{\mathtt{c}}_1$ $\mathcal{N}$-formula $\theta^{\mathcal{N}}_i$. We then get that $\varphi^\tau(\bar{x})$ is itself equivalent to a $\Sigma^{\mathtt{c}}_1$

$\mathcal{N}$-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, the set of indices of computable ω-presentations satisfying φ is Σ^0_{α}.*

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

$$\bigvee\!\!\!\bigvee_{e\in\mathbb{N}} (x = e)\ \wedge\ \varphi^{\mathcal{N},e}.$$ □

Lemma V.8. *Given an ordinal α, the set of indices of computable well-ordering 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 is $\Sigma^0_{2\gamma}$.

Proof. This follows immediately from the previous lemma using the computably infinitary formulas we defined in Lemmas II.4 and II.5. □

Theorem V.9. *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 $\mathcal{O}_{\mathrm{wo}\leq\alpha} = \{e : \mathcal{L}_e \prec \alpha\}$ and that $\{\mathcal{L}_e : e \in \mathbb{N}\}$ is a computable enumeration of the computable linear orderings defined in Lemma I.27.

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.17.

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 disjunction 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, $\bigwedge\!\!\!\bigwedge_{e\in I} \varphi_e^{\Sigma^{\mathtt{c}}_\alpha}$ is equivalent to $\bigwedge\!\!\!\bigwedge_{e\in\mathbb{N}}(\psi_e \rightarrow \varphi_e^{\Sigma^{\mathtt{c}}_\alpha})$, which is $\Pi^{\mathtt{c}}_\alpha$. □

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 14). 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.11. Given a computable linear ordering $\mathcal{L}$, a *jump hierarchy* on $\mathcal{L}$ is a set $H \subseteq L \times \mathbb{N}$ such that

$$\text{(JH)} \qquad (\forall a \in \mathcal{L}) \quad H^{[a]} = (H^{[<_\mathcal{L} a]})',$$

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 $0_{\mathcal{L}}, 1_{\mathcal{L}}, 2_{\mathcal{L}}, \ldots$ 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.12. If $\mathcal{L}$ is a computable well-ordering, the set $0^{(\mathcal{L})}$ is Δ^1_1 (and hence hyperarithmetic). This is because, for $k \in \mathcal{L} \times \mathbb{N}$,

$$\begin{aligned} k \in 0^{(\mathcal{L})} &\iff (\exists H \subseteq L \times \mathbb{N}) \ \ H \text{ is a jump hierarchy on } \mathcal{L} \text{ and } k \in H \\ &\iff (\forall H \subseteq L \times \mathbb{N}) \ \ \text{if } H \text{ is a jump hierarchy on } \mathcal{L}, \text{ then } k \in H, \end{aligned}$$

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.13. 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.15 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.14. *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 to use effective transfinite recursion (Theorem I.30) 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 $H^{[\gamma]}$. We would then get 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

$$x \in X' \iff \bigvee\!\!\!\bigvee_{\substack{\sigma\in 2^{<\mathbb{N}}\\ \Phi^{\sigma}_x(x)\downarrow}} \sigma \subseteq X$$

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:

$$\psi(x) \iff \bigvee\!\!\!\bigvee_{\substack{\sigma\in 2^{<\mathbb{N}}\\ \Phi^{\sigma}_x(x)\downarrow}} \left(\bigwedge_{\substack{i<|\sigma|\\ \sigma(i)=1}} \theta^{\Sigma}(i) \wedge \bigwedge_{\substack{i<|\sigma|\\ \sigma(i)=0}} \neg\theta^{\Pi}(i) \right).$$

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

$$H^{[<\gamma]} = \bigoplus_{\beta\in\mathcal{L}\restriction\gamma} H^{[\beta]},$$

[†]The indices $\alpha+1$ and $1+\alpha$ may seem to be off. Unfortunately, the $0^{(\beta)}$ and the Σ^0_β hierarchies are defined to cause 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.15).

recall that we are using effective transfinite recursion, and we have access to a computable index for $f \upharpoonright \mathcal{L}_{(<\gamma)}$ to get $\Sigma^{\mathtt{c}}_{1+\beta}$ indices for each $H^{[\beta]}$ for $\beta < \gamma$. We can easily transform a $\Sigma^{\mathtt{c}}_{1+\beta}$ index to both a $\Sigma^{\mathtt{c}}_{1+\gamma}$ index and a $\Pi^{\mathtt{c}}_{1+\gamma}$ index for each $\beta < \gamma$, and thus obtain a $\Delta^{\mathtt{c}}_{1+\gamma}$ index for $H^{[<_{\mathcal{L}}\gamma]}$. □

