Università degli Studi di Udine

Ph.D. Course in Computer Science, Mathematics and Physics

Many Problems, Different Frameworks

Classification of Problems in Computable Analysis and Algorithmic Learning Theory

Candidate:
Vittorio Cipriani

Supervisor:
Prof. Alberto Marcone

Co-Supervisor:
Dr. Luca San Mauro

Ph.D. Cycle XXXV

Author’s e-mail: vittorio.cipriani17@gmail.com

Dipartimento di Matematica e Informatica
Università degli Studi di Udine
Via delle Scienze, 206
33100 Udine
Italia

Abstract

In this thesis, we study the complexity of some mathematical problems, in particular those arising in *computable analysis* and *algorithmic learning theory for algebraic structures*. We highlight that our study is not limited to these two areas: indeed, in both cases, the results we obtain are tightly connected to ideas and tools coming from different areas of mathematical logic, including for example descriptive set theory and reverse mathematics.

After giving the necessary preliminaries, the rest of the thesis is divided into two parts one concerning computable analysis and the other algorithmic learning theory for algebraic structures. In the first part we start studying the uniform computational strength of the Cantor-Bendixson theorem in the Weihrauch lattice. This work falls into the program connecting reverse mathematics and computable analysis via the framework of Weihrauch reducibility. We concentrate on problems related to perfect subsets of Polish spaces, studying the perfect set theorem, the Cantor-Bendixson theorem, and various problems arising from them. In the framework of reverse mathematics, these theorems are equivalent respectively to ATR_0 and $\mathbf{\Pi}^1_1\text{−}\mathsf{CA}_0$ and, as far as we know, this is the first systematic study of problems at the level of $\mathbf{\Pi}^1_1\text{−}\mathsf{CA}_0$ in the Weihrauch lattice. We show that the strength of some of the problems we study depends on the topological properties of the Polish space under consideration, while others have the same strength once the space is rich enough. The first part continues considering problems related to (induced) subgraphs. We provide results on the (effective) Wadge complexity of sets of graphs, that are also used to determine the Weihrauch degree of certain decision problems. The decision problems we consider are defined for a fixed graph G, and they take as input a graph H, answering whether G is an (induced) subgraph of H: we also consider the opposite problem (i.e. answering whether H is an induced subgraph of G). Our study in this context is not limited to decision problems, and we also study the Weihrauch degree of problems that, for a fixed graph G and given in input a graph H such that G is an (induced) subgraph H, they output a copy of G in H. In both cases, we highlight differences and analogies between the subgraph and the induced subgraph relation.

In the second part, we introduce algorithmic learning theory, and we present the framework we use to study the learnability of families of algebraic structures: here, given a countable family of pairwise nonisomorphic structures $\mathfrak{K}$, a learner receives larger and larger pieces of an arbitrary copy of a structure in $\mathfrak{K}$ and, at each stage, is required to output a conjecture about the isomorphism type of such a structure. We say that $\mathfrak{K}$ is learnable if there exists a learner which eventually stabilizes to a correct guess. The framework was lacking a method for comparing the complexity of nonlearnable families, and so we propose a solution to this problem using tools coming from invariant descriptive set theory. To do so, we first prove that a family of structures is learnable if and only if its learning domain is continuously reducible to the relation E_0 of eventual agreement on infinite binary sequences and then, replacing E_0 with Borel equivalence relations of higher complexity, we obtain a new hierarchy of learning problems. This leads to the notion of *E-learnability*, where a family of structures $\mathfrak{K}$ is E-learnable, for a Borel equivalence relation E, if there is a continuous reduction from the isomorphism relation associated with $\mathfrak{K}$ to E. It is then natural to ask how the notion of E-learnability interacts with "classical" learning paradigms. We conclude the second part (and the overall thesis) studying the number of mind changes that a learner needs to learn a given family, both from a topological and a combinatorial point of view, and studying how the complexity of a learner (in terms of Turing reducibility) affects the number of mind changes for learning a given family.

Acknowledgments

Everybody knows that is better to write down your ideas as soon as they come up to your mind, before you forget them. I don't usually do this, and I have not done this for this section as well: my apologies to those I'll forget to mention!

A huge thanks is for Alberto: he welcomed me in Udine, and he never loosed his patience despite (unintentionally) I did my "best": so many conversations like **A**:"What you are trying to say?" **V**: *staring into space* **A**: (sighing) "Do you mean this?" **V**:"Exactly!" **A**:"Write it down". He patiently gave me the direction on how a mathematician should be: now, my job to apply them.

We continue with my co-supervisor, Luca. If I am here, doing a Ph.D. it's mainly his merit (or fault). He was my Master degree's supervisor, he advised me to go to Udine, we have two articles together (other beautiful results are coming) and he is the co-supervisor of my thesis. Basically he is constantly guiding me even if he is always in different parts of the world[1]. I'll never forget the Laugavegur trail in Iceland: amazing trail that we successfully finished despite the unmentionable that will always be our shadow.

Now it's Manlio's turn. I do care to mention that he can be considered as another supervisor, maybe the most bully one: I am not overstating when I say that without him, it would have been hard to submit this thesis. It was a pleasure to work with him, being also such a good teacher, finding always the way to make me understand things (and trust me, is not an easy task). I beg you Manlio to be patient in the next conferences if you want food, or we will end up like in Leeds.

Thanks to Arno: he welcomed me in Swansea, despite the Covid era, and I had a great time there. I am glad that I could work him: its contribution to the results in this thesis were fundamental. I take the chance to thank Stuart, the perfect landlord: he basically did everything he could to help me, especially during my Covid days (thanks also to his family, for the lovely Christmas day!). Then thanks to all the Swansea group for making my stay in Swansea so good. A special mention to Leo, for welcoming us in his house and another special thanks to Giovanni. I was lucky to end up in Swansea in the same period he was, and noticed that he survived to the Laugavegur adventure as well! A special thank goes to Kira: I am grateful to have met you, and you are really a special person to me. I wish you the best, you deserve!

Thanks to Nikolay for the work we have done together: I really hope to finally meet you in some conference!

Thanks to Fernando and Seba, I wish you all the best for this adventure, and come on Fernando, you have the "sguardo del poeta" don't beat yourself up in these hard times. Thanks to Andrea: especially for helping me in supporting AS Roma, sometimes is hard. I am just sorry that you are with someone "laziale", but despite this, it was a great time also with Monica; thank you for the lovely cup. Getting back more serious, it was a pleasure to share the office with you guys.

And now the last academic-related thanks. I announced that is not possible to make a ranking of the people I have to thank: but without Davide and Martina, I would probably not have finished this adventure, to you goes the biggest thanks. They started with me this "strazio" as Davide would say. Everybody knows how people around you daily, can make your life easy or not: they made it the best I could ask for. I needed to erase all the stuff I have written for them, otherwise this part would have been unacceptably long. Our friendship will go over this Ph.D. maybe this

[1]legends say that Luca can be in finitely many places in the same time (he is working towards the infinite case).

time, with a proper car, I know.

Thanks to the Cas'aupa's friends: these years were though, and having a group like this was more than helpful. Thanks to Simone for the time spent together (sometimes hard, again due to AS Roma) but Totti and Kety are guiding us. Thanks to Elisa (gnu gnu), to Valerio and Francesca, hope your son Sfugliatiell will not destroy your plants, to Poz, to il comandante to ErBright, Sara and Lucia and to the many I am forgetting: please forgive me, you are a huge group!

Thanks to Letizia and Luca. Thank you for basically letting me be your son (and sorry that this son disappeared in the last period). It was great to spend these years with you, and thanks for helping me in my Padel career. You had really a special role in this adventure!

Thanks to Emanuele, Mirko and Salvatore, the perfect flatmates. Thanks for teaching me Sicilian, and thanks for being always so kind. I am really sorry that we met just a few times in this year: you were able to make the first lockdown funny, and I am really grateful for that.

No, I have not forgotten you: the last Udine-related person (not exactly person) I have met, right now, among the most important ones. Thank Sara & Birillo. I deliberately omitted too personal feelings in this section, and I'll keep doing so with you. Thanks for what you are giving to me. You are one of the strongest person I have met, and I know that you'll reach what you want (with Biri of course). I know it will not be easy, distance is always a problem, but you are worth all the efforts (despite your being Friulana!). Biri, a special thanks is for you. You helped me to write the thesis in the hardest days, with your *etci* and your cuteness.

Thanks to Fabrizio, one of the deepest friendships. You always welcomed me in Ferrara, always having such a great time with you. It's not granted what you have done for me: there are many things that I have to learn from you, not only music related, thank you.

Now Pic. She unreasonably hated me during the firsts' year of high school (maybe just because I was not carrying the books, I don't know), but our friendship makes me proud. It's not easy to make lasting and deep friendships when you are always away: there are so many difficulties, but you are the perfect example that it's possible.

And now my bros. I need to mention them one by one, they deserve: marchetto, tommy-lu-sard, sendro, ciop, seres, aaron, cardarè, gianlù, steichoc. Being away from you, is probably the thing I have hated the most of this period. After my first year of Master degree when I first spend time away from you, I understood that you are really a second family. I really miss you, and thanks for making everything "as-nothing-has-changed" when I come back home. A special mention goes also to Spullo (the dangerous man) and Co.: you are my favorite place in Macerata.

Well last but not least the most obvious one, my family. They know me maybe better than everyone else, and I have simply no words to thank them, they know, sometimes I should show it more. Thanks again for everything you have done for me, and you know it's a lot. I feel sorry for being so distant: this year I can personally thank also my brother, Pari and Arianna. I hope that, who knows in which country, we will finally be less distant, the time you have spent here in these days made me understand, again, how much I miss you.

Contents

Introduction **1**

I Preliminaries **5**
- I.1 Spaces, sequences, trees, and graphs . . . 5
  - I.1.1 Trees . . . 6
  - I.1.2 Graphs . . . 8
- I.2 Computability: from $\mathbb{N}$ to $\mathbb{N}^{\mathbb{N}}$ and represented spaces . . . 10
- I.3 Descriptive set theory . . . 14
  - I.3.1 Boldface hierarchy . . . 16
  - I.3.2 Effective Descriptive Set Theory . . . 19
  - I.3.3 (Effective) Wadge reducibility . . . 22
- I.4 Classification problems and complexity of structures . . . 26
  - I.4.1 Invariant descriptive set theory . . . 26
  - I.4.2 Computable structure theory . . . 30
- I.5 Reverse mathematics . . . 32
- I.6 Weihrauch reducibility . . . 34

1 Comparing problems via Weihrauch reducibility **43**

The Weihrauch lattice, with a focus on $\mathbf{\Pi}^1_1-\mathsf{CA}_0$ **45**

II The Cantor-Bendixson theorem **47**
- II.1 The perfect set theorem in $\mathbb{N}^{\mathbb{N}}$ and $2^{\mathbb{N}}$ . . . 48
  - II.1.1 Perfect sets . . . 48
  - II.1.2 Listing problems . . . 50
- II.2 Functions arising from the Cantor-Bendixson Theorem . . . 53
  - II.2.1 Perfect kernels . . . 53
  - II.2.2 Scattered lists . . . 57
  - II.2.3 The full Cantor-Bendixson theorem . . . 63
- II.3 What happens in arbitrary computable metric spaces . . . 64
  - II.3.1 Perfect sets . . . 65
  - II.3.2 (Weak) lists . . . 66
  - II.3.3 The Cantor-Bendixson theorem . . . 69
- II.4 Conclusions and open questions . . . 71

III The (induced) subgraph problem **73**
- III.1 Introduction to the problem . . . 73
- III.2 The represented spaces of graphs . . . 76
- III.3 Effective Wadge complexity of sets of graphs . . . 77
  - III.3.1 Deciding (induced) subgraphs problems in the Weihrauch lattice . . . 84
  - III.3.2 The "opposite" problem . . . 86
- III.4 Searching the (induced) subgraph . . . 89
  - III.4.1 The induced subgraph problem for infinite graphs . . . 92
  - III.4.2 The subgraph problem: when $R_\omega \subseteq_{\mathbf{s}} G$ . . . 94
  - III.4.3 A particular case: $\mathsf{S\text{-}Copy}_{R_\omega}$ . . . 96
  - III.4.4 Other subgraph problems . . . 105

III.5 Conclusions and open problems 107

2 Algorithmic learning theory of algebraic structures 109

The purpose, the paradigm and the learning criteria 111

IV Learning families of algebraic structures with the help of Borel equivalence relations 115

IV.1 **InfEx**-learnability and E_0 117

IV.2 A novel approach: E-learnability 118

IV.2.1 When oracle equivalence relations don't help: E_1 and E_2 119

IV.2.2 Characterizing the learning power of E_0^ω 121

IV.2.3 A syntactic characterization of E_0^ω-learnability 124

IV.3 Learning with the help of Z_0 and $=^+$ 128

IV.3.1 Z_0-learning 128

IV.3.2 $=^+$-learning 130

V Calculating the Mind Change Complexity of Learning Algebraic Structures 131

V.1 A topological characterization of the mind change complexity 131

V.2 Learnability and posets 135

V.3 Learner's complexity and number of mind changes 136

V.4 Conclusions and open questions (Chapter IV and Chapter V) 138

Bibliography 139

Introduction

The thesis' title suggests that we are going to classify *problems*: the Ph.D. field spoils that they are *mathematical* ones. Even if we heavily restricted the possible meanings of "problem", since mathematics tends to be rigorous (and hopefully this thesis as well), we still have to make precise what do we mean by "classify a mathematical problem". From now on in this introduction, since we clarified the context, we use the word "problem" to indicate a mathematical one. As people interested in mathematical logic, when we face a problem we want to know how complicated it is: hence, for us, classifying a problem means understanding its *complexity*. Towards this goal, researchers in this area developed several frameworks capturing different nuances of complexity. Of course, one could have settled the arriving point of this study at "we consider the problem in this framework easy/difficult" and be happy with this information. But one could also start considering an endless chain of questions: is this problem easy/difficult in framework 1? And in framework 2? ... And in framework n? ... Does the fact that this problem is easy/difficult in framework n, implies that the same problem is easy/difficult in framework m? What if we restrict the possible inputs to the problem? What if we restrict the possible outputs? What if both? Why is this problem so easy/difficult? If this problem is easy/difficult, does this mean that this other problem is easy/difficult as well? .... This is the approach chosen by mathematical logic, and the one we consider in this thesis: the most stressful but, from our humble point of view, also the most interesting one.

The subtitle of the thesis anticipates that the problems we classify and the frameworks we use come from *computable analysis* and *algorithmic learning theory*, but our classification is not confined just to these two areas. For example, we use different notions of complexity and frameworks from *descriptive set theory*, introduced in §I.3.

One of the approaches coming from descriptive set theory is the study of the complexity of a problem in terms of the complexity of its definition. More precisely, this area studies "definable" sets in topological (in particular, Polish) spaces classifying them with respect to their topological properties and the complexity of the formulas defining them. This gives rise to different hierarchies like the *boldface hierarchy* (defined in §I.3.1) and its effective counterpart, the *lightface hierarchy* (see §I.3.2). How these hierarchies are structured and which interesting sets inhabit their levels are a matter of study for people working in this field.

The approach we just described provides an "absolute" complexity of a problem. If we want to compare a "relative" complexity of problems, the notion of *reducibility* comes in handy. Informally, given a notion of complexity c, we say that a problem P is c-reducible to a problem Q if P is easier (in the sense of c) than Q. If also the converse holds, we say that the two problems are c-equivalent. We usually call the class containing all the c-equivalent problems a c-*degree*. Classifying problems under a fixed notion of reducibility induces a structure among them that can be studied locally (i.e. which concrete problems belong to a certain degree) or globally (i.e. exploring the algebraic properties and the degree structure). The comparison of problems via reducibilities is common to many frameworks. For example, going on with descriptive set theory, we have the notion of *(effective) Wadge reducibility* (defined in §I.3.3) that compares sets via continuous (computable) functions: this is studied, for example, in Chapter III.

Reducibilities are also common in *computability theory*. This well known subject, starting with the seminal work of Turing, succeeded in formalizing the informal notion of what is an "algorithm" and suggested a way to compare problems algorithmically. That is, if solving the problem Q we can solve problem P, in some sense P is algorithmically easier than Q. Classical computability theory deals with objects that can be coded by natural numbers. In this context the coding is unproblematic, but when we study the computability of problems involving object that cannot be coded by natural numbers, how we code (*represent*) them play a crucial role. This is (one of the) topic(s) of computable analysis, presented in §I.2. Here, an infinite object is represented via an infinite sequence of natural numbers, on which we can define a natural notion of computability: this induces the notion of *represented space* (see Definition $I.2.6$).

Once we represented an infinite object, we can formalize a problem as a *multi-valued function*, i.e. a function with multiple outputs (see Definition $I.2.3$). Multi-valued functions can be compared via the framework of *Weihrauch reducibility*, the uniform counterpart of Turing reducibility. As first noticed in [GM09], Weihrauch reducibility provides also a bridge between computable analysis and *reverse mathematics*, the discipline aiming to determine which axioms are "necessary" to prove certain theorems of "ordinary mathematics" (see §I.5 for an introduction to the subject). Many theorems are of the form "for all x, if $\varphi(x)$ holds then $\psi(x, y)$ holds" and this formulation has a natural translation as a computational problem: given in input x such that $\varphi(x)$ holds the solutions are those ys such that $\psi(x, y)$ holds. We have already suggested that we can formalize problems as multi-valued functions, and here comes the connection between computable analysis and reverse mathematics: we can compare theorems via the framework of Weihrauch reducibility. The interplay between these two areas is discussed at the beginning of Part 1 and Chapter II (and partly Chapter III) fall in this area.

In the second part of the thesis, we consider different problems, namely *learning problems*. We are in the area of algorithmic learning theory, the research program aiming to formally "describe" an empirical phenomenon (see the beginning of Part 2 for an introduction to the subject). This area provides different frameworks depending on the scenario one wants to model, but the core idea in all of them can be informally described as follows. There is an agent (e.g. a Turing machine) that we call a *learner* and a learning problem consisting of a set of possible answers to a given empirical inquiry: step by step the learner receives hints on such an inquiry and tries to converge to the correct solution. The paradigm we use for modeling learning of algebraic structures (Definition 2) borrows ideas from computable structure theory (see §I.4.2 for a brief presentation of this subject), the field of computable mathematics studying the interplay between the complexity of a mathematical structure and its structural/algorithmic properties. One of our main results in this part shows that we can characterize the (non) learnability of a family of structures using ideas coming from *invariant descriptive set theory* and, in particular, the study of *definable equivalence relations* (see §I.4). This area studies *classification problems* that are formalized by pairs (X, E), for a set X and an equivalence relation E, satisfying certain properties, and compares them via a notion of reducibility between equivalence relations. A particular subclass of such problems are *isomorphism problems*: informally, what we show is that learnability can be described as the study of the complexity of some particular isomorphism problems.

To summarize: in this thesis, we aim to classify the complexity of mathematical problems in computable analysis and algorithmic learning theory (for algebraic structures) showing also the interplay with other disciplines like (effective) descriptive set theory and reverse mathematics. The thesis consists of five chapters:

- in Chapter I, we give (most of) the preliminaries for the rest of the thesis. In §I.1 we fix some general notations, with particular attention to trees and graphs. In §I.2 we introduce the notion of computability on Baire space and represented spaces while §I.3 presents the basic concepts of (effective) descriptive set theory. In §I.4 we turn our attention to invariant descriptive set theory and computable structure theory. Finally, §I.5 and §I.6 speak about reverse mathematics and Weihrauch reducibility respectively. In particular, in the latter we introduce the notion of *finitary part* of a problem, that is tightly connected to other concepts studied in Weihrauch reducibility like the first-order part and the deterministic part of a problem. This notion was introduced together with Arno Pauly while working on topics presented in Chapter III.

After the first chapter, the thesis splits in two parts: Part 1 focuses on Weihrauch reducibility and Part 2 focuses on algorithmic learning theory for algebraic structures. Each of the remaining chapters has its own introduction, but we rapidly give an outline of the topics. Part 1 consists of Chapter II and Chapter III:

- Chapter II is based on a joint work with Alberto Marcone and Manlio Valenti. Here we study the Cantor-Bendixson theorem in the Weihrauch lattice continuing the program connecting Weihrauch reducibility and reverse mathematics: as far as we know, this is the first systematic study of problems at the level of $\mathbf{\Pi}^1_1{-}\mathsf{CA}_0$ in the Weihrauch lattice.
- Chapter III is based on a joint work with Arno Pauly and considers problems related to (induced) subgraphs via (effective) Wadge reducibility and Weihrauch reducibility. We highlight analogies and differences between the subgraph and the induced subgraph relation, and in particular we solve a couple of open questions left open in [BHW21].

Part 2 consists of Chapter IV and Chapter V: both chapters are based on joint works with Nikolay Bazhenov and Luca San Mauro.

- In Chapter IV we introduce a method for calibrating the complexity of nonlearnable families. To do so, we borrow ideas from invariant descriptive set theory, offering a new hierarchy to classify the complexity of learning problems for algebraic structures. This gives us a notion of reducibility between learning problems and gives a new characterization of learnability in descriptive set-theoretic terms.
- In Chapter V we study the number of mind changes that a learner makes while learning a given family. We give two different characterizations of the mind change complexity, a topological one and a combinatorial one. Finally, we study how the complexity of a learner, defined in terms of Turing reducibility, affects the number of mind changes required to learn a given family: this suggests the notion of *learning degree* of a family of structures.

I

Preliminaries

We warn the reader that we do not mean to give an exhaustive presentation of the topics presented in this chapter. We assume the reader to be familiar with basic notions of computability (as presented for example in [Rog87]), but we (at least try) to give the necessary background to make the reading smooth and to clearly fix the notations for the next chapters.

I.1 Spaces, sequences, trees, and graphs

We use the following abbreviations for quantifiers:

$$\exists^\infty n := (\forall n \in \mathbb{N})(\exists m \geqslant n) \text{ and } \forall^\infty n := (\exists n \in \mathbb{N})(\forall m \geqslant n).$$

Given a set X we denote by $\mathcal{P}(X)$ the power set of X. We denote the natural numbers $\{0, 1, 2, \dots\}$ with $\mathbb{N}$, the integers with $\mathbb{Z}$, the rationals with $\mathbb{Q}$ and the reals with $\mathbb{R}$. Furthermore, let

- $\mathbb{N}^\mathbb{N}$ denote the *Baire space*, i.e. the space of functions from $\mathbb{N}$ to $\mathbb{N}$, and
- $2^\mathbb{N}$ denote *Cantor space*, i.e. the space of functions from $\mathbb{N}$ to 2.

Both spaces come with the natural product topology where $\mathbb{N}$ and 2 are endowed with the discrete one. In both cases their topology is generated by the following metric:

$$d(p,q) := \begin{cases} \frac{1}{\min\{n : p(n) \neq q(n)\}+1} & \text{if } p \neq q, \\ 0 & \text{otherwise.} \end{cases}$$

Let σ be a finite sequence of natural numbers (respectively, of 0's and 1's): the basic open sets of $\mathbb{N}^\mathbb{N}$ ($2^\mathbb{N}$) are of the form $N_\sigma := \{p \in \mathbb{N}^\mathbb{N} : \sigma \sqsubset p\}$ ($N_\sigma := \{p \in 2^\mathbb{N} : \sigma \sqsubset p\}$). It is easy to notice that in both cases the complement of N_σ is also closed, and hence each N_σ is clopen. We denote by id the identity function on $\mathbb{N}^\mathbb{N}$. We usually refer to an element of $\mathbb{N}^\mathbb{N}$ or $2^\mathbb{N}$ as an *infinite sequence*.

A function f from a space X to a space Y is denoted by $f :\subseteq X \to Y$: the symbol "$\subseteq$" denotes that f is partial, and, if absent, it means that f is total.

Let $\mathbb{N}^n$ be the set of finite sequences of natural numbers of length n, where the length is denoted by $|\cdot|$. If $n = 0$, $\mathbb{N}^0 = \{\langle\rangle\}$, where $\langle\rangle$ is the empty sequence: in general, given $i_0, \dots, i_{n-1} \in \mathbb{N}$, we denote by $\langle i_0, \dots, i_{n-1}\rangle$ the finite sequence in $\mathbb{N}^n$ having digits $i_0, \dots, i_{n-1}$. For $\sigma \in \mathbb{N}^{<\mathbb{N}}$ and $m \leqslant |\sigma|$, let $\sigma[m] := \langle\sigma(0), \dots, \sigma(m-1)\rangle$. Given $\sigma, \tau \in \mathbb{N}^{<\mathbb{N}}$, we use $\sigma \sqsubseteq \tau$ to say that σ is an *initial segment* of τ (equivalently, τ an *extension* of σ), i.e. $\sigma = \tau[m]$ for some $m \leqslant |\tau|$. We use the symbol $\sqsubset$ in case $\sigma \sqsubseteq \tau$ and $|\sigma| < |\tau|$, and in case $\sigma \not\sqsubseteq \tau$ and $\tau \not\sqsubseteq \sigma$ we say that σ and τ are *incomparable* (in symbols, $\sigma \mid \tau$). The *concatenation* of two finite sequences σ, τ is denoted by $\sigma^\frown\tau$, but often we just write $\sigma\tau$. The same symbol is also used for the concatenation of a finite and an infinite sequence. For $n, k \in \mathbb{N}$, we denote by n^k the sequence made of k many n's: in case $k = 1$ we just write n, and we use $n^\mathbb{N}$ to denote the infinite sequence with constant value n. For $\sigma \in \mathbb{N}^{<\mathbb{N}}$ and $p \in \mathbb{N}^\mathbb{N}$ we denote by σ^- and p^- the result of deleting the first digit of the sequence. Given $\sigma, \tau \in \mathbb{N}^n$, we define the *join* of σ and τ as $\sigma \oplus \tau := \langle\sigma(0), \tau(0), \dots, \sigma(n-1), \tau(n-1)\rangle$. The same definition easily generalizes to infinite sequences and to countably many (finite or infinite) sequences of the same length in a straightforward way.

Remark I.1.1. We fix a bijection between $\mathbb{N}^{<\mathbb{N}}$ and $\mathbb{N}$: to avoid too much notation we do not introduce a specific symbol for this bijection, but we identify a sequence with the number representing it. It should be clear from the context whether we are referring to a finite sequence or to the number representing it. We can safely assume that such a bijection enjoys the following natural properties: given $\sigma, \tau \in \mathbb{N}^{<\mathbb{N}}$ and $i \in \mathbb{N}$, the functions $\sigma \mapsto |\sigma|$, $(\sigma, \tau) \mapsto \sigma^\frown\tau$, $(\sigma, i) \mapsto \sigma(i)$ and the function that, given as input n outputs the code of the string containing just n are computable. We also assume that if $\sigma \sqsubset \tau$ then $\sigma < \tau$.

I.1.1 Trees

A *tree* T is a nonempty subset of $\mathbb{N}^{<\mathbb{N}}$ closed under initial segments; in case T is a subset of $2^{<\mathbb{N}}$, we call T a *binary tree*. We say that $f \in \mathbb{N}^{\mathbb{N}}$ is a *path* through T if for all $n \in \mathbb{N}$, $f[n] \in T$ where, as for finite sequences, $f[n] = \langle f(0), \dots, f(n-1)\rangle$. We denote by $[T]$ the *body* of T, that is the set of paths through T. We say that a tree T is *ill-founded* if and only if there exists at least one path in $[T]$ and *well-founded* otherwise. Given $\sigma \in T$ we define the *tree of extensions of* σ *in* T as $T_\sigma := \{\tau : \tau \sqsubseteq \sigma \lor \tau \sqsupseteq \sigma\}$. We say that T is *perfect* if every element of T has (at least) two incompatible extensions in T, that is, $(\forall \sigma \in T)(\exists \tau, \tau' \in T)(\sigma \sqsubset \tau \land \sigma \sqsubset \tau' \land \tau \mid \tau')$. It is straightforward that the body of a nonempty perfect tree has uncountably many paths. Given a tree T, the largest perfect subtree S of T is called the *perfect kernel* of T while $[T] \setminus [S] \subseteq \mathbb{N}^{\mathbb{N}}$ is called the scattered part of T. We call T *pruned* if every $\sigma \in T$ has a proper extension. Moreover, if $[T]$ is perfect and T is pruned then T is a perfect tree.

Remark I.1.2. It is useful to notice that, for a binary tree T, if $|[T]| > \aleph_0$ then there must uncountably many paths with infinitely many ones. In other words, it can't be the case that all the paths in $[T]$ are eventually zero paths, as it is straightforward to notice that they are only countably many.

We now define and study the properties of some operations between trees that are heavily used especially in Chapter II and also in Chapter III. Given trees T and S, we define the *disjoint union* of T and S as $T \sqcup S = \{\langle\rangle\} \cup \{\langle 0\rangle\tau : \tau \in T\} \cup \{\langle 1\rangle\tau : \tau \in S\}$. Of course, this is still a tree, and it has the property that $T \sqcup S$ is ill-founded if and only if at least one of T and S is ill-founded. The construction can be easily generalized to countably many trees letting $\bigsqcup_{i\in\mathbb{N}} T^i := \{\langle\rangle\} \cup \{\langle i\rangle\tau : \tau \in T^i \land i \in \mathbb{N}\}$, and we still have that $\bigsqcup_{i\in\mathbb{N}} T^i$ is ill-founded if and only if there exists i such that T^i is ill-founded. We also define the *binary disjoint union* as $\bigsqcup^{\mathrm{b}}_{i\in\mathbb{N}} T^i := \{\langle\rangle\} \cup \{0^i\langle 1\rangle\tau : \tau \in T^i \land i \in \mathbb{N}\}$.

Remark I.1.3. Notice that if all the T^i's are binary trees, also $\bigsqcup^{\mathrm{b}}_{i\in\mathbb{N}} T^i$ is and, regardless the ill-foundedness/well-foundedness of the T^i's, $0^{\mathbb{N}} \in \left[\bigsqcup^{\mathrm{b}}_{i\in\mathbb{N}} T^i\right]$. Moreover, $\left|\left[\bigsqcup^{\mathrm{b}}_{i\in\mathbb{N}} T^i\right]\right| = 1 + \sum_{i\in\mathbb{N}} \left|\left[T^i\right]\right|$. In particular, $\left|\left[\bigsqcup^{\mathrm{b}}_{i\in\mathbb{N}} T^i\right]\right| > 1$ if and only if there exists an $i \in \mathbb{N}$ such that T^i is ill-founded.

We now turn our attention to another operation on trees, namely *interleaving*. Given trees T and S, the interleaving between T and S is $T \oplus S := \{\sigma \oplus \tau : |\sigma| = |\tau| \land \sigma \in T \land \tau \in S\}$. Clearly, $T \oplus S$ is a tree, and it is ill-founded if and only if both T and S are ill-founded. This construction can be generalized to countably many trees in a straightforward way, and we use a notation such as $\oplus_{i\in\mathbb{N}} T^i$.

We often use the interleaving $\mathsf{Expl}(T) := T \oplus 2^{<\mathbb{N}}$, which we call the *explosion* of T.

Sometimes it is useful to be able to "translate" back and forth between sequences of natural numbers and binary sequences.

Definition I.1.4. We define:

- $\rho_{2^\mathbb{N}} \colon \mathbb{N}^{<\mathbb{N}} \to 2^{<\mathbb{N}}$ by
$$\rho_{2^\mathbb{N}}(\sigma) := 0^{\sigma(0)}10^{\sigma(1)}1 \dots 10^{|\sigma|-1}1.$$
In particular, $\rho_{2^\mathbb{N}}(\langle\rangle) := \langle\rangle$;
- $\rho_{\mathbb{N}^\mathbb{N}} \colon 2^{<\mathbb{N}} \to \mathbb{N}^{<\mathbb{N}}$ by
$$\rho_{\mathbb{N}^\mathbb{N}}(\tau) := \begin{cases} \rho_{2^\mathbb{N}}^{-1}(\tau[n_\tau + 1]) & \text{if } (\exists i)(\tau(i) = 1) \text{ where } n_\tau := \max\{i : \tau(i) = 1\}; \\ \langle\rangle & \text{if } (\forall i)(\tau(i) = 0). \end{cases}$$

The two functions defined above have the following properties:

- $\rho_{2^\mathbb{N}}$ is injective;
- $\rho_{\mathbb{N}^\mathbb{N}}(\rho_{2^\mathbb{N}}(\sigma)) = \sigma$;
- $\sigma \sqsubset \sigma'$ if and only if $\rho_{2^\mathbb{N}}(\sigma) \sqsubset \rho_{2^\mathbb{N}}(\sigma')$;
- if $\tau \sqsubset \tau'$ then $\rho_{\mathbb{N}^\mathbb{N}}(\tau) \sqsubseteq \rho_{\mathbb{N}^\mathbb{N}}(\tau')$.

We are now able to "translate" back and forth between trees on $\mathbb{N}$ and binary trees. We use the same symbols $\rho_{2^\mathbb{N}}$ and $\rho_{\mathbb{N}^\mathbb{N}}$ as the context explains which function we are using.

Definition I.1.5. Let $T \subseteq 2^{<\mathbb{N}}$ and $S \subseteq \mathbb{N}^{<\mathbb{N}}$ be trees. We define:

- $\rho_{\mathbb{N}^\mathbb{N}}(T) := \{\sigma \in \mathbb{N}^{<\mathbb{N}} : \rho_{2^\mathbb{N}}(\sigma) \in T\}$;
- $\rho_{2^\mathbb{N}}(S) := \{\tau \in 2^{<\mathbb{N}} : \rho_{\mathbb{N}^\mathbb{N}}(\tau) \in S\}$.

Remark I.1.6. Notice that, since $\rho_{\mathbb{N}^\mathbb{N}}(\sigma 0^n) = \rho_{\mathbb{N}^\mathbb{N}}(\sigma)$ for every n, if $\sigma \in \rho_{\mathbb{N}^\mathbb{N}}(T)$ then $\sigma 0^\mathbb{N} \in [\rho_{\mathbb{N}^\mathbb{N}}(T)]$. It is straightforward to check that $\rho_{\mathbb{N}^\mathbb{N}}(T) = \{\rho_{\mathbb{N}^\mathbb{N}}(\tau) \in \mathbb{N}^{<\mathbb{N}} : \tau \in T\}$. On the other hand, for most trees, $S \subseteq \mathbb{N}^{<\mathbb{N}}$, $\rho_{2^\mathbb{N}}(S) \neq \{\rho_{2^\mathbb{N}}(\tau) \in 2^{<\mathbb{N}} : \tau \in S\}$ as the latter is not even a tree.

The back-and-forth translations between sequences in $\mathbb{N}^\mathbb{N}$ and $2^\mathbb{N}$ are also denoted by the same function symbols used for finite sequences and for trees: again the context clarifies which one we are using.

Definition I.1.7.

- $\rho_{2^\mathbb{N}} \colon \mathbb{N}^\mathbb{N} \to 2^\mathbb{N}$ is defined by
$$\rho_{2^\mathbb{N}}(p) := \bigcup_{n \in \mathbb{N}} \rho_{2^\mathbb{N}}(p[n]) = 0^{p(0)}10^{p(1)} \dots 10^{p(n)}1 \dots;$$
- $\rho_{\mathbb{N}^\mathbb{N}} :\subseteq 2^\mathbb{N} \to \mathbb{N}^\mathbb{N}$ has domain $\{q : (\exists^\infty i)(q(i) = 1)\}$ and is defined by
$$\rho_{\mathbb{N}^\mathbb{N}}(q) := \bigcup_{n \in \mathbb{N}} \rho_{\mathbb{N}^\mathbb{N}}(q[n]).$$

In both definitions, the union makes sense because the finite sequences are comparable.

Notice that all the functions $\rho_{\mathbb{N}^\mathbb{N}}$ and $\rho_{2^\mathbb{N}}$ we defined are computable. For the functions on finite sequences, this means usual Turing computability.

Lemma I.1.8. *The following lemma summarizes the fundamental properties of $\rho_{2^{\mathbb{N}}}$ and $\rho_{\mathbb{N}^{\mathbb{N}}}$ for infinite sequences and trees.*

1. *The range of $\rho_{2^{\mathbb{N}}}$ is $\{q \in 2^{\mathbb{N}} : (\exists^\infty i)(q(i) = 1)\}$;*
2. *$\rho_{\mathbb{N}^{\mathbb{N}}}(\rho_{2^{\mathbb{N}}}(p)) = p$ for every $p \in \mathbb{N}^{\mathbb{N}}$.*
3. *$\rho_{2^{\mathbb{N}}}(\rho_{\mathbb{N}^{\mathbb{N}}}(q)) = q$ for every $q \in \operatorname{dom}(\rho_{\mathbb{N}^{\mathbb{N}}})$.*
4. *If $S \subseteq \mathbb{N}^{<\mathbb{N}}$, $p \in [S] \iff \rho_{2^{\mathbb{N}}}(p) \in [\rho_{2^{\mathbb{N}}}(S)]$ and hence $[\rho_{2^{\mathbb{N}}}(S)] \subseteq \{\rho_{2^{\mathbb{N}}}(p) : p \in [S]\} \cup \{q : (\forall^\infty i)(q(i) = 0)\}$ so that $|[\rho_{2^{\mathbb{N}}}(S)]| \leqslant \aleph_0 \iff |[S]| \leqslant \aleph_0$.*
5. *If $T \subseteq 2^{<\mathbb{N}}$ and $q \in \operatorname{dom}(\rho_{\mathbb{N}^{\mathbb{N}}})$ we have that $q \in [T] \iff \rho_{\mathbb{N}^{\mathbb{N}}}(q) \in [\rho_{\mathbb{N}^{\mathbb{N}}}(T)]$.*
6. *If $T \subseteq 2^{<\mathbb{N}}$ and $p \in \mathbb{N}^{\mathbb{N}}$ then $p \in [\rho_{\mathbb{N}^{\mathbb{N}}}(T)] \iff \rho_{2^{\mathbb{N}}}(p) \in [T]$.*

Proof. The proofs are straightforward from the definitions above. □

Lemma I.1.9. *If T is a binary tree such that $[T]$ is perfect then $[\rho_{\mathbb{N}^{\mathbb{N}}}(T)]$ is perfect as well. Furthermore, if T is a perfect tree, then $\rho_{\mathbb{N}^{\mathbb{N}}}(T)$ is also a perfect tree.*

Proof. Let T be a binary tree such that $[T]$ is perfect. We show that no $f \in [\rho_{\mathbb{N}^{\mathbb{N}}}(T)]$ is isolated, i.e. $(\forall n)(\exists g \in [\rho_{\mathbb{N}^{\mathbb{N}}}(T)])(f[n] \sqsubset g \land f \neq g)$. Fix n: by Lemma *I*.1.8(6) we get that $\rho_{2^{\mathbb{N}}}(f) \in [T]$ and, in particular, $\sigma := \rho_{2^{\mathbb{N}}}(f[n]) \in T$. Since $[T]$ is perfect, by Remark *I*.1.2, there exists $h \in [T]$ with infinitely many ones such that $\sigma \sqsubset h$ and $\rho_{2^{\mathbb{N}}}(f) \neq h$. By Lemma *I*.1.8(5) $\rho_{\mathbb{N}^{\mathbb{N}}}(h) \in [\rho_{\mathbb{N}^{\mathbb{N}}}(T)]$ and letting $g := \rho_{\mathbb{N}^{\mathbb{N}}}(h)$ we reach the conclusion.

In case T is a perfect tree, it suffices to show that $\rho_{\mathbb{N}^{\mathbb{N}}}(T)$ is pruned. Suppose there exists $\sigma \in \rho_{\mathbb{N}^{\mathbb{N}}}(T)$ with no extensions in $\rho_{\mathbb{N}^{\mathbb{N}}}(T)$. Then $\tau := \rho_{2^{\mathbb{N}}}(\sigma)$ belongs to T and the only path in T extending τ is of the form $\tau 0^{\mathbb{N}}$, contradicting the perfectness of T. □

Notice that if $S \subseteq \mathbb{N}^{<\mathbb{N}}$ is a perfect tree it may be the case that $[\rho_{2^{\mathbb{N}}}(S)]$ is not perfect, e.g. let $S = \{\sigma \in \mathbb{N}^{<\mathbb{N}} : \sigma(0) = 0\}$ and notice that $0^{\mathbb{N}}$ is isolated in $[\rho_{2^{\mathbb{N}}}(S)]$.

I.1.2 Graphs

A graph G is a pair (V, E) where V is the set of *vertices* and E is a binary relation on $V \times V$; a pair $(v, w) \in E$ is called an *edge*.

Remark I.1.10. In this thesis, when we say graph, we always assume that is countable, undirected, and without self-loops: that is, $V \subseteq \mathbb{N}$ and E satisfies anti-reflexivity and symmetry.

Given a graph G we denote the set of vertices and the set of edges respectively with $V(G)$ and $E(G)$. Given graphs G and H, we write $G \cong H$ to denote that G and H are isomorphic, and we often say that "G is a copy of H" or vice versa. We say that a graph G is *finite* if $V(G)$ is finite, infinite otherwise. Given $v \in V(G)$, let $\{w \in V(G) : (v, w) \in E(G)\}$ the set of *neighbors* of v in G, and define the *degree* of v in G as $\deg^G(v) := |\{w : (v, w) \in E(G)\}|$.

For a graph G and $n > 0$, a *ray of length* n is a sequence of distinct vertices $v_0, \dots, v_n \in V(G)$ such that for every $i < n$, $(v_i, v_{i+1}) \in E(G)$: we say that v_0 and v_n *endpoints* of the ray. Given $u, v, w \in V(G)$, we say that v *and* u *are ray-connected in* G *through* w, denoted by $v \leftrightsquigarrow^G_w u$, if there exists a ray of finite length $n > 0$ in G, with endpoints v and u containing w: we write $v \leftrightsquigarrow^G_{\neg w} u$ to denote that v and u are ray-connected but no ray of finite length with endpoints u and v contains w. If we simply want to say that v and u are ray-connected in G, we drop the subscript w (or $\neg w$). Notice that $v \leftrightsquigarrow^G u$ by a path of length 1 if and only if $(v, w) \in E(G)$. We say that a graph G is *connected* if $(\forall v, u \in V(G))(v \leftrightsquigarrow^G u)$.