THEOREM V.15. *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.30) to define a computable function $f\colon \mathcal{L} \times \mathbb{N} \to \mathbb{N}$ that assigns to each $\Sigma^{\mathtt{c}}_{1+\alpha}$-propositional sentence $\varphi_e^{\Sigma^{\mathtt{c}}_{1+\alpha}}$ over the empty language (as in Theorem V.3) a number $f(\alpha, e)$ such that

$$\varphi_e^{\Sigma^{\mathtt{c}}_{1+\alpha}} \text{ holds if and only if } \langle \alpha, f(\alpha, e)\rangle \in 0^{(\mathcal{L})}.$$

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 $f \upharpoonright \alpha \times \omega$.

Recall that $\varphi_e^{\Sigma^{\mathtt{c}}_{1+\alpha}}$, the eth $\Sigma^{\mathtt{c}}_{1+\alpha}$-sentence over the empty language, was defined as[¶]

$$\bigvee\!\!\!\!\bigvee_{\substack{\langle m, 1+\gamma\rangle \in W_e \\ 1+\gamma < 1+\alpha}} \varphi_m^{\Pi^{\mathtt{c}}_{1+\gamma}}.$$

Thus, $\varphi_e^{\Sigma^{\mathtt{c}}_{1+\alpha}}$ holds if and only if

$$\exists \gamma, m \in \mathcal{L} \upharpoonright \alpha \times \omega \quad (\langle m, 1+\gamma\rangle \in W_e \text{ and } \langle \gamma, f(\gamma, m)\rangle \in 0^{(\alpha)}).$$

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 $\varphi_e^{\Sigma^{\mathtt{c}}_{1+\alpha}}$ 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.16. *A set $A \subseteq \mathbb{N}$ is hyperarithmetic if and only if $A \leq_T 0^{(\mathcal{L})}$ for some computable well-ordering $\mathcal{L}$.*

[‡]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 $\Pi^{\mathtt{c}}_0$ formulas do not show up in the disjunction.

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.17). However, the isomorphism between two ω-presentations of an ordinal may be quite hard to compute, though fortunately not extraordinarily hard. We will see that $0^{(\alpha)}$ itself can compute such isomorphisms; that is just what we need to compute one jump hierarchy from the other.

Lemma V.17. *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

$$H_\beta^{[<_\beta f(a)]} = \bigoplus_{d \in \beta \restriction f(a)} H_\beta^{[d]} = \bigcup_{d \in \beta \restriction f(a)} \{d\} \times H_\beta^{[d]} = \bigcup_{c \in \alpha \restriction a} \{f(c)\} \times \Phi_{i_c}^{H_\alpha^{[c]}}.$$

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

$$H_\beta^{[<_\beta f(a)]} = \bigcup_{c \in \alpha \restriction a} \{f(c)\} \times \Phi_{i_c}^{H_\alpha^{[c]}} \leq_{T \text{ via } e} \bigcup_{c \in \alpha \restriction a} \{c\} \times H_\alpha^{[c]} = H_\alpha^{[<_a a]}.$$

(Notice that this would not work if we did not assume f was computable, as we will in the more general case.) Once we have e, let i_a be an index for the Turing reduction

$$H_\beta^{[<_\beta f(a)]\prime} \leq_{T \text{ via } i_a} H_\alpha^{[<_a a]\prime}. \qquad \square$$

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:

Lemma V.18. *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 in which case we are 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.)

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 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.19. *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

[‖]Recall that $f \restriction D$ is the partial function obtained by restricting f to the domain D.

above, we have a computable sequence of indices e_a such that $f \upharpoonright \alpha_{(<a)} \leq_{T \text{ via } e_a} H_\alpha^{[a]}$.