We continue defining three particular types of graphs, namely *rays*, *complete graphs*, and *cycles*. Notice that the ray we are defining now is the graph having as vertex set an initial segment of $\mathbb{N}$ and defined as the ray above. Formally, for $n > 0$ and $m > 2$ we define R_n, K_n and C_m as the graphs having the same vertex set $\{i : i \leqslant n\}$ and $\{i : i \leqslant m\}$. The edge sets are respectively: $E(R_n) = \{(i, i{+}1) : i < n\}$, $E(K_n) = \{(i, j) : i \neq j \wedge i, j < n\}$ and $E(C_m) = E(R_{m-1}) \cup \{(m{-}1, 0)\}$. It is immediate that $C_3 = K_3$. Notice that R_n and K_n generalize to the infinite case: that is, R_ω and K_ω are the graphs having as vertex set $\mathbb{N}$ and as edge set respectively $\{(i, i+1) : i \in \mathbb{N}\}$ and $\{(i, j) : i \neq j \wedge i, j \in \mathbb{N}\}$. Another infinite generalization of R_n we use is the "two-way infinite ray" L, where $V(\mathsf{L}) = \mathbb{N}$ and $E(\mathsf{L}) = \{(0, 1)\} \cup \{(2i, 2i + 2) : i \in \mathbb{N}\} \cup \{(2i + 1, 2i + 3) : i \in \mathbb{N}\}$.

Given countably many graphs $\{G_i : i \in \mathbb{N}\}$ we define the *disconnected union* $\bigotimes_{i \in \mathbb{N}} G_i$ as follows:

$$V(\bigotimes_{i \in \mathbb{N}} G_i) := \bigcup_{i \in \mathbb{N}} \{\langle i, v\rangle : v \in V(G_i)\} \text{ and } E(\bigotimes_{i \in \mathbb{N}} G_i) := \{(\langle i, v\rangle, \langle i, w\rangle) : (v, w) \in E(G_i)\}.$$

In case the disconnected union involves at most three graphs G_0, G_1, G_2, we write $G_0 \otimes G_1$ and $G_0 \otimes G_1 \otimes G_2$ respectively. We denote by $\overset{\infty}{\bigotimes} G$ the disconnected union of countably many copies of G.

Another operation on graphs we use is the *connected union* (denoted by $\bigodot_{i \in \mathbb{N}} G_i$) in which, intuitively, for every i, G_i and G_{i+1} share a unique common vertex, different from the one shared between G_{i+1} and G_{i+2}. Formally, given countably many graphs $\{G_i : i \in \mathbb{N}\}$ (for simplicity assume $|V(G_i)| \geqslant 3$ for every i), let $\mathsf{v_i} := \min\{v : v \in V(G_i)\}$ and, if G_i is finite, let $\mathsf{w_i} := \max\{v : v \in V(G)\}$, otherwise let $\mathsf{w_i} := \min\{v : v \in V(G_i) \setminus \{\mathsf{v}_i\}\}$. Then let,

$$V(\bigodot_{i \in \mathbb{N}} G_i) := V(\bigotimes_{i \in \mathbb{N}} G_i) \setminus \big\{\{\langle i, \mathsf{v}_i\rangle : i > 0\} \cup \{\langle i, \mathsf{w}_i\rangle : i \in \mathbb{N}\}\big\} \bigcup \big\{\langle \mathsf{w}_i, \mathsf{v}_{i+1}\rangle : i \in \mathbb{N}\big\},$$

$$E(\bigodot_{i \in \mathbb{N}} G_i) := \{(\langle i, v\rangle, \langle i, w\rangle) : v, w \notin \{\mathsf{v}_i, \mathsf{w}_i\} \wedge (v, w) \in E(G_i)\} \bigcup$$
$$\big\{\big(\langle \mathsf{w}_i, \mathsf{v}_{i+1}\rangle, \langle i, u\rangle\big) : (\mathsf{w}_i, u) \in E(G_i) \vee (\mathsf{v}_{i+1}, u) \in E(G_{i+1})\big\}.$$

Figure I.1: On the left side, the disconnected union $\bigotimes_{i>2} C_i$ of all cyclic graphs, shown up to C_5. On the right side, the connected union $\bigodot_{i>2} C_i$ of all cyclic graphs, shown up to C_5: starting from the left one, the two red vertices denote respectively the vertices $\langle \mathsf{w}_3, \mathsf{v}_4\rangle$ and $\langle \mathsf{w}_4, \mathsf{v}_5\rangle$.

As for the disconnected union, we write $G_0 \odot G_1$ and $G_0 \odot G_1 \odot G_2$ in case the connected union is defined only on two or three graphs. Notice that $\mathsf{L} \cong R_\omega \odot R_\omega$.

In §I.1.1 we have already defined what a tree is: it is easy to notice that a tree is a particular type of graph, and hence when considering a tree, adjusting some detail, we can choose at our convenience if we want to refer to it as a set of finite sequences or as a *graph theoretic tree*. A graph-theoretic tree is a connected graph in which any two vertices are connected by exactly one path: in other words, they are connected graphs that do not contain any cycle. The graph theoretic tree is v-*rooted* if there is a distinguished vertex $v \in V(G)$, namely the *root* of T. So, as anticipated, given a tree $T \in \mathbf{Tr}$ we can translate it as a v-rooted graph theoretic tree G where $v = \langle\rangle$, $V(G) = T$ and $E(G) = \{(\sigma, \tau) : \sigma\tau \in T\}$. Conversely, we can translate a v_0-rooted graph theoretic tree G into a tree T, identifying any $v \in V(G)$ with the sequence $\langle v_0, \ldots, v_n\rangle$, where $v_0, \ldots, v_n$ are the vertices of the unique path from v_0 to v_n and $v_n = v$. Notice that both translations are computable relative to the (graph-theoretic) tree (in case we have to translate T into a v-rooted tree just let $v = \langle\rangle$).

In case we drop the assumption that the graph-theoretic tree is connected, we obtain a *forest*: in other words, a forest, is the disconnected union of countably many graph-theoretic trees.

We give the definition of *subgraph* and the *induced subgraph* relations, whose computational properties are studied in Chapter III.

Definition I.1.11. Given two graphs G and H we say that:

- H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$;
- H is an *induced subgraph* of G if H is a subgraph of G and $E(H) = E(G) \cap (V(H) \times V(H))$.

Given graphs G and H, we use the following abbreviations:

- $G \subseteq_{\mathbf{s}} H :\iff (\exists G' \subseteq H)(G' \text{ is a subgraph of } H)$;
- $G \subseteq_{\mathbf{is}} H :\iff (\exists G' \subseteq H)(G' \text{ is an induced subgraph of } H)$.

Finally, given a graph G and $V \subseteq V(G)$, we define the *graph induced by V on G*, denoted by $G_{\restriction V}$ as the graph having $V(G_{\restriction V}) := V$ and $E(G_{\restriction V}) := E(G) \cap (V \times V)$.

I.2 Computability: from $\mathbb{N}$ to $\mathbb{N}^{\mathbb{N}}$ and represented spaces

In this section, we define the notion of computability on different objects proceeding in this order:

$$\text{Computability on } \mathbb{N} \Rightarrow \text{ Computability on } \mathbb{N}^{\mathbb{N}} \Rightarrow \text{ Computability on } \textit{represented spaces}.$$

As already mentioned at the beginning of §I.1, we assume the reader to be familiar with computability on $\mathbb{N}$, that is the classical notion of computability for functions and subsets of $\mathbb{N}$. Notice that this notion of computability is also known as *Type-1 computability*: the notion of *Type-2 computability* generalizes computability on $\mathbb{N}$ to infinite objects, first to functions and subsets of $\mathbb{N}^{\mathbb{N}}$ and finally to *represented spaces*. For more on these topics, we refer the reader to:

- Weihrauch's book "Computable Analysis: An Introduction" ([Wei13]).

The classical framework in which Type-1 computability is developed is the well known Turing machine. We think of it as consisting of an input tape and an output tape and, if the computation requires it, an oracle tape in which the oracle is stored: its precise definition is not important as, by the celebrated Church-Turing thesis, any reasonable computational model yields to the same notion of computability. A function $f :\subseteq \mathbb{N} \to \mathbb{N}$ is Type-1 computable if there is a Turing machine that, given in input n outputs in a finite amount of stages $f(n)$ and stops. The "given in input" part means that n is written in the input tape: then the Turing machine reads n and *converges* to $f(n)$, i.e. after finitely many stages writes $f(n)$ in the output tape and stops. The well known fact that it is possible to code finite strings over some countable alphabet Σ to $\mathbb{N}$ and vice versa tells us that computability can be extended to functions $f :\subseteq \Sigma^{<\mathbb{N}} \to \Sigma^{<\mathbb{N}}$: to summarize, we can say that we have a robust notion of computability for those objects that can be coded with natural numbers.

For infinite objects, the first step is to define a notion of computability on $\mathbb{N}^{\mathbb{N}}$. The model of computation is a *Type-2 machine*. A Type-2 machine is a Turing machine with a read-only input tape, a write-only output tape, a working tape and, if the computation requires it, an oracle tape: again, its formal definition, even if it requires more attention than the one given for the classical Turing machine, is not important. Given a function $f :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$, the input for the Type-2 machine is an infinite sequence of natural numbers. This is the first important difference: while in the context of Type-1 computability, a Turing machine in order to compute a function needs to read the input and write the output in a finite amount of steps, in Type-2 computability, after finitely many steps a Type-2 machine can only read a finite portion of the input. In other words, a Type-2 machine *is always expected to run forever*, something that was forbidden in Type-1 computability.

We say that f is Type-2 computable if there is a Type-2 machine that, taking as input longer and longer prefixes of the input, outputs longer and longer approximations of the output. We now give a more precise definition of Type-2 computability: from now on we drop the "Type-2", and we just write computable.

Definition I.2.1. A function $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is computable if there is a computable function $f :\subseteq \mathbb{N}^{<\mathbb{N}} \to \mathbb{N}^{<\mathbb{N}}$ such that

- if $\sigma \sqsubseteq \tau$ then $f(\sigma) \sqsubseteq f(\tau)$ (monotonicity);
- $F(p) = \bigcup_{\sigma \sqsubset p} f(\sigma)$ (F-approximation).

Since a point $p \in \mathbb{N}^{\mathbb{N}}$ is a function from $\mathbb{N}$ to $\mathbb{N}$, when we say that p is *computable* we mean that it is computable as a function. By [Wei13, Lemma 2.1.11] F is computable by a Type-2 machine if and only if F is computable in the sense of Definition $I.2.1$: the same definition highlights that the computation strictly depends on finite sequences, and this guarantees both that the computation can be simulated by an ordinary computer and that computable functions are also continuous. Notice that the converse is not true, however, the following well known theorem holds.

Theorem I.2.2 (Folklore)**.** *A function $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is continuous, if and only if it is computable relatively to an oracle $X \subseteq \mathbb{N}^{\mathbb{N}}$.*

We now switch our attention to *multi-valued functions*.

Definition I.2.3. A (possibly partial) multi-valued function $f :\subseteq X \rightrightarrows Y$ is a function $f : X \to \mathcal{P}(Y)$ where $\mathrm{dom}(f) := \{x \in X : f(x) \neq \emptyset\}$ and $\mathsf{range}(f) := Y$.

To simplify the notation, whenever $f(x) = \{y\}$ we just write $f(x) = y$. In case $f(x)$ is a singleton for every $x \in X$, we identify f with the (partial) function mapping every $x \in \mathrm{dom}(f)$, to the unique y such that $y \in f(x)$. In Definition $I.2.3$, we could have defined f as a relation $f \subseteq X \times Y$, but it is more convenient to use this formulation so that we can think of f as a *computational problem*. That is, we consider $\mathrm{dom}(f)$ as the set of admissible instances of f and, given $x \in \mathrm{dom}(f)$, we consider $f(x) \subseteq Y$ as the set of possible outputs on input x.

Remark I.2.4. From now on, the terms (computational) problem and multi-valued function are synonyms, and we use them interchangeably at our convenience.

An important difference between multi-valued functions and relations is how the composition behaves.

Definition I.2.5. Let $f :\subseteq X \rightrightarrows Y$ and $g :\subseteq Y \to Z$ be multi-valued functions. We define the composition $g \circ f :\subseteq X \rightrightarrows Z$ with domain $\{x \in \mathrm{dom}(f) : f(x) \subseteq \mathrm{dom}(g)\}$ as

$$g \circ f(x) := \{z \in Z : (\exists y \in Y)(y \in f(x) \land z \in g(y))\}.$$

In standard relations, the domain of the composition is not restricted as in the definition above. On the other hand, for multi multi-valued functions, the domain restriction ensures that we can apply g to any solution of $f(x)$ and $g \circ f$ is still a multi-valued function.

We are now ready to extend the notion computability on $\mathbb{N}^{\mathbb{N}}$ to the broader context of *represented spaces*.

Definition I.2.6. A represented space $\mathbf{X}$ is a pair (X, δ_X) where X is a set and $\delta_X :\subseteq \mathbb{N}^{\mathbb{N}} \to X$ is a (possibly partial) surjective function called *representation map*. We say that $p \in \mathbb{N}^{\mathbb{N}}$ is a *name* for x if $\delta_X(p) = x$.

It is worth mentioning that δ_X is neither required to be injective nor total: that is, an element of X may have multiple names, and not every $p \in \mathbb{N}^{\mathbb{N}}$ is a name for an element in X. The notion of represented space allows us to transfer the notion of computability introduced in Definition $I.2.1$ to arbitrary spaces. The computational task "given as input an f-instance x, compute an f-solution y" becomes "given in input a *name* for an f-instance x, compute a *name* for an f-solution y". This is made precise with the notion of *realizer*.

Definition I.2.7. Let $\mathbf{X}$ and $\mathbf{Y}$ be represented spaces. A partial function $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is a *realizer* of $f :\subseteq \mathbf{X} \to \mathbf{Y}$ (in symbols, $F \vdash f$) if and only if

$$(\forall p \in \operatorname{dom}(f \circ \delta_X))(\delta_Y(F(p)) \in f(\delta_X(p))).$$

Equivalently, $F \vdash f$ if, for all $p \in \operatorname{dom}(f \circ \delta_X)$, the following diagram commutes.

$$\begin{array}{ccc} \mathbb{N}^{\mathbb{N}} & \xrightarrow{F} & \mathbb{N}^{\mathbb{N}} \\ \downarrow{\scriptstyle \delta_X} & & \downarrow{\scriptstyle \delta_Y} \\ X & \overset{f}{\rightrightarrows} & Y \end{array}$$

We are now ready to define computability for partial multi-valued functions on represented spaces.

Definition I.2.8. Let $\mathbf{X}$ and $\mathbf{Y}$ be represented spaces. A partial multi-valued function $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$ is called (δ_X, δ_Y)-computable (respectively, (δ_X, δ_Y)-realizer-continuous) if it has a computable (continuous) realizer. A point $x \in X$ is called δ_X*-computable* if it has a computable name. In case the representation maps are clear from the context, we omit them, and we just write "computable" and "realizer-continuous".

We define the following operations between represented spaces $\mathbf{X}$ and $\mathbf{Y}$:

- $\mathbf{X} \times \mathbf{Y} := (X \times Y, \delta_{X \times Y})$ where $\delta_{X \times Y} :\subseteq \mathbb{N}^{\mathbb{N}} \to X \times Y$ is defined as
$$\delta_{X \times Y}(p \oplus q) := (\delta_X(p), \delta_Y(q));$$
- $\mathbf{X}^* := (X^*, \delta_{X^*})$ where $\delta_{X^*} :\subseteq \mathbb{N}^{\mathbb{N}} \to X^*$ is defined as
$$\delta_{X^*}(\langle n \rangle^\frown (p_1 \oplus \ldots \oplus p_n)) := (n, \delta_X(p_1), \ldots, \delta_X(p_n));$$
- $\mathbf{X}^{\mathbb{N}} := (X^{\mathbb{N}}, \delta_{X^{\mathbb{N}}})$ where $\delta_{X^{\mathbb{N}}} :\subseteq \mathbb{N}^{\mathbb{N}} \to X^{\mathbb{N}}$ is defined as
$$\delta_{X^{\mathbb{N}}}(p_1 \oplus p_2 \oplus \ldots) := (\delta_X(p_1), \delta_X(p_2), \ldots);$$
- $\mathbf{X} \bigsqcup \mathbf{Y} := (X \sqcup Y, \delta_{X \sqcup Y})$ where, $X \sqcup Y := (\{0\} \times X) \cup (\{1\} \times Y)$ is the disjoint union of sets, and $\delta_{X \sqcup Y} :\subseteq \mathbb{N}^{\mathbb{N}} \to X \sqcup Y$ is such that, given $i \in \{0, 1\}$,
$$\delta_{X \sqcup Y}(\langle i \rangle^\frown p) := \begin{cases} \delta_X(p) & \text{if } i = 0 \\ \delta_Y(p) & \text{if } i = 1. \end{cases}$$

We now introduce the *jump* of a represented space.

Definition I.2.9. Let $\mathbf{X} = (X, \delta_X)$ be a represented space: we define the jump of $\mathbf{X}$ as the represented space $\mathbf{X}' := (X, \delta'_X)$ where δ'_X takes as input a sequence of elements of $\mathbb{N}^{\mathbb{N}}$ converging to some $p \in \mathrm{dom}(\delta_X)$ and outputs $\delta_X(p)$.

Some well known represented spaces

We now introduce some represented spaces that play a crucial role in this thesis. Many of them have a natural representation: this is the case for $\mathbb{N}^{\mathbb{N}}$, $2^{\mathbb{N}}$ and $\mathbb{N}$, i.e. let $\delta_{\mathbb{N}^{\mathbb{N}}} := \mathsf{id}$ (where id is the identity function on $\mathbb{N}^{\mathbb{N}}$), $\delta_{2^{\mathbb{N}}} := \mathsf{id}{\restriction}2^{\mathbb{N}}$ (denoting id restricted to elements in $2^{\mathbb{N}}$) and $\delta_{\mathbb{N}}$ as $p \mapsto p(0)$.

One of the most common representations of $\mathbb{R}$ is the *Cauchy representation*. We fix a computable enumeration $(q_i)_{i\in\mathbb{N}}$ of $\mathbb{Q}$, and we define the Cauchy representation $\delta_{\mathbb{R}}$ having domain $\{p \in \mathbb{N}^{\mathbb{N}} : (\forall j)(\forall i > j)(|q_{p(i)} - q_{p(j)}| < 2^{-j})\}$ and letting $\delta_{\mathbb{R}}(p) := \lim q_{p(n)}$. From now on, the symbols $\mathbb{N}^{\mathbb{N}}$, $2^{\mathbb{N}}$, $\mathbb{N}$ and $\mathbb{R}$ indicate also the represented spaces: $(\mathbb{N}^{\mathbb{N}}, \delta_{\mathbb{N}^{\mathbb{N}}})$, $(2^{\mathbb{N}}, \delta_{2^{\mathbb{N}}})$, $(\mathbb{N}, \delta_{\mathbb{N}})$ and $(\mathbb{R}, \delta_{\mathbb{R}})$: it should be clear from the context whether we are referring to the represented space or the space itself. Notice that the classical notion of computability on $\mathbb{N}$ and $\mathbb{R}$ (as spaces) coincide with the one defined in Definition $I.2.8$ on represented spaces. Now we give the definition of *computable metric space*.

Definition I.2.10 ([Wei13, Definition 8.1.2]). Let $\mathcal{X} = (X, d, \alpha)$ be a separable[a] metric space, where $d : X \times X \to \mathbb{R}$ is the distance function and $\alpha : \mathbb{N} \to X$ is a function enumerating a dense subset of X.

- We define the *Cauchy representation* on X as the map $\delta_X :\subseteq \mathbb{N}^{\mathbb{N}} \to X$ with domain

$$\mathrm{dom}(\delta_X) := \{p \in \mathbb{N}^{\mathbb{N}} : (\forall n)(\forall m > n)(d(\alpha(p(n)) - \alpha(p(m))) < 2^{-n})\}$$

  as $\delta_X(p) := \lim_{n\to\infty} \alpha(p(n))$.

- We say that $\mathcal{X}$ is a *computable metric space* if the set

$$\{(i, j, n, m) \in \mathbb{N}^4 : q_i < d(\alpha(n), \alpha(m)) < q_j\}$$

  is computably enumerable.

[a] a topological space (X, τ) is *separable* if it contains a countable dense subset (i.e. a countable set that meets every nonempty open set of the space).

From now on we always assume that computable metric spaces are represented by this representation. For convenience, we fix a computable enumeration $(B_i)_{i\in\mathbb{N}}$ of all basic open sets of $\mathcal{X}$, where the ball $B_{\langle n,m\rangle}$ is centered in $\alpha(n)$ and has radius q_m.

Notice that for $\mathbb{R}$, the topological notion of continuity coincides with realizer-continuity. This nice property, which is not guaranteed for all represented spaces, intuitively tells us that the topological structure of the space agrees with the computational one. It is often convenient to deal with representations satisfying this requirement, namely *admissible representations*. Before giving the formal definition, we introduce a way to compare different representations.

Definition I.2.11 ([Zie07, Definition 2.3.2]). Given two represented spaces $\mathbf{X}$ and $\mathbf{Y}$ with $X \subseteq Y$ we say that $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ *translates* δ_X to δ_Y if and only if $(\forall p \in \mathrm{dom}(\delta_X))(\delta_X(p) = \delta_Y \circ F(p))$. We say that

- δ_X is *computably reducible* to δ_Y (in symbols, $\delta_X \leqslant \delta_Y$) if there exists a computable translation from δ_X to δ_Y;

- δ_X is *continuously reducible* to δ_Y (in symbols $\delta_X \leqslant_t \delta_Y$) if there exists a continuous translation from δ_X to δ_Y.

If $\delta_Y \leqslant \delta_X$ (respectively, $\delta_Y \leqslant_t \delta_X$) as well, we say that δ_X and δ_Y are computably equivalent (continuously equivalent), in symbols $\delta_X \equiv \delta_Y$ ($\delta_X \equiv_t \delta_Y$).

Definition I.2.12 ([Sch21, Definition 3.4])**.** Let (X, τ) be a topological space. A representation map δ_X of X is *admissible with respect to* τ if it is continuous and for all continuous representations $\dot{\delta}_X$ of X, $\dot{\delta}_X \leqslant_t \delta_X$.

In case there is no ambiguity on the topology we just say that a representation is admissible.

Notice that every represented space can be endowed with a topology induced by the representation map, i.e. the final topology.

Definition I.2.13. Given a represented space $\mathbf{X}$ the final topology, denoted with $\mathcal{O}(\mathbf{X})$, is the finest topology making δ_X continuous and is defined as

$$\mathcal{O}(\mathbf{X}) := \{U \subseteq X : (\exists V \subseteq \mathbb{N}^{\mathbb{N}})(V \text{ is open and } \delta_X^{-1}(U) = V \cap \operatorname{dom}(\delta_X))\}.$$

In §I.1 we introduced trees and graphs. We denote by $\mathbf{Tr}$ and $\mathbf{Tr}_2$ the represented spaces of trees on $\mathbb{N}$ and binary trees respectively: for both the representation map is the characteristic function. For graphs, we use two different representations, but we postpone their definitions to §III.2. Additional results about represented spaces are given in §I.3.2, in which we study represented spaces arising from descriptive set theory.

I.3 Descriptive set theory

Descriptive set theory is the area of mathematical logic that studies "definable" sets in topological spaces, in particular in *Polish spaces*. Thanks to its different applications to different mathematical fields, even outside mathematical logic (e.g., combinatorics, topology, analysis etc. . .), descriptive set theory has become one of the main areas of research of set theory and mathematical logic in general. Roughly speaking, we can divide descriptive set theory into three main areas tightly connected to each other:

(*i*) the study of *regularity properties* that, informally, are properties that "well behaved" sets should have. Using the Axiom of Choice it is possible to define pathological subsets of $\mathbb{R}$ that do not have such properties, hence, researchers started a systematic study of classes of sets in which these pathologies can be avoided. The study of the interconnections between the topological properties of sets and the complexity of their definitions gives rise to the *boldface hierarchy* that consists of the *Borel hierarchy* and the *projective hierarchy.*

(*ii*) The study of *reducibilities between sets*, in which sets are compared with respect to some notion of reducibility: informally, a set is reducible to another if it is "simpler" (with respect to the corresponding reduction). This comparison gives rise to an order (i.e. a hierarchy) between sets and, in this context, the most studied reduction is *Wadge reducibility.*

(*iii*) *Invariant descriptive set theory* and in particular the area that studies *definable equivalence relations* to classify the complexity of mathematical problems. Indeed, many mathematical problems can be naturally represented via some equivalence relation on a set (we call them *classification problems*), and we can compare them via some suitable notion of reducibility: the most studied is *Borel reducibility*, but in this thesis, we are more interested in *continuous reducibility.*

Notice that all the areas described above have their effective analogs: this chapter, for (i) and (ii), considers them as well.

We postpone the results on (iii) to §I.4, and now we give the necessary preliminaries about (i), (ii) and their effective analogs. For these topics, we refer the reader to two standard textbooks, namely,

1. Kechris' book "Classical Descriptive Set Theory" ([Kec12]),
2. Moschovakis' book "Effective Descriptive Set Theory" ([Mos82]),

and to some unpublished notes by Louveau ([Lou17]).

As already mentioned, descriptive set theory is mostly developed in Polish spaces. We say that (X, τ) is *completely metrizable* if there exists a metric d on X such that (X, d) is complete, i.e. every Cauchy sequence of elements of X converges in X.

Definition I.3.1 ([Kec12, Definition 3.1])**.** A Polish space is a separable completely metrizable space.

Sometimes we also refer to *computable Polish spaces*, where a computable Polish space is a computable metric space $\mathcal{X} = (X, d, \alpha)$ (see Definition I.2.10) such that the metric d is complete.

Two of the most fundamental Polish spaces in descriptive set theory have already been introduced in §I.1, and they are the Baire space ($\mathbb{N}^{\mathbb{N}}$) and the Cantor space ($2^{\mathbb{N}}$). The following is a well known and useful characterization of their closed subsets.

Theorem I.3.2 ([Kec12, Proposition 2.4])**.** *A set $F \subseteq \mathbb{N}^{\mathbb{N}}$ (respectively, $2^{\mathbb{N}}$) is closed if and only if there is a tree $T \subseteq \mathbb{N}^{<\mathbb{N}}$ ($2^{<\mathbb{N}}$) such that $F = [T]$.*

The next theorem, called "transfer theorem" in [Mos82] suggests a useful property of $\mathbb{N}^{\mathbb{N}}$: in a sense made precise below, we can transfer every Polish space in $\mathbb{N}^{\mathbb{N}}$.

Theorem I.3.3 ([Kec12, Theorem 7.9], [Mos82, Theorem 1G.2])**.** *For every Polish space X there is a closed set $A \subseteq \mathbb{N}^{\mathbb{N}}$ and a continuous bijection $s : A \to X$. If X is nonempty then s extends to a continuous surjection from $\mathbb{N}^{\mathbb{N}}$ to X.*

We say that a subset of a topological space X is G_δ if it is the countable intersection of open sets of X.

Theorem I.3.4 ([Kec12, Exercise 3.12])**.** *The Baire space is homeomorphic to a G_δ subspace of $2^{\mathbb{N}}$.*

In general, if P is a subset of a topological space $\mathcal{X}$, a point $x \in P$ is a *limit point of* P if for every open set U with $x \in U$ there is a distinct point $y \in P \cap U$: otherwise, we call x *isolated in* P. A subset of a topological space is *perfect* if it is closed and has no isolated points. Notice that every nonempty perfect subset of a Polish space has the cardinality of the continuum. The following theorem informally says that the Cantor space witnesses if a nonempty Polish space is perfect or not.

Theorem I.3.5 ([Kec12, Theorem 6.2])**.** *For every nonempty perfect Polish space X there is an embedding of $2^{\mathbb{N}}$ into X.*

Combining Theorem I.3.5 and Theorem I.3.4 we obtain that every uncountable Polish space contains a G_δ subspace homeomorphic to $\mathbb{N}^{\mathbb{N}}$. We cannot replace "G_δ" with "closed" as $\mathbb{N}^{\mathbb{N}}$ is not compact (a subset of a topological space X is *compact* if every open cover of X has a finite subcover), however, we have the following useful fact due to Hurewicz[1]. We say that a set A in a topological space X is K_σ if $A = \bigcup_{n\in\mathbb{N}} K_n$ where K_n is a compact subset of X ([Kec12, Definition 5.2]).

Theorem I.3.6 ([Kec12, Theorem 7.10])**.** *Let X be a Polish space. Then X contains a closed subspace homeomorphic to $\mathbb{N}^\mathbb{N}$ if and only if X is not K_σ.*

We conclude this section, with the well known *Cantor-Bendixson theorem.* This theorem, beyond being a celebrated and important result in descriptive set theory, is also the main topic of Chapter II.

Theorem I.3.7 ([Kec12, Theorem 6.4])**.** *Every closed subset C of a Polish space X can be uniquely written as the disjoint union of a perfect set P and a countable set S. We call P the perfect kernel of C and S the scattered part of C.*

When $X = 2^\mathbb{N}$ or $X = \mathbb{N}^\mathbb{N}$, by Theorem I.3.2, we can rephrase the Cantor-Bendixson theorem saying that all but countably many paths through a tree $T \subseteq \mathbb{N}^{<\mathbb{N}}$ (respectively, $T \subseteq 2^{<\mathbb{N}}$) belong to a perfect subtree S of T. Again, S (which is unique) is called the *perfect kernel* of T, and the set of missing paths is the *scattered part* of T.

Theorem I.3.7 can be restated also in another form, and to do so, we first need to define the *Cantor-Bendixson derivative* for a topological space X.

Definition I.3.8 ([Kec12, Definition 6.10])**.** For any topological space X, let $X' := \{x \in X : x \text{ is a limit point of } X\}$ be the Cantor-Bendixson derivative of X. Using transfinite recursion we define the iterated Cantor-Bendixson derivatives X^α where α is an ordinal as follows:

- $X^0 := X$,
- $X^{\alpha+1} := (X^\alpha)'$,
- $X^\lambda := \bigcap_{\alpha<\lambda} X^\alpha$, if λ is limit.

Definition I.3.8 gives us also another formulation of being perfect: namely a subset $P \subseteq X$ is perfect if and only if $P = P'$. We now restate Theorem I.3.7.

Theorem I.3.9 ([Kec12, Theorem 6.11])**.** *Let X be a Polish space. For some ordinal $\alpha_0 < \omega_1$, $X^\alpha = X^{\alpha_0}$ for all $\alpha \geqslant \alpha_0$ and X^{α_0} is the perfect kernel of X and $X \backslash X^{\alpha_0}$ is the scattered part of X.*

For a Polish space X, we call the α_0 in Theorem I.3.9 the *Cantor-Bendixson rank* of X. Notice that X is countable if and only if $X^{\alpha_0} = \varnothing$. These notions play an important role in Chapter V, in which we define an operation strictly related to the Cantor-Bendixson derivative.

I.3.1 Boldface hierarchy

The σ-algebra on a set X is a nonempty collection of subsets of X that is closed under complement, countable unions, and countable intersection. For a topological space (X, τ), we denote with $\mathbf{Bor}(X)$ the *Borel subsets* of X, and we define them as the smallest σ-algebra containing the open sets.

Borel subsets of topological spaces can be stratified accordingly to their topological complexity: this gives rise to the *Borel hierarchy.* Usually, the classical definition of boldface hierarchy (see for

[1]Notice that this is a special case of a more general statement, see [Kec12, Theorem 21.18].

example [Kec12, §11.B]) is given for Polish spaces: the definition we use here extends to topological spaces in general. This is achieved by slightly modifying the classical definition of $\mathbf{\Sigma}^0_\xi(X)$ sets so that we handle the case of non-Hausdorff spaces, where an open set is not always the union of closed sets[2]. Notice that, in case the topological space is Hausdorff, the definition we use here and the classical one coincide [CH20, §2.1.1].

Let ω_1 be the first uncountable ordinal: for any $1 \leqslant \xi < \omega_1$, we define by transfinite recursion the Borel hierarchy on X as follows. We define $\mathbf{\Sigma}^0_1(X) := \{A : A \text{ is open}\}$ and, for $\xi > 1$,

- $\mathbf{\Sigma}^0_\xi(X) := \left\{\bigcup_{n\in\mathbb{N}} A_n \backslash B_n : n \in \mathbb{N} \wedge A_n, B_n \in \mathbf{\Sigma}^0_{\xi_n}(X) \wedge \xi_n < \xi\right\}$,
- $\mathbf{\Pi}^0_\xi(X) := \{X\backslash A : A \in \mathbf{\Sigma}^0_\xi(X)\}$.

In case X is Polish and $\xi = n$ for some natural number $n > 1$, the first item, as in [Mos82, §1B], can be rewritten as

$$\mathbf{\Sigma}^0_n(X) := \{A \subseteq X : (\exists Q \subseteq \mathbf{\Pi}^0_n(X \times \mathbb{N}))(A = \{x \in X : (\exists n)((x, n) \in Q)\})\}.$$

Finally, for $\xi > 0$, we define the *ambiguous* classes of X as $\mathbf{\Delta}^0_\xi(X) := \mathbf{\Pi}^0_\xi(X) \cap \mathbf{\Sigma}^0_\xi(X)$.

By the definitions, it is easy to observe that $\mathbf{\Pi}^0_1(X)$ is the class of closed sets of X, $\mathbf{\Delta}^0_1(X)$ are the clopen ones. Notice also that the class $\mathbf{\Sigma}^0_2(X)$ (countable unions of closed subsets of X) is also denoted with $F_\sigma(X)$, while $\mathbf{\Pi}^0_2(X)$ (countable intersection of open subsets of X) is what before Theorem $I.3.4$ we denoted with $G_\delta(X)$. From now on, in case the result does not depend on the space, we just write $\mathbf{\Sigma}^0_n$, and we refer to it as a *boldface class* (or simply *class*) letting $\mathbf{\Sigma}^0_n := \{A : A \in \mathbf{\Sigma}^0_n(X) \text{ for some space } X\}$ (similarly for $\mathbf{\Pi}^0_n$ and $\mathbf{\Delta}^0_n$).

The next theorem states that every Borel set of a space X can be obtained, starting from open sets, iterating the operations of complement, countable union, and countable intersection for less than ω_1 steps i.e. is in one of the classes defined above.

Theorem I.3.10 ([Kec12, §11.B, Theorem 22.4])**.** *Given a space X and $\xi > 0$,*

$$\mathbf{Bor}(X) = \bigcup_{\xi<\omega_1} \mathbf{\Sigma}^0_\xi = \bigcup_{\xi<\omega_1} \mathbf{\Pi}^0_\xi = \bigcup_{\xi<\omega_1} \mathbf{\Delta}^0_\xi.$$

If X is Polish and uncountable, the hierarchy does not collapse at any level $\xi < \omega_1$.

Borel classes have nice closure properties and some of them are summarized in the next theorem.

Theorem I.3.11. *For $\xi < \omega_1$, the classes $\mathbf{\Sigma}^0_\xi$, $\mathbf{\Pi}^0_\xi$ and $\mathbf{\Delta}^0_\xi$ are closed under finite unions, finite intersections, and continuous preimages. Furthermore, $\mathbf{\Sigma}^0_\xi$ is closed under countable unions, $\mathbf{\Pi}^0_\xi$ is closed under countable intersections and $\mathbf{\Delta}^0_\xi$ is closed under complement.*

We continue defining the *projective hierarchy.* The Borel hierarchy was defined for topological spaces in general: here we restrict our attention to Polish spaces. The first levels of the projective hierarchy play a major role in this thesis and in descriptive set theory in general: they are the *analytic* sets (denoted by $\mathbf{\Sigma}^1_1$) and the *co-analytic* sets (denoted by $\mathbf{\Pi}^1_1$).

Definition I.3.12. Given a Polish space X, a set $A \subseteq X$ is called analytic (denoted by $A \in \mathbf{\Sigma}^1_1(X)$) if there is a Polish space Y and a continuous function $f : Y \to X$ such that $\mathsf{range}(f) = A$. The complement of an analytic set is called co-analytic and the family of co-analytic sets of X is denoted by $\mathbf{\Pi}^1_1(X)$.

The following proposition collects some useful characterizations of analytic subsets of a Polish space X that are used extensively in the next chapters.

[2] Recall that a space is *Hausdorff* if every two different points of X have disjoint open neighborhoods.

Proposition I.3.13 ([Kec12, Exercise 14.3]). *Let X be Polish and let $A \subseteq X$ nonempty. Then, A is analytic if and only if there is a Polish space Y and some $Q \in \mathbf{Bor}(X \times Y)$ such that $A = \{x \in X : (\exists y \in Y)((x, y) \in Q)\}$;*

By Theorems $I.3.3$ and $I.3.4$, in the proposition above, we can replace Y with $\mathbb{N}^{\mathbb{N}}$ and assume $Q \in \mathbf{\Pi}^0_1(X \times Y)$, or we can replace Y with $2^{\mathbb{N}}$ and assume $Q \in \mathbf{\Pi}^0_2(X \times Y)$.

In Theorem $I.3.2$ we gave a characterization of closed subsets of Baire space in terms of trees: now we give a tree characterization of (co-)analytic subsets of Baire space[3].

Theorem I.3.14 ([Kec12, §25.2], [Mos82, essentially Theorem 4A.1]). *Given $A \subseteq \mathbb{N}^{\mathbb{N}}$,*

$$A \text{ is analytic} \iff (\exists T \subseteq \mathbb{N}^{<\mathbb{N}} \times \mathbb{N}^{<\mathbb{N}})(A = \{x : (\exists y)((x, y) \in [T])\}),$$

$$A \text{ is co-analytic} \iff (\exists T \subseteq \mathbb{N}^{<\mathbb{N}} \times \mathbb{N}^{<\mathbb{N}})(A = \{x : (\forall y)((x, y) \notin [T])\}).$$

Given a Polish space X and $n > 0$, we now define the *projective hierarchy* on X as follows:

- $\mathbf{\Sigma}^1_{n+1}(X) := \{A \subseteq X : (\exists Q \subseteq \mathbf{\Pi}^1_n(X \times \mathbb{N}^{\mathbb{N}}))(A = \{x \in X : (\exists y \in \mathbb{N}^{\mathbb{N}})((x, y) \in Q)\})\}$;
- $\mathbf{\Pi}^1_{n+1}(X) := \{X \backslash A : A \in \mathbf{\Sigma}^1_n(X)\}$.

As for the Borel hierarchy, we define the *ambiguous projective classes* $\mathbf{\Delta}^1_n(X) := \mathbf{\Sigma}^1_n(X) \cap \mathbf{\Pi}^1_n(X)$ and, in case the result we mention is independent of the space, we drop the X. The following is an analog of Theorem $I.3.10$.

Proposition I.3.15. *The classes $\mathbf{\Sigma}^1_n$, $\mathbf{\Pi}^1_n$, and $\mathbf{\Delta}^1_n$ are closed under countable intersections, countable unions, and continuous preimages. Furthermore, $\mathbf{\Sigma}^1_n$ is closed under continuous images (in particular, projections, i.e., existential quantification over Polish spaces), $\mathbf{\Pi}^1_n$ is closed under co-projections (i.e., universal quantification over Polish spaces) and $\mathbf{\Delta}^1_n$ is closed under complements.*

The following well known theorem is due to Suslin.

Theorem I.3.16 ([Lou17, Theorem 14.11 and Corollary 26.2]). *For every uncountable Polish space X,*

$$\mathbf{Bor}(X) = \mathbf{\Delta}^1_1(X) \subset \mathbf{\Sigma}^1_1(X).$$

We now state the *uniformization problem.* Given two sets X, Y and $P \subseteq X \times Y$, a *uniformization* of P is a subset $P^* \subseteq P$ such that for all $x \in X$, $(\exists y)(P(x, y)) \iff (\exists! y)(P^*(x, y))$. In other words, P^* is the graph of a function f with domain $\{x : (\exists y)(P(x, y))\}$ such that $f(x) \in \{y : P(x, y)\}$ for every $x \in A$. Using the axiom of choice we can uniformize any P by some P^*; on the other hand, it frequently occurs that, given a definable set P, obtaining a definable set P^* is a difficult task. We say that a boldface class $\mathbf{\Gamma}$ has the *uniformization property* if every $P \subseteq \mathbf{\Gamma}(X \times Y)$ can be uniformized by some P^* in $\mathbf{\Gamma}$. The following theorem is also known as the Novikov-Kondo-Addison uniformization theorem and is very useful in proving Proposition $I.6.28$.

Theorem I.3.17 ([Mos82, Theorem 4E.4]). *The class $\mathbf{\Pi}^1_1$ has the uniformization property.*

[3] While Theorem $I.3.2$ characterizes also closed subsets of $2^{\mathbb{N}}$, Theorem $I.3.14$ does not characterize analytic subsets of $2^{\mathbb{N}}$: the reason resides in the complexity of ill-foundedness for binary trees, but we discuss this later (see Theorem $I.3.39$).

I.3.2 Effective Descriptive Set Theory

In this section, we introduce the effective counterpart of descriptive set theory. Here sets are not classified with respect to their topological properties, but with respect to their computability-theoretic properties, and this classification yields the *lightface hierarchy*, also known as *Kleene's hierarchy*. We introduce it starting from the *Kleene arithmetical hierarchy* (the effective counterpart of the finite levels of the Borel hierarchy), and we skip the definition of the *Kleene hyperaritmetical hierarchy* (the effective counterpart of the transfinite levels of the Borel hierarchy). This choice is justified by the fact that its definition is a bit technical and in this thesis we do not deal with the Kleene hyperarithmetical hierarchy: the interested reader is referred to [Lou17, Section 5.2] where such a hierarchy is defined via Borel codes. We define the Kleene arithmetical hierarchy for *effective second-countable spaces*: its definition is from [Lou17, §2.3.1] but notice that the author refers to such spaces as *basic spaces*.

Definition I.3.18. An effective second-countable space is a pair $(X, (B_n^X)_{n\in\mathbb{N}})$ where

- X is a second-countable space (i.e. it admits a countable base),
- $(B_n^X)_{n\in\mathbb{N}}$ is the *effective basis*, i.e. an enumeration of a countable basis of the topology of X such that there is a computably enumerable relation $R \in \mathbb{N}^3$ satisfying

$$x \in B_n^X \cap B_m^X \iff (\exists i)(x \in B_i^X \wedge R(m, n, i)).$$

Given an effective second-countable space $(X, (B_n^X)_{n\in\mathbb{N}})$ if $Y \subset X$ comes with the induced topology, for every n, given $B_n^Y := B_n^X \cap Y$, we have that $(Y, (B_n^Y)_{n\in\mathbb{N}})$ is an effective second-countable space as well (see [Lou17, §2.3.1]). Furthermore, given a countable sequence of effective second-countable spaces we can define an effective basis for their product (see [Lou17, Proposition 2.3.1]).

It is easy to verify that any computable metric space (X, d, α) (see Definition I.2.10) is also effectively second-countable: furthermore, we also have a canonical choice for an effective basis, letting $B_{\langle n,m\rangle} := B(\alpha(n), q_m)$ (recall that q_m is the m-th element of the computable enumeration of $\mathbb{Q}$). The following remark is essentially from [VM21, §1.2.1].

Remark I.3.19. Notice that usually effective descriptive set theory is developed in the context of *recursively presented metric spaces* (see for example [Lou17, 2.2.1] and [Mos82, §3B]). Given a separable metric space $\mathcal{X} = (X, d, \alpha)$, where d is a distance function and $\alpha : \mathbb{N} \to X$ is a dense sequence in X, we say that α is a *recursive presentation of* X if the following relations are computable:

$$P^{d,X}(i, j, k) :\iff d(\alpha(i), \alpha(j)) \leqslant q_k \text{ and } Q^{d,X}(i, j, k) :\iff d(\alpha(i), \alpha(j)) < q_k.$$

Being a recursively presented metric space is a strictly stronger condition than being a computable metric space (see [GKP17, Observation 2.4 and Exercise 2.5]). Despite this, given a computable metric space, $X = (X, d, \alpha)$ there is a computable real $\beta \leqslant 1$ such that the computable metric space $X' := (X, \beta d, \alpha)$ is a recursively presented metric space, and there is a computable bijection $X \mapsto X'$ with computable inverse (see [GKP17, Theorem 2.10]).

For an effective second countable space $(X, (B_n)_{n\in\mathbb{N}})$, we define the Kleene arithmetical hierarchy as follows. Starting from $\Sigma_1^0(X) := \{A \subseteq X : A = \bigcup_{i\in\mathbb{N}} B_{\varphi(i)}$ for some computable $\varphi : \mathbb{N} \to \mathbb{N}\}$, for $n > 0$, and given an effective indexing $(A_{\langle n,i\rangle})_{i\in\mathbb{N}}$ of the $\Sigma_n^0(X)$ classes,

- $\Sigma_{n+1}^0(X) := \{A \subseteq X : \bigcup_{k\in\mathbb{N}} A_{\langle n,\varphi(2k+1)\rangle} \setminus A_{\langle n,\varphi(2k)\rangle}$ for some computable $\varphi : \mathbb{N} \to \mathbb{N}\}$;
- $\Pi_{n+1}^0(X) := \{X \setminus A : A \in \Sigma_{n+1}^0(X)\}$.

Notice that if X is a computable metric space, the class $\Sigma_n^0(X)$ can be also defined as follows:

$$\Sigma_n^0(X) := \{A \subseteq X : (\exists Q \in \Pi_n^0(X \times \mathbb{N}))(A = \{x \in X : (\exists n)((x, n) \in Q)\})\}.$$

We usually refer to $\Sigma^0_1(X)$ and $\Pi^0_1(X)$ as *effectively open* and *effectively closed*, and we define the ambiguous classes as $\Delta^0_n(X) := \Sigma^0_n(X) \cap \Pi^0_n(X)$.

Remark I.3.20. The choice of the terms "effectively open" and "effectively closed" is just to highlight the connection to the boldface classes $\boldsymbol{\Sigma}^0_1(X)$ and $\boldsymbol{\Pi}^0_1(X)$. Indeed, $\Sigma^0_1(X)$ is not always a topology: in other words, the lightface classes are not topological properties, in the sense that they heavily depend on the space X and on the specific basis we have chosen.

We continue defining the *effective projective hierarchy*, (also known as *Kleene's analytical hierarchy*) for computable metric spaces. The first levels are:

- $\Sigma^1_1(X) := \{A \subseteq X : (\exists Q \in \Pi^0_1(X \times \mathbb{N}^\mathbb{N}))(A = \{x \in X : (\exists y \in \mathbb{N}^\mathbb{N})(x, y) \in Q\})\}$;
- $\Pi^1_1(X) := \{X \backslash A : A \in \Sigma^1_1(X)\}$.

In general:

- $\Sigma^1_n(X) := \{A \subseteq X : (\exists Q \in \Pi^0_n(X \times \mathbb{N}^\mathbb{N}))(A = \{x \in X : (\exists y \in \mathbb{N}^\mathbb{N})(x, y) \in Q\})\}$;
- $\Pi^1_n(X) := \{X \backslash A : A \in \Sigma^1_n(X)\}$;

and the ambiguous classes are defined as $\Delta^1_n(X) := \Sigma^1_n(X) \cap \Pi^1_n(X)$. In analogy to the boldface hierarchy, the $\Sigma^1_1(X)$ sets are defined as *lightface analytic* and $\Pi^1_1(X)$ as *lightface co-analytic* and, in case the results we mention are independent of the space, we just write Σ^i_n for $i < 2$, and we refer to it as a *lightface class*, similarly for Π^i_n and Δ^i_n.

Theorem I.3.21 ([Mos82, Corrolary 3E.2, Theorem 3G.2])**.** *The lightface (projective) classes are closed under finite union, intersection, and computable preimages. Furthermore,*

- Σ^0_ξ *is closed under computable unions and* Π^0_ξ *is closed under computable intersection;*
- Σ^1_n *is closed under projections on* $\mathbb{N}^\mathbb{N}$ *and* Π^1_n *is closed under co-projections on* $\mathbb{N}^\mathbb{N}$.
- Δ^0_ξ *and* Δ^1_n *are closed under complements.*

Notice that the definitions of uniformization of a set and of uniformization property for boldface classes given in §I.3 hold also here. This is the effective analog of the Kondo-Addison-Novikov uniformization theorem given in the previous section.

Theorem I.3.22 ([Mos82, Theorem 3E.4])**.** *The class* Π^1_1 *has the uniformization property.*

We introduce the following notation. Let X and Y be effective second-countable spaces and Γ be a lightface class. Given $y \in Y$ we define the *relativization of* Γ *to* y, denoted with Γ^y, so that $A \in \Gamma^y(X)$ if there exists some $B \subseteq Y \times X$ such that $x \in X \iff (y, x) \in Y \times X$.

The following is the *effective perfect set theorem* by Harrison: we can also interpret it as the fact that countable Σ^1_1 sets are actually simpler.

Theorem I.3.23 ([Mos82, 4F.1])**.** *Let* X *be a computable metric space and let* $P \in \Sigma^1_1(X)$*. Then either* P *has only* Δ^1_1 *points, or* P *has nonempty perfect subset. In particular, if* $P \in \Sigma^1_1(X)$ *and* $|P| \leqslant \aleph_0$*, then* $P \subseteq \Delta^1_1(X)$*. The same holds in the relativized version.*

Normal forms

We now provide useful normal forms for lightface and boldface classes.

Theorem I.3.24 ([Mos82, §3E.3]). *Let Y be a Polish space and $n \geqslant 1$: a subset $P \subseteq Y$ is $\Sigma^0_n(Y)$ if and only if there is some $S \in \Gamma(Y \times \bigtimes_{i<n} \mathbb{N})$ such that*

$$x \in P \iff (\exists t_1)(\forall t_2)(\exists t_3)(\forall t_4)\ldots(\forall t_{n-1})((x, t_1, \ldots, t_{n-1}) \in S), \tag{1}$$

where $\Gamma = \Sigma^0_1$ if n is odd and $\Gamma = \Pi^0_1$ otherwise.

If $Y = \bigtimes_{i<n} X_i$ where either $X_i = \mathbb{N}$ or $\mathbb{N}^{\mathbb{N}}$, then $P \subseteq Y$ is $\Sigma^0_n(Y)$ if and only if there is a computable set $R \subseteq Y \times \bigtimes_{i\leqslant n} \mathbb{N}$ such that

$$x \in P \iff (\exists t_1)(\forall t_2)(\exists t_3)(\forall t_4)\ldots(\mathsf{Q} t_n)((x, t_1, \ldots, t_n) \in R). \tag{2}$$

where $\mathsf{Q} = \exists$ if n is odd and $\mathsf{Q} = \forall$ otherwise.

Similar normal forms hold for $\Pi^0_n(Y)$ classes: here, the alternations of the quantifiers start with $\forall$ and in (1) $\Gamma = \Pi^0_1$ if n is odd and $\Gamma = \Sigma^0_1$ otherwise; in (2) $\mathsf{Q} = \forall$ if n is odd and $\mathsf{Q} = \exists$ otherwise.

The following generalizes Theorem I.3.24 to the lightface projective classes.

Theorem I.3.25 ([Mos82, §3E.3]). *Let X be a Polish space and $n \geqslant 1$: a subset $P \subseteq X$ is $\Sigma^1_n(X)$ if and only if there is some $S \in \Sigma^0_1(X \times \times_{i<n} \mathbb{N}^{\mathbb{N}})$ such that*

$$x \in P \iff (\exists y_1)(\forall y_2)(\exists y_3)(\forall y_4)\ldots(\mathsf{Q} y_n)((x, y_1, \ldots, y_n) \in S),$$

where $\mathsf{Q} = \exists$ if n is odd and $\mathsf{Q} = \forall$ otherwise. We can assume S to be computable if $X = \times_{i<n} X_i$ where either $X_i = \mathbb{N}$ or $\mathbb{N}^{\mathbb{N}}$. Similar normal forms hold for $\Pi^0_n(X)$ classes: here, the alternations of the quantifiers start with $\forall$ and in (2) $\mathsf{Q} = \forall$ if n is odd and $\mathsf{Q} = \exists$ otherwise.

The following theorem connects the lightface and boldface classes.

Theorem I.3.26 (essentially [Mos82, 3E.4]). *Given a second-countable space X, $A \in \mathbf{\Gamma}(X)$ if and only if there exists $p \in \mathbb{N}^{\mathbb{N}}$ such that $A \in \Gamma^p$.*

In particular, Theorems I.3.24 and I.3.25 hold also in the relativized form, and so we can easily derive a normal form also for the boldface classes (see also [Mos82, Exercise 1B.6] in which the normal forms are given directly for the boldface classes).

Represented spaces and descriptive set theory

Given a separable metric space $\mathcal{X} = (X, d, \alpha)$ and $k > 0$, using the inductive definition of Borel sets, we can define the represented spaces $\mathbf{\Sigma}^0_k(\mathcal{X})$, $\mathbf{\Pi}^0_k(\mathcal{X})$ and $\mathbf{\Delta}^0_k(\mathcal{X})$: this shows that the Borel classes can be naturally considered as represented spaces. Recall that in §I.2 after Definition I.2.10 we have fixed a sequence $(B_i)_{i\in\mathbb{N}}$ of all basic open sets of $\mathcal{X}$.

Definition I.3.27 ([Bra04, Definition 3.1]). For any separable metric space $\mathcal{X} = (X, d, \alpha)$ and for any $k > 0$, the represented spaces $(\mathbf{\Sigma}^0_k(\mathcal{X}), \delta_{\mathbf{\Sigma}^0_k(\mathcal{X})})$, $(\mathbf{\Pi}^0_k(\mathcal{X}), \delta_{\mathbf{\Pi}^0_k(\mathcal{X})})$ and $(\mathbf{\Delta}^0_k(\mathcal{X}), \delta_{\mathbf{\Delta}^0_k(\mathcal{X})})$ are defined inductively as follows:

- $\delta_{\mathbf{\Sigma}^0_1(\mathcal{X})}(p) := \bigcup_{i\in\mathbb{N}} B_{p(i)}$;
- $\delta_{\mathbf{\Pi}^0_k(\mathcal{X})} := X \setminus \delta_{\mathbf{\Sigma}^0_k(\mathcal{X})}(p)$;
- $\delta_{\mathbf{\Sigma}^0_{k+1}(\mathcal{X})}(p_0 \oplus p_1 \oplus \ldots) := \bigcup_{i\in\mathbb{N}} \delta_{\mathbf{\Pi}^0_k(\mathcal{X})}(p_i)$.

It is worth noticing that a name for a closed set is in fact a name for its complement: that is a name for a closed set is a list of open balls whose union is its complement. This representation of $\mathbf{\Pi}^0_1(\mathcal{X})$ is also known as *negative representation of closed sets* of $\mathcal{X}$ and, as usual in the literature, we denote this represented space by $\mathcal{A}_-(\mathcal{X})$. We can represent the class $\mathbf{\Sigma}^1_1(\mathcal{X})$ of analytic subsets of $\mathcal{X}$ by defining a name for $A \in \mathbf{\Sigma}^1_1(\mathcal{X})$ as a name for a closed set $C \subseteq \mathcal{X} \times \mathbb{N}^\mathbb{N}$ such that $A := \{x : (\exists y)((x,y) \in C)\}$. Then, a name for a coanalytic set $B \in \mathbf{\Pi}^1_1(\mathcal{X})$ is just a name for its complement. Recall from §I.2 that $\mathbf{Tr}$ and $\mathbf{Tr}_2$ denote respectively the represented spaces of trees on $\mathbb{N}$ and binary trees where the representation map is, in both cases, the characteristic function. In case $\mathcal{X} = \mathbb{N}^\mathbb{N}$, by Theorem *I*.3.2, there exists a surjective function $[\cdot] : \mathbf{Tr} \to \mathcal{A}_-(\mathbb{N}^\mathbb{N})$ defined by $T \mapsto [T]$ that is computable with multi-valued computable inverse: the same holds if $X = 2^\mathbb{N}$ considering $\mathbf{Tr}_2$ instead of $\mathbf{Tr}$. This means that the negative representation of a closed subset C of $\mathbb{N}^\mathbb{N}$ (respectively $2^\mathbb{N}$) is equivalent (in the sense of Definition *I*.2.11) to the one given by the characteristic function of a (binary) tree T such that $[T] = C$. We refer to the latter representation as the *tree representation.* By Theorem *I*.3.14 we obtain a similar result for $\mathbf{\Sigma}^1_1(\mathbb{N}^\mathbb{N})$: namely we can define a name for $A \in \mathbf{\Sigma}^1_1(\mathcal{X})$ as a name for a tree $T \subseteq \mathbb{N}^{<\mathbb{N}} \times \mathbb{N}^{<\mathbb{N}}$ such that $A = \{x : (\exists y)((x,y) \in [T])\}$ (similarly for $\mathbf{\Pi}^1_1(\mathbb{N}^\mathbb{N})$).

Collecting results that can be found in [Bra04, §3], we obtain the following proposition.

Proposition I.3.28. *Let $\mathcal{X} = (X, d, \alpha)$ be a separable metric space. For every $A \subseteq X$,*

$$A \in \Sigma^0_k(Y) \iff A \in \mathbf{\Sigma}^0_k(Y) \text{ and } A \text{ has a computable } \delta_{\mathbf{\Sigma}^0_k(Y)}\text{-name.}$$

We take the chance to introduce another (represented) space, namely the *Sierpiński space* $\mathbb{S} := \{0_\mathbb{S}, 1_\mathbb{S}\}$ endowed with the topology $\{\varnothing, \{1_\mathbb{S}\}, 0_\mathbb{S}\}$. Usually, $0_\mathbb{S}$ and $1_\mathbb{S}$ are denoted respectively with 0 (or $\bot$) and 1 (or $\top$), but we prefer to use this notation to avoid any confusion in the next chapters. The representation map we associate to $\mathbb{S}$ is the following:

$$\delta_\mathbb{S}(p) := \begin{cases} 1_\mathbb{S} & \text{if } (\exists i)(p(i) \neq 0), \\ 0_\mathbb{S} & \text{if } p = 0^\mathbb{N}. \end{cases}$$

The reason why we introduce now such a space is that we could have defined the sets in Definition *I*.3.27 using it. Indeed, given a represented space $\mathbf{X}$ and given the final topology $\mathcal{O}(\mathbf{X})$ on X induced by δ_X (see Definition *I*.2.13), we have that $U \in \mathcal{O}(\mathbf{X})$ if and only if the characteristic function $\chi_U : \mathbf{X} \to \mathbb{S}$ is realizer-continuous (see Definition *I*.2.8 and [Pau16, §4]). Hence, we can represent $U \in \mathcal{O}(\mathbf{X})$ by χ_U: then using the jumps of the Sierpiński space we can define all the sets in Definition *I*.3.27 as promised (the details of this are omitted, but we refer the reader to [Bra04, Pau16]). Similar ideas apply also to the lightface case: namely, for a represented space $\mathbf{X}$ we have that $U \in \Sigma^0_1(\mathbf{X})$ if and only if the characteristic function $\chi_U : \mathbf{X} \to \mathbb{S}$ is computable. As for the boldface case, the jumps of the Sierpiński space allow us to obtain the lightface classes as defined in Proposition *I*.3.28.

I.3.3 (Effective) Wadge reducibility

We now introduce a notion of complexity between sets of topological spaces, namely *Wadge reducibility* and its effective counterpart.

Definition I.3.29 ([Kec12, Definition 21.13])**.** Let X, Y be topological spaces and let $A \subseteq X$, $B \subseteq Y$. We say that A is *Wadge reducible* to B (in symbols, $A \leqslant_\mathbf{W} B$) if there is a continuous function $f : X \to Y$ such that $x \in A \iff f(x) \in B$.

Informally, if $A \leqslant_\mathbf{W} B$ it means that A is "simpler" than B. Notice that $\leqslant_\mathbf{W}$ is reflexive and transitive (hence, a quasi-order) and the corresponding equivalence classes are called the *Wadge degrees.*

Definition I.3.30 ([Kec12, Definition 22.9])**.** Let $\mathbf{\Gamma}$ be a boldface class and X and Y be Polish spaces with X being zero-dimensional.[a] We say that $B \subseteq Y$ $\mathbf{\Gamma}$-*hard* if, for any $A \in \mathbf{\Gamma}(X)$, $A \leqslant_{\mathbf{W}} B$: if in addition $B \in \mathbf{\Gamma}(Y)$, we say that B is $\mathbf{\Gamma}$-*complete.*

[a] A topological space X is *zero-dimensional* if it Hausdorff and has a basis consisting of clopen sets.

For any class of sets $\mathbf{\Gamma}$ let $\check{\mathbf{\Gamma}} := \{X \backslash A : A \in \mathbf{\Gamma}(X)\}$. Notice that while Definition $I.3.29$ is given for arbitrarily Polish spaces, Definition $I.3.30$ is restricted to Polish spaces where the space of the underlying set on the left-hand-side of the Wadge reduction is zero-dimensional. Note that if $\mathbf{\Gamma}$ is not an ambiguous class in a zero-dimensional Polish space and it is closed under continuous preimages, no $\mathbf{\Gamma}$-hard set is in $\check{\mathbf{\Gamma}}$.

Remark I.3.31. If B is $\mathbf{\Gamma}$-hard, then the complement of B is $\check{\mathbf{\Gamma}}$-hard. Furthermore, if B is $\mathbf{\Gamma}$-hard and $B \leqslant_{\mathbf{W}} A$, then A is $\mathbf{\Gamma}$-hard as well. All these considerations still hold if we replace hardness with completeness. This gives us a useful technique to show that a set A is $\mathbf{\Gamma}$-hard (respectively, $\mathbf{\Gamma}$-complete): "take an already known $\mathbf{\Gamma}$-hard ($\mathbf{\Gamma}$-complete) set B and show that $B \leqslant_{\mathbf{W}} A$".

Proposition I.3.32 ([Kec12, Exercise 22.11 and Exercise 24.20])**.** *Let X be a Polish space. For every $\xi \geqslant 1$ and every $A \subseteq X$,*

$$A \in \mathbf{\Sigma}^0_\xi \backslash \mathbf{\Pi}^0_\xi \iff A \text{ is } \mathbf{\Sigma}^0_\xi\text{-complete.}$$

The same statement holds interchanging $\mathbf{\Sigma}^0_\xi$ and $\mathbf{\Pi}^0_\xi$.

We mention that the previous proposition cannot be extended to $\mathbf{\Sigma}^1_1$-complete and $\mathbf{\Pi}^1_1$-complete sets, and the corresponding statement is equivalent to the principle of $\mathbf{\Sigma}^1_1$-determinacy (see [Kec12, Theorem 26.4]).

We now define the effective counterpart of Wadge reducibility: the following definition can be found for example in [CH20, §3].

Definition I.3.33. Let X and Y be two effective second-countable spaces. Given $A \subseteq X$ and $B \subseteq Y$ we say that A is *effectively Wadge reducible* to B (in symbols, $A \leqslant_{\mathsf{EW}} B$) if there is a computable $f : X \to Y$ such that $x \in A \iff f(x) \in B$. For a lightface class Γ, and an effective Polish space Y we say that $B \subseteq Y$ is Γ-*hard* if, for every $A \in \Gamma(\mathbb{N}^\mathbb{N})$, $A \leqslant_{\mathsf{EW}} B$. In case B is Γ-hard and $B \in \Gamma(Y)$, then we say that B is Γ-*complete.*

We now give some examples of complete sets that are useful in the next chapters.

Remark I.3.34. All the results in the remaining part of this section are stated for lightface classes and with respect to effective Wadge reducibility: on the other hand, all the results still hold replacing "lightface" with "boldface" and "effective Wadge" with "Wadge". These results can be found, sometimes with different terminology, for example in [Mos82, Kec12].

Lemma I.3.35. *The following classification results hold:*

(i) The set $P_1 := \{p \in 2^\mathbb{N} : (\exists i)(p(i) = 1)\}$ is Σ^0_1 complete;

(ii) The set $P_2 := \{p \in 2^\mathbb{N} : (\forall^\infty i)(p(i) = 1)\}$ is Σ^0_2 complete;

Proof. The fact that $\{p \in 2^{\mathbb{N}} : (\exists i)(p(i) = 1)\}$ is Σ^0_1-complete is straightforward, while the proof that $\{p \in 2^{\mathbb{N}} : (\forall^\infty i)(p(i) = 1)\}$ is Σ^0_2-complete can be found in [Kec12, Exercise 23.1] (the statement is given for the boldface case, but the same proof also shows the lightface case). □

We aim to generalize Lemma *I*.3.35, defining complete sets of (products of) $2^{\mathbb{N}}$ for all levels of the Kleene arithmetical hierarchy.

Theorem I.3.36. *Let $k > 0$. Then,*

$$P_{2k+1} := \{p \in 2^{\mathbb{N}^{k+1}} : (\exists n_0)(\exists^\infty n_1)\dots(\exists^\infty n_k)(p(n_0,\dots,n_k) = 1)\} \text{ is } \Sigma^0_{2k+1}\text{-complete}$$

$$P_{2k+2} := \{p \in 2^{\mathbb{N}^{k+1}} : (\exists^\infty n_0)(\exists^\infty n_1)\dots(\exists^\infty n_k)(p(n_0,\dots,n_k) = 1)\} \text{ is } \Pi^0_{2k+2}\text{-complete.}$$

The proof of the theorem above is a direct consequence of Lemmas *I*.3.35 and *I*.3.38. Before proving the first lemma we need the following result: notice that the proofs of both Lemmas *I*.3.37 and *I*.3.38 exploit ideas by Solecki (see [Kec12, §Notes and Hints 23.5(i)]).

Lemma I.3.37. *Let $\mathcal{X}$ be a computable metric space:*

(i) every $A \in \Pi^0_{n+2}(\mathcal{X})$ can be rewritten as $\bigcap_{k\in\mathbb{N}} B'_k$ with $B'_k \in \Sigma^0_{n+1}(\mathcal{X})$ and $B'_k \supseteq B'_{k+1}$

(ii) if $n > 2$, every $A \in \Sigma^0_{n+1}(\mathcal{X})$ can be rewritten as $\bigcup_{n\in\mathbb{N}} B'_k$ with $B'_k \in \Pi^0_n(\mathcal{X})$ and for every $i \neq j$, $B'_i \cap B'_k = \varnothing$.

Proof. For (i), by definition, $B = \bigcap_{k\in\mathbb{N}} B_k$ where $B_k \in \Sigma^0_{n+1}(\mathcal{X})$. Letting $B'_0 := B_0$ and for $k > 0$, $B'_k := \bigcup_{j\leqslant k} B_k$, it is straightforward to check that $B = \bigcap_{k\in\mathbb{N}} B'_k$ where $B'_k \in \Sigma^0_{n+1}(\mathcal{X})$ and $B'_k \supseteq B'_{k+1}$.

For (ii), let $A \in \Sigma^0_{n+1}(\mathcal{X})$: by definition, $A := \bigcup_{n\in\mathbb{N}} B_k$ with $B_k \in \Pi^0_n(\mathcal{X})$. For every k, let $D_k := \bigcup_{i<k} B_i$ and notice that $D_k \in \Pi^0_n(\mathcal{X})$. By (i), $D_k = \bigcap_{m\in\mathbb{N}} C^k_m$ where $C^k_m \in \Sigma^0_{n-1}(\mathcal{X})$ and $C^k_m \supseteq C^k_{m+1}$. Without loss of generality, we can also assume $C^k_0 = X$. It is easy to check that $A := \bigcup_{k\in\mathbb{N}} B_k \backslash D_k$ and that all $B_n \backslash \bigcup_{i<n} B_i$ are pairwise disjoint. Since $C^k_0 = \mathcal{X}$ and $C^k_m \supseteq C^k_{m+1}$, we obtain that $B_k \backslash D_k = \bigcup_{m\in\mathbb{N}} (B_k \bigcap (C^k_m \backslash C^k_{m+1}))$. Notice that for every k, m, the $(B_k \bigcap (C^k_m \backslash C^k_{m+1}))$'s are pairwise disjoint, and it is easy to check that they are in $\Pi^0_n(\mathcal{X})$. To conclude the proof, notice that $A = \bigcup_{k,m\in\mathbb{N}} B_k \backslash (C^k_m \backslash C^k_{m+1})$ i.e., $B'_k = B_k \bigcap (C^k_m \backslash C^k_{m+1})$. □

Lemma I.3.38. *Let $n > 1$ and let X be a metrizable space and let A be a Π^0_n-complete set of X. Then,*

- *$A^0_\infty := \{(x_i)_{i\in\mathbb{N}} \in X^{\mathbb{N}} : (\exists i)(x_i \in A)\}$ is Σ^0_{n+1}-complete*
- *$A^1_\infty := \{(x_i)_{i\in\mathbb{N}} \in X^{\mathbb{N}} : (\exists^\infty i)(x_i \in A)\}$ is Π^0_{n+2}-complete.*

Proof. The fact that A^0_∞ and A^1_∞ are respectively Σ^0_{n+1} and Π^0_{n+2} is immediate. We now show that for any computable metric space Y,

(i) for any $B^0 \in \Sigma^0_{n+1}(Y)$, $B \leqslant_{\mathsf{EW}} A^0_\infty$ and,

(ii) for any $B^0 \in \Pi^0_{n+2}(Y)$, $B \leqslant_{\mathsf{EW}} A^1_\infty$,

concluding that A^0_∞ and A^1_∞ are respectively Σ^0_{n+1}-complete and Π^0_{n+2}-complete For (i), notice that by Lemma *I*.3.37(i) $B^0 = \bigcup_{j\in\mathbb{N}} B^0_j$ where the $B^0_j \in \Pi^0_n(Y)$ are pairwise disjoint. By hypothesis A is a Π^0_n-complete set: let f_j be the reduction witnessing $B^0_j \leqslant_{\mathsf{EW}} A$. We claim that $g : Y \to X^{\mathbb{N}}$, $y \mapsto f_j(y)$ witnesses $B^0 \leqslant_{\mathsf{EW}} A^0_\infty$. This is straightforward as:

- $y \in B^0 \iff (\exists j)(y \in B^0_j) \iff (\exists j)(f_j(y) \in A) \iff g(y) \in A^0_\infty$;
- $y \notin B^0 \iff (\forall j)(y \in B^0_j) \iff (\forall j)(f_j(y) \notin A) \iff g(y) \notin A^0_\infty$;

For (ii), notice that, by Lemma $I.3.37$, $B^1 = \bigcap_{k\in\mathbb{N}} B^1_k$ with $B^1_k \in \Sigma^0_{n+1}(Y)$. Furthermore, for every k, B^1_k can be rewritten as $\bigcap_{j\in\mathbb{N}} C_{k,j}$ where the $C_{k,j} \in \Pi^0_n(Y)$ are pairwise disjoint.

Recall that, by hypothesis, A is a Π^0_n-complete set and let $f_{k,j}$ be the reduction witnessing $C_{k,j} \leqslant_{\mathsf{EW}} A$. We claim that $g : Y \to X^{\mathbb{N}}$, $y \mapsto (f_{k,j}(y))_{k,j\in\mathbb{N}}$ witnesses $B^1 \leqslant_{\mathsf{EW}} A^1_\infty$. To prove this notice that:

- $y \in B^1 \iff (\forall k)(\exists j)(y \in C_{k,j}) \iff (\exists^\infty \langle k,j\rangle)(f_{k,j}(y) \in A) \iff g(y) \in A^0_\infty$ and
- $y \notin B^1 \iff (\exists k)(y \notin B^1_k)$. By hypothesis, $(\forall k)(B^0_k \supseteq B^1_{k+1})$, hence we obtain that $(\forall k' \geqslant k)(y \notin B^1_k)$ and so $(\exists k)(\forall k' \geqslant k)(\forall j)(y \notin C_{k',j})$, i.e. $(\exists k)(\forall k' \geqslant k)(\forall j)(f_{k,j}(y) \notin A)$. Since, by hypothesis, there are only finitely many j's such that $y \in \bigcap_{i<k} \bigcup_{j\in\mathbb{N}} f_{i,j}(y) \in A$, we conclude that $g(y) \notin A^0_\infty$.

This concludes the proof. □

We now move to the complexity of subsets of trees.

Theorem I.3.39. *The following classification results hold:*

(i) *The set* $\mathcal{IF} := \{T \in \mathbf{Tr} : T$ *is ill-founded*$\}$ *is* Σ^1_1*-complete, while* $\mathcal{WF} := \{T \in \mathbf{Tr} : T$ *is well-founded*$\}$ *is* Π^1_1*-complete. In contrast,* $\mathcal{IF}_2 := \mathcal{IF} \cap \mathbf{Tr}_2$ *is* Π^0_1*-complete and* $\mathcal{WF}_2 := \mathcal{WF} \cap \mathbf{Tr}_2$ *is* Σ^0_1*-complete.*

(ii) *The set* $\mathcal{T}^{>\aleph_0} := \{T \in \mathbf{Tr} : |[T]| > \aleph_0\}$ *is* Σ^1_1*-complete, while* $\mathcal{T}^{\leqslant\aleph_0} := \{T \in \mathbf{Tr} : |[T]| \leqslant \aleph_0\}$ *is* Π^1_1*-complete. In this case,* $\mathcal{T}^{>\aleph_0}_2 := \mathcal{T}^{>\aleph_0} \cap \mathbf{Tr}_2$ *is* Σ^1_1*-complete as well and* $\mathcal{T}^{\leqslant\aleph_0}_2 := \mathcal{T}^{\leqslant\aleph_0} \cap \mathbf{Tr}_2$ *is also* Π^1_1*-complete.*

(iii) *The set* $\mathcal{UB} := \{T \in \mathbf{Tr} : |[T]| = 1\}$ *is* Π^1_1*-complete. In contrast,* $\mathcal{UB}_2 := \mathcal{UB} \cap \mathbf{Tr}_2$ *is* Π^0_2*-complete.*

Proof. To show that $\mathcal{IF}$ is Σ^1_1-complete see [Kec12, Theorem 27.1]: the theorem states the boldface case, but its proof works also in the lightface one. If $T \subseteq 2^{<\mathbb{N}}$, notice that, by König's lemma, $T \in \mathcal{IF}_2$ if and only if $(\forall n)(\exists \tau \in 2^n)(\tau \in T)$; hence $\mathcal{IF}_2$ is Π^0_1 and completeness is straightforward. It follows immediately that $\mathcal{WF}$ is Π^1_1-complete and $\mathcal{WF}_2$ is Σ^0_1-complete.

To prove that $\mathcal{T}^{>\aleph_0}$ is Σ^1_1-complete notice that, by the Cantor-Bendixson theorem for trees $T \in \mathcal{T}^{>\aleph_0}$ if and only if $(\exists S \subseteq T)(S$ is nonempty and perfect): the latter is a Σ^1_1 formula, hence it remains to show that $\mathcal{T}^{>\aleph_0}$ is complete for Σ^1_1 sets. This is immediate as $T \in \mathcal{IF} \iff \mathsf{Expl}(T) \in \mathcal{T}^{>\aleph_0}$. The proof for $\mathcal{T}^{>\aleph_0}_2$ is similar, and it follows immediately that $\mathcal{T}^{\leqslant\aleph_0}$ and $\mathcal{T}^{\leqslant\aleph_0}_2$ are Π^1_1-complete.

To prove that $\mathcal{UB}$ is Π^1_1-complete notice that, by the effective perfect set theorem (see [Mos82, Theorem 4F.1]), $T \in \mathcal{UB}$ if and only if

$$(\exists p \in \mathsf{HYP}(\mathbb{N}^{\mathbb{N}}))(p \in [T]) \land (\forall \tau, \tau')(\tau \mid \tau' \implies T_\tau \in \mathcal{WF} \lor T_{\tau'} \in \mathcal{WF}),$$

where $\mathsf{HYP}(\mathbb{N}^{\mathbb{N}})$ is the set of hyperarithmetical elements in $\mathbb{N}^{\mathbb{N}}$. Notice that the formula is Π^1_1: indeed, the second conjunct is clearly Π^1_1 and by Kleene's quantification theorem (see [Mos82, Theorem 4D.3]), the first conjunct is Π^1_1 as well. It remains to show that $\mathcal{UB}$ is complete for Π^1_1 sets. To do so, it suffices to notice that $T \in \mathcal{WF}$ if and only if $S \in \mathcal{UB}$, where $S := \{0^n : n \in \mathbb{N}\} \sqcup T$ (indeed, $[S] = \{0^{\mathbb{N}}\} \cup \{1p : p \in [T]\}$). If $T \subseteq 2^{<\mathbb{N}}$, notice that $T \in \mathcal{UB}_2$ if and only if

$$T \in \mathcal{IF}_2 \land (\forall \tau, \tau')(\tau \mid \tau' \implies T_\tau \in \mathcal{WF}_2 \lor T_{\tau'} \in \mathcal{WF}_2).$$

The formula is clearly Π^0_2 and proving completeness is straightforward. □

I.4 Classification problems and complexity of structures

I.4.1 Invariant descriptive set theory

In this section, we briefly discuss the last of the three areas of descriptive set theory mentioned in §I.3, namely invariant descriptive set theory, the discipline studying definable equivalence relations. For more on this topic, the reader is referred to a classical textbook in this area, i.e.

- Gao's book "Invariant descriptive set theory" ([Gao09]).

This area of descriptive set theory, as the ones mentioned in the previous sections, confirms the interdisciplinary nature of this subject. Indeed, many equivalence relations, before being studied in invariant descriptive set theory, had already been considered by other areas of mathematics. Furthermore, many mathematical problems can be expressed as *classification problems*. Formally, a classification problem is a pair (X, E), where X is a nonempty set and E an equivalence relation on X; a solution for (X, E) is a pair (I, φ) where I is a set and $\varphi : X \to I$ is a map assigning to each object in X an element of I, i.e. $(\forall x, y \in X)(xEy \iff \varphi(x) = \varphi(y))$. In some sense, invariant descriptive set theory is the "complexity theory" of equivalence relations where the complexity is measured by means of *reductions*. Given an equivalence relation E on a set X and $x \in X$, let $[x]_E := \{x' \in X : xEx'\}$ be the E-equivalence class of $x \in X$.

Definition I.4.1 ([Gao09, Definition 5.1.1])**.** Given two sets X and Y, let E be an equivalence relation on X and F be an equivalence relation on Y. We say that E is *reducible* to F if there is a reduction from E to F, i.e. there exists a map $f : X \to Y$ such that $xEy \iff f(x)Ff(y)$

Intuitively, if E is reducible to F then E is at most as complex as F. Considering classification problems and reductions without any constraint is not of great interest: indeed, any classification problem has a solution letting $I := X/E$ and $\varphi(x) := [x]_E$ and, using the Axiom of Choice, for showing that E is reducible to F, it is enough showing that there are as many E-equivalence classes as F-equivalence classes. In other words, the reducibility order on equivalence relations is the order on the cardinalities of the quotient space. It becomes much more interesting if we add definability/algorithmic requirements both on the classification problems and the reductions.

In the context of invariant descriptive set theory, an equivalence relation is defined over *standard Borel spaces* and the reduction is usually Borel. Recall from I.3.1 that, given a topological space X, $\mathbf{Bor}(X)$ is the smallest σ-algebra containing the open sets. Then, a *Borel space* is a pair $(X, \mathbf{Bor}(X))$ where X is a set and $\mathbf{Bor}(X)$ is the σ-algebra on X and $(X, \mathbf{Bor}(X))$ is called *standard* if it is isomorphic to $(Y, \mathbf{Bor}(Y))$ for some Polish space Y ([Gao09, Definition 1.4.2]). Standard Borel spaces enjoy nice closure properties: indeed, if $(X, \mathbf{Bor}(X))$ is a standard Borel space and $B \subseteq X$ is Borel, then the subspace B with the inherited Borel structure is standard Borel as well ([Gao09, Theorem 1.4.4]).

We introduce the following notations: given two standard Borel spaces X and Y, E an equivalence relation on X, and F an equivalence relation on Y. We say that E is *Borel reducible* to F, in symbols $E \leqslant_B F$ if there is a Borel reduction from E to F. Similarly, we say that E is *continuously reducible* to F, in symbols $E \leqslant_c F$, if there is a continuous reduction from E to F.

We continue introducing some well known benchmark equivalence relations that play a major role in invariant descriptive set theory and, in particular, in Part 2. All the equivalence relations we consider are Borel, i.e. given an equivalence relation E on a standard Borel space X we have that E is a Borel subset of $X \times X$. Notice that the equivalence relations we consider are all defined on (products of) $2^{\mathbb{N}}$. We start from the *identity on* $2^{\mathbb{N}}$ and the *eventual agreement on* $2^{\mathbb{N}}$. Given $p, q \in 2^{\mathbb{N}}$,

$$p \; Id \; q \iff (\forall n)(p(n) = q(n)) \text{ and } p \; E_0 \; q \iff (\exists m)(\forall n \geqslant m)(p(n) = q(n)).$$

We mention that E_0 plays a pivotal role in this subject: for example, it is the main character of the celebrated *Glimm-Effros dichotomy* . An equivalence relation E on a standard Borel space X is *smooth* if $E \leqslant_B Id$.

Theorem I.4.2. *[HKL90] Let E be a Borel equivalence relation. Then either E is smooth, or else $E_0 \leqslant_B E$.*

Notice also that E_0 is an *hyperfinite* equivalence relation. Let E be an equivalence relation on a standard Borel space X. We say that E is *finite* if every E-equivalence class is finite, and we say that E is hyperfinite if there are finite Borel equivalence relations $(E_n)_{n\in\mathbb{N}}$ such that $E := \bigcup_{n\in\mathbb{N}} E_n$ ([Gao09, Definition 7.2.1]). It is easy to notice that $E_0 = \bigcup_{n\in\mathbb{N}} E_n$, where, given $p, q \in 2^{\mathbb{N}}$, E_n is defined as $p\ E_n\ q \iff (\forall m \geqslant n)(p(m) = q(m))$. We mention that E_0 is the "canonical" hyperfinite equivalence relation: indeed the Dougherty-Jackson-Kechris theorem (see [Gao09, Theorem 7.2.3]) implies that an equivalence relation is hyperfinite if and only if $E \leqslant_B E_0$. In other words, combining this result with Theorem $I.4.2$, we have that, up to Borel reducibility, E_0 is the only hyperfinite and not smooth equivalence relation.

The next equivalence relation is defined on elements of $2^{\mathbb{N}\times\mathbb{N}}$. Given $p, q \in 2^{\mathbb{N}\times\mathbb{N}}$,

$$p\ E_1\ q \iff (\forall^\infty n \in \mathbb{N})(\forall i)(p(n,i) = q(n,i)).$$

As E_0 is the "canonical" hyperfinite equivalence relation, E_1 is the "canonical" *hypersmooth* equivalence relation, where an equivalence relation E on a standard Borel space X is hypersmooth if $E = \bigcup_{n\in\mathbb{N}} E_n$, and, for every n, E_n is smooth and $E_n \subseteq E_{n+1}$. Indeed, by [Gao09, Proposition 8.1.4], E is hypersmooth if and only if $E \leqslant_B E_1$.