As in Lemma V.17, 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 $c \in \alpha \upharpoonright a$. 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.17 that

$$H_\beta^{[<_\beta f(a)]} = \bigcup_{c \in \alpha \upharpoonright a} \{f(c)\} \times \Phi_{i_c}^{H_\alpha^{[c]}}.$$

So, using $f \upharpoonright \alpha_{(<a)}$ and the sequence $\{i_c : c \in \alpha \upharpoonright a\}$, we can compute $H_\beta^{[<_\beta f(a)]}$ from $H_\alpha^{[<_\alpha a]}$. However, we know that $f \upharpoonright \alpha_{(<a)}$ 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 in which case it is and to find b in the latter case.

Case (1): If a is a limit ordinal, we saw in the proof of Lemma V.18 that $f \upharpoonright \alpha_{(<a)} \leq_T H_\alpha^{[<_\alpha a]}$.

Case (2): If not, and $a = b + 1$, then we know that $f \upharpoonright \alpha_{(<b)} \leq_T H_\alpha^{[b]} \leq_T H_\alpha^{[<_\alpha a]}$. 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

$$\bigcup_{c \in \alpha \upharpoonright a} \{f(c)\} \times \Phi_{i_c}^{H_\alpha^{[c]}} \leq_T H_\alpha^{[<_\alpha a]}.$$

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.20. *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 *hyperarithmetic* 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.21. *Every hyperarithmetically infinitary formula is equivalent to a computably infinitary formula.*

The rest of this section is dedicated to proving this theorem.

First, every hyperarithmetically infinitary formula is an X-computably 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 $\Sigma^{\mathtt{c}X}_{\alpha}$ index for some X-computable well-ordering α. Recall that the $\Sigma^{\mathtt{c}X}_{\alpha}$ formula with index e (denoted $\varphi_e^{\Sigma^{\mathtt{c}X}_{\alpha}}$) is the disjunction of the $\exists$-over-$\Pi^{\mathtt{c}X}_{<\alpha}$ formulas with indices in W_e^X. That is,

$$\varphi_e^{\Sigma^{\mathtt{c}X}_{\alpha}} \quad \text{is} \quad \bigvee\!\!\!\!\bigvee_{\substack{\langle i,\beta\rangle \in W_e^X \\ \beta<\alpha}} \exists \bar{y} \;\; \varphi_i^{\Pi^{\mathtt{c}X}_{\beta}}(\bar{x}, \bar{y}).$$

(In Section III.2, we also used a subindex j describing the arity of the formula. We omit it here to simplify the notation.)

Let $\mathcal{H}$ be a hyperarithmetic well-ordering extending α, so that we can think of α as a member of $\mathcal{H}$. We want to show that every $\Sigma^{\mathtt{c}X}_{\beta}$ formula, for $\beta \in \mathcal{H}$, is equivalent to a computable one. There are two obstacles: first, the infinitary disjunctions and conjunctions are not c.e. but X-c.e., and second, the ordinals $\mathcal{H} \restriction \beta$ 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 $\Sigma^{\mathtt{c}}_{\alpha}$ formulas is equivalent to a $\Sigma^{\mathtt{c}}_{\alpha}$ formula. Before that, let us resolve the second issue:

By Spector's theorem (Theorem IV.18), there is a computable well-ordering $\mathcal{K}$ isomorphic to $\mathcal{H}$.

Furthermore, the isomorphism $h\colon \mathcal{H} \to \mathcal{K}$ is hyperarithmetic: Recall from Lemma II.18 that there are computably infinitary formulas

$\psi_\gamma(x)$ for $\gamma \in \mathcal{K}$ such that

$$\mathcal{H} \models \psi_\gamma(\alpha) \iff \mathcal{H} \upharpoonright \alpha \cong \mathcal{K} \upharpoonright \gamma \iff h(\alpha) = \gamma.$$

The formulas ψ_γ are defined computably uniformly in $\gamma \in \mathcal{K}$. Since $\mathcal{H}$ 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 $\mathcal{H}$ to $\mathcal{K}$.

Lemma V.22. *Every $\Sigma^{\mathtt{c}X}_{<\mathcal{H}}$ formula is equivalent to a $\Sigma^{\mathtt{c}Z}_{<\mathcal{K}}$ formula.*[**]

Proof. Using Z-effective transfinite recursion (Theorem I.30), define a Z-computable function $g\colon \mathcal{H} \times \mathbb{N} \to \mathbb{N}$ that, for each $\alpha \in \mathcal{H}$ and $e \in \mathbb{N}$, produces an index $g(\alpha, e)$ for a $\Sigma^{\mathtt{c}Z}_{h(\alpha)}$ formula equivalent to the eth $\Sigma^{\mathtt{c}X}_{\alpha}$ formula. That is, $g(\alpha, e)$ will be defined so that

$$\varphi_{g(\alpha,e)}^{\Sigma^{\mathtt{c}Z}_{h(\alpha)}} \iff \varphi_e^{\Sigma^{\mathtt{c}X}_{\alpha}}.$$

For this, let $g(\alpha, e)$ be an index for the Z-c.e. given by

$$W^Z_{g(\alpha,e)} \quad = \quad \{\langle g(\beta, i), h(\beta)\rangle : \beta \in \mathcal{H} \upharpoonright \alpha \text{ and } \langle i, \beta\rangle \in W^X_e\},$$

so we get that

$$\varphi_{g(\alpha,e)}^{\Sigma^{\mathtt{c}Z}_{h(\alpha)}} \quad \text{is} \quad \bigvee\!\!\!\bigvee_{\substack{\langle i,\beta\rangle \in W^X_e \\ \beta<\alpha}} \exists \bar{y}\ \ \varphi_{g(\beta,i)}^{\Pi^{\mathtt{c}Z}_{h(\beta)}}(\bar{x}, \bar{y}),$$

which by transfinite induction is equivalent to $\varphi_e^{\Sigma^{\mathtt{c}X}_{\alpha}}$. □

The next step is to show that every $\Sigma^{\mathtt{c}Z}_{<\mathcal{K}}$ formula is equivalent to a computable infinitary formula — now knowing that $\mathcal{K}$ is computable. Let π by a computable ordinal that is large enough that Z is Δ^0_π.[††]

Lemma V.23. *For every $\gamma \in \mathcal{K}$, every $\Sigma^{\mathtt{c}Z}_{\gamma}$ formula is equivalent to a $\Sigma^{\mathtt{c}}_{\pi+\gamma}$ formula.*

Proof. We use effective transfinite recursion (Theorem I.30) to define a function $f\colon \mathcal{K} \times \mathbb{N} \to \mathbb{N}$ such that, for $\gamma \in \mathcal{K}$ and $e \in \mathbb{N}$, $f(\gamma, e)$ is an index for a $\Sigma^{\mathtt{c}}_{\pi+\gamma}$ formula equivalent to the eth $\Sigma^{\mathtt{c}Z}_{\gamma}$ formula; that is, $f(\gamma, e)$ will be defined so that

$$\varphi_{f(\gamma,e)}^{\Sigma^{\mathtt{c}}_{\pi+\gamma}} \iff \varphi_e^{\Sigma^{\mathtt{c}Z}_{\gamma}}.$$

[**]Recall that a $\Sigma^{\mathtt{c}X}_{<\mathcal{H}}$ formula is a $\Sigma^{\mathtt{c}X}_{\beta}$ formula for some $\beta \in \mathcal{H}$.

[††]So that W^Z_e is Σ^0_π for all e.

Recall that we defined

$$\varphi_e^{\Sigma^{\mathtt{c}Z}_\gamma} \quad \text{as} \quad \bigvee\!\!\!\bigvee_{\substack{\langle i,\delta\rangle \in W_e^Z \\ \delta \in \mathcal{K}\restriction\gamma}} \exists \bar{y} \;\; \varphi_i^{\Pi^{\mathtt{c}Z}_\delta}.$$

Using the same idea as in Lemma V.10, this is equivalent to

$$(2) \qquad \bigvee\!\!\!\bigvee_{\langle i,\delta\rangle \in \mathbb{N}\times\mathcal{K}\restriction\gamma} \langle i,\delta\rangle \in W_e^Z \;\wedge\; \exists \bar{y} \;\; \varphi_i^{\Pi^{\mathtt{c}Z}_\delta}.$$