Let $p, q \in 2^{\mathbb{N}}$:

$$p\ E_2\ q \iff \sum_{k=0}^{\infty} \frac{(p\triangle q)(k)}{k+1} < \infty \text{ and } p\ Z_0\ q \iff \lim_{k\to\infty} \frac{\text{card}(\{i \leqslant k : p\triangle q(i) = 1\})}{k+1} = 0,$$

where $p\Delta q$ is the *symmetric difference of p and q* defined as $p\Delta q(i) = 1 :\iff p(i) \neq q(i)$.

Before defining the last two equivalence relations we consider in this thesis, we introduce two well known operators, namely the *Friedman-Stanley jump* and the *power operator.*

Definition I.4.3 ([Gao09, Definition 8.3.1])**.** Let E be a Borel equivalence relation on a standard Borel space X. The Friedman–Stanley jump of E, denoted by E^+, is the equivalence relation on $X^{\mathbb{N}}$ defined by

$$(x_n)_{n\in\mathbb{N}}\ E^+\ (y_n)_{n\in\mathbb{N}} :\iff \{[x_n]_E : n \in \mathbb{N}\} = \{[y_n]_E : n \in \mathbb{N}\}.$$

It is easy to notice that if E is a Borel equivalence relation, E^+ is a Borel equivalence relation as well. Indeed,

$$(x_n)_{n\in\mathbb{N}} E^+ (y_n)_{n\in\mathbb{N}} \iff (\forall n)(\exists m)(x_n\ E\ y_m) \wedge (\forall m)(\exists n)(x_n\ E\ y_m).$$

In general, if E is a $\mathbf{\Sigma}^0_\gamma$ equivalence relation for some ordinal $\gamma < \omega_1$, E^+ is a $\mathbf{\Pi}^0_{\gamma+2}$ one and notice that $E <_B E^+$ ([Gao09, Theorem 8.3.6]).

Definition I.4.4 ([Gao09, Definition 8.5.1])**.** Let E be a Borel equivalence relation on a standard Borel space X. The power of E, denoted by E^ω, is the equivalence relation on $X^{\mathbb{N}}$ defined by

$$(x_n)_{n\in\mathbb{N}}\ E^\omega\ (y_n)_{n\in\mathbb{N}} \iff (\forall n)(x_n\ E\ y_n).$$

Notice that, in contrast to the Friedman-Stanley jump, this is not a jump operator as $E^\omega \equiv_B (E^\omega)^\omega$. Notice that if E is a $\mathbf{\Pi}^0_\gamma$ equivalence relation for some ordinal $\gamma < \omega_1$, then E^ω is a $\mathbf{\Pi}^0_\gamma$ equivalence relation as well. In Chapter IV we consider E_0^ω and $=^+$: sometimes in the literature these equivalence relations are denoted respectively with E_3 and E_{set}.

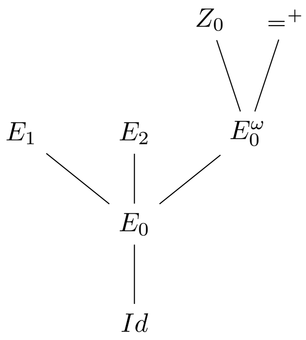

Figure I.2: Arrows represent continuous (and in particular, Borel) reductions. All the reductions are strict.

In invariant descriptive set theory and, in particular, in this thesis, an important subclass of classification problems is given by *isomorphism problems*. Descriptive set theorists have put serious effort into ranking the complexity of isomorphism problems for various familiar classes of countable structures such as groups, graphs, trees, linear orderings, and Boolean algebras (see [FS89, Mek81, CG01]).

Before giving some results on isomorphism problems, we give some basic definitions to fix the notations. Recall that a language L is a set of constants, operation symbols, and relation symbols of finite arity.

Definition I.4.5. An *L-structure* $\mathcal{A}$ consists of a nonempty set A called the *domain* of $\mathcal{A}$ together with an interpretation of the symbols of L, i.e. a function mapping

(i) every constant $c \in L$ into an element $c^{\mathcal{A}}$ of A,

and for any positive integer n,

(ii) every n-ary function symbol f of L into an n-ary function $F^{\mathcal{A}}$ of A

(iii) every n-ary relation symbol R of L into an n-ary relation $R^{\mathcal{A}}$ of A,

Definition I.4.6. Let $\mathcal{A}$ and $\mathcal{B}$ two structures in a language L. A *homomorphism* of $\mathcal{A}$ into $\mathcal{B}$ is a function f from A into B such that

(i) for every constant $c \in L$, $f(c^{\mathcal{A}}) = c^{\mathcal{B}}$;

(ii) for every positive integer n, for every n-ary function symbol F in L and for every sequence $(a_1, \ldots, a_n) \in A^n$, $f(F^A(a_1, \ldots, a_n)) = F^B(f(a_1, \ldots, a_n))$;

(iii) for every positive integer n, for every n-ary relation symbol R in L and for every sequence $(a_1, \ldots, a_n) \in A^n$, if $(a_1, \ldots, a_n) \in R^{\mathcal{A}}$ then $f(a_1, \ldots, a_n) \in R^{\mathcal{B}}$.

In case f is injective and in (iii), for every $(a_1, \ldots, a_n) \in A^n$, $f(a_1, \ldots, a_n) \in R^{\mathcal{B}}$ implies $(a_1, \ldots, a_n) \in R^{\mathcal{A}}$, then f is also an *embedding* from $\mathcal{A}$ into $\mathcal{B}$ (in symbols, $\mathcal{A} \hookrightarrow \mathcal{B}$). If f is a surjective embedding then we say that $\mathcal{A}$ and $\mathcal{B}$ are isomorphic (in symbols, $\mathcal{A} \cong \mathcal{B}$).

The isomorphism relation between structures is an equivalence relation and its equivalence classes (among structures) are called *isomorphism types*.

We fix a countable language L that, for simplicity, we assume to be relational, i.e. let $L := (R_i)_{i\in I}$ where $I \subseteq \mathbb{N}$, R_i is an n_i-ary relation symbol and let $Mod(L)$ be the space of L-structures on $\mathbb{N}$. Every element in $Mod(L)$ can be viewed as an element of the *space of countably infinite structures* defined as

$$X_L := \prod_{i\in I} 2^{\mathbb{N}^{n_i}}.$$

Indeed, we have a bijection that associates any $x := (x_i)_{i\in\mathbb{N}} \in X_L$ to $\mathcal{A}_x \in Mod(L)$, i.e. the (countable) model coded by x. Notice that, for every $i \in I$ and $\sigma \in \mathbb{N}^{n_i}$, $R_i^{\mathcal{A}_x}(\sigma) \iff x_i(\sigma) = 1$. This correspondence allows us to use at our convenience equivalently either $Mod(L)$ or X_L. Furthermore, notice that, if $L \neq \emptyset$, X_L (and hence $Mod(L)$) is homeomorphic to $2^{\mathbb{N}}$ and hence is a compact Polish space. We define the *logic action* of S_∞ (i.e. the group of permutations on $\mathbb{N}$) on $Mod(L)$ as follows. Given $\mathcal{A}, \mathcal{B} \in Mod(L)$, for every $i \in I$ and $(k_1, \dots, k_{n_i}) \in \mathbb{N}^{n_i}$, let $g \cdot \mathcal{A} = \mathcal{B}$ if and only if

$$R_i^{\mathcal{B}}(k_1, \dots, k_{n_i}) \iff R_i^{\mathcal{A}}(g^{-1}(k_1), \dots, g^{-1}(k_{n_i})).$$

In other words, $g \cdot \mathcal{A} = \mathcal{B}$ if and only if g is an isomorphism between $\mathcal{A}$ and $\mathcal{B}$. Clearly, g is a continuous action, and the associated equivalence relation (i.e. called *orbit equivalence relation*) is just the isomorphism relation on $Mod(L)$, i.e. $(\exists g \in S^\infty)(g \cdot \mathcal{A} = \mathcal{B}) \iff \mathcal{A} \cong \mathcal{B}$. We mention the following theorem by Scott, which can be obtained as a corollary from a result by Miller ([Kec12, Theorem 15.14]).

Theorem I.4.7 ([Kec12, Theorem 16.6]). *Given a nonempty countable language L, for any $\mathcal{A} \in Mod(L)$, $\{\mathcal{B} : \mathcal{A} \cong \mathcal{B}\}$ is Borel.*

We now introduce the basics of *infinitary logic*. Given a language L, we denote by $\mathcal{L}_{\omega_1\omega}$ the infinitary language over L defined as the finitary language except for the fact that we can take conjunctions and disjunctions of any size less than ω_1, the number of free variables has cardinality less than ω and formulas with $\forall$'s or $\exists$'s are of any length less than ω.

Definition I.4.8. Given a language L, $\mathcal{L}_{\omega_1\omega}$ is the smallest class such that:

- all finitary quantifier free L-formulas are in $\mathcal{L}_{\omega_1\omega}$;
- if φ is in $\mathcal{L}_{\omega_1\omega}$, then so are $(\forall x)\varphi$ and $(\exists x)\varphi$.
- Let $\bar{x}$ be a finite tuple of variables and $S \subseteq \mathcal{L}_{\omega_1\omega}$ a countable set of formulas with free variables in $\bar{x}$. Then, both the infinitary disjunction of formulas in S (denoted by $\bigvee\!\!\!\bigvee_{\varphi\in S} \varphi$) and the infinitary conjunction of formulas in S (denoted by $\bigwedge\!\!\!\bigwedge_{\varphi\in S}\varphi$) are in $\mathcal{L}_{\omega_1\omega}$.

We now make an example of a class of structures that is not axiomatizable in finite order logic. For instance, Take the language of graphs $L := \{E\}$ where E is the edge relation: applying the compactness theorem of first order logic, it is possible to show that the class of connected graphs is not axiomatizable by finitary first order logic. On the other hand, the following infinitary sentence does the job, i.e. a graph G is connected if and only if

$$(\forall v, w \in V(G))\bigwedge\!\!\!\bigwedge_{n\in\mathbb{N}}(\exists u_1, \dots, u_n \in V(G))((v, u_1) \in E(G) \wedge (u_1, u_2) \in E(G) \wedge \dots (u_n, w) \in E(G)).$$

The following results have applications also in invariant descriptive set theory, highlighting the pivotal role of infinitary logic. For an equivalence relation E on a set X, we say that $A \subseteq X$ is *E-invariant* if $x \in A$ and xEy imply $y \in A$. The following is the well known Lopez-Escobar theorem.

Theorem I.4.9 ([Kec12, Theorem 16.8]). *The $\cong$-invariant subsets of X_L are exactly those of the form $\{x : \mathcal{A}_x \models \varphi\}$, where φ is a formula without free variables of $L_{\omega_1\omega}$.*

The following is the Scott isomorphism theorem, telling us that countable structures are uniquely identified by formulas of $L_{\omega_1\omega}$ called *Scott sentences.*

Theorem I.4.10 ([Kec12, Corollary 16.10]). *For every countable structure $\mathcal{A}$ of L, there is a corresponding formula $\varphi_{\mathcal{A}}$ without free variables of $L_{\omega_1\omega}$ (i.e. the corresponding Scott sentence) such that for any countable structure $\mathcal{B}$ of L, $\mathcal{B} \cong \mathcal{A} \iff \mathcal{B} \models \varphi_{\mathcal{A}}$.*

Since it is not the purpose of this thesis, we do not make explicit here the construction of a Scott sentence, but we just mention that it is obtained via a transfinite process in which, at each stage, the formulas are, in some sense, approximations of the final product.

In first order logic, the complexity of a formula is measured by counting the alternation of quantifiers. For infinitary formulas the approach is similar but, when counting alternation, infinitary disjunctions are treated as existential ones, while infinitary conjunctions are as universal ones. We give the formal definition.

Definition I.4.11. Let α be an ordinal. A formula is $\Sigma^{\mathsf{inf}}_\alpha$ if it is of the form $\bigwedge\!\!\!\bigwedge_{i\in\mathbb{N}} \exists \bar{x}_i \varphi_i(\bar{x}_i, \bar{y})$ where the formulas φ_i are Π^{inf}_β for some $\beta < \alpha$. Similarly, a formula is $\Pi^{\mathsf{inf}}_\alpha$ if it is of the form $\bigvee\!\!\!\bigvee_{i\in\mathbb{N}} \forall \bar{x}_i \varphi_i(\bar{x}_i, \bar{y})$ where the formulas φ_i are $\Sigma^{\mathsf{inf}}_\beta$ for some $\beta < \alpha$. We denote by Σ^{inf}_0 and Π^{inf}_0 the finitary quantifier-free formulas.

Hence, the connectedness of graphs is axiomatizable via infinitary Π^{inf}_2 formulas.

We conclude this section mentioning that $L_{\omega_1\omega}$ infinitary logic is very useful in characterizing the syntactic properties of countable structures: on the other hand, if we want to study the computational properties of structures, as first noticed by Ash in [Ash86], the appropriate language is *computably infinitary language* that we discuss in the next section.

I.4.2 Computable structure theory

Computable structure theory is the field of computable mathematics studying the interplay between the complexity of a mathematical structure (i.e. how hard is to describe it and compute it) and its structural/algorithmic properties. The mathematical structures considered have (usually countable) domains consisting of relations, functions, and constants (e.g. graphs, linear orderings, rings etc. . .) and among the tools that are used to measure the complexity of mathematical structures, we can find for example Turing reducibility and the (hyper)arithmetic hierarchy. For this subject, we refer the reader to:

1. Ash and Knight's book "Computable Structures and the Hyperarithmetical Hierarchy" ([CJAJFK00]) and,
2. Montálban' books "Computable structure theory: Within the arithmetic" [Mon21] and "Computable structure theory: Beyond the arithmetic " ([Mon]).

We start introducing the notion of *atomic diagram.* Given an L-structure for some countable and relational language L, using some standard bijection from $\mathbb{N}^{<\mathbb{N}}$ to $\mathbb{N}$, we can identify any atomic sentence about any L-structure with a natural number. We define the atomic diagram $\mathcal{D}(\mathcal{A})$ of a structure $\mathcal{A}$, as the set of $n \in \mathbb{N}$ such that n is the code of an atomic L-sentence true in $\mathcal{A}$ or the negation of an atomic L-sentence that is false in $\mathcal{A}$. Now, we can identify any structure $\mathcal{A}$ via some $p \in 2^{\mathbb{N}}$ where $p(i) = 1$ if and only if the i-th atomic L-sentence is true in $\mathcal{A}$. Representing a structure via an element in $2^{\mathbb{N}}$ gives us a natural way to compare the complexity of L-structures.

We can assign to every structure some Turing degree $\mathbf{d}$: a structure $\mathcal{A}$ is $\mathbf{d}$-computable if $\mathcal{D}(\mathcal{A})$ is a $\mathbf{d}$-computable subset of $\mathbb{N}$. Given a structure $\mathcal{A}$ we denote by $\mathcal{A}\upharpoonright_n$ the finite substructure that is the restriction of $\mathcal{A}$ to the domain $\{0,\dots,n\}$. Notice that any computable structure $\mathcal{A}$ in a relational language is such that $\mathcal{A} = \bigcup_{n\in\mathbb{N}} \mathcal{A}\upharpoonright_n$ where $\mathcal{A}\upharpoonright_n \subseteq \mathcal{A}\upharpoonright_{n+1}$.

As noticed at the end of the previous subsection, in order to study the computational properties of structures we need a computably infinitary language. The *computable infinitary formulas* are the computable counterparts of $\Sigma^{\mathsf{inf}}_\alpha$ and $\Pi^{\mathsf{inf}}_\alpha$ formulas, and we denote them by Σ^{c}_n and Π^{c}_n. These consist of $L_{\omega_1\omega}$ formulas that are defined as in Definition I.4.8, but requiring infinite conjunctions and disjunctions to range over some c.e. set I of (computable) formulas. Notice that computable infinitary formulas can be relativized to an arbitrary oracle X: the class of X*-computable infinitary* Σ^{inf}_n *formulas* (respectively, Π^{inf}_n formulas) are denoted by $\Sigma^{c,X}_n$ ($\Pi^{c,X}_n$): the definition is the same of computable infinitary ones with the only exception that conjunctions and disjunctions range over an X-c.e. set I.

Notice that the definition of X-computable infinitary formulas we have just given, despite being the natural generalization of Definition I.4.11, it is not mathematically precise when $\alpha \geqslant 2$. Indeed, elements of a c.e. set are (objects that can be coded by) natural numbers: this is unproblematic when $\alpha \in \{0,1\}$ but for $\alpha \geqslant 2$, a computable infinitary $\Sigma^{\mathsf{inf}}_\alpha$ and $\Pi^{\mathsf{inf}}_\alpha$ formulas are obtained by c.e. sets of infinitary formulas and hence we need to code such formulas via natural numbers. This can be done by assigning codes to infinitary formulas in normal form, but, since this process is not important for the next topics, we prefer to stick with the informal but intuitive definition given above, referring the reader to [AK00, Chapter 7].

We move on to defining a *Turing computable embedding*, showing its interactions with computable infinitary formulas: recalling that all of our structures have domain $\mathbb{N}$,

Definition I.4.12 ([CCKM04, Definition 2])**.** A Turing computable embedding of $\mathfrak{K}$ into $\mathfrak{K}'$ is a computable functional Φ such that

- for every $\mathcal{A} \in \mathfrak{K}$, there exists $\mathcal{B} \in \mathfrak{K}'$ such that $\Phi^{D(\mathcal{A})} = \chi_{D(\mathcal{B})}$;
- if $\mathcal{A}, \mathcal{B} \in \mathfrak{K}$, we have that $\mathcal{A} \cong \mathcal{B} \iff \Phi(\mathcal{A}) \cong \Phi(\mathcal{B})$;

If such an embedding exists we denote it by $\mathfrak{K} \leqslant_{tc} \mathfrak{K}'$

In some sense, Turing computable embedding is the analog of a Borel reduction where the reduction map is computable instead of Borel and the equivalence relations are restricted to isomorphism relations.

Definition I.4.13 ([KMB07, Pullback Theorem §2])**.** Suppose that $\mathfrak{K}_0 \leqslant_{tc} \mathfrak{K}_1$ via a Turing operator Φ. Then for any computable infinitary sentence Ψ in the language of $\mathfrak{K}_1$, one can effectively find a computable infinitary sentence Ψ^* in the language of $\mathfrak{K}_0$ such that for all $\mathcal{A} \in \mathfrak{K}_0$ we have $\mathcal{A} \models \Psi^*$ if and only if $\Phi(\mathcal{A}) \models \Psi$. Moreover, for a nonzero $\alpha < \omega_1^{CK}$, if Ψ is a $\Sigma^{\mathsf{c}}_\alpha$ formula (Π^{c}_α) then so is Ψ^*.

In [BFSM20, §3.1] the authors showed that Definition I.4.13 admits a full relativization.

Theorem I.4.14 (Relativized Pullback Theorem)**.** *Suppose that $X \subseteq \omega$ and $\mathfrak{K}_0 \leqslant_{tc} \mathfrak{K}_1$ via a Turing X-operator Φ. Then, for any X-computable infinitary sentence ψ in the language of $\mathfrak{K}_1$, one can find, effectively with respect to X, an X-computable sentence $\psi^\star$ in the language of $\mathfrak{K}_0$ such that, for all $\mathcal{A} \in \mathfrak{K}_0$, we have $\mathcal{A} \models \psi^\star \Leftrightarrow \Phi(\mathcal{A}) \models \psi$.*

I.5 Reverse mathematics

Reverse mathematics is the field of mathematical logic and, in particular, of the foundations of mathematics, that aims to find which set existence axioms are required for proving a theorem of "ordinary mathematics". In the first chapter of ([Sim09]), Simpson refers to ordinary (or not set theoretic) mathematics as "that body of mathematics which is prior to or independent of the introduction of abstract set theoretic concepts". This includes theorems coming from different areas of mathematics like analysis, number theory, and some topology. The results contained in this section are mainly from

1. Simpson's book "Subsystem of Second Order Arithmetic" ([Sim09]) and
2. Dzhafarov and Mummert's book "Reverse Mathematics: Problems, Reductions, and Proofs" ([DM22]),

and we refer the reader to these books for more on this topic.

The field starts in [Fri75] where Friedman asks which formal systems isolate the "essential" axioms needed to prove them. The fascinating intuition is that "when the theorem is proved from the right axioms, the axioms can be proved from the theorem". More precisely, working in a "weak" axiom system B, given a theorem T we want to answer two questions:

1. which is the weakest axiom system R that we need to add to B in order to prove T?
2. Using B and T, can we recover all the axioms in R?

If the answer to both questions above is positive, we say that T and R are equivalent over B: this, in some sense, tells us that R contains the "essential" axioms needed to prove T. When proving such an equivalence, the direction answering the second question is usually called the "reversal", suggesting the process of retrieving back the axioms which we used to prove the theorem from the theorem itself: this also justifies the name "reverse mathematics".

Reverse mathematics usually is developed in the context of *second order arithmetic*. Indeed, second order arithmetic is powerful enough to formalize most theorems of ordinary mathematics but not so powerful to forbid the identification of the essential axioms needed to prove a theorem as set theory does.

The language of second order arithmetic L_2 is a two-sorted language which means that it has two different kinds of variables: *first order variables* (or number variables) ranging over $\mathbb{N}$, and *second order* variables ranging over subsets of $\mathbb{N}$. The signature of L_2 consists of:

- constant first order symbols 0 and 1;
- binary function symbols $+$ and $\cdot$ on first order variables;
- binary relation symbols $<$ and $=$ for first order variables;
- a set membership relation $\in$ taking a first order term and a second order term.

The first order terms (intended to denote $\mathbb{N}$) are the first order variables, 0, 1 and, if t_1 and t_2 are first order terms, then so are $t_1 + t_2$ and $t_1 \cdot t_2$. Given t_1 and t_2 first order terms and X a second order variable, the atomic formulas are $t_1 = t_2$, $t_2 < t_2$ and $t_1 \in X$. Their meanings are the classical one, i.e. addition and multiplication on $\mathbb{N}$, equality on $\mathbb{N}$, the "less than" relation on $\mathbb{N}$, and the membership of a natural number to a set. Formulas are built from atomic ones using propositional connectives and the quantifiers "$\forall$" and "$\exists$" ranging on $\mathbb{N}$ and subsets of $\mathbb{N}$.

Definition I.5.1 ([Sim09, Definition I.2.2]). A model for L_2 is a 7-tuple

$$(|M|, \mathcal{S}_M, +_M, \cdot_M, 0_M, 1_M, <_M),$$

where $|M|$ is the range of the number variables, $\mathcal{S}_M$ is a set of subsets of $|M|$ serving as the range of the set variables, $+_M$ and $\cdot_M$ are binary operations on $|M|$, 0_M and 1_M are distinguished elements of $|M|$, and $<_M$ is a binary relation on $|M|$. We always assume that the sets $|M|$ and $\mathcal{S}_M$ are disjoint and nonempty. Formulas of L_2 are interpreted in M in the obvious way.

We are now ready to define full second order arithmetic and its subsystems. Before doing so we give the following definition.

Definition I.5.2 ([DM22, Definition 5.3.5 and 5.3.6])**.** Let φ be a formula of L_2:

- the *induction axiom* for φ is the universal closure of

$$(\varphi(0) \wedge (\forall x)(\varphi(x) \implies \varphi(x+1))) \implies (\forall x)(\varphi(x)),$$

- the *comprehension axiom* for φ is the universal closure of

$$(\exists X)(\forall x)(x \in X \iff \varphi(x)]$$

  where X is a set variable not mentioned in φ and the formula φ which may have free set variables, which serve as parameters relative to which X is defined.

Then, given a collection Γ of formulas of L_2,

- $\mathsf{I}\Gamma$ is the axiom scheme consisting of the induction axiom for every $\varphi \in \Gamma$ and
- Γ-CA is the axiom scheme consisting of the comprehension axiom for every $\varphi \in \Gamma$.

We define the theory of second order arithmetic as the theory consisting of the basic axioms of Peano arithmetic (capturing the basic properties of $\mathbb{N}$ as a discrete ordered semiring), the induction axiom scheme for all formulas in L_2 and the comprehension axiom scheme for all formulas in L_2

Since second order arithmetic is a strong theory, in the sense that numerous theorems that can be formalized in L_2 are provable in it, to grasp the strength of theorems we restrict to fragments of second order arithmetic obtained by weakening the collection of comprehension and induction axioms that may be used. These are *subsystems* of second order arithmetic

A well known empirical fact in this field is the so called *big five phenomenon*. Namely, many theorems of ordinary mathematics happen to be equivalent to one of the five subsystems of second order arithmetic that we describe below.

RCA_0 (Recursive Comprehension Axiom)

This is the weakest subsystem of the big five and, roughly speaking, corresponds to a formalization of computable mathematics. Usually, it is assumed to be the base theory B we mentioned above. It consists of the basic axioms of Peano arithmetic plus Δ^0_1-comprehension scheme and Σ^0_1-induction scheme.

WKL_0 (Weak König Lemma)

This is the subsystem consisting of RCA_0 plus Weak König Lemma, the statement asserting "every infinite binary tree has an in infinite path".

ACA_0 (Arithmetic Comprehension Axiom)

It consists of WKL_0 plus the arithmetic comprehension scheme: in other words, it consists of all the axioms asserting the existence of any set which is arithmetically definable from given sets.

ATR_0 (Arithmetical Transfinite Recursion)

It consists of ACA_0 plus the statement saying that every arithmetic formula (or equivalently, the Turing jump operator) can be iterated along any countable well-order starting at any set.

$\mathbf{\Pi}^1_1$−CA_0 ($\mathbf{\Pi}^1_1$-comprehension axiom)

It consists of ATR_0 plus the $\mathbf{\Pi}^1_1$-comprehension scheme.

In particular, ATR_0 and $\mathbf{\Pi}^1_1$−CA_0, turn out to be equivalent to theorems arising in classical descriptive set theory related to perfect subsets of Polish spaces that will be the main characters of Chapter II.

We highlight that not all the theorems of ordinary mathematics are equivalent to one of the big five axiom systems described above. The most famous example of a theorem falling outside the big-five is Ramsey's theorem for pairs: from the proof of this fact, a variety of "natural" mathematical principles have been shown to be equivalent to none of the big-five, yielding the so called "reverse mathematics zoo".

I.6 Weihrauch reducibility

Weihrauch reducibility is a way to compare the uniform computational strength of partial multi-valued functions between represented spaces, i.e. problems. That is if a problem f is *Weihrauch reducible* to g it means that f can be computed by *exactly one* application of g, modulo some computable modification that is needed to "adjust" the input for g and the output for f. For more on Weihrauch reducibility, we refer the reader to Weihrauch's book [Wei13] (already mentioned in §I.2) and to

- Brattka, Gherardi and Pauly's chapter "Weihrauch complexity in computable analysis" in "Handbook of computability and complexity in analysis" ([BGP21]).

We start by giving the formal definition of Weihrauch reducibility.

Definition I.6.1. Let $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$ and $\mathbf{W}$ be represented spaces and $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$, $g :\subseteq \mathbf{Z} \rightrightarrows \mathbf{W}$ be partial multi-valued functions. We say that f is *Weihrauch reducible* to g, (in symbols $f \leqslant_{\mathrm{W}} g$[a]) if there exists computable $\Phi, \Psi :\subseteq \mathbb{N}^\mathbb{N} \to \mathbb{N}^\mathbb{N}$ such that

$$(\forall G \vdash g)(\Psi(\mathsf{id}(\cdot) \oplus G(\Phi(\cdot))) \vdash f).$$

We say that f is *strongly Weihrauch reducible* to g (in symbols,$f \leqslant_{\mathrm{sW}} g$), if there exists computable $\Phi, \Psi :\subseteq \mathbb{N}^\mathbb{N} \to \mathbb{N}^\mathbb{N}$ such that $(\forall G \vdash g)(\Psi(G(\Phi(\cdot)))) \vdash f$.

[a] Unfortunately, Wadge reducibility and Weihrauch reducibility are denoted with the same letter. On the other hand, we use a different style: Wadge reducibility is denoted by $\leqslant_{\mathbf{W}}$ and Weihrauch reducibility by $\leqslant_{\mathrm{W}}$.

Usually, the maps Φ and Ψ are called *forward* and *backward* functional respectively, and they play the roles of "pre-processing" and "post-processing" phases of the computation. The partial multi-valued function g plays the role of the oracle: in other words, the definition above can be restated as follows.

Definition I.6.2 (Definition I.6.1, restated)**.** Let $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$, $\mathbf{W}$ be represented spaces and $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$, $g :\subseteq \mathbf{Z} \rightrightarrows \mathbf{W}$ be partial multi-valued functions. Then f is Weihrauch reducible (respectively, strongly Weihrauch reducible) to a problem g, if there are computable maps $\Phi, \Psi :\subseteq \mathbb{N}^\mathbb{N} \to \mathbb{N}^\mathbb{N}$ such that

- for every name p_x for some $x \in \mathrm{dom}(f)$, $\Phi(p_x) = p_z$, where p_z is a name for some $z \in \mathrm{dom}(g)$ and,

- for every name p_w for some $w \in g(z)$, $\Psi(p_x \oplus p_w) = p_y$ (or just $\Psi(p_w) = p_y$ in case is a strong Weihrauch reduction) where p_y is a name for $y \in f(x)$.

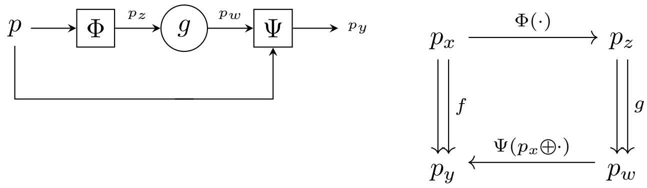

Figure I.3: Two diagrams graphically representing the Weihrauch reduction $f \leqslant_W g$. In particular, the right-hand-side diagram justifies the terms "forward" and "backward" functionals used for Φ and Ψ.

The difference between Weihrauch and strong Weihrauch reducibility is in the fact that, in the latter, the map Ψ does not have access to the name p of the input of f. Clearly strong Weihrauch reducibility implies Weihrauch reducibility, but the converse is not true in general.

Weihrauch reducibility and strong Weihrauch reducibility are reflexive and transitive hence they induce the equivalence relations $\equiv_W$ and $\equiv_{sW}$: that is $f \equiv_W g$ if and only if $f \leqslant_W g$ and $g \leqslant_W f$ (similarly for $\leqslant_{sW}$). The $\equiv_W$-equivalence classes are called *Weihrauch degrees* (similarly the $\equiv_{sW}$-equivalence classes are called *strong Weihrauch degrees*). Both the Weihrauch degrees and the strong Weihrauch degrees form lattices (see [BGP21, Thm. 3.9 and Thm. 3.10]).

There are several natural operations on problems which also lift to the $\equiv_W$-degrees and the $\equiv_{sW}$-degrees: we mention below the ones we need. Let $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$ and $g :\subseteq \mathbf{Z} \rightrightarrows \mathbf{W}$. Then we define

- the *parallel product* $f \times g :\subseteq \mathbf{X} \times \mathbf{Z} \rightrightarrows \mathbf{Y} \times \mathbf{W}$ where

$$\mathrm{dom}(f \times g) := \mathrm{dom}(f) \times \mathrm{dom}(g) \text{ and } (f \times g)(x, z) := f(x) \times g(z);$$

- the *finite parallelization* $f^* :\subseteq \mathbf{X}^* \to \mathbf{Y}^*$ where

$$\mathrm{dom}(f^*) := \bigcup_{n \in \mathbb{N}} (\{n\} \times \mathrm{dom}(f)^n) \text{ and } f^*(n, (x_i)_{i<n}) := \{(y_i)_{i<n} : (\forall i < n)(y_i \in f(x_i))\};$$

- the *infinite parallelization* $f :\subseteq \mathbf{X}^{\mathbb{N}} \rightrightarrows \mathbf{Y}^{\mathbb{N}}$ where

$$\mathrm{dom}(\widehat{f}) := \mathrm{dom}(f)^{\mathbb{N}} \text{ and } \widehat{f}((x_i)_{i \in \mathbb{N}}) := \{(y_i)_{i \in \mathbb{N}} : (\forall i)(y_i \in f(x_i))\}.$$

- the *co-product* $f \sqcup g :\subseteq \mathbf{X} \times \mathbf{Z} \rightrightarrows \mathbf{Y} \times \mathbf{W}$ where given $i \in \{0, 1\}$

$$\mathrm{dom}(f \sqcup g) := \mathrm{dom}(f) \sqcup \mathrm{dom}(g) \text{ and } (f \sqcup g)(i, x) := \begin{cases} \{0\} \times f(x) & \text{if } i = 0 \\ \{1\} \times g(x) & \text{if } i = 1 \end{cases}$$

- generalizing the co-product, given a family of problems $\{f_i : i \in \mathbb{N}\}$ with $f_i :\subseteq \mathbf{X}_i \rightrightarrows \mathbf{Y}_i$, the *countable co-product* $\bigsqcup_{i \in \mathbb{N}} f_i :\subseteq \bigcup_{i \in \mathbb{N}} \{i\} \times X_i \rightrightarrows \bigcup_{i \in \mathbb{N}} \{i\} \times Y_i$ where

$$\mathrm{dom}(\bigsqcup_{i \in \mathbb{N}} f_i) := \bigcup_{i \in \mathbb{N}} \{i\} \times \mathrm{dom}(f_i) \text{ and } \big(\bigsqcup_{i \in \mathbb{N}} f_i\big)(i, x) := \{i\} \times f_i(x).$$

Informally, the first three operators defined above, capture respectively the idea of using f and g in parallel, using f a finite (but given in the input) number of times in parallel and using f

countably many times in parallel. The last two capture respectively the idea of computing exactly one between f and g and computing exactly one f_i.

We now define the *finite unbounded parallelization*: the definition we use, with a slightly different notation, was recently given by Soldà and Valenti. This operator generalizes the finite parallelization operator, defined above, relaxing the requirement that the number of instances is part of the input for the problem.

Definition I.6.3 ([SV22])**.** For every $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$, define the $f^{u*} :\subseteq \mathbb{N} \times \mathbb{N}^{\mathbb{N}} \times \mathbf{X} \rightrightarrows (\mathbb{N}^{\mathbb{N}})^{<\mathbb{N}}$ as follows:

- instances are triples $(e, w, (x_n)_{n\in\mathbb{N}})$ such that $(x_n)_{n\in\mathbb{N}} \in \mathrm{dom}(\widehat{f})$ and for each sequence $(q_n)_{n\in\mathbb{N}}$ with $\delta_Y(q_n) \in f(x_n)$, there is a $k \in \mathbb{N}$ such that $\Phi_e(w, q_0 \oplus \ldots \oplus q_{k-1})(0)\downarrow$ in k steps;
- a solution for $(e, w, (x_n)_{n\in\mathbb{N}})$ is a finite sequence $(q_n)_{n<k}$ such that for every $n < k$, $\delta_Y(q_n) \in f(x_n)$ and $\Phi_e(w, q_0 \oplus \ldots \oplus q_{k-1})(0)\downarrow$ in k steps.

Informally, f^{u*} takes as input a Turing functional with a parameter plus an input for $\widehat{f}$ and outputs "sufficiently many" names for solutions where "sufficiently many" is determined by the convergence of the Turing functional in input. This operator is particularly useful in Theorem I.6.10 where we characterize the first-order part (see Definition I.6.8) of problems that are Weihrauch equivalent to the parallelization of functions with codomain $\mathbb{N}$.

We proceed to give the definition of *cylinder*. Intuitively, a cylinder f is a problem that from the outputs recovers the input, i.e. for every $x \in \mathrm{dom}(f)$, given $f(x)$ we can compute x.

Definition I.6.4 ([BG09, Definition 3.4])**.** We say that a multi-valued function f is a *cylinder* if $\mathsf{id} \times f \leqslant_{\mathrm{sW}} f$.

Notice that, in general, we only have that $f \leqslant_{\mathrm{sW}} \mathsf{id} \times f$: as a counterexample to the converse reduction, take any total constant $f : \mathbb{N} \to \mathbb{N}$. Another useful observation is that, since id is clearly a cylinder, for any problem f, $\mathsf{id} \times f$ is a cylinder and hence, since $\mathsf{id} \times f \equiv_{\mathrm{W}} f$ is trivial, we get that every Weihrauch degree has a representative which is a cylinder. The notion of cylinder gives a simple but very useful connection between Weihrauch and strong Weihrauch reducibility, as stated in the next theorem.

Theorem I.6.5 ([BG09, Cor. 3.6])**.** *If f is a cylinder, then $g \leqslant_{\mathrm{W}} f$ if and only if $g \leqslant_{\mathrm{sW}} f$.*

This theorem is useful for establishing nonreductions because, if f is a cylinder, then it suffices to diagonalize against all strong Weihrauch reductions from g to f in order to show that $g \nleqslant_{\mathrm{W}} f$. Cylinders are also useful when working with *compositional products.*

The compositional product was introduced in [BGM12] and proved to be well-defined in [BP16, Corollary 3.7], and it captures the idea of applying g and then applying f. One could observe that in Definition I.2.5 we have already defined the composition $f \circ g$, which, at a first sight, could seem that it works for our purposes. Unfortunately, its definition does not match the above intuitive idea of applying first g and then f: indeed, $f \circ g$ is well-defined if and only if $\mathsf{range}(g) \subseteq \mathrm{dom}(f)$ and this does not happen in many cases.

Definition I.6.6 ([BGM12, Definition 4.1])**.** Let f and g be multi-valued functions. The compositional product $f * g$ is defined as

$$f * g \equiv_{\mathrm{W}} \max_{\leqslant_{\mathrm{W}}}\{f_0 \circ g_0 \,:\, f_0 \leqslant_{\mathrm{W}} f \wedge g_0 \leqslant_{\mathrm{W}} g\}.$$

Differently from the operations defined above and in Definition $I.6.3$, this operator does not map f and g to a unique multi-valued function, but it maps f and g to a Weihrauch degree. For readability, we always use this abuse of notation: namely, when we write $h \leqslant_W f * g$ we mean h is Weihrauch reducible to any problem in the Weihrauch degree defined by $f*g$. In proving statements involving compositional products, one direction is usually harder than the other. Namely, to prove that $h \leqslant_W f * g$, it suffices to prove that $h \leqslant_W f \circ \Phi_e \circ g$, for some computable functional Φ_e. Proving the opposite direction, i.e. $f * g \leqslant_W h$, is the difficult task: the following proposition, also known as *cylindrical decomposition*, is a helpful tool to solve this problem.

Proposition I.6.7. *Given f, g problems and F, G cylinders such that $F \equiv_W f$ and $G \equiv_W g$, there exists a computable Φ_e such that $f * g \equiv_W F \circ \Phi_e \circ G$.*

We have already observed that for every function f, $f \equiv_W \mathsf{id} \times f$ where $\mathsf{id} \times f$ is clearly a cylinder: hence $f * g \equiv_W (\mathsf{id} \times f) \circ \Phi_e \circ (\mathsf{id} \times g)$. Therefore, we can assume $f * g$ is a cylinder.

For each problem f, we denote by $f^{[n]}$ the n-fold iteration of the compositional product of f with itself, i.e., $f^{[1]} = f$, $f^{[2]} = f * f$, and so on.

Many (non) reductions in Chapter II and Chapter III, follow from the characterization of the *first-order part* ${}^1 f$ (introduced in [DSY23] and extensively studied in [SV22]) and the *deterministic part* $\mathsf{Det}(f)$ of a problem f (defined in [GPV21]). We start discussing the first-order part of a problem, Its definition is one of the outcomes coming from the investigation of the interplay between reverse mathematics and computable analysis (see the introduction to Chapter II for a discussion on this research program). Indeed, the first-order part of a problem takes inspiration from the *first-order part* of a theorem, a notion studied in reverse mathematics that captures the strongest "number-theoretic result" that one can derive from a particular theorem. In similar fashion, the first-order part of a problem represent the strongest problem with codomain $\mathbb{N}$ that is Weihrauch reducible to that problem.

Definition I.6.8. [SV22, Definition 2.2] We say that a computational problem $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$ is first-order if there is a computable injection $Y \to \mathbb{N}$ with computable inverse. For every problem $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$, the first-order part of f is the multi-valued function ${}^1 f :\subseteq \mathbb{N} \times \mathbb{N}^\mathbb{N} \times \mathbf{X} \rightrightarrows \mathbb{N}$ defined as follows:

- instances are triples (e, w, x) such that $x \in \mathrm{dom}(f)$ and for every $y \in f(x)$ and every name p_y for y, $\Phi_e(w \oplus p_y)(0) \downarrow$;
- a solution for (e, w, x) is any n such that there is a name p_y for a solution $y \in f(x)$ with $\Phi_e(w \oplus p_y)(0) \downarrow = n$.

The following theorem provides a convenient characterization of the first-order part of a problem that is used extensively in the next chapters.

Theorem I.6.9 ([SV22])**.** *For every problem f,*

$$^1 f \equiv_W \max_{\leqslant_W}\{g : g \text{ is first-order and } g \leqslant_W f\}.$$

The following theorem relates the first-order part with the unbounded finite parallelization.

Theorem I.6.10 ([SV22, Theorem 5.7])**.** *For every first-order f, ${}^1(\widehat{f}) \equiv_W f^{u*}$.*

In the same spirit of the first-order part definition, one can define other "parts" of a problem which capture the strongest problem with a restricted codomain which Weihrauch reduces to the problem: this is the case for the deterministic part of a problem, i.e. the strongest single-valued function which Weihrauch reduces to the problem.

Definition I.6.11. [GPV21, Definition 3.1] Let $\mathbf{X}$ be a represented space and $f :\subseteq \mathbf{Y} \rightrightarrows \mathbf{Z}$ be a multi-valued function. We define $\mathsf{Det}_{\mathbf{X}}(f) :\subseteq \mathbb{N} \times \mathbb{N}^{\mathbb{N}} \times \mathbf{Y} \to \mathbf{X}$ by

$$\mathsf{Det}_{\mathbf{X}}(f)(e, w, y) = x :\iff (\forall z \in \delta_Z^{-1}(f(y)))(\delta_X(\Phi_e(p \oplus z) = y)).$$

The domain of $\mathsf{Det}_{\mathbf{X}}(f)$ is maximal for this to be well-defined. We just write $\mathsf{Det}(f)$ for $\mathsf{Det}_{\mathbb{N}^{\mathbb{N}}}(f)$.

As we did for the first-order part, also for the deterministic part we mostly use the characterization given by the next theorem, instead of dealing with the definition above.

Theorem I.6.12 ([GPV21, Theorem 3.2])**.** *For every problem f,*

$$\mathsf{Det}(f) \equiv_{\mathrm{W}} \max_{\leqslant_{\mathrm{W}}}\{g : g :\subseteq X \to \mathbb{N}^{\mathbb{N}} \land g \leqslant_{\mathrm{W}} f\}.$$

The (k-)finitary part of a problem

These notions were introduced jointly with Arno Pauly, and, so far, they do not have explicitly appeared in the literature.

For $k > 0$, we denote with $\mathbf{k}$ the space consisting of $\{0, \dots, k-1\}$ with the discrete topology: the **k**-*finitary part* and the *finitary part* of a problem f captures respectively the most complex problem with codomain $\{0, \dots, k-1\}$ and the most complex problem with finite codomain. We start from the **k**-finitary part of a problem, whose definition follows the same pattern of Definition *I*.6.8.

Definition I.6.13. For every problem $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$, the **k**-finitary part of f is the multi-valued function $\mathrm{Fin}_{\mathbf{k}} :\subseteq \mathbb{N} \times \mathbb{N}^{\mathbb{N}} \times \mathbf{X} \rightrightarrows \mathbf{k}$ defined as follows:

- instances are pairs (e, w, x) such that $x \in \mathrm{dom}(f)$ and for every $y \in f(x)$ and every name p_y for y, $\Phi_e(w \oplus p_y)(0) \downarrow< k$;
- a solution for (e, w, x) is any $n < k$ such that there is a name p_y for a solution $y \in f(x)$ with $\Phi_w(p_y)(0) \downarrow= n$.

Proposition I.6.14. *For every problem f,* $\mathrm{Fin}_{\mathbf{k}} \equiv_{\mathrm{W}} \max_{\leqslant_{\mathrm{W}}}\{g :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbf{k} \mid g \leqslant_{\mathrm{W}} f\}$.

Proof. The proof is analogous to [GPV21, Theorem 3.2] (see also the comment in [SV22] after Proposition 3.2). □

We now define the finitary part of a problem f.

Definition I.6.15. For every problem f, we define the *finitary part* of f as

$$\mathrm{Fin}(f) := \bigsqcup_{k \geqslant 1} \mathrm{Fin}_{\mathbf{k}}(f).$$

We highlight that, despite its intuitive meaning is reminiscent of the first-order/deterministic/**k**-finitary part of a problem, the definition of the finitary part of a problem is very different. Indeed, we do not have a similar characterization to the ones given in Theorems *I*.6.9 and *I*.6.12 and Proposition *I*.6.14: in particular, the codomain of the finitary part is not even finite.

The following proposition is immediate.

Proposition I.6.16. *For every problem f,* $\mathrm{Fin}(f) \leqslant_{\mathrm{W}} {}^{1}f$.

We want to show that, for some f's, the reduction in Proposition I.6.16 can be strict (Proposition I.6.19). Before doing so we define when a problem is *join-irreducible* and the *cardinality of a problem.*

Definition I.6.17 ([BDP12, Definition 5.4])**.** A problem f is called join-irreducible, if

$$f \equiv_{\mathrm{W}} \bigsqcup_{i \in \mathbb{N}} f_i \implies (\exists n_0)(f \equiv_{\mathrm{W}} f_{n_0}).$$

Definition I.6.18 ([BGH15, Definition 3.5])**.** For every problem $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$ we denote by $\#f$ the maximal cardinality (if it exists) of a set $M \subseteq \mathrm{dom}(f)$ such that $\{f(x) : x \in M\}$ contains pairwise disjoint sets.

It is easy to see that for every problem f and g, $f \leqslant_{\mathrm{sW}} g \implies \#f \leqslant \#g$ [BGH15, Proposition 3.6].

Proposition I.6.19. $\mathrm{Fin}(\mathsf{C}_{\mathbb{N}}) <_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} {}^{1}\mathsf{C}_{\mathbb{N}}$.

Proof. The reduction and the equivalence are immediate.

For strictness, suppose $\mathsf{C}_{\mathbb{N}} \leqslant_{\mathrm{W}} \bigsqcup_{k \geqslant 1} \mathrm{Fin}_{\mathbf{k}}(\mathsf{C}_{\mathbb{N}})$. Since $\mathsf{C}_{\mathbb{N}}$ is join-irreducible ([BDP12, Corollary 5.6]), we obtain that if $\mathsf{C}_{\mathbb{N}} \leqslant_{\mathrm{W}} \bigsqcup_{k \in \mathbb{N}} \mathrm{Fin}_{\mathbf{k}}(\mathsf{C}_{\mathbb{N}})$, then there must exist some $k \in \mathbb{N}$ with $\mathsf{C}_{\mathbb{N}} \leqslant_{\mathrm{W}} \mathrm{Fin}_{\mathbf{k}}(\mathsf{C}_{\mathbb{N}})$. Since $\#\mathrm{Fin}_{\mathbf{k}}(\mathsf{C}_{\mathbb{N}}) < \#\mathsf{C}_{\mathbb{N}}$, by [BGH15, Proposition 3.6] this cannot be the case. ☐

Other useful operations on problems do not lift to Weihrauch degrees (i.e. applying the operation to equivalent problems does not always produce equivalent problems).

The first such operation is the *jump of a problem*: recall the definition of jump of a represented space from Definition I.2.9.

Definition I.6.20. Given a problem $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$ its jump $f' :\subseteq \mathbf{X}' \rightrightarrows \mathbf{Y}$ is defined as $f'(x) := f(x)$.

In other words, f' is the following task: given a sequence that converges to a name for an instance of f, produces a solution for that instance. Unfortunately, the jump on the Weihrauch degrees does not have the properties that we usually require for a jump operator: on the other hand, it behaves better in the strong Weihrauch degrees.

Proposition I.6.21. *For all problems f and g, $f \leqslant_{\mathrm{sW}} f'$ and $f \leqslant_{\mathrm{sW}} g \implies f' \leqslant_{\mathrm{sW}} g'$.*

The proposition above tells us, in particular, that the jump is monotone (and hence degree-theoretic) with respect to strong Weihrauch reducibility. On the other hand, it is easy to show examples of problems f such that $f \equiv_{\mathrm{sW}} f'$, e.g. take as f any constant map. As we said the situation is even worse for Weihrauch degrees: the jump does not lift to Weihrauch degrees and, for example, it is possible that $f \leqslant_{\mathrm{W}} g$ but $g' <_{\mathrm{W}} f'$. We use $f^{(n)}$ to denote the n-th iterate of the jump applied to f.

In [BHK17, Theorem 11], the authors proved the following theorem that is a sort of an inverse of Proposition I.6.21. Notice that "relative to the halting problem" in the next theorem and in similar statements means that the forward and backward functionals of the Weihrauch reduction $f \leqslant_{\mathrm{W}} g$ have access to the halting problem: a similar definition holds replacing "halting problem" with any oracle.

Theorem I.6.22 (Jump inversion theorem)**.** *$f' \leqslant_W g' \implies f \leqslant_W g$ relative to the halting problem.*

The theorem above is particularly useful in Chapter III.

We now introduce the *totalization of a problem* and the *completion of a problem.* These two operators are different ways of making a partial multi-valued function total; neither of them lifts to Weihrauch degrees.

Definition I.6.23. Given a partial multi-valued function $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$ the totalization of f is the total multi-valued function $\mathsf{T}f$ defined as

$$\mathsf{T}f(x) := \begin{cases} f(x) & \text{if } x \in \text{dom}(f), \\ Y & \text{otherwise.} \end{cases}$$

For more details on the totalization we refer the reader to [BG21].

To define the completion of a problem f we need to first introduce the completion of a represented space. We adopt the following notation: given $p \in \mathbb{N}^\mathbb{N}$ we define $\hat{p}_n$ to be $\langle\rangle$ if $p(n) = 0$, $\langle p(n) - 1\rangle$ otherwise; then $p - 1$ is the concatenation of all the $\hat{p}_n$'s.

Definition I.6.24. For a represented space $\mathbf{X} = (X, \delta_X)$ we define its completion as $\overline{\mathbf{X}} = (\overline{X}, \delta_{\overline{X}})$ where $\overline{X} = X \cup \{\bot\}$ with $\bot \notin X$ and $\delta_{\overline{X}} : \mathbb{N}^\mathbb{N} \to \overline{X}$ is the total function defined by

$$\delta_{\overline{X}}(p) := \begin{cases} \delta_X(p-1) & \text{if } p - 1 \in \text{dom}(\delta_X) \\ \bot & \text{otherwise.} \end{cases}$$

Let $f :\subseteq \mathbf{X} \rightrightarrows \mathbf{Y}$ be a multi-valued function. We define the completion of f as the total multi-valued function $\overline{f} : \overline{\mathbf{X}} \rightrightarrows \overline{\mathbf{Y}}$ such that

$$\overline{f}(x) = \begin{cases} f(x) & \text{if } x \in \text{dom}(f), \\ \overline{Y} & \text{otherwise.} \end{cases}$$

Well known problems in the Weihrauch lattice

We now introduce some problems in the Weihrauch lattice highlighting some results on them. Let $\mathsf{lim}_\mathbf{X} :\subseteq (\mathbf{X})^\mathbb{N} \to \mathbf{X}$, $(p_n)_{n\in\mathbb{N}} \mapsto \lim p_n$ be the single-valued function whose domain consists of all converging sequences in $\mathbf{X}$. In case $\mathbf{X} = \mathbb{N}^\mathbb{N}$, we just write lim instead of $\mathsf{lim}_{\mathbb{N}^\mathbb{N}}$. The following proposition connects f and its jump via lim.

Proposition I.6.25 ([BGM12, Corollary 5.16 and 5.17])**.** *For every problem f, $f' \leqslant_W f * \mathsf{lim}$ and, if f is a cylinder, $f' \equiv_W f * \mathsf{lim}$.*

Notice that $f * \mathsf{lim} \leqslant_W f$ does not hold in general. The function lim has also another convenient characterization used both in Chapter II and Chapter III, namely the Turing jump operator $\mathsf{J} : \mathbb{N}^\mathbb{N} \to \mathbb{N}^\mathbb{N}$, $p \mapsto p'$. By [BDP12, Lemma 8.9], $\mathsf{lim} \equiv_{sW} \mathsf{J}$.

We continue with *decision problems*, i.e. problems having codomain $\{0, 1\}$. We begin presenting those solving Γ-complete or $\check{\Gamma}$-complete decision problem.

Definition I.6.26 (Limited principle of omniscience)**.** The problem $\mathsf{LPO} : 2^\mathbb{N} \to \{0, 1\}$ is defined as $\mathsf{LPO}(p) := 1$ if and only if $(\exists n)(p(n) = 1)$.

It is immediate from the definition that we can think of LPO as the problem of answering yes or no to a $\Sigma_1^{0,p}$ or $\Pi_1^{0,p}$ question.

Similarly, for every $n > 0$ the function $\mathsf{LPO}^{(n)} : \mathbb{N}^\mathbb{N} \to \{0,1\}$ answers yes or no to a $\Sigma_{n+1}^{0,p}$ or $\Pi_{n+1}^{0,p}$ question. Usually, in case $n = 1$ or $n = 2$ we write LPO' and LPO''.

Notice that LPO and lim are closely related as the next theorem shows.

Theorem I.6.27 ([BGP21, Theorem 6.7 and Proposition 6.10])**.** *For every n, $\mathsf{lim}^{(n)} \equiv_{\mathrm{sW}} \widehat{\mathsf{LPO}^{(n)}}$.*

To conclude the definitions of Γ-complete or $\check{\Gamma}$-complete decision problems, we introduce $\mathsf{WF} : \mathbf{Tr} \to \{0,1\}$ as $\mathsf{WF}(T) = 1$ if and only if $T \in \mathcal{WF}$. Analogously to LPO, we can think of WF as the problem answering yes or no to questions which are Π_1^1 or Σ_1^1 in the input. In the literature, WF was introduced under different names: the same notation appears in [Hir19, SV22], while [BG21] uses WFT and [KMP20, MV21] use $\chi_{\Pi_1^1}$.

We anticipate that $\widehat{\mathsf{WF}}$ is the natural candidate to represent $\mathbf{\Pi}_1^1{-}\mathsf{CA}_0$ in the Weihrauch lattice, but we postpone the discussion on the program connecting reverse mathematics and computable analysis to Part 1 and in particular Chapter II. We just mention a convenient characterization of $\widehat{\mathsf{WF}}$ in terms of *hyperjump* of a set. The hyperjump of $A \subseteq \mathbb{N}$ can be defined, following [Rog87, Definition 4.12], as

$$\mathsf{HJ}(A) := \{z : \varphi_z^A \text{ is the characteristic function of a well-founded tree}\}.$$

The well known fact that $\mathsf{HJ}(A)$ is a $\Pi_1^{1,A}$-complete subset of the natural numbers implies that $\widehat{\mathsf{WF}} \equiv_{\mathrm{sW}} \mathsf{HJ}$. We continue the discussion on problems in the Weihrauch lattice corresponding to statements that, in reverse mathematics, are equivalent to $\mathbf{\Pi}_1^1{-}\mathsf{CA}_0$ in Chapter II, and we move to other well-studied problems.

We now move our attention to *choice problems*, which have emerged as very significant milestones in the Weihrauch lattice. For a computable metric space $\mathcal{X}$ and a class $\mathbf{\Gamma}$ as the ones in Definition *I*.3.27, let $\mathbf{\Gamma}\text{-}\mathsf{C}_\mathcal{X} :\subseteq \mathbf{\Gamma}(\mathcal{X}) \rightrightarrows \mathcal{X}$ be the problem that given as input a nonempty set $A \in \mathbf{\Gamma}(\mathcal{X})$ outputs a member of A. When $\mathbf{\Gamma} = \mathbf{\Pi}_1^0$ we just write $\mathsf{C}_\mathcal{X}$, and we denote by $\mathsf{C}_\mathbf{k}$ the choice problem on $\mathbf{k} := \{0, \dots, k-1\}$. The same problem with domain restricted to singletons is denoted by $\mathbf{\Gamma}\text{-}\mathsf{UC}_\mathcal{X}$. It is well known that for every $n > 0$, $(\mathbf{\Pi}_n^0\text{-}\mathsf{C}_\mathbb{N})' \equiv_{\mathrm{W}} \mathbf{\Pi}_{n+1}^0\text{-}\mathsf{C}_\mathbb{N}$.

Using the tree representation of closed sets, $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ can be formulated as the problem of computing a path through some $T \in \mathcal{IF}$; $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ is the same problem with domain restricted to $\mathcal{UB}$. Notice that both problems are closed under compositional product by [BDP12, Theorem 7.3]: furthermore $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathbf{\Sigma}_1^1\text{-}\mathsf{C}_{\mathbb{N}^\mathbb{N}}$, $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathbf{\Sigma}_1^1\text{-}\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ (see [KMP20]) and, similarly, $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathbf{\Sigma}_1^1\text{-}\mathsf{C}_{\mathbb{N}^\mathbb{N}}$. As noticed in [KMP20] and mentioned in the introduction, $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ and $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ are among the problems that correspond to ATR_0. We need the following proposition.

Proposition I.6.28. $\widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_\mathbb{N}} \equiv_{\mathrm{W}} \widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{C}_\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{UC}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \widehat{\mathbf{\Sigma}_1^1\text{-}\mathsf{C}_\mathbb{N}} <_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$.

Proof. The reduction $\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_\mathbb{N} \leqslant_{\mathrm{W}} \mathbf{\Pi}_1^1\text{-}\mathsf{C}_\mathbb{N}$ is trivial; by the effective version of the Novikov-Kondo-Addison uniformization theorem [Mos82, Theorem 4E.4], we obtain $\mathbf{\Pi}_1^1\text{-}\mathsf{C}_\mathbb{N} \leqslant_{\mathrm{W}} \mathbf{\Pi}_1^1\text{-}\mathsf{UC}_\mathbb{N}$. Hence, $\widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{C}_\mathbb{N}} \equiv_{\mathrm{W}} \widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_\mathbb{N}}$.

By [KMP20, Theorem 3.11] $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_2}$ (in [KMP20] $\widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_2}$ is denoted by $\mathbf{\Delta}_1^1\text{-}\mathsf{CA}$). Since $\widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_2} \leqslant_{\mathrm{W}} \widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_\mathbb{N}}$ is trivial, we obtain $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_\mathbb{N}}$. To complete the proof of the equivalence between the first three problems, it suffices to show that $\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_\mathbb{N} \leqslant_{\mathrm{W}} \widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_2}$ and then notice that $\widehat{\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_2}$ is parallelizable. We can think of an input for $\mathbf{\Pi}_1^1\text{-}\mathsf{UC}_\mathbb{N}$ as a sequence $(T^i)_{i\in\mathbb{N}} \in \mathbf{Tr}^\mathbb{N}$ such that exactly one T^i belongs to $\mathcal{WF}$. For every i, we compute the pair of trees (S^i, R^i) where $S^i := \bigoplus_{j\leqslant i} T^j$ and $R^i := \bigoplus_{j>i} T^j$: notice that exactly one of S^i

and $R^i \in \mathcal{WF}$. We can view the sequence $(S^i, R^i)_{i\in\mathbb{N}}$ as an instance of $\widehat{\mathbf{\Pi}^1_1\text{-}\mathsf{UC}_2}$. Finally, let $n := \min\left\{m : \widehat{\mathbf{\Pi}^1_1\text{-}\mathsf{UC}_2}((S^i, R^i)_{i\in\mathbb{N}}))(m) = 0\right\}$. Clearly $T^n \in \mathcal{WF}$.

The last two (strict) reductions are [KMP20, Theorem 4.3] and [ADK21, Theorem 3.34]. □

For further reference, we collect here some facts which are implicit in the literature.

Proposition I.6.29. *If* $\mathbf{\Gamma} \in \{\mathbf{\Sigma}, \mathbf{\Pi}, \mathbf{\Delta}\}$ *and* $\mathbf{\Lambda} \in \{\mathbf{\Pi}, \mathbf{\Delta}\}$ *then*

$$^1\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \equiv_\mathrm{W} \mathbf{\Gamma}^1_1\text{-}\mathsf{UC}_\mathbb{N} \equiv_\mathrm{W} (\mathbf{\Gamma}^1_1\text{-}\mathsf{UC}_\mathbb{N})^{u*} \equiv_\mathrm{W} \mathbf{\Lambda}^1_1\text{-}\mathsf{C}_\mathbb{N} \equiv_\mathrm{W} (\mathbf{\Lambda}^1_1\text{-}\mathsf{C}_\mathbb{N})^{u*} <_\mathrm{W} {}^1\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_\mathrm{W} \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N}.$$

Proof. The equivalence $^1\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_\mathrm{W} \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N}$ is proved in [GPV21, Proposition 2.4]; essentially the same proof shows that $^1\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \equiv_\mathrm{W} \mathbf{\Sigma}^1_1\text{-}\mathsf{UC}_\mathbb{N}$.

If $A \subseteq \mathbb{N}$ is a singleton then $n \in A$ if and only if $(\forall m \neq n)(m \notin A)$. This implies that A is Σ^1_1 if and only if A is Π^1_1 if and only if A is Δ^1_1 and this shows that $\mathbf{\Sigma}^1_1\text{-}\mathsf{UC}_\mathbb{N} \equiv_\mathrm{W} \mathbf{\Pi}^1_1\text{-}\mathsf{UC}_\mathbb{N} \equiv_\mathrm{W} \mathbf{\Delta}^1_1\text{-}\mathsf{UC}_\mathbb{N}$. These equivalences, together with Proposition *I*.6.28 and Theorem *I*.6.10 allow us to derive all the stated equivalent characterizations of $^1\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$.

Since $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ is parallelizable and $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} <_\mathrm{W} \widehat{\mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N}}$ (Proposition *I*.6.28), we obtain that $\mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N} \not\leq_\mathrm{W} \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ and hence $^1\mathsf{UC}_{\mathbb{N}^\mathbb{N}} <_\mathrm{W} {}^1\mathsf{C}_{\mathbb{N}^\mathbb{N}}$. □

Proposition I.6.30. *For every n,*

(i) $^1(\mathsf{lim}^{(n)}) \equiv_\mathrm{W} \mathbf{\Pi}^0_{n+1}\text{-}\mathsf{C}_\mathbb{N} \equiv_\mathrm{W} (\mathsf{LPO}^{(n)})^{u*} <_\mathrm{W} {}^1(\mathsf{lim}^{(n+1)})$;

(ii) $\mathbf{\Pi}^0_{n+1}\text{-}\mathsf{C}_\mathbb{N} \not\leq_\mathrm{W} \mathsf{LPO}^{(n+1)}$;

(iii) $^1(\mathsf{lim}^{(n)}) <_\mathrm{W} \mathbf{\Pi}^1_1\text{-}\mathsf{C}_\mathbb{N} \equiv_\mathrm{W} {}^1\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$.

Proof. The equivalences in (i) are from [SV22, Theorem 7.2]. It follows from [SV22, Theorem 5.10(5)] that $\widehat{\mathsf{LPO}^{(n)u*}} \equiv_\mathrm{W} \widehat{\mathsf{LPO}^{(n)}} \equiv_\mathrm{W} \mathsf{lim}^{(n)}$; since $\mathsf{lim}^{(n)} <_\mathrm{sW} \mathsf{lim}^{(n+1)}$ this implies the nonreductions in (i) and (ii).

The equivalence in (iii) is from Proposition *I*.6.29, while the strictness follows from (i). □

1

Comparing problems via Weihrauch reducibility

The Weihrauch lattice, with a focus on $\mathbf{\Pi}^1_1{-}\mathsf{CA}_0$

In §I.2 and I.6 we have introduced computable analysis and Weihrauch reducibility, while in §I.5 we presented the program of reverse mathematics. The next chapter (and partly Chapter III) explore the interconnections between these two areas. The program aiming to provide a bridge between computable analysis and reverse mathematics via the framework of Weihrauch reducibility was initiated by Gherardi and Marcone in [GM09]. The starting point is noticing that many theorems from "ordinary mathematics" have the following Π^1_2 form

$$(\forall x \in X)(\varphi(x) \implies (\exists y \in Y)(\psi(x, y))).$$

This formulation has a natural translation as a computational problem: given an instance $x \in X$ satisfying $\varphi(x)$, the task is to find a solution $y \in Y$ such that $\psi(x, y)$. A computational problem can be naturally rephrased as a multi-valued function $f :\subseteq X \rightrightarrows Y$ (see Definition I.2.3) where, for every $x \in X$ such that $\varphi(x)$, $f(x) := \{y \in Y : \psi(x, y)\}$. The interpretation of theorems as multi-valued functions/problems, allows us to compare their uniform computational content using the framework of Weihrauch reducibility and this shows a close connection between reverse mathematics and computable analysis. One of the main topics of this part is the prosecution of this program at the "higher levels" of reverse mathematics. Indeed, a main theme in this area is the identification of some "analogs" of the big-five axiom systems of reverse mathematics in the Weihrauch lattice. Most of the work has been done for the first three, and we can summarize the results obtained in this area (informally) as follows:

- RCA_0 "corresponds" to id;
- WKL_0 "corresponds" to WKL (that is Weihrauch equivalent to $\mathsf{C}_{2^\mathbb{N}}$);
- ACA_0 "corresponds" to iterations of the jump of lim.

Less work has been done in finding the Weihrauch "analog" of ATR_0 and $\mathbf{\Pi}^1_1{-}\mathsf{CA}_0$. In [BKMP16], Marcone raised the question "What do the Weihrauch hierarchies look like once we go to very high levels of reverse mathematics strength?": we start discussing the picture for ATR_0. The situation here is more "messy" than the one for RCA_0, WKL_0 and ACA_0, in the sense that ATR_0 has more "natural" analogs of different computational strength in the Weihrauch lattice. One of the reasons is that the translation of a mathematical theorem to a computational problem is not unique. As observed by [KMP20, §6], at the level of ATR_0 many theorems have a disjunctive form $\varphi \vee \psi$, and when formulating them as computational problems, we can interpret them in two different ways, namely either $\neg\varphi \implies \psi$ or $\neg\psi \implies \varphi$. For example, the perfect tree theorem, which in reverse mathematics is equivalent to ATR_0 (see [Sim09, Theorem V.4.3]), states that a tree with uncountably many paths has a perfect subtree. This leads to two different problems:

- given a tree with uncountably many paths output a nonempty perfect subtree of it;
- given a tree with countably many paths output a witness of the countability, i.e. enumerate the paths in it.

In [KMP20, Proposition 6.3, Theorem 6.4], the authors show that the first problem is Weihrauch equivalent to $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$, while (variations of) the second one to $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ and hence, by Proposition I.6.28 they belong to different Weihrauch degrees. In §II.1 and §II.3 we study similar problems related to perfect subsets of arbitrary metric spaces studying how these relate to the problems considered in [KMP20]. Notice that also other theorems at the level of ATR_0 have a different computational strength depending on how they are formulated: this is the case for open determinacy (see [Sim09, Definition V.8.1] for its definition and [KMP20, §6.2] for results on the Weihrauch degree of the corresponding problems) and for the existence of jump hierarchies along well-orders. The problems related to the latter have been introduced and studied in [KMP20] and [Goh19, LG19]. Other examples can be found again in [KMP20] where the authors also considered principles related to the (strong) comparability of well-orders, in [Goh19, LG19, Goh20] where Goh studied the weak comparability of well-orders and the König duality theorem. Anglès D'Auriac and Kihara in [ADK21] studied variants of $\mathbf{\Sigma}^1_1$-$\mathsf{C}_\mathbb{N}$ while Marcone and Valenti in [MV21] studied infinite dimensional generalizations of Ramsey's theorem like the open and clopen Ramsey theorem. To summarize the results we have discussed so far around ATR_0 (and some that we have not mentioned, but that can be found in the papers cited above) we can say that:

- the analogues of ATR_0 are around $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$, $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ and $\mathsf{TC}_{\mathbb{N}^\mathbb{N}}$.

The situation for $\mathbf{\Pi}^1_1-\mathsf{CA}_0$ seems to be clearer: [Sim09, Theorem VI.1.1] tells us that the subsystem of second order arithmetic $\mathbf{\Pi}^1_1-\mathsf{CA}_0$ is equivalent, over RCA_0 to the statement "If $\{T_i \in: i \in \mathbb{N}\}$ is a sequence of trees, then there is a set Z such that, for all $n \in \mathbb{N}$, $n \in Z$ if and only if $[T_n] = \varnothing$". We have already anticipated in §I.6, that such a statement has a natural correspondent in the Weihrauch lattice, that has been studied in some of the papers mentioned above, namely $\widehat{\mathsf{WF}}$. Despite this natural correspondence, we show that, even in this case, we have some differences with respect to the results in reverse mathematics. Other problems at this level, related to trees and graphs, have been studied for example in [Hir19] and [BHW21] but, to the best of our knowledge, Chapter II beyond studying also problems at the level of ATR_0, is the first systematic study of problems at the level of $\mathbf{\Pi}^1_1-\mathsf{CA}_0$ in the Weihrauch lattice. Chapter III is focused on the study of graph related problems in general, and some of these, in reverse mathematics, are equivalent to $\mathbf{\Pi}^1_1-\mathsf{CA}_0$, see [HL96, BHW21].

II

The Cantor-Bendixson theorem

All the results in this chapter are a joint work with Alberto Marcone and Manlio Valenti and can be found in [CMV22].

Here we study the uniform computational strength of theorems arising in classical descriptive set theory related to perfect subsets of Polish spaces. The paper is organized as follows. In §II.1 we study multi-valued functions related to the perfect set and perfect tree theorem in Baire and Cantor space, while in §II.2 we consider problems related to the Cantor-Bendixson theorem in the same setting. In §II.3 we study the problems considered in §II.1 and §II.2 for arbitrary computable metric spaces, while §II.4 lists some open problems that remain to be solved. Figures II.1 and II.2 summarize some of our results. The precise definitions of the various functions are given in due time.

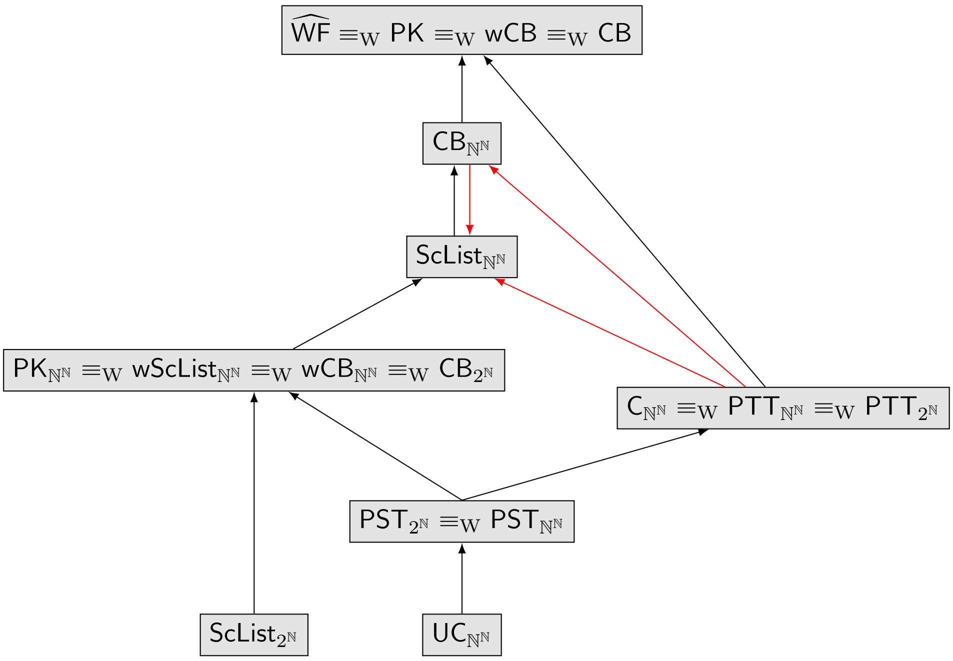

Figure II.1: Some multi-valued functions studied in this chapter. Black arrows represent Weihrauch reducibility in the direction of the arrow. Red arrows mean that the existence of a reduction is still open. If a function cannot be reached from another one following a path of arrows we know that there is no reduction between the two functions.

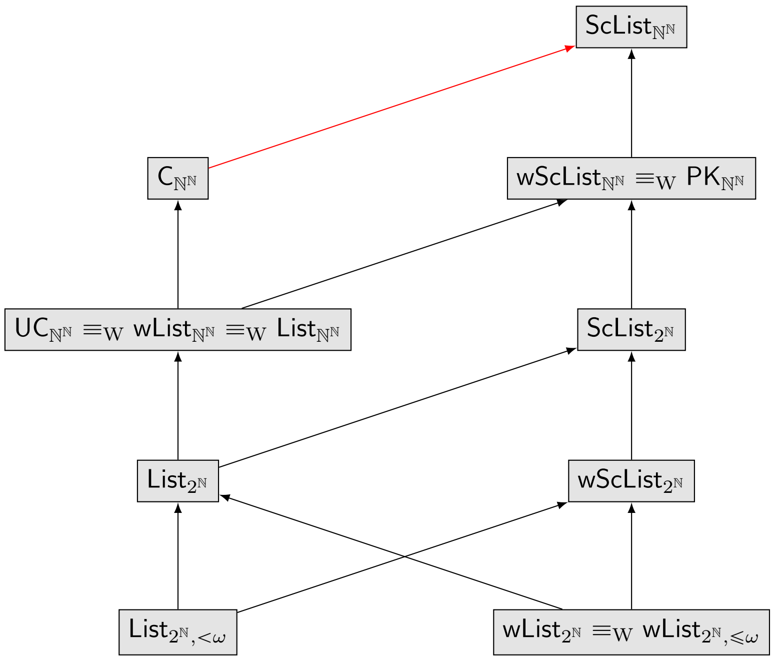

Figure II.2: Multi-valued functions related to listing problems in the Weihrauch lattice. The arrows have the same meaning as in Figure II.1.

II.1 The perfect set theorem in $\mathbb{N}^\mathbb{N}$ and $2^\mathbb{N}$

II.1.1 Perfect sets

The following multi-valued function was introduced and studied in [KMP20].

Definition II.1.1. The multi-valued function $\mathsf{PTT}_1 :\subseteq \mathbf{Tr} \rightrightarrows \mathbf{Tr}$ has domain $\{T \in \mathbf{Tr} : T \in \mathcal{T}^{>\aleph_0}\}$ and is defined by

$$\mathsf{PTT}_1(T) := \{S \in \mathbf{Tr} : S \subseteq T \land S \text{ is perfect}\}.$$

We also study $\mathsf{PTT}_1 \restriction \mathbf{Tr}_2$, the restriction of PTT_1 to $\mathbf{Tr}_2$. We now define the same problem for closed sets.

Definition II.1.2. Let $\mathcal{X}$ be a computable Polish space. The multi-valued function $\mathsf{PST}_\mathcal{X} :\subseteq \mathcal{A}_-(\mathcal{X}) \rightrightarrows \mathcal{A}_-(\mathcal{X})$ has domain $\{A \in \mathcal{A}_-(\mathcal{X}) : |A| > \aleph_0\}$ and is defined as

$$\mathsf{PST}_\mathcal{X}(A) := \{P \in \mathcal{A}_-(\mathcal{X}) : P \subseteq A \land P \text{ is perfect}\}.$$

Using the tree representation of closed sets in $\mathbb{N}^\mathbb{N}$, we can think of a name for an input of $\mathsf{PST}_{\mathbb{N}^\mathbb{N}}$ as $T \in \mathcal{T}^{>\aleph_0}$ and a name for a solution of $\mathsf{PST}_{\mathbb{N}^\mathbb{N}}([T])$ as $S \in \mathbf{Tr}$ such that $[S] \subseteq [T]$ and $[S]$ is perfect. Notice that $\mathsf{PST}_{\mathbb{N}^\mathbb{N}}([T])$ contains $\mathsf{PTT}_1(T)$, but includes also every tree with perfect body contained in $[T]$.

Theorem II.1.3. $\mathsf{PTT}_1 \restriction \mathbf{Tr}_2 \equiv_{\mathrm{sW}} \mathsf{PTT}_1$ *and* $\mathsf{PST}_{2^\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{PST}_{\mathbb{N}^\mathbb{N}}$.

Proof. The reduction $\mathsf{PTT}_1 \restriction \mathbf{Tr}_2 \leqslant_{\mathrm{sW}} \mathsf{PTT}_1$ is trivial.

We now prove that $\mathsf{PST}_{2^\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{PST}_{\mathbb{N}^\mathbb{N}}$. Let $T \in \mathbf{Tr}_2$ and let the forward functional be the identity. Let P be a name for $\mathsf{PST}_{\mathbb{N}^\mathbb{N}}([T])$: even if $P \in \mathbf{Tr}$, notice that $[P]$ is perfect and $[P] \subseteq 2^\mathbb{N}$. Let $\Psi(P) = \{\sigma \in P : \sigma \in 2^{<\mathbb{N}}\}$ and notice that $\Psi(P)$ is a name for an element $\mathsf{PST}_{2^\mathbb{N}}([T])$.

For the other direction, $\mathsf{PTT}_1 \leqslant_{\mathrm{sW}} \mathsf{PTT}_1 \restriction \mathbf{Tr}_2$ is witnessed by the maps $\rho_{2^\mathbb{N}}$ (forward) and $\rho_{\mathbb{N}^\mathbb{N}}$ (backward) from Definition $I.1.5$. Let $T \in \mathcal{T}^{>\aleph_0}$ and let $P \in \mathsf{PTT}_1 \restriction \mathbf{Tr}_2(\rho_{2^\mathbb{N}}(T))$. By Lemma $I.1.9$, $\rho_{\mathbb{N}^\mathbb{N}}(P)$ is a perfect tree. To show that $\rho_{\mathbb{N}^\mathbb{N}}(P) \subseteq T$, it suffices to prove that $[\rho_{\mathbb{N}^\mathbb{N}}(P)] \subseteq [T]$. Let $f \in [\rho_{\mathbb{N}^\mathbb{N}}(P)]$: by Lemma $I.1.8(6)$ we have that $\rho_{2^\mathbb{N}}(f) \in [P] \subseteq [\rho_{2^\mathbb{N}}(T)]$ and by Lemma $I.1.8(4)$ we conclude that $f \in [T]$.

The proof that $\mathsf{PST}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{PST}_{2^\mathbb{N}}$ is similar. □

Lemma II.1.4. $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{sW}} \mathsf{PST}_{\mathbb{N}^\mathbb{N}}$ *and* $\mathsf{PST}_{\mathbb{N}^\mathbb{N}} \not\leqslant_{\mathrm{W}} \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$.

Proof. Since $\mathsf{PST}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{PST}_{2^\mathbb{N}}$ (Theorem $II.1.3$), we prove the lemma with $\mathsf{PST}_{2^\mathbb{N}}$ in place of $\mathsf{PST}_{\mathbb{N}^\mathbb{N}}$.

To show $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{PST}_{2^\mathbb{N}}$, fix $T \in \mathcal{UB}$ and let p_0 be the unique element of $[T]$. Let $S := \rho_{2^\mathbb{N}}(\mathsf{Expl}(T))$ and notice that, by definition of $\mathsf{Expl}(\cdot)$ and by Lemma $I.1.8(4)$, all paths in $[S]$ are either eventually zero or are of the form $\rho_{2^\mathbb{N}}(p_0 \oplus q)$ for some $q \in 2^\mathbb{N}$.

Fix a name P for an element of $\mathsf{PST}_{2^\mathbb{N}}([S])$. We claim that all the paths in $[P]$ are of the form $\rho_{2^\mathbb{N}}(p_0 \oplus q)$ for some $q \in 2^\mathbb{N}$. To prove this, we need to rule out that some eventually zero path belongs to $[P]$. Let $r \in 2^\mathbb{N}$ be of the form $\sigma 0^\mathbb{N}$, where $\sigma = \langle\rangle$ or $\sigma(|\sigma|-1) = 1$. Notice that $\rho_{2^\mathbb{N}}(\rho_{\mathbb{N}^\mathbb{N}}(\sigma)) = \sigma$. Let

$$k := \begin{cases} p_0(|\rho_{\mathbb{N}^\mathbb{N}}(\sigma)|/2) & \text{if } |\rho_{\mathbb{N}^\mathbb{N}}(\sigma)| \text{ is even,} \\ 1 & \text{otherwise,} \end{cases}$$

and set $m = |\sigma| + k + 1$. It suffices to prove that

$$(\forall q \in 2^\mathbb{N})(r[m] \not\sqsubseteq \rho_{2^\mathbb{N}}(p_0 \oplus q)),$$

so that all paths in $[S]$ which extend $r[m]$ are eventually zero and $S_{r[m]} \in \mathcal{T}^{\leqslant\aleph_0}$, which implies $r \notin [P]$. Fix $q \in 2^\mathbb{N}$:

- if $\sigma \not\sqsubseteq \rho_{2^\mathbb{N}}(p_0 \oplus q)$ then $r[m] \not\sqsubseteq \rho_{2^\mathbb{N}}(p_0 \oplus q)$;
- if $\sigma \sqsubset \rho_{2^\mathbb{N}}(p_0 \oplus q)$ then either $\sigma 0^k 1$ or $\sigma 0^{k-1} 1$ (in case $|\rho_{\mathbb{N}^\mathbb{N}}(\sigma)|$ is odd and $q((|\rho_{\mathbb{N}^\mathbb{N}}(\sigma)| - 1)/2) = 0$) is a prefix of $\rho_{2^\mathbb{N}}(p_0 \oplus q)$ which is incomparable with $r[m] = \sigma 0^{k+1}$; hence $r[m] \not\sqsubseteq \rho_{2^\mathbb{N}}(p_0 \oplus q)$ also in this case.

This concludes the proof of the claim.

We show how to computably retrieve p_0 from P. To find $p_0(0)$ we search for n such that

$$(\forall \tau \in 2^{n+1})(P_\tau \in \mathcal{IF}_2 \implies \tau = 0^n 1).$$

Indeed, the previous claim implies that the unique n satisfying this condition is $p_0(0)$. Since $\mathcal{IF}_2$ is a Π^0_1 set (Theorem $I.3.39(i)$), the above condition is Σ^0_1 and at some finite stage we find $p_0(0)$.