Recall that we chose π so that W_e^Z is Σ^0_π. Thus, the formula "$\langle i,\delta\rangle \in W_e^Z$" can be replaced by a $\Sigma^{\mathtt{c}}_\pi$ 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 $\Pi^{\mathtt{c}}_{\pi+\delta}$ formula "$\langle i,\delta\rangle \in W_e^Z \;\wedge\; \varphi_{f(\delta,i)}^{\Pi^{\mathtt{c}}_{\pi+\delta}}$." By the induction hypothesis, the formula (2) is equivalent to

$$\bigvee\!\!\!\bigvee_{\langle i,\delta\rangle \in \mathbb{N}\times\mathcal{K}\restriction\gamma} \exists \bar{y} \; \underbrace{\big(\langle i,\delta\rangle \in W_e^Z \;\wedge\; \varphi_{f(\delta,i)}^{\Pi^{\mathtt{c}}_{\pi+\delta}}\big)}_{\varphi_{\tilde{f}(\delta,i)}^{\Pi^{\mathtt{c}}_{\pi+\delta}}}.$$

Finally, we define $f(\gamma,e)$ as the index for the c.e. set

$$W_{f(\gamma,e)} \quad = \quad \{\langle \tilde{f}(i,\delta), \pi+\delta\rangle : \langle i,\delta\rangle \in \mathbb{N}\times\mathcal{K}\restriction\gamma\}. \qquad \square$$

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, a unique one. The definition was for jump hierarchies over linear orderings in general, but 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 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. We then get that J is Σ^1_1. As we saw in the previous chapter, over every well-ordering there is a jump hierarchy. So, $\mathcal{O}_{\text{wo}}$, the set of indices of computable well-orders, is included in J:

$$\mathcal{O}_{\text{wo}} \subseteq J.$$

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_2^0). 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*

$$\exists X \in HYP\ \psi(X)$$

is equivalent to a Π^1_1 formula too.

Conversely, every Π^1_1 formula $\varphi(Y)$ is equivalent to one of the form $\exists X \in HYP\ (\psi(X))$, 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.27). Notice that for $\varphi_e^{\Sigma^{\mathtt{c}}_a}(x)$ to be an $\mathcal{L}_{\mathtt{c},\omega}$ formula, we need to have $a \in \mathcal{O}_{\text{wo}}$, so 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

$$\exists X \in \text{HYP}\varphi(X) \iff \exists a, e \in \mathbb{N}\ (a \in \mathcal{O}_{\text{wo}}\ \&\ \forall X(\text{if } \varphi_e^{\Sigma^{\mathtt{c}}_a}(x) \text{ defines } X \rightarrow \varphi(X))).$$

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.

For the second part, let $\mathcal{L}$ be a linear ordering, uniformly build out of φ and the parameters in φ, such that φ holds if and only if $\mathcal{L}_\varphi$ is well-ordered. 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

$$\varphi \iff \exists H \in \mathrm{HYP}\ (H \text{ is a jump hierarchy over } \mathcal{L}_\varphi). \qquad \square$$

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. *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, $e_0 \in X'_{i_0}$, 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 exit $i_2 > i_1$ 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

$$\Phi^X_{f(e)}(n)\downarrow \iff \exists k > 0\ e \notin \Gamma_k(X).$$

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}, and thus will allow us to fix indices for all computable ordinals. 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 $\mathcal{O}_{\text{wo}}$. The set S is Π^1_1-over-hyp, and hence by the Spector–Gandy Theorem VI.3, it is Σ^1_1 — it must *overspill*. That is, since $S \supseteq \mathcal{O}_{\text{wo}}$ and S is Σ^1_1, we must have $\mathcal{O}_{\text{wo}} \subsetneq S$. Any element of $S \setminus \mathcal{O}_{\text{wo}}$ 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} the the $\leq_{\mathbb{N}}$-least $b \in A$, $b <_L a_n$ 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 shows 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. Consider the set $\mathbb{K}$ of order types of the intervals $[a, b]$ for $a <_L b$ with $a \sim b$ — they are all computable well-orderings, so we can think of $\mathbb{K}$ as a subset of ω_1^{CK}. If $\mathbb{K}$ has an upper bound $\alpha < \omega_1^{CK}$, then $\{a \in L : a <_L b \wedge a \sim b\}$ is hyperarithmetic, as an element $a <_L b$ would be $\sim$-equivalent to b if and only if $[a, b] \prec \alpha$, which we know is $\Sigma^{\mathtt{c}}_{2\cdot\alpha}$ decidable (Lemma II.5). Let us now show $\mathbb{K}$ is bounded below ω_1^{CK}. If not, there would be infinitely many order types in $\mathbb{K}$. Let $\mathbb{K}_\omega$ be the first ω-many elements of $\mathbb{K}$. We claim that $\mathbb{K}_\omega$ is Σ^1_1: An ω-presentation β is in $\mathbb{K}_\omega$ if and only if there are finitely many initial segments $\beta_1 < \beta_2 < \cdots < \beta_n = \beta$ such that for every $a <_L b$, either $[a, b] \cong \beta_i$ for some $i \leq n$, or $\beta + 1 \preccurlyeq [a, b]$. By the Σ^1_1 bounding theorem IV.13, $\mathbb{K}_\omega$ must be bounded below ω_1^{CK}.[†] Let α be the supremum of $\mathbb{K}$. We now have that the set $\{a <_L b : [a, b] + 1 \preccurlyeq \alpha\}$ is hyperarithmetic and has no least element.