Suppose now that we computed the first i coordinates of p_0, i.e. $p_0[i]$. We generalize the previous strategy to compute $p_0(i)$. Let

$$A_i := \{0^{p_0(0)} 1 \xi_0 0^{p_0(1)} 1 \xi_1 \dots \xi_{i-2} 0^{p_0(i-1)} 1 \xi_{i-1} \in P : (\forall j < i)(\xi_j \in \{1, 01\})\}$$

(recall that $\rho_{2^{\mathbb{N}}}(0) = 1$ and $\rho_{2^{\mathbb{N}}}(1) = 01$). Informally, the ξ_j's come from the interleaving of sequences in T with sequences in $2^{<\mathbb{N}}$. Notice that A_i is finite and nonempty; moreover, there exists $\sigma \in A_i$ which is the prefix of some path in $[P]$. We search for n satisfying the Σ^0_1 property

$$(\forall \sigma \in A_i)(\forall \tau \in 2^{n+1})(P_{\sigma\tau} \in \mathcal{IF}_2 \implies \tau = 0^n 1)$$

As before the claim implies that the unique n satisfying this condition is $p_0(i)$. The main difference with the case $i = 0$ is that it may be the case that different sequences in A_i are prefixes of paths in $[P]$ that come from the interleaving of p_0 with different elements of $2^{\mathbb{N}}$. However, any such σ provides the correct $n = p_0(i)$. This concludes the proof of the reduction.

To show that $\mathsf{PST}_{2^{\mathbb{N}}} \not\leqslant_{\mathrm{W}} \mathsf{UC}_{\mathbb{N}^{\mathbb{N}}}$, recall from §I.6 that $\mathsf{lim} * \mathsf{UC}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{UC}_{\mathbb{N}^{\mathbb{N}}}$, so it suffices to show that $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{lim} * \mathsf{PST}_{2^{\mathbb{N}}}$. From [KMP20, Proposition 6.3] and Theorem *II*.1.3, it follows that $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{PTT}_1 \equiv_{\mathrm{W}} \mathsf{PTT}_1 \restriction \mathbf{Tr}_2$ and hence it is enough to show that $\mathsf{PTT}_1 \restriction \mathbf{Tr}_2 \leqslant_{\mathrm{W}} \mathsf{lim} * \mathsf{PST}_{2^{\mathbb{N}}}$. From [NP19], we know that lim is equivalent to the function that prunes elements in $\mathbf{Tr}_2$. So let $T \in \mathcal{T}_2^{>\aleph_0}$ be the input of $\mathsf{PTT}_1 \restriction \mathbf{Tr}_2$ and let P be a name for an element of $\mathsf{PST}_{2^{\mathbb{N}}}([T])$: pruning P with lim is enough to obtain a perfect subtree of T. □

Recall that trees are represented via their characteristic functions and notice that $s \in 2^{<\mathbb{N}}$ is a prefix of a (name for a) tree if and only if $\{\tau : s(\tau) = 1\}$ is a tree, which is a computable property.

Lemma II.1.5. ${}^1\mathsf{PST}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathbf{\Pi}^1_1\text{-}\mathsf{C}_{\mathbb{N}}$.

Proof. The fact that $\mathbf{\Pi}^1_1\text{-}\mathsf{C}_{\mathbb{N}} \leqslant_{\mathrm{W}} {}^1\mathsf{PST}_{\mathbb{N}^{\mathbb{N}}}$ follows from the fact that ${}^1\mathsf{UC}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathbf{\Pi}^1_1\text{-}\mathsf{C}_{\mathbb{N}}$ by Proposition *I*.6.29 and $\mathsf{UC}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{sW}} \mathsf{PST}_{\mathbb{N}^{\mathbb{N}}}$ by Lemma *II*.1.4.

For the opposite direction, suppose that f is a first-order problem such that $f \leqslant_{\mathrm{W}} \mathsf{PST}_{\mathbb{N}^{\mathbb{N}}}$ as witnessed by the computable maps Φ and Ψ. Let p be a name for an input x of f. Then $\Phi(p) = T$ where $T \in \mathcal{T}^{>\aleph_0}$. Consider the set

$$\mathbf{Prefixes} := \{s : s \text{ is a prefix of a tree } \wedge \Psi(p[|s|], s)(0)\downarrow \wedge (\forall \tau)(s(\tau) = 0 \implies T_\tau \in \mathcal{T}^{\leqslant\aleph_0})\}.$$

Since $\mathcal{T}^{\leqslant\aleph_0}$ is a Π^1_1 set (see Theorem *I*.3.39(*ii*)), **Prefixes** is a $\Pi^{1,T}_1$ subset of $\mathbb{N}$.

We prove that **Prefixes** is nonempty. Let q be a name for the perfect kernel of $[T]$, which belongs to $\mathsf{PST}_{\mathbb{N}^{\mathbb{N}}}([T])$. Let t be the least stage such that $\Psi(p[t], q[t])(0)\downarrow$. Then, if $q(\tau) = 0$ then $T_\tau \in \mathcal{T}^{\leqslant\aleph_0}$ (otherwise T_τ contains some perfect subset of T contradicting that q is a name for the perfect kernel of $[T]$). This proves that $q[t] \in \mathbf{Prefixes}$.

Thus, **Prefixes** is a valid input for $\mathbf{\Pi}^1_1\text{-}\mathsf{C}_{\mathbb{N}}$. The argument above shows that every $s \in$ **Prefixes** is a prefix of a name for the perfect kernel of $[T]$, which belongs to $\mathsf{PST}_{\mathbb{N}^{\mathbb{N}}}([T])$. Since f is first-order, for any such s, $\Psi(p[|s|], s)(0) \in f(x)$. This shows that $f \leqslant_{\mathrm{W}} \mathbf{\Pi}^1_1\text{-}\mathsf{C}_{\mathbb{N}}$. □

Our results about $\mathsf{PST}_{2^{\mathbb{N}}}$ and $\mathsf{PST}_{\mathbb{N}^{\mathbb{N}}}$ are summarized in the following theorem.

Theorem II.1.6. $\mathsf{UC}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{sW}} \mathsf{PST}_{2^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{PST}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{sW}} \mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$

Proof. The first strict reduction and the first equivalence were proven in Lemma *II*.1.4 and Theorem *II*.1.3 respectively. The last reduction follows from the fact $\mathsf{PTT}_1 \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ ([KMP20, Proposition 6.3]) and a solution for PTT_1 is also a solution for $\mathsf{PST}_{\mathbb{N}^{\mathbb{N}}}$. Strictness follows by Lemma *II*.1.5 as, by Proposition *I*.6.29, $\mathbf{\Pi}^1_1\text{-}\mathsf{C}_{\mathbb{N}} <_{\mathrm{W}} {}^1\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$. □

II.1.2 Listing problems

We now move our attention to the functions that, given in input a countable closed set of a computable metric space, output a list of all its elements. There are different possible meanings of the word 'list', and these correspond to different functions. For Baire and Cantor space, some of these functions were already introduced and studied in [KMP20]. For trees and closed sets, we made a distinction between the perfect tree and the perfect set theorem; on the other hand, if $T \in \mathbf{Tr}$ and $A \in \mathcal{A}_-(\mathbb{N}^\mathbb{N})$ are such that $A = [T]$ then listing the elements in $[T]$ and listing the elements in A are the same problem.

We generalize [KMP20, Definition 6.1] from $\mathbb{N}^\mathbb{N}$ to an arbitrary computable metric space.

Definition II.1.7. Let $\mathcal{X}$ be a computable metric space. The two multi-valued function $\mathsf{wList}_\mathcal{X} :\subseteq \mathcal{A}_-(\mathcal{X}) \rightrightarrows (2 \times \mathcal{X})^\mathbb{N}$ and $\mathsf{List}_\mathcal{X} :\subseteq \mathcal{A}_-(\mathcal{X}) \rightrightarrows \mathbb{N} \times (2 \times \mathcal{X})^\mathbb{N}$ with the same domain $\{A \in \mathcal{A}_-(\mathcal{X}) : |A| \leqslant \aleph_0\}$ are defined by

$$\mathsf{wList}_\mathcal{X}(A) := \{(b_i, x_i)_{i\in\mathbb{N}} : A = \{x_i : b_i = 1\}\},$$

$$\mathsf{List}_\mathcal{X}(A) := \{(n, (b_i, x_i)_{i\in\mathbb{N}}) : A = \{x_i : b_i = 1\} \wedge ((n = 0 \wedge |A| = \aleph_0) \vee (n > 0 \wedge |A| = n - 1))\}.$$

In the remaining part of this chapter, we sometimes need to consider finite prefixes $\sigma \in \mathbb{N}^{<\mathbb{N}}$ of some infinite sequence $f \in \mathbb{N}^\mathbb{N}$ defined as the join of countably many infinite sequences, i.e. $f := \oplus_{i\in\mathbb{N}} g_i$ where $g_i \in \mathbb{N}^\mathbb{N}$: furthermore given σ, we want to retrieve the prefix of some specific g_i. For these purposes, we give the following definition, related to the notation $\sigma := \mathsf{dvt}(\tau_0, \dots, \tau_m)$ that can be found in [MV21, Theorem 4.11].

Definition II.1.8. Given $\sigma \in \mathbb{N}^{<\mathbb{N}}$, we define $\ell_\sigma := \min\{i : \langle i, 0\rangle \geqslant |\sigma|\}$. Then, for every $i < \ell_\sigma$, we define $\pi_i(\sigma) := \langle \sigma(\langle i, j\rangle) : \langle i, j\rangle < |\sigma|\rangle$ where $|\pi_i(\sigma)| = \max\{j : \langle i, j\rangle < |\sigma|\}$.

Informally, thinking of σ as the prefix of some $f := \oplus_{i\in\mathbb{N}} g_i$, we obtain that ℓ_σ is the least i such that σ contains a prefix of g_i and $\pi_i(\sigma)$ is the prefix of g_i contained in σ.

Remark II.1.9. Notice that there is a slight difference between List (there was no subscript there because only $\mathbb{N}^\mathbb{N}$ was considered) in [KMP20, Definition 6.1] and our definition of $\mathsf{List}_{\mathbb{N}^\mathbb{N}}$. Indeed, $\mathsf{List} :\subseteq \mathcal{A}_-(\mathbb{N}^\mathbb{N}) \rightrightarrows (\mathbb{N}^\mathbb{N})^\mathbb{N}$ is defined by stipulating that $(n, (p_i)_{i\in\mathbb{N}}) \in \mathsf{List}(A)$ if and only if either $n = 0$, $A = \{p_i : i \in \mathbb{N}\}$ and $p_i \neq p_j$ for every $i \neq j$, or else $n > 0$, $|A| = n - 1$ and $A = \{p_i : i < n - 1\}$.

In particular, the output of List is always injective: this version is apparently stronger than $\mathsf{List}_{\mathbb{N}^\mathbb{N}}$ because the latter allows repeating elements of the form $(1, p_i)$ for $p_i \in A$. We briefly discuss why $\mathsf{List} \equiv_{\mathrm{sW}} \mathsf{List}_{\mathbb{N}^\mathbb{N}}$.

We claim that given $(n, (b_i, p_i)_{i\in\mathbb{N}}) \in \mathsf{List}_{\mathbb{N}^\mathbb{N}}(A)$ we can compute some I such that $(n, I) \in \mathsf{List}(A)$. Let $L := (b_i p_i)_{i\in\mathbb{N}}$ and recall that $(b_i p_i)^- = p_i$. At any finite stage s we inspect the finite prefix $L[s]$ of L where $\pi_i(L[s])^- \sqsubset p_i$. We start listing $\pi_i(L[s])^-$ in I when we see that $\pi_i(L[s])(0) = 1$ and $\pi_i(L[s])^- \nsubseteq \pi_j(L[s])^-$ for every $j < i$ such that $\pi_j(L[s])(0) = 1$.

If $n > 0$ (i.e. A is finite) after we listed $n - 1$ elements we can add to I any element in $\mathbb{N}^\mathbb{N}$ having as first digit 0. If $n = 0$ (i.e. A is infinite) then we always find new elements to list in I, and we continue forever. Since we are listing each p_i only if $p_i \neq p_j$ for every $j < i$ with $b_j = 1$, we have that I lists injectively all the elements of A.

We now focus on listing problems in Cantor space, comparing $\mathsf{wList}_{2^\mathbb{N}}$ and $\mathsf{List}_{2^\mathbb{N}}$ with the analogous problems in Baire space and with the functions $\mathsf{List}_{2^\mathbb{N},<\omega}$ and $\mathsf{wList}_{2^\mathbb{N},\leqslant\omega}$ considered in [KMP20, §6.1].

Definition II.1.10. The multi-valued functions

$$\mathsf{List}_{2^\mathbb{N},<\omega} :\subseteq \mathcal{A}_-(2^\mathbb{N}) \rightrightarrows (2^\mathbb{N})^{<\mathbb{N}} \text{ and } \mathsf{wList}_{2^\mathbb{N},\leqslant\omega} :\subseteq \mathcal{A}_-(2^\mathbb{N}) \rightrightarrows (2^\mathbb{N})^\mathbb{N}$$

have domains $\{A \in \mathcal{A}_-(2^\mathbb{N}) : |A| < \aleph_0\}$ and $\{A \in \mathcal{A}_-(2^\mathbb{N}) : |A| \leqslant \aleph_0 \wedge A \neq \emptyset\}$ respectively and are defined by

$$\mathsf{List}_{2^{\mathbb{N}},<\omega}(A) := \{(p_i)_{i<n} : A = \{p_i : i < n\}\};$$
$$\mathsf{wList}_{2^{\mathbb{N}},\leqslant\omega}(A) := \{(p_i)_{i\in\mathbb{N}} : A = \{p_i : i \in \mathbb{N}\}\}.$$

The following theorem establishes the relations between the listing problems defined so far: the results stated in this theorem are collected with other ones in Figure II.2.

Theorem II.1.11.

- $\mathsf{List}_{2^{\mathbb{N}},<\omega} \mid_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}},\leqslant\omega} \equiv_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}}}$ *and*
- $\mathsf{wList}_{2^{\mathbb{N}},\leqslant\omega}, \mathsf{List}_{2^{\mathbb{N}},<\omega} <_{\mathrm{W}} \mathsf{List}_{2^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{UC}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{wList}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{List}_{\mathbb{N}^{\mathbb{N}}}$.

Since all the problems involved are cylinders all the reductions mentioned above are strong.

Proof. The fact that $\mathsf{List}_{2^{\mathbb{N}},<\omega} \mid_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}},\leqslant\omega}$ is [KMP20, Corollary 6.15].

To prove that $\mathsf{wList}_{2^{\mathbb{N}},\leqslant\omega} \leqslant_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}}}$, let $A \in \mathcal{A}_-(2^{\mathbb{N}})$ be countable and nonempty and let $(b_i, p_i)_{i\in\mathbb{N}} \in \mathsf{wList}_{2^{\mathbb{N}}}(A)$. Then, $\{p_i : b_i = 1\}$ can be easily rearranged, and possibly duplicated, to produce an element of $\mathsf{wList}_{2^{\mathbb{N}},\leqslant\omega}(A)$.

For the opposite direction, let $A \in \mathcal{A}_-(2^{\mathbb{N}})$ be countable and possibly empty. Define $A' := \{0^{\mathbb{N}}\} \cup \{1x : x \in A\}$: A' is still countable but nonempty, i.e. a suitable input for $\mathsf{wList}_{2^{\mathbb{N}},\leqslant\omega}$. Let $(p_i)_{i\in\mathbb{N}} \in \mathsf{wList}_{2^{\mathbb{N}},\leqslant\omega}(A')$: then $(p_i(0), p_i^-)_{i\in\mathbb{N}} \in \mathsf{wList}_{2^{\mathbb{N}}}(A)$.

The reductions $\mathsf{wList}_{2^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{List}_{2^{\mathbb{N}}}$ and $\mathsf{List}_{2^{\mathbb{N}},<\omega} \leqslant_{\mathrm{W}} \mathsf{List}_{2^{\mathbb{N}}}$ are immediate. Strictness follows from incomparability of $\mathsf{wList}_{2^{\mathbb{N}}}$ and $\mathsf{List}_{2^{\mathbb{N}},<\omega}$.

By Remark *II*.1.9 and [KMP20, Theorem 6.4] we obtain that $\mathsf{UC}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{wList}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{List}_{\mathbb{N}^{\mathbb{N}}}$.

The reduction $\mathsf{List}_{2^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{List}_{\mathbb{N}^{\mathbb{N}}}$ is obvious. To prove that $\mathsf{List}_{\mathbb{N}^{\mathbb{N}}} \nleqslant_{\mathrm{W}} \mathsf{List}_{2^{\mathbb{N}}}$ we recall that by Proposition *I*.6.30(*i*) and (*iii*) $\boldsymbol{\Pi}^0_3\text{-}\mathsf{C}_{\mathbb{N}} <_{\mathrm{W}} \boldsymbol{\Pi}^1_1\text{-}\mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} {}^1\mathsf{UC}_{\mathbb{N}^{\mathbb{N}}}$. It thus suffices to show that ${}^1\mathsf{List}_{2^{\mathbb{N}}} \leqslant_{\mathrm{W}} \boldsymbol{\Pi}^0_3\text{-}\mathsf{C}_{\mathbb{N}}$.

Let f be a first-order function and suppose that $f \leqslant_{\mathrm{W}} \mathsf{List}_{2^{\mathbb{N}}}$ as witnessed by the maps Φ and Ψ. Let p be a name for an input of f. Then $\Phi(p) = T$ where $T \in \mathcal{T}_2^{\leqslant\aleph_0}$. Let $\varphi(n, T)$ be the formula $(\exists\sigma_0, \dots, \sigma_{n-1})(\forall i \neq j < n)(\sigma_i \mid \sigma_j \wedge T_{\sigma_i} \in \mathcal{IF}_2)$. Notice that $\mathcal{IF}_2$ is a Π^0_1 set (see Theorem *I*.3.39(*i*)) and hence φ is Σ^0_2.

Let **Prefixes** be the set of all $(n, (\tau, \sigma)) \in \mathbb{N} \times 2^{<\mathbb{N}} \times 2^{<\mathbb{N}}$ such that

- $|\tau| = \ell_\sigma$;
- $\Psi\big(p[\sigma], \langle\tau(i), \pi_i(\sigma)\rangle_{i<\ell_\sigma}\big)(0)\downarrow$;
- $(\forall i < \ell_\sigma)(\tau(i) = 1 \implies T_{\pi_i(\sigma)} \in \mathcal{IF}_2)$;
- $(n = 0 \wedge (\forall k)(\varphi(k, T))) \vee (n > 0 \wedge \varphi(n-1, T) \wedge \neg\varphi(n, T))$.

Elements in $\mathbb{N} \times 2^{<\mathbb{N}} \times 2^{<\mathbb{N}}$ can be coded as natural numbers, hence **Prefixes** is a $\Pi^{0,T}_3$ subset of $\mathbb{N}$.

We claim that **Prefixes** $\neq \emptyset$, so that **Prefixes** is a valid input for $\boldsymbol{\Pi}^0_3\text{-}\mathsf{C}_{\mathbb{N}}$. Let (n, L) be an element of $\mathsf{List}_{2^{\mathbb{N}}}([T])$ and let s be the least stage such that $\Psi\big(p[s], \langle n, L[s]\rangle\big)(0)\downarrow$. Then $n = 0$ implies that $|[T]| = \aleph_0$, while if $n > 0$ then $|[T]| = n - 1$. Furthermore, $L[s]$ is of the form $(\tau(i), \pi_i(\sigma))_{i<\ell_\sigma}$ for some $(\tau, \sigma) \in 2^{<\mathbb{N}} \times 2^{<\mathbb{N}}$ such that $|\tau| = \ell_\sigma$. It is immediate that if $\tau(i) = 1$ then $\pi_i(\sigma)$ is the initial segment of a path in T. We obtain that $(n, (\tau, \sigma)) \in$ **Prefixes**, and this concludes the proof of the claim. The same argument shows that every $(n, (\tau, \sigma)) \in$ **Prefixes** computes a prefix for an element of $\mathsf{List}_{2^{\mathbb{N}}}([T])$. Since f is first-order, $\Psi\big(p[\sigma], \langle n, (\tau(i), \pi_i(\sigma))\rangle_{i<\ell_\sigma}\big)(0) \in f(x)$. This shows that $f \leqslant_{\mathrm{W}} \boldsymbol{\Pi}^0_3\text{-}\mathsf{C}_{\mathbb{N}}$. □

Lemma II.1.12. $\mathsf{wList}_{2^{\mathbb{N}}}$ *is parallelizable.*

Proof. It suffices to show that $\widehat{\mathsf{wList}_{2^{\mathbb{N}}}} \leqslant_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}}}$. Given $(T^n)_{n\in\mathbb{N}}$, an input for $\widehat{\mathsf{wList}_{2^{\mathbb{N}}}}$, i.e. a sequence of binary trees with countable body, compute $T := \bigsqcup^{\mathrm{b}}_{n\in\mathbb{N}} T^n$ and notice that $T \in \mathbf{Tr}_2$ (see Remark *I*.1.3). Given $(b_i, p_i) \in \mathsf{wList}_{2^{\mathbb{N}}}(T)$, it is straightforward to check that $\{(b_i, p_i) : b_i = 1 \land 0^n 1 \sqsubset p_i\} \in \mathsf{wList}_{2^{\mathbb{N}}}(T^n)$. □

We conclude this section by characterizing the first-order part of $\mathsf{wList}_{2^{\mathbb{N}}}$.

Lemma II.1.13. ${}^1\mathsf{wList}_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$.

Proof. We first show that ${}^1\mathsf{wList}_{2^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$. Let f be a first-order function such that $f \leqslant_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}}}$ as witnessed by the maps Φ and Ψ. Let p be a name for an input of f. Then $\Phi(p) = T$ where $T \in \mathcal{T}_2^{\leqslant\aleph_0}$. Let **Prefixes** be the set of all $(\tau, \sigma) \in 2^{<\mathbb{N}} \times 2^{<\mathbb{N}}$ such that

- $|\tau| = \ell_\sigma$;
- $\Psi\big(p[\sigma], \langle \tau(i), \pi_i(\sigma)\rangle_{i<\ell_\sigma}\big)(0)\downarrow$ in $|\sigma|$ steps;
- $(\forall i < \ell_\sigma)(\tau(i) = 1 \implies T_{\pi_i(\sigma)} \in \mathcal{IF}_2)$.

Elements in $2^{<\mathbb{N}} \times 2^{<\mathbb{N}}$ can be coded as natural numbers and since $\mathcal{IF}_2$ is a Π^0_1 set (Theorem *I*.3.39(*i*)), **Prefixes** is a $\Pi^{0,T}_1$ subset of $\mathbb{N}$.

We claim that **Prefixes** $\neq \varnothing$, so that **Prefixes** is a valid input for $\mathsf{C}_{\mathbb{N}}$. Let S be an element of $\mathsf{wList}_{2^{\mathbb{N}}}([T])$ and let n be the least stage such that $\Psi(p[n], S[n])(0)\downarrow$. Then $S[n]$ is of the form $\langle \tau(i), \pi_i(\sigma)\rangle_{i<\ell_\sigma}$ for some $(\tau, \sigma) \in 2^{<\mathbb{N}} \times 2^{<\mathbb{N}}$ such that $|\tau| = \ell_\sigma$. It is immediate that if $\tau(i) = 1$ then $\pi_i(\sigma)$ is the initial segment of a path in T. We obtain that $(\tau, \sigma) \in$ **Prefixes**, and this concludes the proof of the claim.

The same argument shows that every $(\tau, \sigma) \in$ **Prefixes** computes a prefix of a name for $\mathsf{wList}_{2^{\mathbb{N}}}([T])$. Since f is first-order, $\Psi\big(p[\sigma], \langle \tau(i), \pi_i(\sigma)\rangle_{i<\ell_\sigma}\big)(0) \in f(x)$. This shows that $f \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$.

We now show that $\mathsf{C}_{\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}}}$. Let $A \in \mathcal{A}_-(\mathbb{N})$ be nonempty, and let $A^c[s]$ denote the enumeration of the complement of A up to stage s. We compute the tree

$$T := \{\langle\rangle\} \cup \{0^n 1 0^s : n \notin A^c[s]\}.$$

Notice that for every n, $n \in A$ if and only if $0^n 1 0^{\mathbb{N}} \in [T]$. Given $(b_i, p_i)_{i\in\mathbb{N}} \in \mathsf{wList}_{2^{\mathbb{N}}}([T])$, we computably search for some i such that $b_i = 1$ and $0^n 1 \sqsubset p_i$ for some $n \in \mathbb{N}$ (such an i exists because A is nonempty). By construction, $n \in \mathsf{C}_{\mathbb{N}}(A)$. □

II.2 Functions arising from the Cantor-Bendixson Theorem

II.2.1 Perfect kernels

We now move to the study of functions related to the perfect kernel.

Definition II.2.1. Let $\mathsf{PK} : \mathbf{Tr} \to \mathbf{Tr}$ be the total single-valued function defined as $\mathsf{PK}(T) := S$ where S is the perfect kernel of T. We denote by $\mathsf{PK}{\restriction}\mathbf{Tr}_2$ the restriction of PK to $\mathbf{Tr}_2$.

Similarly, for a computable Polish space $\mathcal{X}$, let $\mathsf{PK}_{\mathcal{X}} : \mathcal{A}_-(\mathcal{X}) \to \mathcal{A}_-(\mathcal{X})$ be the total single-valued function defined as $\mathsf{PK}_{\mathcal{X}}(A) := P$ where P is the perfect kernel of A.

Notice that in [Hir19], Hirst already introduced PK, proving the following theorem.

Theorem II.2.2. $\widehat{\mathsf{WF}} \equiv_{\mathrm{sW}} \mathsf{PK}$.

The following proposition summarizes some well known facts about the relationship between $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ and (the parallelization of) WF.

Proposition II.2.3. $\mathsf{WF} \not\leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ *and* $\mathsf{C}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{sW}} \widehat{\mathsf{WF}}$.

Proof. The fact that $\mathsf{WF} \not\leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ was already noticed in [KMP20, page 1033].

To show that $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \widehat{\mathsf{WF}}$ it suffices to notice that $\mathsf{PTT}_1 \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ ([KMP20, Proposition 6.3]) and PK clearly computes PTT_1. The strictness of the reduction is immediate by the first part of this proposition and the fact that $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ is parallelizable. □

Recall from I.6 that J and HJ are respectively the Turing jump operator and the hyperjump operator, and that $\mathsf{lim} \equiv_{\mathrm{sW}} \mathsf{J}$ and $\widehat{\mathsf{WF}} \equiv_{\mathrm{sW}} \mathsf{HJ}$.

Proposition II.2.4. $\mathsf{lim} * \widehat{\mathsf{WF}} \not\leqslant_{\mathrm{W}} \widehat{\mathsf{WF}}$ *and hence* $\widehat{\mathsf{WF}}$ *is not closed under compositional product.*

Proof. Towards a contradiction, suppose that $\mathsf{lim} * \widehat{\mathsf{WF}} \leqslant_{\mathrm{W}} \widehat{\mathsf{WF}}$. By the definition of compositional product (Definition $I.6.6$) and the facts that $\mathsf{lim} \equiv_{\mathrm{sW}} \mathsf{J}$ and $\mathsf{J} \circ \widehat{\mathsf{WF}}$ is defined, let Φ and Ψ witness $\mathsf{J} \circ \widehat{\mathsf{WF}} \leqslant_{\mathrm{W}} \widehat{\mathsf{WF}}$.

Let $(T^i)_{i\in\mathbb{N}}$ be a computable list of all computable elements $\mathbf{Tr}$ and notice that

$$\widehat{\mathsf{WF}}((T^i)_{i\in\mathbb{N}}) \equiv_T \mathsf{HJ}(\varnothing).$$

Then, $\Phi((T^i)_{i\in\mathbb{N}})$ is a computable list of trees and $\widehat{\mathsf{WF}}(\Phi((T^i)_{i\in\mathbb{N}}))$ is Turing reducible to $\mathsf{HJ}(\varnothing) \equiv_T \mathsf{HJ}(\Phi((T^i)_{i\in\mathbb{N}}))$. Therefore, $\Psi\big((T^i)_{i\in\mathbb{N}}, \widehat{\mathsf{WF}}(\Phi((T^i)_{i\in\mathbb{N}}))\big) \leqslant_T \mathsf{HJ}(\varnothing)$ as well. On the other hand, $(\mathsf{J} \circ \widehat{\mathsf{WF}})((T^i)_{i\in\mathbb{N}})$ computes the Turing jump of $\mathsf{HJ}(\varnothing)$, which is not Turing reducible $\mathsf{HJ}(\varnothing)$. □

As we did when dealing with $\mathsf{PST}_{\mathbb{N}^\mathbb{N}}$ and $\mathsf{PST}_{2^\mathbb{N}}$, to study $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ and $\mathsf{PK}_{2^\mathbb{N}}$ we use the tree representation of $\mathcal{A}_-(\mathbb{N}^\mathbb{N})$ and $\mathcal{A}_-(2^\mathbb{N})$. The following is the analog of Theorem $II.1.3$.

Proposition II.2.5. $\mathsf{PK}\restriction\mathbf{Tr}_2 \equiv_{\mathrm{sW}} \mathsf{PK}$ *and* $\mathsf{PK}_{2^\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$.

Proof. We follow the pattern of the proof of Theorem $II.1.3$.

$\mathsf{PK}\restriction\mathbf{Tr}_2 \leqslant_{\mathrm{sW}} \mathsf{PK}$ is trivial and $\mathsf{PK}_{2^\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ is witnessed by the same functionals of the proof of $\mathsf{PST}_{2^\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{PST}_{\mathbb{N}^\mathbb{N}}$ in Theorem $II.1.3$. Given $T \in \mathbf{Tr}_2$, and a name P for $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}([T])$ we get $[\Psi(P)] = [P]$. Hence, $\Psi(P)$ is a name for $\mathsf{PK}_{2^\mathbb{N}}(T)$.

For the opposite directions, we only deal with $\mathsf{PK} \leqslant_{\mathrm{sW}} \mathsf{PK}\restriction\mathbf{Tr}_2$, as the proof of $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{PK}_{2^\mathbb{N}}$ follows the same pattern. Again, the reduction is witnessed by the same functionals of the analogous proof in Theorem $II.1.3$. Fix $T \in \mathbf{Tr}$ and let $P := \mathsf{PK}(\rho_{2^\mathbb{N}}([T]))$. As before, set $\Psi(P) := \rho_{\mathbb{N}^\mathbb{N}}(P)$. To prove the reduction, it suffices to show that $|[T] \setminus [\rho_{\mathbb{N}^\mathbb{N}}(P)]| \leqslant \aleph_0$. We claim that $[T] \setminus [\rho_{\mathbb{N}^\mathbb{N}}(P)] \subseteq \{\rho_{\mathbb{N}^\mathbb{N}}(q) : q \in [\rho_{2^\mathbb{N}}(T)] \setminus [P] \wedge (\exists^\infty i)(q(i) = 1)\}$, which completes the proof as the set on the right-hand side is countable. If $p \in [T] \setminus [\rho_{\mathbb{N}^\mathbb{N}}(P)]$, then by Lemma $I.1.8(4)$ and (6) we have that $\rho_{2^\mathbb{N}}(p) \in [\rho_{2^\mathbb{N}}(T)] \setminus [P]$. Moreover, $q := \rho_{2^\mathbb{N}}(p)$ has infinitely many ones and, by Lemma $I.1.8(2)$, $p = \rho_{\mathbb{N}^\mathbb{N}}(q)$. □

Proposition II.2.6. $\mathsf{PK}_{2^\mathbb{N}}$ *and* $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ *are (strongly) parallelizable.*

Proof. To prove the statement, by Proposition $II.2.5$, it is enough to show that $\widehat{\mathsf{PK}_{\mathbb{N}^\mathbb{N}}} \leqslant_{\mathrm{sW}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. Given $(T^i)_{i\in\mathbb{N}}$ an input for $\widehat{\mathsf{PK}_{\mathbb{N}^\mathbb{N}}}$, let P be a name for $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}([\bigsqcup_{i\in\mathbb{N}} T^i])$. By the definitions of perfect kernel and disjoint union of trees, we have that for every i, $\{\sigma : i^\frown\sigma \in P\}$ is a name for $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}([T^i])$. □

Definition II.2.7. We define the multi-valued function $\mathsf{WF}_\mathbb{S} : \mathbf{Tr} \to \mathbb{S}$ as

$$\mathsf{WF}_\mathbb{S}(T) := \begin{cases} 1_\mathbb{S} & \text{if } T \in \mathcal{WF}, \\ 0_\mathbb{S} & \text{if } T \in \mathcal{IF}. \end{cases}$$

The next proposition shows that the main functions we consider in this section are cylinders, which implies that most reductions we obtain in this section are strong.

Proposition II.2.8. $\widehat{\mathsf{WF}_\mathbb{S}}$*,* $\widehat{\mathsf{WF}}$*,* PK*,* $\mathsf{PK}\restriction\mathbf{Tr}_2$*,* $\mathsf{PK}_{2^\mathbb{N}}$ *and* $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ *are cylinders.*

Proof. All six functions are parallelizable (this is either obvious or is a consequence of Proposition $II.2.6$ and Theorem $II.2.2$) and hence it is enough to show that id strongly Weihrauch reduces to each of them. As $\widehat{\mathsf{WF}_\mathbb{S}} \leqslant_{\mathrm{sW}} \widehat{\mathsf{WF}} \equiv_{\mathrm{sW}} \mathsf{PK} \equiv_{\mathrm{sW}} \mathsf{PK}\restriction\mathbf{Tr}_2$ (Proposition $II.2.5$ and Theorem $II.2.2$) and $\mathsf{PK}_{2^\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ (Proposition $II.2.5$), it suffices to show that $\mathsf{id} \leqslant_{\mathrm{sW}} \widehat{\mathsf{WF}_\mathbb{S}}$ and $\mathsf{id} \leqslant_{\mathrm{sW}} \mathsf{PK}_{2^\mathbb{N}}$.

For the first reduction let p be an input for id. For any $i, j \in \mathbb{N}$ let

$$T^{\langle i,j\rangle} := \begin{cases} \varnothing & \text{if } p(i) = j, \\ 2^{<\mathbb{N}} & \text{if } p(i) \neq j. \end{cases}$$

Let $\widehat{\mathsf{WF}_\mathbb{S}}((T^{\langle i,j\rangle})_{i,j\in\mathbb{N}}) = (a_{\langle i,j\rangle})_{i,j\in\mathbb{N}}$. To compute $p(i)$ we search for the unique j such that $a_{\langle i,j\rangle} = 1_\mathbb{S}$ (recall that the set of names for $1_\mathbb{S}$ is $\mathbf{\Sigma}^0_1$).

For the second reduction, recall that by Lemma $II.1.4$, $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{PST}_{2^\mathbb{N}}$, clearly $\mathsf{PST}_{2^\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{PK}_{2^\mathbb{N}}$ and $\mathsf{id} \leqslant_{\mathrm{sW}} \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$. □

We now give a useful characterization of $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$.

Theorem II.2.9. $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \widehat{\mathsf{WF}_\mathbb{S}}$.

Proof. Recall that by Proposition $II.2.5$, $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{PK}_{2^\mathbb{N}}$ so that it suffices to show that $\mathsf{PK}_{2^\mathbb{N}} \equiv_{\mathrm{W}} \widehat{\mathsf{WF}_\mathbb{S}}$.

Let $T \in \mathbf{Tr}_2$ be a name for an input of $\mathsf{PK}_{2^\mathbb{N}}$. Notice that $\{\sigma : T_\sigma \in \mathcal{T}_2^{\leqslant\aleph_0}\}$ is $\Pi^{1,T}_1$ (see Theorem $I.3.39(ii)$) and hence, using Theorem $I.3.39(i)$, we can compute from T a sequence $(S(\sigma))_{\sigma\in 2^{<\mathbb{N}}} \in \mathbf{Tr}^\mathbb{N}$ such that $S(\sigma) \in \mathcal{WF}$ if and only if $T_\sigma \in \mathcal{T}_2^{\leqslant\aleph_0}$. Let $A := \{p \in 2^\mathbb{N} : (\forall n)(\mathsf{WF}_\mathbb{S}(S(p[n])) = 0_\mathbb{S})\}$: since the set of names for $0_\mathbb{S}$ is $\mathbf{\Pi}^0_1$, $A \in \mathcal{A}_-(2^\mathbb{N})$ and we can compute $U \in \mathbf{Tr}_2$ with $[U] = A$. Notice that for any $\tau \in U$, τ is a prefix of a path through $[U]$ if and only if $T_\tau \in \mathcal{T}^{>\aleph_0}$. Therefore, U is a name for $\mathsf{PK}_{2^\mathbb{N}}([T])$.

To show that $\widehat{\mathsf{WF}_\mathbb{S}} \leqslant_{\mathrm{W}} \mathsf{PK}_{2^\mathbb{N}}$, as $\mathsf{PK}_{2^\mathbb{N}}$ is parallelizable it suffices to prove that $\mathsf{WF}_\mathbb{S} \leqslant_{\mathrm{W}} \mathsf{PK}_{2^\mathbb{N}}$. Let $T \in \mathbf{Tr}$ be an input for $\mathsf{WF}_\mathbb{S}$ and notice that $T \in \mathcal{WF}$ if and only if $\mathsf{Expl}(T) \in \mathcal{WF}$ if and only if $\rho_{2^\mathbb{N}}(\mathsf{Expl}(T)) \in \mathcal{T}^{\leqslant\aleph_0}$. Hence, if S is a name for $\mathsf{PK}_{2^\mathbb{N}}([\rho_{2^\mathbb{N}}(\mathsf{Expl}(T))])$, $T \in \mathcal{WF}$ if and only if $S \in \mathcal{WF}_2$. Since $\mathcal{WF}_2$ is a Σ^0_1 set (see Theorem *I*.3.39(*i*)), given S we can uniformly compute a name for $\mathsf{WF}_{\mathbb{S}}(T)$. □

Proposition II.2.10. ${}^1\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathbf{\Pi}^1_1\text{-}\mathsf{C}_\mathbb{N} <_W {}^1\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N} <_W {}^1\widehat{\mathsf{WF}} \equiv_W \mathsf{WF}^{u*}$.

Proof. Since clearly $\mathsf{PST}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$, by Lemma *II*.1.5 we obtain $\mathbf{\Pi}^1_1\text{-}\mathsf{C}_\mathbb{N} \equiv_W {}^1\mathsf{PST}_{\mathbb{N}^\mathbb{N}} \leqslant_W {}^1\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. For the opposite direction, notice that the proof of ${}^1\mathsf{PST}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathbf{\Pi}^1_1\text{-}\mathsf{C}_\mathbb{N}$ in Lemma *II*.1.5 actually shows that ${}^1\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathbf{\Pi}^1_1\text{-}\mathsf{C}_\mathbb{N}$. In fact, the definition of **Prefixes** works also if $T \in \mathcal{T}^{\leqslant\aleph_0}$, and we already considered only prefixes of names for the perfect kernel of T.

Proposition *I*.6.29 tells us that $\mathbf{\Pi}^1_1\text{-}\mathsf{C}_\mathbb{N} <_W {}^1\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N}$. Moreover, $\mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N} <_W \mathsf{WF}^{u*}$ as $\widehat{\mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N}} \leqslant_W \mathsf{C}_{\mathbb{N}^\mathbb{N}} <_W \widehat{\mathsf{WF}}$ by Propositions *I*.6.29 and *II*.2.3 and the fact that $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ is parallelizable. On the other hand, ${}^1\widehat{\mathsf{WF}} \equiv_W \mathsf{WF}^{u*}$ is an instance of Theorem *I*.6.10. □

Theorem II.2.11. $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} <_W \widehat{\mathsf{WF}} \equiv_W \mathsf{PK} \leqslant_W \lim * \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$.

Proof. Given $T \in \mathbf{Tr}$, $\mathsf{PK}(T)$ is a name for $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}([T])$: therefore $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{PK}$; strictness follows from Proposition *II*.2.10. By Theorem *II*.2.2, $\widehat{\mathsf{WF}} \equiv_W \mathsf{PK}$.

To prove the last reduction, notice that $\mathsf{PK}\restriction\mathbf{Tr}_2 \equiv_W \widehat{\mathsf{WF}}$ (Proposition *II*.2.5) and $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{PK}_{2^\mathbb{N}}$ (Theorem *II*.2.2), hence, to finish the proof, it suffices to show that $\mathsf{PK}\restriction\mathbf{Tr}_2 \leqslant_W \lim * \mathsf{PK}_{2^\mathbb{N}}$. From [NP19], we know that $\lim$ is equivalent to the function that prunes a binary tree. So let $T \in \mathbf{Tr}_2$ and let P be a name for $\mathsf{PK}_{2^\mathbb{N}}([T])$: pruning P with $\lim$ is enough to obtain $\mathsf{PK}\restriction\mathbf{Tr}_2(T)$. □

We do not know whether $\mathsf{PK} \equiv_W \lim * \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ (see Question *II*.4.1).

Proposition II.2.12. $\mathsf{PST}_{\mathbb{N}^\mathbb{N}} <_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}} \mid_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$.

Proof. The fact that $\mathsf{PST}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ is trivial. By Theorem *II*.2.11, $\widehat{\mathsf{WF}} \leqslant_W \lim * \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ while, by the closure of $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ under compositional product, we get $\lim * \mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$: hence by Proposition *II*.2.3 $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \nleqslant_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ and a fortiori $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \nleqslant_W \mathsf{PST}_{\mathbb{N}^\mathbb{N}}$. For the opposite nonreduction, just notice that by Proposition *II*.2.10 we have that ${}^1\mathsf{C}_{\mathbb{N}^\mathbb{N}} \nleqslant_W {}^1\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. □

Proposition II.2.13. $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} <_W \widehat{\mathsf{TC}_{\mathbb{N}^\mathbb{N}}}$ *and hence the reduction* $\mathsf{WF}_\mathbb{S} \leqslant_W \mathsf{TC}_{\mathbb{N}^\mathbb{N}}$ *in [BG21, Proposition 11.4(1)] is actually strict.*

Proof. From $\mathsf{WF}_\mathbb{S} \leqslant_W \mathsf{TC}_{\mathbb{N}^\mathbb{N}}$ using Theorem *II*.2.9, we obtain $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \leqslant_W \widehat{\mathsf{TC}_{\mathbb{N}^\mathbb{N}}}$. Strictness follows from Proposition *II*.2.12 as $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \nleqslant_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ but clearly $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{TC}_{\mathbb{N}^\mathbb{N}}$. □

We end the subsection by characterizing the deterministic part of $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$.

Proposition II.2.14. $\mathsf{Det}(\mathsf{PK}_{\mathbb{N}^\mathbb{N}}) \equiv_W \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$.

Proof. For the right-to-left direction, notice that $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ is single-valued and, by Lemma *II*.1.4 and Proposition *II*.2.12 we have that $\widehat{\mathsf{UC}_{\mathbb{N}^\mathbb{N}}} <_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. For the converse, observe that ${}^1\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathbf{\Pi}^1_1\text{-}\mathsf{C}_\mathbb{N}$ (Proposition *II*.2.10) and $\widehat{\mathbf{\Pi}^1_1\text{-}\mathsf{C}_\mathbb{N}} \equiv_W \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ (Proposition *I*.6.28). This, together with $\mathsf{Det}(\mathsf{PK}_{\mathbb{N}^\mathbb{N}}) \leqslant_W \widehat{{}^1\mathsf{PK}_{\mathbb{N}^\mathbb{N}}}$ ([GPV21, Corollary 3.7]), concludes the proof. □

II.2.2 Scattered lists

We now introduce the problems of listing the scattered part of a closed subset of a computable Polish space. Their definition is similar to Definition *II*.1.7: the crucial difference is that the domain includes all closed sets A of the computable Polish space $\mathcal{X}$ (not only the countable ones as in Definition *II*.1.7) and we ask for the list of the elements of the scattered part of A.

Definition II.2.15. Let $\mathcal{X}$ be a computable Polish space. We define three multi-valued functions $\mathsf{wScList}_\mathcal{X} : \mathcal{A}_-(\mathcal{X}) \rightrightarrows 2 \times \mathcal{X}^\mathbb{N}$, $\mathsf{ScCount}_\mathcal{X} : \mathcal{A}_-(\mathcal{X}) \to \mathbb{N}$ and $\mathsf{ScList}_\mathcal{X} : \mathcal{A}_-(\mathcal{X}) \rightrightarrows \mathbb{N} \times (2 \times \mathcal{X})^\mathbb{N}$ by

$$\mathsf{wScList}_\mathcal{X}(A) := \{(b_i, x_i)_{i\in\mathbb{N}} : A \setminus \mathsf{PK}_\mathcal{X}(A) = \{x_i : b_i = 1\}\},$$

$$\mathsf{ScCount}_\mathcal{X}(A) := \begin{cases} 0 & \text{if } A\setminus\mathsf{PK}_\mathcal{X}(A) \text{ is infinite,} \\ |A\setminus\mathsf{PK}_\mathcal{X}| + 1 & \text{if } A\setminus\mathsf{PK}_\mathcal{X}(A) \text{ is finite.} \end{cases}$$

$$\mathsf{ScList}_\mathcal{X}(A) := \mathsf{wScList}_\mathcal{X}(A) \times \mathsf{ScCount}_\mathcal{X}(A).$$

Remark II.2.16. Notice that if a closed set A of some T_1 topological space has a finite set F of isolated points, then $A\setminus F$ is perfect, and hence the scattered part of A is F. Equivalently, if the scattered part of A is infinite then it contains infinitely many isolated points. Moreover, the set of isolated points is always dense in the scattered part.

With a similar proof to that Proposition *II*.2.6, we obtain the following.

Proposition II.2.17. $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$ *and* $\mathsf{wScList}_{2^\mathbb{N}}$ *are parallelizable.*

One of the main results of this subsection is that $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. We first prove the easier direction.

Lemma II.2.18. $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$.

Proof. By Theorem *II*.2.9 we get $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \equiv_W \widehat{\mathsf{WF}_\mathbb{S}}$ and, by Proposition *II*.2.17, $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$ is parallelizable. So it suffices to show that $\mathsf{WF}_\mathbb{S} \leqslant_W \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$.

Given an input $T \in \mathbf{Tr}$ for $\mathsf{WF}_\mathbb{S}$, let $S := \lfloor\flat\rfloor_{i\in\mathbb{N}} \mathsf{Expl}(T) \in \mathbf{Tr}$. If $T \in \mathcal{WF}$ then $\mathsf{Expl}(T) \in \mathcal{WF}$, so that by Remark *I*.1.3, $[S] = \{0^\mathbb{N}\}$. If instead $T \in \mathcal{IF}$, $[\mathsf{Expl}(T)]$ is perfect and therefore $0^\mathbb{N} \in \mathsf{PK}_{\mathbb{N}^\mathbb{N}}([S])$ and $[S]$ is perfect, so that the scattered part of $[S]$ is empty.

Hence, for every $(b_i, x_i)_{i\in\mathbb{N}} \in \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}([S])$ we obtain $\mathsf{WF}_\mathbb{S}(T) = 1_\mathbb{S}$ if and only if $(\exists i)(b_i = 1)$. Hence, a name for $\mathsf{WF}_\mathbb{S}(T)$ can be uniformly computed from $(b_i, x_i)_{i\in\mathbb{N}}$. □

We split the proof of $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ in several lemmas. Before stating them we discuss some results in [KMP20, §6.1] and correct an error there.

Remark II.2.19. Theorem 6.4 of [KMP20] states (in our notation) that $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{wList}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{List}_{\mathbb{N}^\mathbb{N}}$. The main ingredient of the proof of $\mathsf{wList}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ is a variant of the Cantor-Bendixson derivative that allows to carry out the process in a Borel way for countable closed sets. A single step of this process is called *one-step* mCB-*certificate* ([KMP20, Definition 6.5]) and all steps are then "collected" in a *global* mCB-*certificate* ([KMP20, Definition 6.6]).

Definition II.2.20 ([KMP20, Definition 6.5])**.** Let $A \in \mathcal{A}_-(\mathbb{N}^\mathbb{N})$. A *one-step* mCB-*certificate* for A is some $c = ((\sigma_i^c)_{i\in\mathbb{N}}, (b_i^c)_{i\in\mathbb{N}}, (p_i^c)_{i\in\mathbb{N}}) \in (\mathbb{N}^{<\mathbb{N}}, 2, \mathbb{N}^\mathbb{N})^\mathbb{N}$ where

- for all $i \neq j$, $\sigma_i^c \not\sqsubseteq \sigma_j^c$ and if $i < j$ then $\sigma_i^c < \sigma_j^c$;
- there exists i such that $b_i^c = 1$,
- Let $\mathsf{HYP}(A)$ be the set of hyperarithmetical elements of A. For all i:
  - if $b_i^c = 1$, then $p_i^c \in A$ and $\sigma_i^c \sqsubset p_i^c$,
  - if $b_i^c = 0$, then $(\forall p \in \mathsf{HYP}(A))(\sigma_i^c \not\sqsubseteq p)$ and $p_i^c = 0^\mathbb{N}$,
  - $(\forall p, q \in \mathsf{HYP}(A))(p, q \in A \wedge \sigma_i^c \sqsubset p, q \implies p = q)$,
- for every $\sigma \in \mathbb{N}^{<\mathbb{N}}$, if $(\forall i \in \mathbb{N})(\sigma_i^c \not\sqsubseteq \sigma)$ then $(\exists p, q \in A)(p \neq q \wedge \sigma \sqsubset p, q)$.

For a one-step mCB-certificate c for A, the residue of c is $\{p \in A : (\forall i \in \mathbb{N})(\sigma_i^c \not\sqsubseteq p)\}$.

By [KMP20, Lemma 6.8], every nonempty countable $A \in \mathcal{A}_-(\mathbb{N}^\mathbb{N})$ has a one-step mCB-certificate c: moreover the residue of c is the Cantor-Bendixson derivative of A. Furthermore, if $|A| \leqslant \aleph_0$ then the one-step mCB-certificate of A is unique.

Definition II.2.21 ([KMP20, Definition 6.6], corrected)**.** A *global* mCB-*certificate* for $A \in \mathcal{A}_-(\mathbb{N}^\mathbb{N})$ is indexed by some initial $I \subseteq \mathbb{N}$ and consists of a sequence $(c_n)_{n\in I}$ and a strict linear ordering $\lhd$ on I with minimum n_0 (if nonempty) such that, denoting with A_i the residue of c_i:

- c_{n_0} is a one-step mCB-certificate for A;
- for every $n \in I \backslash \{n_0\}$, c_n is a one-step mCB-certificate for $\bigcap_{i\lhd n} A_i$;
- for all $p \in \mathsf{HYP}(A)$, if $p \in A$ then $(\exists i \in I)(p \in A_i)$;
- for every $n, m \in I$, if $n < m$ then $\sigma_{h(n)}^{c_n} < \sigma_{h(m)}^{c_m}$ where $h(n) := \min\{i : b_i^{c_n} = 1\}$.

Observe that Definition $II.2.21$ differs from [KMP20, Definition 6.6] by the addition of the last requirement, which ensures that [KMP20, Corollary 6.9] holds: if $|A| \leqslant \aleph_0$, then A has a unique global mCB-certificate. Indeed, the last condition forces a specific ordering (determined by the code of the first finite sequence that is the prefix of an isolated path) on the sequence of the one-step mCB-certificates, avoiding the different codings of the sequence allowed in [KMP20, Definition 6.6] because it was possible to permute the ordering.

Notice that, assuming $|A| \leqslant \aleph_0$, the global mCB-certificate of A computes a list of the paths in A. Since the global mCB-certificate of a countable closed set A is a $\Sigma_1^{1,A}$ singleton, we obtain, as in [KMP20, Theorem 6.4], that $\mathsf{wList}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$.

The following lemmas involve the completion of a problem (Definition $I.6.24$).

Lemma II.2.22. *The function* $F : \mathbb{N} \times (\mathbb{N}^\mathbb{N} \cup \mathbb{N}^{<\mathbb{N}}) \to \overline{\mathbb{N}^\mathbb{N}}$ *such that for all* $e \in \mathbb{N}$ *and* $p \in \mathbb{N}^\mathbb{N} \cup \mathbb{N}^{<\mathbb{N}}$

$$F(e,p) = \begin{cases} \Phi_e(p) & \text{if } p \in \mathbb{N}^\mathbb{N} \text{ and } p \in \operatorname{dom}(\Phi_e), \\ \bot & \text{otherwise,} \end{cases}$$

is computable.

Proof. A computable realizer Φ' for F can be defined recursively as follows. Suppose we have defined $\Phi'(e,p)[s]$ and let $t_s = |\{t < s : \Phi_e(p)(t) > 0\}|$. Then set

$$\Phi'(e,p)(s) = \begin{cases} \Phi_e(p)(t_s) + 1 & \text{if } \Phi_e(p)(t_s) \downarrow \text{ in less than } s \text{ steps,} \\ 0 & \text{otherwise.} \end{cases}$$

It is easy to check that this works. □

Lemma II.2.23. $\overline{\mathsf{wList}_{\mathbb{N}^\mathbb{N}}} \leqslant_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$.

Proof. Since $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{PK}_{2^\mathbb{N}}$ is parallelizable (Propositions *II*.2.5 and *II*.2.6), it suffices to show that $\overline{\mathsf{wList}_{\mathbb{N}^\mathbb{N}}} \leqslant_W \widehat{\mathsf{PK}_{2^\mathbb{N}}}$.

By [BG21, Lemma 5.1] $\mathcal{A}_-(\mathbb{N}^\mathbb{N})$ is multi-retraceable, i.e. there is a computable multi-valued function $r : \overline{\mathcal{A}_-(\mathbb{N}^\mathbb{N})} \rightrightarrows \mathcal{A}_-(\mathbb{N}^\mathbb{N})$ such that its restriction to $\mathcal{A}_-(\mathbb{N}^\mathbb{N})$ is the identity. Given $A \in \overline{\mathcal{A}_-(\mathbb{N}^\mathbb{N})}$, let G be the set of global mCB-certificates of $r(A) \in \mathcal{A}_-(\mathbb{N}^\mathbb{N})$. By [KMP20, Lemma 6.7], G is a $\Sigma_1^{1,A}$ subset of $\mathbb{N}^\mathbb{N}$ and, in case $|r(A)| \leqslant \aleph_0$, by [KMP20, Corollary 6.9], G is a singleton.

Let T be a name for G as described in §I.6, i.e. $T \in \mathbf{Tr}$ is such that $G = \{x : (\exists y)(\forall n)(x[n] \oplus y[n] \in T)\}$. Then, for every m, compute the tree

$$T^{\oplus m} := \mathbb{N}^m \cup \{\sigma : \langle \sigma(m), \dots, \sigma(|m|-1)\rangle \in T \wedge (\forall i < m)(m+2i < |\sigma| \implies \sigma(i) = \sigma(m+2i))\}.$$

Notice that $[T^{\oplus m}] = \{\langle p(0), p(2), \dots, p(2m-2)\rangle p : p \in [T]\}$; therefore, every path in $[T^{\oplus m}]$ begins with the first m coordinates of some element of the analytic set G. In particular, if $G = \{p_0\}$ then every path in $[T^{\oplus m}]$ extends $p_0[m]$. Let U^m be a name for $\mathsf{PK}_{2^\mathbb{N}}(\rho_{2^\mathbb{N}}(\mathsf{Expl}(T^{\oplus m})))$. If $G = \{p_0\}$ then every path in $[U^m]$ extends $\rho_{2^\mathbb{N}}(p_0[m] \oplus \sigma)$ for some $\sigma \in 2^m$. We now describe how to compute an element $x \in \mathbb{N}^\mathbb{N} \cup \mathbb{N}^{<\mathbb{N}}$ from the sequence $(U^m)_{m \in \mathbb{N}}$ such that $x = p_0$ when $G = \{p_0\}$.

The procedure is similar to the one used in the proof of Lemma *II*.1.4. Looking first at U^1, we search for n_0 such that

$$(\forall \tau \in 2^{n_0+1})(U^1_\tau \in \mathcal{IF}_2 \implies \tau = 0^{n_0}1).$$

Since $\mathcal{IF}_2$ is Σ^0_1 (Theorem *I*.3.39(*i*)), the above condition is Π^0_1. If we find such an n_0, we set $x(0) = n_0$ and we move to the next step. Notice that, in case $G = \{p_0\}$, the unique n_0 satisfying the above condition is $p_0(0)$.

Suppose we have computed $x[m-1] := n_0 n_1 \dots n_{m-1}$. We generalize the previous strategy to compute n_m. Let

$$A_m = \{0^{n_0}1\xi_0 0^{n_1}1\xi_1 \dots \xi_{m-2}0^{n_{m-1}}1\xi_{m-1} \in U^m : (\forall j < m)(\xi_j \in \{1, 01\})\}.$$

We search for n_m satisfying the Σ^0_1 property

$$(\forall \sigma \in A_m)(\forall \tau \in 2^{n_m+1})(U^m_{\sigma\tau} \in \mathcal{IF}_2 \implies \tau = 0^{n_m}1)$$

As before, if we find such an n_m we let $x(m) := n_m$ and move to the next step. Again, if $G = \{p_0\}$, the unique n_m satisfying the above condition is $p_0(m)$.

The proof of [KMP20, Theorem 6.4] gives us a computable function Φ_e such that, in case $|r(A)| \leqslant \aleph_0$, $\Phi_e(x)$ is a name for a member of $\mathsf{wList}_{\mathbb{N}^\mathbb{N}}(r(A))$. If F is the function of

Lemma $II.2.22$ then, identifying the completion of the codomain of $\mathsf{wList}_{\mathbb{N}^\mathbb{N}}$ with $\overline{\mathbb{N}^\mathbb{N}}$, we obtain $F(e,x) \in \overline{\mathsf{wList}_{\mathbb{N}^\mathbb{N}}}(r(A))$.

Summing up, if $A \in \mathcal{A}_-(\mathbb{N}^\mathbb{N})$ is countable then $F(e,x)$ is a name in $\overline{\mathbb{N}^\mathbb{N}}$ for an element of $\mathsf{wList}_{\mathbb{N}^\mathbb{N}}(A)$. If instead $A \in \mathcal{A}_-(\mathbb{N}^\mathbb{N})$ does not belong to $\operatorname{dom}(\mathsf{wList}_{\mathbb{N}^\mathbb{N}})$, then $F(e,x)$ is anyway a name for some member of $\overline{\mathbb{N}^\mathbb{N}}$. □

Lemma II.2.24. $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \widehat{\mathsf{WF}_\mathbb{S}} \times \widehat{\overline{\mathsf{wList}_{\mathbb{N}^\mathbb{N}}}}$.

Proof. Let $T \in \mathbf{Tr}$ be a name for an element of $\mathcal{A}_-(\mathbb{N}^\mathbb{N})$ and recall that, by Theorem $I.3.39(ii)$, $\{\sigma : T_\sigma \in \mathcal{T}^{\leqslant\aleph_0}\}$ is $\Pi^{1,T}_1$. Hence, using Theorem $I.3.39(i)$, we can compute $(S(\sigma))_{\sigma\in\mathbb{N}^{<\mathbb{N}}} \in \mathbf{Tr}^\mathbb{N}$ such that $S(\sigma) \in \mathcal{WF}$ if and only if $T_\sigma \in \mathcal{T}^{\leqslant\aleph_0}$. For any $\sigma \in T$, let $S(\sigma)$ and $[T_\sigma]$ be the inputs for the σ-th instance of $\mathsf{WF}_\mathbb{S}$ and $\overline{\mathsf{wList}_{\mathbb{N}^\mathbb{N}}}$ respectively. Let $L_\sigma \in \overline{\mathsf{wList}_{\mathbb{N}^\mathbb{N}}}([T_\sigma])$.

For any σ, when we see that $\mathsf{WF}_\mathbb{S}(S(\sigma)) = 1_\mathbb{S}$ then we know that $S(\sigma) \in \mathcal{WF}$ and hence $T_\sigma \in \mathcal{T}^{\leqslant\aleph_0}$: we need to include a list of $[T_\sigma]$ in $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}}([T])$. We compute a name for $L \in \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}([T])$ combining all L_σ such that $\mathsf{WF}_\mathbb{S}(S(\sigma)) = 1_\mathbb{S}$. □

Theorem II.2.25. $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$.

Proof. The right-to-left direction is Lemma $II.2.18$. For the opposite direction, notice that, by Theorem $II.2.9$ and Lemma $II.2.23$, we have that $\mathsf{WF}_\mathbb{S}, \overline{\mathsf{wList}_{\mathbb{N}^\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$, By Lemma $II.2.24$ and the fact that $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ is parallelizable (Proposition $II.2.6$), we conclude that $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. □

We now study $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$ and show that it lies strictly between $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ and $\widehat{\mathsf{WF}}$.

Lemma II.2.26. $\mathsf{WF}^* \leqslant_{\mathrm{W}} \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$.

Proof. Let $(T^m)_{m\leqslant n} \in \mathbf{Tr}^n$ be an input for WF^*. For every T^m compute the tree $S^m := \{j^n\tau : j < 2^m \land n \in \mathbb{N} \land \tau \in \mathsf{Expl}(T^m)\}$. Notice that if $T^m \in \mathcal{WF}$ then $\mathsf{Expl}(T^m) \in \mathcal{WF}$, and, in this case, $|[S^m]| = 2^m$. On the other hand, if $T^m \in \mathcal{IF}$ then $[S^m]$ is perfect. Now let $k := \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}\left(\left[\bigsqcup_{m\leqslant n} S^m\right]\right)$. Notice that $k > 0$ (as the scattered part of $\left[\bigsqcup_{m\leqslant n} S^m\right]$ is always finite) and

$$k - 1 = \sum_{T^m\in\mathcal{WF}} 2^m.$$

Hence, the binary expansion of $k-1$ contains the information about which T^m's are well-founded. □

We do not know more about the Weihrauch degree of $\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$ (see Question $II.4.2$).

Theorem II.2.27. $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$.

Proof. Since $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$ is trivial, Theorem $II.2.25$ immediately implies the reduction.

For strictness, first notice that $\mathsf{WF} \nleqslant_{\mathrm{W}} {}^1\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$: indeed, by Proposition $II.2.3$ and the fact that WF is first-order, we obtain that $\mathsf{WF} \nleqslant_{\mathrm{W}} {}^1\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ and by Proposition $II.2.10$, we get that ${}^1\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} {}^1\mathsf{C}_{\mathbb{N}^\mathbb{N}}$. Hence,$\mathsf{WF} \nleqslant_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. On the other hand, clearly $\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$ and $\mathsf{WF}^* \leqslant_{\mathrm{W}} \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$ by Lemma $II.2.26$, so that $\mathsf{WF} \leqslant_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$. □

Combining this with the fact that $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} \mid_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ (Proposition II.2.12), we immediately obtain that $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} \not\leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$. On the other hand, we do not know if $\mathsf{C}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$ or $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \mid_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$ (see Question II.4.4).

Remark II.2.28. By Theorem I.3.39(iii), given a tree T, $\{\sigma : |[T_\sigma]| = 1\}$ is $\Pi^{1,T}_1$ if $T \in \mathbf{Tr}$ and $\Pi^{0,T}_2$ if $T \in \mathbf{Tr}_2$. Let $\varphi(n,T) := (\exists \sigma_0,\dots,\sigma_{n-1})(\forall i \neq j < n)(\sigma_i \mid \sigma_j \wedge |[T_{\sigma_i}]| = 1)$: this formula, asserting that T has at least n isolated points, is Π^1_1 if $T \in \mathbf{Tr}$, and Σ^0_3 if $T \in \mathbf{Tr}_2$. Notice that, by Remark II.2.16, the scattered part of $[T]$ has at least n elements if and only if $\varphi(n,T)$ holds. Therefore, the scattered part of $[T]$ is infinite if and only if $(\forall n)(\varphi(n,T))$, which is Π^1_1 if $T \in \mathbf{Tr}$ and Π^0_2 if $T \in \mathbf{Tr}_2$.

Theorem II.2.29. $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \widehat{\mathsf{WF}}$.

Proof. To prove the reduction notice that $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}} \times \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$. By Theorem II.2.25 and Theorem II.2.11, $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \widehat{\mathsf{WF}}$. As $\widehat{\mathsf{WF}}$ is clearly parallelizable, it suffices to prove that $\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \widehat{\mathsf{WF}}$. Let $T \in \mathbf{Tr}$ be a name for an input of $\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$: by Remark II.2.28 we can compute $(S^n)_{n\in\mathbb{N}} \in \mathbf{Tr}^\mathbb{N}$ such that $S^0 \in \mathcal{WF}$ if and only if the scattered part of $[T]$ has infinitely many elements and, for $n > 0$, $S^n \in \mathcal{WF}$ if and only if the scattered part of $[T]$ has at least n elements. Then,

$$\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}([T]) = \begin{cases} 0 & \text{if } \mathsf{WF}(S^0) = 1, \\ \min\{n > 0 : \mathsf{WF}(S^n) = 0\} & \text{if } \mathsf{WF}(S^0) = 0. \end{cases}$$

For strictness, notice that $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{wScList}_{\mathbb{N}^\mathbb{N}} * \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$. By Theorem 3.9 of [GPV21], $\mathsf{Det}(f * g) \leqslant_{\mathrm{W}} \mathsf{Det}(f) * g$ and so

$$\mathsf{Det}(\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} * \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}) \leqslant_{\mathrm{W}} \mathsf{Det}(\mathsf{wScList}_{\mathbb{N}^\mathbb{N}}) * \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{UC}_{\mathbb{N}^\mathbb{N}} * \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}},$$

where the equivalence follows from Theorem II.2.25 and Proposition II.2.14. Since the output of $\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$ is a natural number and the solution of $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ is always hyperarithmetical relative to the input ([KMP20, Corollary 3.4]), while $\widehat{\mathsf{WF}}$ has instances with no hyperarithmetical solutions in the input, we conclude that $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} * \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \mathsf{Det}(\widehat{\mathsf{WF}}) \equiv_{\mathrm{W}} \widehat{\mathsf{WF}}$ (the equivalence is immediate as $\widehat{\mathsf{WF}}$ is single-valued). Therefore, $\widehat{\mathsf{WF}} \not\leqslant_{\mathrm{W}} \mathsf{wScList}_{\mathbb{N}^\mathbb{N}} * \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$ and, a fortiori, $\widehat{\mathsf{WF}} \not\leqslant_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$. □

We now move our attention to listing problems of the scattered part of closed subsets of Cantor space. From Theorem II.2.11 and Theorem II.2.25 notice that $\mathsf{WF} \leqslant_{\mathrm{W}} \mathsf{LPO} * \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$. As the next lemma shows, to compute WF it suffices to compose $\mathsf{wScList}_{2^\mathbb{N}}$ with a function slightly stronger than LPO.

Lemma II.2.30. $\mathsf{WF} \leqslant_{\mathrm{W}} \mathsf{LPO}' * \mathsf{wScList}_{2^\mathbb{N}}$

Proof. Given an input $T \in \mathbf{Tr}$ for WF, we can compute $S := \biguplus_{n\in\mathbb{N}} (\rho_{2^\mathbb{N}}(\mathsf{Expl}(T)))$. Let $(b_i, p_i)_{i\in\mathbb{N}} \in \mathsf{wScList}_{2^\mathbb{N}}(S)$. Then

$$\begin{aligned} T \in \mathcal{WF} &\iff \mathsf{Expl}(T) \in \mathcal{WF} \\ &\iff \rho_{2^\mathbb{N}}(\mathsf{Expl}(T)) \in \mathcal{T}_2^{\leqslant\aleph_0} \\ &\iff (\exists i)(b_i = 1 \wedge p_i = 0^\mathbb{N}). \end{aligned}$$

The last condition is Σ^0_2 and so LPO' suffices to establish from $(b_i, p_i)_{i\in\mathbb{N}}$ whether $T \in \mathcal{WF}$. $\square$

Lemma II.2.31. $\boldsymbol{\Pi}^0_2\text{-}\mathsf{C}_\mathbb{N} \equiv_\mathrm{W} {}^1\mathsf{wScList}_{2^\mathbb{N}}$.

Proof. For the left-to-right direction, since $\mathsf{wScList}_{2^\mathbb{N}}$ is parallelizable (Proposition *II*.2.17), it suffices to show that $\boldsymbol{\Pi}^0_2\text{-}\mathsf{C}_\mathbb{N} \leqslant_\mathrm{W} \widehat{\mathsf{wScList}_{2^\mathbb{N}}}$. An input for $\boldsymbol{\Pi}^0_2\text{-}\mathsf{C}_\mathbb{N}$ is a nonempty set $A \in \boldsymbol{\Pi}^0_2(\mathbb{N})$. By Lemma *I*.3.35, we can uniformly find a sequence $(p_n)_{n\in\mathbb{N}}$ of elements of $2^\mathbb{N}$ such that $n \in A \iff (\exists^\infty i)(p_n(i) = 0)$.

For every n, let

$$T^n := \{\sigma \in 2^{<\mathbb{N}} : (\forall i < |\sigma|)(p_n(i) = 0 \implies (\forall j < i)(\sigma(j) = 0))\}.$$

Notice that if $(\exists^\infty i)(p_n(i) = 0)$ then $[T^n] = \{0^\mathbb{N}\}$, while $[T^n]$ is perfect otherwise.

Given $((b_{i,n}, p_{i,n})_{i\in\mathbb{N}})_{n\in\mathbb{N}} \in \widehat{\mathsf{wScList}_{2^\mathbb{N}}}((T^n)_{n\in\mathbb{N}})$ notice that, for every n, $n \in A$ if and only if there exists i such that $b_{i,n} = 1$. Hence, we can find $n \in A$ simply by searching for a pair i, n such that $b_{i,n} = 1$.

For the other direction, let f be a first-order function and suppose that $f \leqslant_\mathrm{W} \mathsf{wScList}_{2^\mathbb{N}}$ is witnessed by the maps Φ and Ψ. Let p be a name for an input of f and let $\Phi(p) = T \in \mathbf{Tr}_2$. Recall that Definition *II*.1.8 introduced ℓ_σ and $\pi_i(\sigma)$ for $\sigma \in 2^{<\mathbb{N}}$. Let **Prefixes** be the set of all $(\sigma, \tau) \in 2^{<\mathbb{N}} \times 2^{<\mathbb{N}}$ such that

$$|\tau| = \ell_\sigma \wedge \Psi\big(p[|\sigma|], \langle \tau(i), \pi_i(\sigma)\rangle_{i<\ell_\sigma}\big)(0)\downarrow \wedge (\forall i < \ell_\sigma)(\tau(i) = 1 \implies T_{\pi_i(\sigma)} \in \mathcal{UB}_2).$$

Elements in $2^{<\mathbb{N}} \times 2^{<\mathbb{N}}$ can be coded as natural numbers. Since $\mathcal{UB}_2$ is a Π^0_2 set (Theorem *I*.3.39(*iii*)), **Prefixes** is a $\Pi^{0,T}_2$ subset of $\mathbb{N}$. It is immediate that every $(\tau, \sigma) \in$ **Prefixes** is a prefix of a name for $\mathsf{wScList}_{2^\mathbb{N}}([T])$, and hence $f \leqslant_\mathrm{W} \boldsymbol{\Pi}^0_2\text{-}\mathsf{C}_\mathbb{N}$ follows immediately if **Prefixes** $\neq \emptyset$.

To show this, let $L \in \mathsf{wScList}_{2^\mathbb{N}}([T])$ and n be such that $\Psi(p[n], L[n])(0)\downarrow$. Then $L[n] = \langle \tau(i), \pi_i(\sigma)\rangle_{i<\ell_\sigma}$ for some $(\tau, \sigma) \in 2^{<\mathbb{N}} \times 2^{<\mathbb{N}}$. If $\tau(i) = 1$ then $\pi_i(\sigma)$ is a prefix of a member of the scattered part of $[T]$ and, by Remark *II*.2.16, there exists $\xi_i \sqsupseteq \pi_i(\sigma)$ such that $T_{\xi_i} \in \mathcal{UB}_2$. Let (τ', σ') be such that $\tau' \sqsupseteq \tau$, $\sigma' \sqsupseteq \sigma$, $(\forall i < \ell_\sigma)(\tau(i) = 1 \implies \pi_i(\sigma') \sqsupseteq \xi_i)$ and $(\forall i < \ell_{\sigma'})(i \geqslant \ell_\sigma \implies \tau'(i) = 0)$. Then $(\tau', \sigma') \in$ **Prefixes**. $\square$

We collect in the next theorem our results about the problems of listing the scattered part of a closed set; these results are also summarized in Figure II.2.

Theorem II.2.32. *The following relations hold:*

(i) $\mathsf{List}_{2^\mathbb{N},<\omega}, \mathsf{wList}_{2^\mathbb{N}} <_\mathrm{W} \mathsf{wScList}_{2^\mathbb{N}}$*, while* $\mathsf{List}_{2^\mathbb{N}}$ *and* $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ *are both Weihrauch incomparable with* $\mathsf{wScList}_{2^\mathbb{N}}$*;*

(ii) $\mathsf{List}_{2^\mathbb{N}}, \mathsf{wScList}_{2^\mathbb{N}} <_\mathrm{W} \mathsf{ScList}_{2^\mathbb{N}}$ *and* $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \mid_\mathrm{W} \mathsf{ScList}_{2^\mathbb{N}}$*;*

(iii) $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}, \mathsf{ScList}_{2^\mathbb{N}} <_\mathrm{W} \mathsf{wScList}_{\mathbb{N}^\mathbb{N}} <_\mathrm{W} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$ *while* $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \mid_\mathrm{W} \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$ *and* $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} \nleqslant_\mathrm{W} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$*.*

Proof. (i) By [KMP20, Proposition 6.11] and Lemma *II*.2.31, $\mathsf{List}_{2^\mathbb{N},<\omega} \equiv_\mathrm{W} \boldsymbol{\Pi}^0_2\text{-}\mathsf{C}_\mathbb{N} \equiv_\mathrm{W} {}^1\mathsf{wScList}_{2^\mathbb{N}}$. The reduction $\mathsf{wList}_{2^\mathbb{N}} \leqslant_\mathrm{W} \mathsf{wScList}_{2^\mathbb{N}}$ is obvious. From Theorem *II*.1.11 we know that $\mathsf{List}_{2^\mathbb{N},<\omega} \mid_\mathrm{W} \mathsf{wList}_{2^\mathbb{N}}$ and hence $\mathsf{wScList}_{2^\mathbb{N}} \nleqslant_\mathrm{W} \mathsf{List}_{2^\mathbb{N},<\omega}$ and $\mathsf{wScList}_{2^\mathbb{N}} \nleqslant_\mathrm{W} \mathsf{wList}_{2^\mathbb{N}}$.

For the incomparabilities, recall that, by Theorem *II*.1.11, $\mathsf{List}_{2^\mathbb{N}} \leqslant_\mathrm{W} \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$: therefore it suffices to show that $\mathsf{wScList}_{2^\mathbb{N}} \nleqslant_\mathrm{W} \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ and $\mathsf{List}_{2^\mathbb{N}} \nleqslant_\mathrm{W} \mathsf{wScList}_{2^\mathbb{N}}$.

By Lemma *II*.2.30, $\mathsf{WF} \leqslant_\mathrm{W} \mathsf{LPO}' * \mathsf{wScList}_{2^\mathbb{N}}$ while, since $\mathsf{WF} \nleqslant_\mathrm{W} \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ (Proposition $II.2.3$) and $\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$ is closed under compositional product, we obtain that $\mathsf{WF} \not\leqslant_W \mathsf{LPO}' * \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$. Hence, $\mathsf{wScList}_{2^\mathbb{N}} \not\leqslant_W \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$.

To show that $\mathsf{List}_{2^\mathbb{N}} \not\leqslant_W \mathsf{wScList}_{2^\mathbb{N}}$ we prove that ${}^1\mathsf{List}_{2^\mathbb{N}} \not\leqslant_W {}^1\mathsf{wScList}_{2^\mathbb{N}}$. Since, by Lemma $II.2.31$ and Proposition $I.6.30(ii)$, ${}^1\mathsf{wScList}_{2^\mathbb{N}} \equiv_W \mathbf{\Pi}^0_2\text{-}\mathsf{C}_\mathbb{N} \not\leqslant_W \mathsf{LPO}''$, it suffices to show $\mathsf{LPO}'' \leqslant_W \mathsf{List}_{2^\mathbb{N}}$. By Theorem $I.3.36$, we can view an input for LPO'' as $q \in 2^{\mathbb{N}\times\mathbb{N}}$

$$\mathsf{LPO}''(q) = 1 \iff (\exists^\infty n)(\forall i)(q(n,i) = 0).$$

Given q, we compute $(T^n)_{n\in\mathbb{N}} \in \mathbf{Tr}^\mathbb{N}$ defined as $T^n := \{0^s : (\forall i < s)(q(n,i) = 0)\}$. Notice that $T^n \in \mathcal{IF}_2 \iff (\forall i)(q(n,i) = 0)$ and given $T' := \bigsqcup^{\mathsf{b}}_{n\in\mathbb{N}} T^n$ it is easy to check that $|[T']| = \aleph_0 \iff \mathsf{LPO}''(q) = 1$. Since the information about the cardinality of $[T']$ is included in $\mathsf{List}_{2^\mathbb{N}}([T'])$, this concludes the reduction $\mathsf{LPO}'' \leqslant_W \mathsf{List}_{2^\mathbb{N}}$.

(ii) The reductions $\mathsf{List}_{2^\mathbb{N}}, \mathsf{wScList}_{2^\mathbb{N}} \leqslant_W \mathsf{ScList}_{2^\mathbb{N}}$ are immediate and, since we just showed that $\mathsf{List}_{2^\mathbb{N}} \mid_W \mathsf{wScList}_{2^\mathbb{N}}$, they are strict.

Combining the facts that $\mathsf{wScList}_{2^\mathbb{N}} <_W \mathsf{ScList}_{2^\mathbb{N}}$ and $\mathsf{wScList}_{2^\mathbb{N}} \not\leqslant_W \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$, we conclude that $\mathsf{ScList}_{2^\mathbb{N}} \not\leqslant_W \mathsf{UC}_{\mathbb{N}^\mathbb{N}}$.

To show that $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \not\leqslant_W \mathsf{ScList}_{2^\mathbb{N}}$, we first prove that $\mathsf{ScCount}_{2^\mathbb{N}} \leqslant_W \mathbf{\Pi}^0_4\text{-}\mathsf{C}_\mathbb{N}$. Given $T \in \mathbf{Tr}_2$, let

$$A := \{n : (n > 0 \implies \varphi(n-1,T) \wedge \neg\varphi(n,T)) \wedge (n = 0 \implies (\forall k)(\varphi(k,T)))\}$$

where $\varphi(n,T) := (\exists \sigma_0, \dots, \sigma_{n-1})(\forall i \neq j < n)(\sigma_i \mid \sigma_j \wedge T_{\sigma_i} \in \mathcal{UB}_2)$. Using Remark $II.2.28$, it is easy to check that $\varphi(n,T)$ is Σ^0_3 and hence A is a $\Pi^{0,T}_4$ subset of $\mathbb{N}$. Notice that A is a singleton and, by Remark $II.2.16$, the unique $n \in A$ is the correct answer for $\mathsf{ScCount}_{2^\mathbb{N}}$.

As $\mathsf{ScList}_{2^\mathbb{N}} \leqslant_W \mathsf{wScList}_{2^\mathbb{N}} \times \mathsf{ScCount}_{2^\mathbb{N}} \leqslant_W \mathsf{wScList}_{2^\mathbb{N}} * \mathsf{ScCount}_{2^\mathbb{N}}$ we have that

$${}^1\mathsf{ScList}_{2^\mathbb{N}} \leqslant_W {}^1(\mathsf{wScList}_{2^\mathbb{N}} * \mathsf{ScCount}_{2^\mathbb{N}}) \leqslant_W {}^1\mathsf{wScList}_{2^\mathbb{N}} * \mathsf{ScCount}_{2^\mathbb{N}},$$

where the second reduction follows from [SV22, Proposition 4.1(4)] which states that ${}^1(f * g) \leqslant_W {}^1f * g$ for any f and g. By Lemma $II.2.31$ and the fact that $\mathsf{ScCount}_{2^\mathbb{N}} \leqslant_W \mathbf{\Pi}^0_4\text{-}\mathsf{C}_\mathbb{N}$ we get that ${}^1\mathsf{ScList}_{2^\mathbb{N}} \leqslant_W \mathbf{\Pi}^0_2\text{-}\mathsf{C}_\mathbb{N} * \mathbf{\Pi}^0_4\text{-}\mathsf{C}_\mathbb{N} \equiv_W \mathbf{\Pi}^0_4\text{-}\mathsf{C}_\mathbb{N}$ (the last equivalence follows from [SV22, Theorem 7.2]). By Proposition $I.6.30(i)$ and (iii) and Proposition $I.6.29$, we know that $\mathbf{\Pi}^0_4\text{-}\mathsf{C}_\mathbb{N} <_W \mathbf{\Pi}^1_1\text{-}\mathsf{C}_\mathbb{N} \equiv_W {}^1\mathsf{UC}_{\mathbb{N}^\mathbb{N}}$, hence $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \not\leqslant_W \mathsf{ScList}_{2^\mathbb{N}}$.

(iii) By Proposition $II.2.12$ and Theorem $II.2.25$, we have that $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} <_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$. Moreover, in the proof of (ii), we showed $\mathsf{ScCount}_{2^\mathbb{N}} \leqslant_W \mathbf{\Pi}^0_4\text{-}\mathsf{C}_\mathbb{N}$, which by Proposition $II.2.10$ implies $\mathsf{ScCount}_{2^\mathbb{N}} \leqslant_W \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$. Since $\mathsf{ScList}_{2^\mathbb{N}} \leqslant_W \mathsf{ScCount}_{2^\mathbb{N}} \times \mathsf{wScList}_{2^\mathbb{N}}$ and $\mathsf{wScList}_{2^\mathbb{N}} \leqslant_W \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$ is immediate, we obtain $\mathsf{ScList}_{2^\mathbb{N}} \leqslant_W \mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \times \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$. Recalling that by Proposition $II.2.17$ $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$ is parallelizable, we have that $\mathsf{ScList}_{2^\mathbb{N}} \leqslant_W \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$. Since $\mathsf{UC}_{\mathbb{N}^\mathbb{N}} \mid_W \mathsf{ScList}_{2^\mathbb{N}}$ by the previous item, we also deduce that the reduction is strict.

By the fact that $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ (Theorem $II.2.25$), we have that Theorem $II.2.27$ and Proposition $II.2.12$ imply respectively $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} <_W \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$ and $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \mid_W \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$. These two relationships imply that $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} \not\leqslant_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$. □

II.2.3 The full Cantor-Bendixson theorem

The following functions formulate the Cantor-Bendixson theorem as a problem.

Definition II.2.33. Let $\mathcal{X}$ be a computable Polish space. We define two multi-valued functions

$\mathsf{wCB}_{\mathcal{X}} : \mathcal{A}_{-}(\mathcal{X}) \rightrightarrows \mathcal{A}_{-}(\mathcal{X}) \times (2 \times \mathcal{X})^{\mathbb{N}}$ and $\mathsf{CB}_{\mathcal{X}} : \mathcal{A}_{-}(\mathcal{X}) \rightrightarrows \mathcal{A}_{-}(\mathcal{X}) \times (\mathbb{N} \times (2 \times \mathcal{X})^{\mathbb{N}})$ by

$$\mathsf{wCB}_{\mathcal{X}}(A) := \mathsf{PK}_{\mathcal{X}}(A) \times \mathsf{wScList}_{\mathcal{X}}(A) \text{ and } \mathsf{CB}_{\mathcal{X}}(A) := \mathsf{PK}_{\mathcal{X}}(A) \times \mathsf{ScList}_{\mathcal{X}}(A).$$

The multi-valued functions $\mathsf{wCB} : \mathbf{Tr} \rightrightarrows \mathbf{Tr} \times (2 \times \mathbb{N}^{\mathbb{N}})^{\mathbb{N}}$ and $\mathsf{CB} : \mathbf{Tr} \rightrightarrows \mathbf{Tr} \times (\mathbb{N} \times (2 \times \mathbb{N}^{\mathbb{N}})^{\mathbb{N}})$ are defined similarly, substituting $\mathsf{PK}_{\mathcal{X}}$ with PK and $(\mathsf{w})\mathsf{List}_{\mathcal{X}}$ with $(\mathsf{w})\mathsf{List}_{\mathbb{N}^{\mathbb{N}}}$ in the definitions above.