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 $a < b$ 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. □

We call $\omega_1^{CK} + \omega_1^{CK} \cdot \mathbb{Q}$ the *Harrison linear ordering*. By the previous theorems, it is 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 $\omega_1^{CK} + 1$.*

[†]To apply Σ^1_1 bounding, we need to think of $\mathbb{K}_\omega$ as a set of reals. We're thus thinking of $\mathbb{K}_\omega$ as the set of all ω-presentations of ordinals whose order types are are among the first ω many in $\mathbb{K}$.

[‡]Show that the set of $b \in L$ for which there is a hyperarithmetic descending sequence starting at b is Π^1_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 $\Pi^{\mathtt{in}}_{\omega_1^{CK}}$ and not less.

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 relative to X, that is, it has order type $\omega_1^X(1+\mathbb{Q})$ 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.20). Thus, we can build a tree 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. In other words, we consider the tree corresponding to the Π^0_1 set of reals $Y\oplus Z\in\omega^\omega$ such that Z is a witness for the $\Sigma^{1,X}_1$ formula that says that Y is not hyperarithmetic (Corollary IV.6). This tree is not well-founded, as there are lots of Y's which are not hyperarithmetic in X. But it has no path hyperarithmetic in X. Its Kleene–Brouwer ordering of this tree is then ill-founded (Definition **??**). Furthermore, if we look into the proof of Theorem I.24, 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's 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

[§]By a copy of ω_1^{CK} within $\mathcal{H}$, we mean a maximal interval isomorphic to ω_1^{CK}.

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 is now actually isomorphic to $\omega_1^X(1+\mathbb{Q})$. □

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 if $\omega^{\mathcal{H}} \cong \mathcal{H}$. 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 example. We show below that a computable structure has a computable Scott sentence if and only if its Scott rank is computable.

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 $\bar{b}$,*

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

Furthermore, we can find $\varphi_{\bar{a},\alpha}$ uniformly in $\bar{a}$ and α.

¶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^0_2 set.

Proof. For the transfinite recursion to work, we also need to define $\psi_{\bar{a},\alpha}(\bar{x})$ such that

$$\bar{B} \models \psi_{\bar{a},\alpha}(\bar{b}) \iff (\mathcal{A},\bar{a}) \geq_\alpha (\mathcal{B},\bar{b}).$$

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 computably infinitary Scott sentence if and only if its Scott rank is a computable ordinal.*

Proof. For the left-to-right direction, if $\mathcal{A}$ has a computably infinitary Scott sentence, that sentence must be $\Sigma^{\mathtt{c}}_\alpha$ for some $\alpha < \omega_1^{CK}$, and hence $\mathcal{A}$ has Scott rank below ω_1^{CK}.

For the right-to-left 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 that the $\Pi^{\mathtt{c}}_{2\alpha+6}$ formula $\varphi_{\bar{a},\alpha+3}$ 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 would not be of optimal complexity.

Exercise VI.16. (Alvir, Knight, McCoy [**AKM**]) 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.‖

‖Use Morleyization as in Proposition II.26.

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 computably 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, if $\bar{a} E_0 \bar{b}$ 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 [**MonP1**, 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 computably infinitary sentences, they are isomorphic.*

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 computably infinitary sentences, they must be α-back-and-froth 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 computably 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 jus the tree of descending sequences of $\mathcal{H}$ (see page 28); the *Harrison Boolean algebra* is the interval algebra of $\mathcal{H}$; and the *Harrison 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^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 a 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

$$A_0 \subseteq A_1 \subseteq \cdots \subseteq \mathcal{H}$$

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 $b_0^a < b_1^a < b_2^a < \cdots \to a$ that may be finite or infinite, where b_n^a is the $\leq_{\mathbb{N}}$-least element b such that $b_{n-1}^a < b < a$, if such an element exists. If a is a successor ordinal, we will eventually have $b_n^a = a-1$ and stop finding new elements in the sequence. If a is a limit ordinal, this sequence will be infinite, and for every $c < a$, if $b_n^a \geq_{\mathbb{N}} c$ (in the ordering of $\mathbb{N}$), then either $c \leq_{\mathcal{H}} b_{n-1}^a$ or c would be chosen as b_n^a. 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 in $\mathcal{H}$.

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}:

$$T = \{\langle\langle h_0, n_0\rangle, \langle h_1, n_1\rangle, ..., \langle h_k, n_k\rangle\rangle \in (H \times \mathbb{N})^{<\mathbb{N}} : \\ (\forall i \leq k)\ h_i \in A_i \ \ \&\ \ (\forall i < k)\ h_i >_H h_{i+1}\}.$$

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}. So 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 $h(\sigma) \in \mathcal{H} \smallsetminus \omega_1^{CK}$, 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 $h(\sigma), h(\tau) \in \mathcal{H} \smallsetminus \omega_1^{CK}$, 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 $\mathrm{rk}(T_{\sigma\restriction i}) = \infty$. More generally, if we have two tuples of nodes $\bar{\sigma} = \langle\sigma_1, ..., \sigma_\ell\rangle$ and $\bar{\tau} = \langle\tau_1, ..., \tau_\ell\rangle$, we let $\bar{\sigma}\downarrow$ be the tuple which contains all the initial segments of the nodes in $\bar{\sigma}$, i.e.,

**Recall that we use ω_1^{CK} to denote the well-ordered initial segment of $\mathcal{H}$. Recall also that $T_\sigma = \{\gamma : \sigma^\frown\gamma \in T\}$.

all the nodes of the form $\sigma_j \upharpoonright i$ for $j \leq \ell$ and $i \leq |\sigma_j|$. We then have that

$$(\mathcal{T}, \bar{\sigma}) \cong (\mathcal{T}, \bar{\tau}) \iff$$
$$(\mathcal{T}, \bar{\sigma}\downarrow) \equiv_0 (\mathcal{T}, \bar{\tau}\downarrow) \;\; \& \;\; \forall j \leq \ell \; \forall i \leq |\sigma_j| \; (\mathrm{rk}(T_{\sigma_j \upharpoonright i}) = \mathrm{rk}(T_{\tau_j \upharpoonright i})).$$

Thus, to define the automorphism orbit of any tuple, we need to find the rank 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 computably infinitary sentence that is true only for trees of rank $h(\sigma)$. If $h(\sigma) \in \mathcal{H} \smallsetminus \omega_1^{CK}$, 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 $A_n \smallsetminus \omega_1^{CK}$, then $A_n \cap \omega_1^{CK} = A_n \cap (\mathcal{H} \upharpoonright a_n)$ 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 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. 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). 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, and for every $i < |\bar{\sigma}|$, $T_{\sigma_i \smallsetminus \bar{\sigma}} \leq_\alpha T_{\tau_i \smallsetminus \bar{\tau}}$, where $T_{\sigma_i \smallsetminus \bar{\sigma}}$ 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 $T_{\sigma_i \smallsetminus \bar{\sigma}}$ 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 $T_{\sigma_i \smallsetminus \bar{\sigma}}$ independently. The third observation is that $T_{\sigma_i \smallsetminus \bar{\sigma}} = T_{\sigma_i}$, 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 $A_n \cap \omega_1^{CK}$, 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 structure not $\aleph_0$ categorical for the computably 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-Skolmen Theorem II.57 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 sentence 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 computably 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 computably 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 computably 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 computably 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 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].

[††]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 computably 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.

Corollary VI.23. *(Barwise-Kreisel Compactness Theorem) Let S be a Π^1_1 set of indices of computably 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 can be decomposed as the union $\bigcup_{\xi\in\omega_1^{CK}} S_\xi$ of a nested sequence of sets, where S_ξ is Σ^0_ξ for each $\xi \in \omega_1^{CK}$: Given an m-reduction from S to $\mathcal{O}_{\text{wo}}$, let

$$S_\xi = \{e \in \mathbb{N} : \mathcal{L}_{f(e)} \prec \xi\}.$$

The sets S_ξ are $\Sigma^0_{2\log_\omega(\xi)}$ (by Lemma II.5) and in particular Σ^0_ξ.

We showed in Lemma V.10 that a Σ^0_ξ 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 Σ^0_ξ set. Let φ_ξ be a computable infinitary sentence equivalent to $\bigwedge\!\!\!\bigwedge S_\xi$. 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 S, is satisfiable. □

Exercise VI.24. Prove that the theorem above still holds if the function $f\colon \xi \mapsto \varphi_\xi$ is defined by a 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 $\theta(e, d)$.

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 computably 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_a and must overspill. □

Corollary VI.26. *If a computably 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 that say that the model has Scott rank greater than ξ. To write down such a sentence, we need the characterization of Scott rank in terms of the back-and-forth relations given in Theorem II.62, and then the $\Pi^{\mathtt{c}}_{2\alpha}$ definition of the back-and-forth relations given by spelling out their definition (**??**). 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 Morely [**Mor**] and Barwise [**Bar69**]. Then boldface version are due to Morely [**Mor65**] and López-Escobar [**LE66**]. (They show the Hanf number of $\mathcal{L}_{\omega_1,\omega}$ is $\beth_{\omega_1}$ and that of $\mathcal{L}_{\omega_1,\omega}comp$ is $\beth_{\omega_1^{CK}}$. They use the Erdos-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 [**MonP1**, Chapter **??**].

THEOREM VI.27. *If a computably 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 being a linear ordering $\mathcal{L}$ with a first element 0 and a last element ℓ, which we should think of as a well-order. 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 computably 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 $\mathcal{M}$.

[‡‡] 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 $\bar{a}, \bar{b} \in M^n$.

(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 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 [**MonP1**, Lemma III.15] that there is an automorphism mapping c to d. $\square$

Index

Bibliography

[AGHTT] Rachel Alvir, Noam Grenberg, Matthew Harrison-Trainor, and Dan Turetsky. Scott complexity of countable structures. submitted for publication.

[AK00] C.J. Ash and J. Knight. *Computable Structures and the Hyperarithmetical Hierarchy*. Elsevier Science, 2000.

[AKM] R. Alvir, J. Knight, and C. McCoy. Complexity of scott sentences. submitted for publication.

[AR] Rachael Alvir and Dino Rossegger. The complexity of scott sentences of scattered linear orderings. Submitted for publication.

[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.

[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.

[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.

[End11] Herbert B. Enderton. *Computability theory*. Elsevier/Academic Press, Amsterdam, 2011. An introduction to recursion theory.

[Gan60] R. O. Gandy. Proof of Mostowski's conjecture. *Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys.*, 8:571–575, 1960.

[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.

[Har68] J. Harrison. Recursive pseudo-well-orderings. *Transactions of the American Mathematical Society*, 131:526–543, 1968.

[HT18] Matthew Harrison-Trainor. Scott ranks of models of a theory. *Adv. Math.*, 330:109–147, 2018.

[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.

[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.

[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.

[LE66] E. G. K. Lopez-Escobar. On defining well-orderings. *Fund. Math.*, 59:13–21, 1966.

[Mak81] M. Makkai. An example concerning Scott heights. *J. Symbolic Logic*, 46(2):301–318, 1981.

[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.

[Mon10] Antonio Montalbán. Counting the back-and-forth types. *Journal of Logic and Computability*, page doi: 10.1093/logcom/exq048, 2010.

[Mon15] Antonio Montalbán. A robuster Scott rank. *Proc. Amer. Math. Soc.*, 143(12):5427–5436, 2015.

[MonP1] Antonio Montalbán. Computable structure theory: Beyond the arithmetic. In preparation, P1.

[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.

[Nad74] Mark Nadel. Scott sentences and admissible sets. *Ann. Math. Logic*, 7:267–294, 1974.

[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.

[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.

[Sel71] Alan L. Selman. Arithmetical reducibilities. I. *Z. Math. Logik Grundlagen Math.*, 17:335–350, 1971.

[Soa16] Robert I. Soare. *Turing computability.* Theory and Applications of Computability. Springer-Verlag, Berlin, 2016. Theory and applications.

[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.