Proposition II.2.34. $\mathsf{wCB} \equiv_{\mathrm{W}} \mathsf{CB} \equiv_{\mathrm{W}} \widehat{\mathsf{WF}}$.

Proof. Clearly $\mathsf{PK} \leqslant_{\mathrm{W}} \mathsf{wCB} \leqslant_{\mathrm{W}} \mathsf{CB}$ and since, by Theorem *II*.2.2, $\widehat{\mathsf{WF}} \equiv_{\mathrm{W}} \mathsf{PK}$ we have that $\widehat{\mathsf{WF}} \leqslant_{\mathrm{W}} \mathsf{wCB} \leqslant_{\mathrm{W}} \mathsf{CB}$. For the opposite directions notice that $\mathsf{wCB} \leqslant_{\mathrm{W}} \mathsf{PK} \times \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}}$ and $\mathsf{CB} \leqslant_{\mathrm{W}} \mathsf{PK} \times \mathsf{ScList}_{\mathbb{N}^{\mathbb{N}}}$. By Theorems *II*.2.25 and *II*.2.29, we have that $\mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{W}} \widehat{\mathsf{WF}}$. As $\widehat{\mathsf{WF}}$ is clearly parallelizable this concludes the proof. □

Theorem II.2.35. $\mathsf{wCB}_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{CB}_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{PK}_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{wCB}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{CB}$.

Proof. By Theorem *II*.2.25 and Proposition *II*.2.5, $\mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{PK}_{2^{\mathbb{N}}}$. By Theorem *II*.2.32, $\mathsf{wScList}_{2^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{ScList}_{2^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}}$. Since, by Proposition *II*.2.6, $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$ is parallelizable, we obtain all the equivalences. Also, $\mathsf{ScList}_{\mathbb{N}^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^{\mathbb{N}}}$ is immediate, and $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^{\mathbb{N}}}$ was already proven in Theorem *II*.2.27. To prove $\mathsf{CB}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{CB}$, notice that the reduction is straightforward and $\mathsf{CB}_{\mathbb{N}^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{wCB}_{\mathbb{N}^{\mathbb{N}}} * \mathsf{ScCount}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}} * \mathsf{ScCount}_{\mathbb{N}^{\mathbb{N}}}$. To conclude the proof, observe that $\widehat{\mathsf{WF}} \equiv_{\mathrm{W}} \mathsf{CB}$ (Proposition *II*.2.34) and that $\widehat{\mathsf{WF}} \not\leqslant_{\mathrm{W}} \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}} * \mathsf{ScCount}_{\mathbb{N}^{\mathbb{N}}}$ (see the proof of Theorem *II*.2.29), hence $\mathsf{CB} \not\leqslant_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^{\mathbb{N}}}$. □

The situation here is similar to the one discussed above for $\mathsf{ScList}_{\mathbb{N}^{\mathbb{N}}}$: we do not know if $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^{\mathbb{N}}}$ or $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}} \mid_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^{\mathbb{N}}}$. It is also open whether $\mathsf{ScList}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^{\mathbb{N}}}$ (see Questions *II*.4.4 and *II*.4.3).

In the next theorem, we explore what can be added to $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$ in order to compute $\mathsf{CB}_{\mathbb{N}^{\mathbb{N}}}$.

Theorem II.2.36. $\mathsf{CB}_{\mathbb{N}^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{WF}^{u*} \times \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} * \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$.

Proof. For the first reduction, by Theorem *II*.2.35, $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{wCB}_{\mathbb{N}^{\mathbb{N}}}$ and clearly $\mathsf{CB}_{\mathbb{N}^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{ScCount}_{\mathbb{N}^{\mathbb{N}}} \times \mathsf{wCB}_{\mathbb{N}^{\mathbb{N}}}$: by Theorems *II*.2.29 and *I*.6.10, we obtain that $\mathsf{ScCount}_{\mathbb{N}^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{WF}^{u*}$.

For the second reduction, notice that $\mathsf{C}_{\mathbb{N}}^{u*} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$ ([SV22, Theorem 7.2]). Furthermore, since $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{wCB}_{\mathbb{N}^{\mathbb{N}}}$, $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$ is parallelizable (Proposition *II*.2.6) and $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \widehat{\mathsf{WF}_{\mathbb{S}}}$ (Theorem *II*.2.9), it suffices to show that $\mathsf{WF}^{u*} \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}^{u*} * \widehat{\mathsf{WF}_{\mathbb{S}}}$. The input for WF^{u*} is a sequence $(T^i)_{i \in \mathbb{N}} \in \mathbf{Tr}^{\mathbb{N}}$. Let $(p_i)_{i \in \mathbb{N}}$ be a name for a solution of $\widehat{\mathsf{WF}_{\mathbb{S}}}((T^i)_{i \in \mathbb{N}})$ (recall that the only name for $0_{\mathbb{S}}$ is $0^{\mathbb{N}}$). For every $i \in \mathbb{N}$, the input for the i-th instance of $\mathsf{C}_{\mathbb{N}}$

$$A_i := \{n : (n = 0 \land p_i = 0^{\mathbb{N}}) \lor (n > 0 \land (\exists m < n)(p_i(m) = 1))\}.$$

To conclude the proof it suffices to notice that for any i and $n_i \in \mathsf{C}_{\mathbb{N}}(A_i)$, $T^i \in \mathcal{WF}$ if and only if $n_i \neq 0$. □

II.3 What happens in arbitrary computable metric spaces

In this section, we study the functions connected to the perfect set and Cantor-Bendixson theorems in arbitrary computable metric spaces. We start by collecting some facts about maps between spaces of closed sets.

Proposition II.3.1 ([BDP12, proof of Proposition 3.7]). *Let $\mathcal{X}$ and $\mathcal{Y}$ be computable metric spaces and $s :\subseteq \mathcal{X} \to \mathcal{Y}$ be a computable function with* $\operatorname{dom}(s) \in \Pi^0_1(\mathcal{X})$*: then the function $S : \mathcal{A}_-(\mathcal{Y}) \to \mathcal{A}_-(\mathcal{X})$, $M \mapsto s^{-1}(M)$ is computable as well.*

Definition II.3.2 ([BG08]). Given two represented spaces $\mathcal{X}$ and $\mathcal{Y}$, we say that $\iota : \mathcal{X} \to \mathcal{Y}$ is a *computable embedding* if ι is injective and ι as well as its partial inverse ι^{-1} are computable. If $\mathcal{X}$ is a computable metric space we say that $\mathcal{X}$ is *rich* if there exists a computable embedding of $2^{\mathbb{N}}$ into $\mathcal{X}$.

As observed in [BG08], any computable embedding $\iota : 2^{\mathbb{N}} \to \mathcal{X}$ is such that $\mathsf{range}(\iota) \in \Pi^0_1(\mathcal{X})$. Moreover, by [BG08, Theorem 6.2], any perfect computable metric space $\mathcal{X}$ is rich.

Theorem II.3.3 ([BG08, Theorem 3.7]). *Let $\mathcal{X}$ and $\mathcal{Y}$ be computable metric spaces and $\iota : \mathcal{X} \to \mathcal{Y}$ be a computable embedding with $\mathsf{range}(\iota) \in \Pi^0_1(\mathcal{Y})$. Then the map $J : \mathcal{A}_-(\mathcal{X}) \to \mathcal{A}_-(\mathcal{Y}), A \mapsto \iota(A)$ is computable and admits a partial computable right inverse.*

The following is an analog of [BDP12, Corollaries 4.3 and 4.4].

Lemma II.3.4. *Let $\mathcal{X}$ and $\mathcal{Y}$ be computable Polish spaces and $\iota : \mathcal{X} \to \mathcal{Y}$ be a computable embedding with $\mathsf{range}(\iota) \in \Pi^0_1(\mathcal{Y})$. Let f be any of the following:* PST, $\mathsf{(w)List}$, PK, $\mathsf{(w)ScList}$, $\mathsf{(w)CB}$. *Then $f_{\mathcal{X}} \leqslant_{\mathrm{W}} f_{\mathcal{Y}}$. In particular, $f_{2^{\mathbb{N}}} \leqslant_{\mathrm{W}} f_{\mathcal{Y}}$ for every rich computable metric space $\mathcal{Y}$.*

Proof. By Theorem $II.3.3$, the map $J : \mathcal{A}_-(\mathcal{X}) \to \mathcal{A}_-(\mathcal{Y})$ and its partial inverse are computable. Given $A \in \operatorname{dom}(f_{\mathcal{X}})$, we have that $J(A) \in \operatorname{dom}(f_{\mathcal{Y}})$, as cardinality is preserved by J. Moreover, $J(A)$ is homeomorphic to A. Depending on f, we use combinations of copies of J^{-1} and ι^{-1} to compute from a solution for $f_{\mathcal{Y}}(J(A))$ a solution for $f_{\mathcal{X}}(A)$. For example, considering the case $f = \mathsf{CB}$, we have that if $(B, (n, (b_i, y_i)_{i\in\mathbb{N}})) \in \mathsf{CB}_{\mathcal{Y}}(J(A))$ then $(J^{-1}(B), (n, (b_i, \iota^{-1}(y_i))_{i\in\mathbb{N}})) \in \mathsf{CB}_{\mathcal{X}}(A)$. □

The following lemma is immediate using Lemma $II.3.4$ and either Theorem $II.1.3$ or Proposition $II.2.5$ or Theorem $II.2.35$.

Lemma II.3.5. *Let f be any of the following:* PST, PK, wCB. *For every rich computable Polish space $\mathcal{X}$, $f_{\mathbb{N}^{\mathbb{N}}} \leqslant_{\mathrm{W}} f_{\mathcal{X}}$. If moreover there exists a computable embedding $\iota : \mathcal{X} \to \mathbb{N}^{\mathbb{N}}$ with* $\mathsf{range}(\iota) \in \Pi^0_1(\mathbb{N}^{\mathbb{N}})$, *$f_{\mathcal{X}} \equiv_{\mathrm{W}} f_{\mathbb{N}^{\mathbb{N}}}$.*

II.3.1 Perfect sets

Lemma $II.3.5$ implies that $\mathsf{PST}_{\mathcal{X}} \equiv_{\mathrm{W}} \mathsf{PST}_{\mathbb{N}^{\mathbb{N}}}$ whenever $\mathcal{X}$ is 0-dimensional. We can however obtain this result also for the unit interval.

Theorem II.3.6. $\mathsf{PST}_{[0,1]} \equiv_{\mathrm{W}} \mathsf{PST}_{\mathbb{N}^{\mathbb{N}}}$.

Proof. The right-to-left direction follows from Lemma $II.3.5$.

For the opposite direction, since $\mathsf{PST}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{PST}_{2^{\mathbb{N}}}$ by Theorem $II.1.3$, we show instead that $\mathsf{PST}_{[0,1]} \leqslant_{\mathrm{W}} \mathsf{PST}_{2^{\mathbb{N}}}$.

Let $s_{\mathsf{b}} : 2^{\mathbb{N}} \to [0,1]$ be the computable function that computes a real from its binary expansion:

$$s_{\mathsf{b}}(p) = \sum_{i\in\mathbb{N}} \frac{p(i)}{2^{i+1}}.$$

Notice that s_{b} is not injective (and hence not an embedding) as $s_{\mathsf{b}}(\sigma 01^{\mathbb{N}}) = s_{\mathsf{b}}(\sigma 10^{\mathbb{N}})$ for any $\sigma \in 2^{<\mathbb{N}}$; however these are the only counterexamples to injectivity. In particular, for every $x \in [0,1]$, $|s_{\mathsf{b}}^{-1}(x)| \leqslant 2$. For $\sigma \in 2^{<\mathbb{N}}$ we let $I^\sigma := \{x \in [0,1] : (\forall p \in 2^{\mathbb{N}})(s_{\mathsf{b}}(p) = x \implies \sigma \sqsubset p)\}$. Notice that if σ is not constant then $I^\sigma = (s_{\mathsf{b}}(\sigma 0^{\mathbb{N}}), s_{\mathsf{b}}(\sigma 1^{\mathbb{N}}))$, while $I^{0^n} = [0, s_{\mathsf{b}}(0^n 1^{\mathbb{N}}))$ and $I^{1^n} = (s_{\mathsf{b}}(1^n 0^{\mathbb{N}}), 1]$: all these intervals are open subsets of $[0,1]$.

By Proposition *II*.3.1 given $A \in \mathcal{A}_-([0,1])$ we can compute $s_{\mathsf{b}}^{-1}(A)$.

Although Theorem *II*.3.3 does not apply, we claim that $J : \mathcal{A}_-(2^{\mathbb{N}}) \to \mathcal{A}_-([0,1])$, $C \mapsto s_{\mathsf{b}}(C)$, is computable (notice that, as $2^{\mathbb{N}}$ is compact and s_{b} is continuous, the image of a closed set is closed). To prove that J is computable, we proceed as follows: let $S \in \mathbf{Tr}_2$ be a name for $C \in \mathcal{A}_-(2^{\mathbb{N}})$, i.e. a tree such that $[S] = C$. Recall that, by Theorem *I*.3.39(*i*), $\mathcal{WF}_2$ is a Σ^0_1 set. We compute $B \in \mathcal{A}_-([0,1])$ as follows:

(i) whenever we witness that $S_\sigma \in \mathcal{WF}_2$, we list I^σ in the complement of B;

(ii) whenever we witness that $S_{\sigma 01^i}$ and $S_{\sigma 10^i}$ are in $\mathcal{WF}_2$ for some $i \in \mathbb{N}$, we list in the complement of B the open interval $(s_{\mathsf{b}}(\sigma 01^i 0^{\mathbb{N}}), s_{\mathsf{b}}(\sigma 10^i 1^{\mathbb{N}}))$ which coincides with $I^{\sigma 01^i} \cup I^{\sigma 10^i} \cup \{s_{\mathsf{b}}(\sigma 01^{\mathbb{N}})\}$.

This proves that J is computable: now, we need to check that $B = J(C)$, i.e. for every $x \in [0,1]$, $x \notin B$ if and only if $s_{\mathsf{b}}^{-1}(x) \cap C = \emptyset$.

If $x \notin B$ because $x \in I^\sigma$ for some σ with $S_\sigma \in \mathcal{WF}_2$, then σ is a prefix of every element of $s_{\mathsf{b}}^{-1}(x)$ and hence $s_{\mathsf{b}}^{-1}(x) \cap C = \emptyset$. If $x \notin B$ because $x \in (s_{\mathsf{b}}(\sigma 01^i 0^{\mathbb{N}}), s_{\mathsf{b}}(\sigma 10^i 1^{\mathbb{N}}))$ for some σ and i, then either $x \in I^{\sigma 01^i} \cup I^{\sigma 10^i}$, in which case we can apply the previous argument to one of $\sigma 01^i$ and $\sigma 10^i$, or $x = s_{\mathsf{b}}(\sigma 01^{\mathbb{N}}) = s_{\mathsf{b}}(\sigma 10^{\mathbb{N}})$; in this case we know that both $\sigma 01^{\mathbb{N}}$ and $\sigma 10^{\mathbb{N}}$ do not belong to C.

For the converse, consider first the case where $s_{\mathsf{b}}^{-1}(x) = \{q\}$ and $x \notin \{0,1\}$: then q is not eventually constant. Since $q \notin C$, there exists $\sigma \sqsubset q$ such that $\sigma \notin S$ and hence I^σ is listed in the complement of B. As $q \notin \{\sigma 0^{\mathbb{N}}, \sigma 1^{\mathbb{N}}\}$, we obtain that $x \in I^\sigma$ and hence $x \notin B$. The case in which $x \in \{0,1\}$ is analogous. If $s_{\mathsf{b}}^{-1}(x) = \{q_0, q_1\}$ then, as noticed above, there exists τ such that $q_0 = \tau 01^{\mathbb{N}}$ and $q_1 = \tau 10^{\mathbb{N}}$. Since $q_0, q_1 \notin C$ we have $\tau 01^i, \tau 10^i \notin S$ for some i. Then $x \in (s_{\mathsf{b}}(\tau 01^i 0^{\mathbb{N}}), s_{\mathsf{b}}(\tau 10^i 1^{\mathbb{N}}))$, and this interval is listed in the complement of B by condition (*ii*). Therefore, $x \notin B$.

We now describe the reduction. Given an uncountable $A \in \mathcal{A}_-([0,1])$, we can compute $s_{\mathsf{b}}^{-1}(A) \in \mathcal{A}_-(2^{\mathbb{N}})$ which is uncountable as well. Let $P \in \mathsf{PST}_{2^{\mathbb{N}}}(s_{\mathsf{b}}^{-1}(A))$ and $B = J(P)$. It suffices to show that $B \subseteq A$ and that B is perfect.

If $x \in B$ then there exists $q \in s_{\mathsf{b}}^{-1}(x) \cap P$. Since $P \subseteq s_{\mathsf{b}}^{-1}(A)$ we get that $s_{\mathsf{b}}(q) = x \in A$. This shows that $B \subseteq A$.

It remains to show that B is perfect. Suppose not: then there exists $x \in B$ and some open interval $I \subseteq [0,1]$ such that $I \cap B = \{x\}$. By continuity of s_{b}, $s_{\mathsf{b}}^{-1}(I \cap B)$ is an open set in P which has at most two members; these points are isolated in P, contradicting the perfectness of P. □

Remark II.3.7. Following the ideas of the previous proof and using some extra care it is possible to prove that $\mathsf{PST}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{PST}_{\mathbb{R}}$: replace s_{b} with $s'_{\mathsf{b}} : \mathbb{N} \times 2^{\mathbb{N}} \to \mathbb{R}$ defined by

$$s'_{\mathsf{b}}(n,p) = (-1)^n \cdot \left\lceil \frac{n}{2} \right\rceil + s_{\mathsf{b}}(p).$$

We do not know whether there exist rich computable Polish spaces $\mathcal{X}$ such that $\mathsf{PST}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{PST}_{\mathcal{X}}$ (see Question *II*.4.5).

II.3.2 (Weak) lists

The following classical fact helps in the next proofs.

Theorem II.3.8 ([Mos82, Theorem 3E.6]). *For every computable metric space $\mathcal{X}$ there is a computable surjection $s : \mathbb{N}^{\mathbb{N}} \to \mathcal{X}$ and $A \in \Pi^0_1(\mathbb{N}^{\mathbb{N}})$ such that s is one-to-one on A and $s(A) = \mathcal{X}$.*

Lemma II.3.9. *Let $\mathcal{X}$ be a computable metric space. Then* $(\mathsf{w})\mathsf{List}_{\mathcal{X}} \leqslant_{\mathrm{W}} (\mathsf{w})\mathsf{List}_{\mathbb{N}^{\mathbb{N}}}$.

Proof. We prove only $\mathsf{List}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{List}_{\mathbb{N}^{\mathbb{N}}}$ as the other reduction is similar. Let s and A be as in Theorem *II*.3.8 and s_A be the restriction of s to A. By Proposition *II*.3.1, the function $S_A : \mathcal{A}_-(\mathcal{X}) \to \mathcal{A}_-(\mathbb{N}^{\mathbb{N}})$ such that $S_A(M) = s_A^{-1}(M)$ is computable: hence given $C \in \mathcal{A}_-(\mathcal{X})$ and $(n, (b_i, p_i)_{i\in\mathbb{N}}) \in \mathsf{List}_{\mathbb{N}^{\mathbb{N}}}(S_A(C))$ we have that $(n, (b_i, s(p_i))_{i\in\mathbb{N}}) \in \mathsf{List}_{\mathcal{X}}(C)$. ☐

Lemma II.3.10. *Let $\mathcal{X}, \mathcal{Y}$ be computable metric spaces and $\iota : \mathcal{X} \to \mathcal{Y}$ be a computable embedding with* $\mathsf{range}(\iota) \in \Pi^0_1(\mathcal{Y})$*. Then* $(\mathsf{w})\mathsf{List}_{\mathcal{X}} \leqslant_{\mathrm{W}} (\mathsf{w})\mathsf{List}_{\mathcal{Y}}$*. In particular,* $(\mathsf{w})\mathsf{List}_{2^{\mathbb{N}}} \leqslant_{\mathrm{W}} (\mathsf{w})\mathsf{List}_{\mathcal{Y}}$ *for every rich computable metric space $\mathcal{Y}$.*

Proof. We only prove that $\mathsf{List}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{List}_{\mathcal{Y}}$, the other reduction is similar. By Theorem *II*.3.3 the map $J : \mathcal{A}_-(\mathcal{X}) \to \mathcal{A}_-(\mathcal{Y})$ is computable. Given $A \in \mathrm{dom}(\mathsf{List}_{\mathcal{X}})$ we have that $J(A) \in \mathrm{dom}(\mathsf{List}_{\mathcal{Y}})$: moreover, given $(n, (b_i, p_i))_{i\in\mathbb{N}} \in \mathsf{List}_{\mathcal{Y}}(J(A))$ we have that $(n, (b_i, \iota^{-1}(p_i)))_{i\in\mathbb{N}} \in \mathsf{List}_{\mathcal{X}}(A)$. ☐

Lemma II.3.11. $\mathsf{wList}_{\mathbb{R}} \equiv_{\mathrm{W}} \mathsf{wList}_{[0,1]} \equiv_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}}}$.

Proof. The fact that $\mathsf{wList}_{[0,1]} \leqslant_{\mathrm{W}} \mathsf{wList}_{\mathbb{R}}$ is immediate and $\mathsf{wList}_{2^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{wList}_{[0,1]}$ follows from Lemma *II*.3.10.

Recall that $\mathsf{wList}_{2^{\mathbb{N}}}$ is parallelizable (Lemma *II*.1.12) and notice that it is straightforward to check that $\mathsf{wList}_{\mathbb{R}} \leqslant_{\mathrm{W}} \widehat{\mathsf{wList}_{[0,1]}}$. Hence, it suffices to show that $\mathsf{wList}_{[0,1]} \leqslant_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}}}$. The function s_{b} of the proof of Theorem *II*.3.6 is useful also here. Consider an input $A \in \mathcal{A}_-([0,1])$ and let $(b_i, p_i)_{i\in\mathbb{N}} \in \mathsf{wList}_{2^{\mathbb{N}}}(s_{\mathsf{b}}^{-1}(A))$: it is straightforward to check that $(b_i, s_{\mathsf{b}}(p_i))_{i\in\mathbb{N}}$ is a solution of $\mathsf{wList}_{[0,1]}(A)$. ☐

Notice that the argument above shows that $\mathsf{wList}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}}}$ for every computable metric space $\mathcal{X}$ such that there exists an admissible representation $\delta :\subseteq 2^{\mathbb{N}} \to \mathcal{X}$ with $\mathrm{dom}(\delta) \in \Pi^0_1(2^{\mathbb{N}})$ and such that $|\delta^{-1}(x)| \leqslant \aleph_0$ for every $x \in \mathcal{X}$. In particular $\mathsf{wList}_{[0,1]^d} \equiv_{\mathrm{W}} \mathsf{wList}_{2^{\mathbb{N}}}$ for any $d \in \mathbb{N}$.

Notice that the situation for $\mathsf{List}_{\mathcal{X}}$ is less clear: for example, we do not know if, in contrast to what happens for $\mathsf{wList}_{\mathcal{X}}$, $\mathsf{List}_{2^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{List}_{\mathbb{R}}$ (see Question *II*.4.6).

We now consider listing problems on countable spaces. Let us start from finite spaces: recall from §I.6, for $n > 0$, we denote by $\mathbf{n}$ the space consisting of $\{0, \dots, n-1\}$ with the discrete topology, which is obviously a computable metric space.

Proposition II.3.12. *For every $n > 0$,* $\mathsf{wList}_{\mathbf{n}} \equiv_{\mathrm{W}} \mathsf{List}_{\mathbf{n}} \equiv_{\mathrm{W}} \mathsf{LPO}^n$ *and therefore* $\mathsf{List}_{\mathbf{n}} <_{\mathrm{W}} \mathsf{List}_{\mathbf{n+1}}$.

Proof. The fact that $\mathsf{wList}_{\mathbf{n}} \leqslant_{\mathrm{W}} \mathsf{List}_{\mathbf{n}}$ is trivial. For the converse let $A \in \mathcal{A}_-(\mathbf{n})$ and let $(b_i, m_i)_{i\in\mathbb{N}} \in \mathsf{wList}_{\mathbf{n}}(A)$. Notice that for every $m < n$ exactly one of $m \notin A$ and $(\exists i)(b_i = 1 \land m_i = m)$ holds: since both conditions are Σ^0_1 we can compute whether $m \in A$ or not. This allows us to compute $|A|$ and, together with $(b_i, m_i)_{i\in\mathbb{N}}$ we obtain a name for $\mathsf{List}_{\mathbf{n}}$ (see Remark *II*.1.9).

To show that $\mathsf{wList}_{\mathbf{n}} \leqslant_{\mathrm{W}} \mathsf{LPO}^n$, let $A \in \mathcal{A}_-(\mathbf{n})$ and fix a computable formula φ such that $i \in A$ if and only if $(\forall k)\varphi(i, k, A)$. The input $p_i \in 2^{\mathbb{N}}$ for the i-th instance of LPO is defined by

$p_i(k) = 1$ if and only if $\neg\varphi(i,k,A)$, so that $\mathsf{LPO}(p_i) = 1 \iff i \in A$. For all $i \in \mathbb{N}$ define

$$b_i := \begin{cases} 1 & \text{if } i < n \text{ and } \mathsf{LPO}(p_i) = 1; \\ 0 & \text{otherwise.} \end{cases} \qquad x_i := \begin{cases} i & \text{if } i < n; \\ 0 & \text{otherwise.} \end{cases}$$

Then, $(b_i, x_i)_{i\in\mathbb{N}} \in \mathsf{wList}_{\mathbf{n}}(A)$.

For the opposite direction, we show that $\mathsf{LPO}^n \leqslant_{\mathrm{W}} \mathsf{wList}_{\mathbf{n}}$. Let $(p_j)_{j<n}$ be an input for LPO^n. Consider $A := \{j < n : p_j = 0^{\mathbb{N}}\} \in \mathcal{A}_-(\mathbf{n})$ and let $(b_i, m_i)_{i\in\mathbb{N}} \in \mathsf{List}_{\mathbf{n}}(A)$. Notice that, for every $j < n$, $p_j = 0^{\mathbb{N}}$ if and only if $(\exists i)(b_i = 1 \wedge m_i = j)$. We thus can compute $\mathsf{LPO}(p_j)$ by searching for i such that either $p_j(i) = i$ or $b_i = 1$ and $m_i = j$.

The fact that $\mathsf{List}_{\mathbf{n}} <_{\mathrm{W}} \mathsf{List}_{\mathbf{n+1}}$ follows from $\mathsf{LPO}^n <_{\mathrm{W}} \mathsf{LPO}^{n+1}$ ([BG09, Corollary 6.7]). □

We say that a computable metric space is *effectively countable* if there exists a computable surjection $f : \mathbb{N} \to \mathcal{X}$.

Lemma II.3.13. *For any computable metric space $\mathcal{X}$ which is effectively countable,* $\mathsf{wList}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{lim}$.

Proof. Fix a computable surjection $f : \mathbb{N} \to \mathcal{X}$. Recalling that $\mathsf{lim} \equiv_{\mathrm{W}} \widehat{\mathsf{LPO}}$, the proof is a straightforward generalization of the proof of $\mathsf{wList}_{\mathbf{n}} \leqslant_{\mathrm{W}} \mathsf{LPO}^n$ in Proposition *II*.3.12. □

We say that a computable metric space $\mathcal{X}$ is *effectively infinite* if there exists a computable sequence $(U_i)_{i\in\mathbb{N}}$ of open sets in $\mathcal{X}$ such that $(\forall i)(U_i \nsubseteq \bigcup_{j\neq i} U_j)$.

Lemma II.3.14. *For every countable computable metric space $\mathcal{X}$ which is effectively infinite,* $\mathsf{lim} \leqslant_{\mathrm{W}} \mathsf{wList}_{\mathcal{X}}$.

Proof. Fix a sequence $(U_i)_{i\in\mathbb{N}}$ witnessing that $\mathcal{X}$ is effectively infinite. Recall that $\mathsf{lim} \equiv_{\mathrm{W}} \widehat{\mathsf{LPO}}$ so that it suffices to show $\widehat{\mathsf{LPO}} \leqslant_{\mathrm{W}} \mathsf{wList}_{\mathcal{X}}$.

Given an input $(p_i)_{i\in\mathbb{N}}$ for $\widehat{\mathsf{LPO}}$, let $A := \{x \in \mathcal{X} : (\forall i)(x \in U_i \implies p_i = 0^{\mathbb{N}})\} \in \mathcal{A}_-(\mathcal{X})$ and notice that $|A| \leqslant \aleph_0$ because $\mathcal{X}$ is countable. Notice that $p_i = 0^{\mathbb{N}}$ if and only if $A \cap U_i \neq \emptyset$ (for the forward direction use the existence of $y_i \in U_i$ such that $y_i \notin \bigcup_{j\neq i} U_j$ by definition of effectively infinite).

Fix $(b_i, x_i)_{i\in\mathbb{N}} \in \mathsf{wList}_{\mathcal{X}}(A)$. By the above observation, for every $i \in \mathbb{N}$ we get

$$p_i = 0^{\mathbb{N}} \iff (\exists k)(b_k = 1 \wedge x_k \in U_i).$$

Since we showed the equivalence of the Π^0_1 condition $p_i = 0^{\mathbb{N}}$ with a Σ^0_1 condition, we can compute $\mathsf{LPO}(p_i)$ for every $i \in \mathbb{N}$. □

Many natural countable computable metric spaces, not necessarily Polish, are easily seen to be both effectively countable and effectively infinite. We thus can combine Lemmas *II*.3.13 and *II*.3.14 to obtain $\mathsf{wList}_{\mathcal{X}} \equiv_{\mathrm{W}} \mathsf{lim}$ for several countable spaces, both compact and not compact:

Corollary II.3.15. $\mathsf{wList}_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{wList}_{\mathcal{K}} \equiv_{\mathrm{W}} \mathsf{wList}_{\mathbb{Q}} \equiv_{\mathrm{W}} \mathsf{lim}$, *where* $\mathcal{K} = \{0\} \cup \{2^{-n} : n \in \mathbb{N}\}$.

Proposition II.3.16. $\mathsf{wList}_{\mathbb{N}} <_{\mathrm{W}} \mathsf{List}_{\mathbb{N}}$.

Proof. The fact that $\mathsf{wList}_{\mathbb{N}} \leqslant_W \mathsf{List}_{\mathbb{N}}$ is trivial. For strictness, we show that $\mathsf{LPO}' \leqslant_W \mathsf{List}_{\mathbb{N}}$: since $\mathsf{LPO}' \,|_W\, \mathsf{lim}$ and $\mathsf{wList}_{\mathbb{N}} \equiv_W \mathsf{lim}$ by Corollary $II.3.15$ this suffices to conclude the proof.

We can think of LPO' as the function that, given in input $p \in 2^{\mathbb{N}}$, is such that $\mathsf{LPO}'(p) = 1 \iff (\exists^\infty i)(p(i) = 0)$. For any $p \in 2^{\mathbb{N}}$, let $A := \{i : p(i) = 0\}$: given $(n, (b_i, p_i)_{i\in\mathbb{N}}) \in \mathsf{List}_{\mathbb{N}}(A)$ it is clear that $\mathsf{LPO}'(p) = 1$ if and only if $n = 0$. □

II.3.3 The Cantor-Bendixson theorem

Notice that it makes sense to study $\mathsf{PK}_{\mathcal{X}}$ only when $\mathcal{X}$ is an uncountable computable Polish space: indeed, if $\mathcal{X}$ is countable, $\mathsf{PK}_{\mathcal{X}}$ is the function with constant value $\emptyset$.

Lemma II.3.17. *For any computable Polish space $\mathcal{X}$ the set $\{C \in \mathcal{A}_-(\mathcal{X}) : |C| \leqslant \aleph_0\}$ is Π^1_1.*

Proof. Let s and A be as in Theorem $II.3.8$ and denote by s_A be the restriction of s to A. By Proposition $II.3.1$, the function $S : \mathcal{A}_-(\mathcal{X}) \to \mathcal{A}_-(\mathbb{N}^{\mathbb{N}})$ defined by $S(C) = s_A^{-1}(C)$ is computable. Since s_A is a bijection, we obtain that $|C| = |S(C)|$. Recall from §I.6 that we can represent $S(C)$ via some $T \in \mathbf{Tr}$ such that $S(C) = [T]$. To conclude the proof notice that $|S(C)| \leqslant \aleph_0$ if and only if $T \in \mathcal{T}^{\leqslant\aleph_0}$ and, by Theorem $I.3.39(ii)$, $\mathcal{T}^{\leqslant\aleph_0}$ is a Π^1_1 set. □

Recall that in §I.2 we fixed an enumeration $(B_i)_{i\in\mathbb{N}}$ of all basic open sets of $\mathcal{X}$, where the ball $B_{\langle n,m\rangle}$ is centered in $\alpha(n)$ and has radius q_m.

Theorem II.3.18. *For every rich computable Polish space $\mathcal{X}$, $\mathsf{PK}_{\mathcal{X}} \equiv_W \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$.*

Proof. The right-to-left direction is Lemma $II.3.5$. For the converse reduction, by Theorem $II.2.9$ we have that $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}} \equiv_W \widehat{\mathsf{WF}_{\mathbb{S}}}$, hence it suffices to show that $\mathsf{PK}_{\mathcal{X}} \leqslant_W \widehat{\mathsf{WF}_{\mathbb{S}}}$. Let $C \in \mathcal{A}_-(\mathcal{X})$ be an input for $\mathsf{PK}_{\mathcal{X}}$.

Notice that $|B_{\langle n,m\rangle} \cap C| \leqslant \aleph_0$ if and only if $(\forall \epsilon > 0)(|\{x \in \mathcal{X} : d(x, \alpha(n)) \leqslant q_m - \epsilon\} \cap C| \leqslant \aleph_0)$: as $\{x \in \mathcal{X} : d(x, \alpha(n)) \leqslant q_m - \epsilon\} \cap C$ is a closed set that can be uniformly computed from C, n and m, by Lemma $II.3.17$ we get that $|B_{\langle n,m\rangle} \cap C| \leqslant \aleph_0$ is Π^1_1.

We can therefore compute a sequence $(T^{\langle n,m\rangle})_{n,m\in\mathbb{N}}$ of trees such that $T^{\langle n,m\rangle} \in \mathcal{WF}$ if and only if $|B_{\langle n,m\rangle} \cap C| \leqslant \aleph_0$. Hence, searching the output of $\widehat{\mathsf{WF}_{\mathbb{S}}}((T^{\langle n,m\rangle})_{n,m\in\mathbb{N}})$ for the $\langle n,m\rangle$'s such that $\mathsf{WF}_{\mathbb{S}}(T^{\langle n,m\rangle}) = 1$, we eventually enumerate all the $B_{\langle n,m\rangle}$ such that $B_{\langle n,m\rangle} \cap \mathsf{PK}_{\mathcal{X}}(C) = \emptyset$, thus obtaining a name for $\mathsf{PK}_{\mathcal{X}}(C) \in \mathcal{A}_-(\mathcal{X})$. □

To prove the next Lemma we use ideas from the proof of Lemma $II.2.24$ and Lemma $II.3.9$.

Lemma II.3.19. *For every computable Polish space $\mathcal{X}$, $\mathsf{wScList}_{\mathcal{X}} \leqslant_W \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}}$.*

Proof. We show that $\mathsf{wScList}_{\mathcal{X}} \leqslant_W \widehat{\mathsf{WF}_{\mathbb{S}}} \times \widehat{\overline{\mathsf{wList}_{\mathcal{X}}}} \leqslant_W \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}}$.

The first reduction is obtained by generalizing the proof of Lemma $II.2.24$ (which is the case $\mathcal{X} = \mathbb{N}^{\mathbb{N}}$): given $C \in \mathcal{A}_-(\mathcal{X})$ it suffices to use as input for the $\langle n,m\rangle$-th instances of $\mathsf{WF}_{\mathbb{S}}$ and $\overline{\mathsf{wList}_{\mathcal{X}}}$ respectively a tree $T^{\langle n,m\rangle}$ such that $T^{\langle n,m\rangle} \in \mathcal{WF}$ if and only if $(\forall \epsilon > 0)(|\{x \in \mathcal{X} : d(x, \alpha(n)) \leqslant q_m - \epsilon\} \cap C| \leqslant \aleph_0)$ (see the proof Theorem $II.3.18$) and $\{x \in \mathcal{X} : d(x, \alpha(n)) \leqslant q_m - \epsilon\} \cap C$.

For the second reduction, notice that $\overline{\mathsf{wList}_{\mathcal{X}}} \leqslant_W \overline{\mathsf{wList}_{\mathbb{N}^{\mathbb{N}}}}$ by essentially the same proof of Lemma $II.3.9$. By $\overline{\mathsf{wList}_{\mathbb{N}^{\mathbb{N}}}} \leqslant_W \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}} \equiv_W \widehat{\mathsf{WF}_{\mathbb{S}}}$ (Theorems $II.2.9$ and $II.2.25$ and Lemma $II.2.23$) and the fact that $\mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}}$ is parallelizable (Proposition $II.2.17$), we obtain the reduction. □

The same proof of Lemma $II.3.10$ yields the following Lemma.

Lemma II.3.20. *Let $\mathcal{X}$ and $\mathcal{Y}$ be computable metric spaces and $\iota : \mathcal{X} \to \mathcal{Y}$ be a computable embedding with* $\mathsf{range}(\iota) \in \Pi^0_1(\mathcal{Y})$*. Then* $(\mathsf{w})\mathsf{ScList}_{\mathcal{X}} \leqslant_{\mathrm{W}} (\mathsf{w})\mathsf{ScList}_{\mathcal{Y}}$*. In particular,* $(\mathsf{w})\mathsf{ScList}_{2^{\mathbb{N}}} \leqslant_{\mathrm{W}} (\mathsf{w})\mathsf{ScList}_{\mathcal{Y}}$ *for every rich computable metric space $\mathcal{Y}$.*

Recall that the problems $\mathsf{PK}_{\mathcal{X}}$, where $\mathcal{X}$ is a rich computable Polish space, are all Weihrauch equivalent (Theorem *II*.3.18). Combining Lemmas *II*.3.20 and *II*.3.19, we obtain $\mathsf{wScList}_{2^{\mathbb{N}}} \leqslant_{\mathrm{W}} \mathsf{wScList}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}}$, but we do not know whether for some rich computable Polish space $\mathcal{X}$ both reductions are strict (see Question *II*.4.8).

Theorem II.3.21. *For any rich computable Polish space $\mathcal{X}$,* $\mathsf{wCB}_{\mathcal{X}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$.

Proof. For the left-to-right reduction notice that $\mathsf{wCB}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{PK}_{\mathcal{X}} \times \mathsf{wScList}_{\mathcal{X}}$. By Theorem *II*.3.18 and Lemma *II*.3.19 we know that $\mathsf{PK}_{\mathcal{X}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$ and $\mathsf{wScList}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}}$. Since $\mathsf{wScList}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$ (Theorem *II*.2.25) and $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$ is parallelizable (Proposition *II*.2.6), this concludes the reduction. The other direction follows from the combination of Theorem *II*.3.18 and the fact that $\mathsf{PK}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{wCB}_{\mathcal{X}}$. ☐

In the literature, there are many equivalent definitions of computably compact represented spaces. The following is the most convenient for our purposes.

Definition II.3.22 ([Pau16, §5])**.** A subset K of a represented space $\mathcal{X}$ is computably compact if $\{A \in \mathcal{A}_-(\mathcal{X}) : A \cap K = \varnothing\}$ is Σ^0_1.

Definition II.3.23. A computable metric space $\mathcal{X}$ is computably K_σ if there exists a computable sequence $(K_i)_{i\in\mathbb{N}}$ of nonempty computably compact sets with $\mathcal{X} = \bigcup_{i\in\mathbb{N}} K_i$.

The following remark extends Remark *II*.2.28 to computably K_σ spaces.

Remark II.3.24. Let $\mathcal{X}$ be a computably K_σ space and let $(K_i)_{i\in\mathbb{N}}$ witness this property.

Notice that for $C \in \mathcal{A}_-(\mathcal{X})$, $C = \varnothing$ if and only if $(\forall i)(K_i \cap C = \varnothing)$, i.e. a Π^0_2 condition. Moreover, $C \cap B_{\langle n,m\rangle} = \varnothing$ if and only if $(\forall k)(\{x \in \mathcal{X} : d(x, \alpha(n)) \leqslant q_m - 2^{-k}\} \cap C = \varnothing)$, so that this condition is Π^0_2 as well.

Now, $|C| = 1$ if and only if both $C \neq \varnothing$ and

$$(\forall n, n', m, m')\big(d(\alpha(n), \alpha(n')) \geqslant q_m + q_{m'} \implies B_{\langle n,m\rangle} \cap C = \varnothing \vee B_{\langle n',m'\rangle} \cap C = \varnothing\big) \quad (1)$$

where (1) is a Π^0_2 condition. Now $\left|C \cap B_{\langle n,m\rangle}\right| = 1$ is the conjunction of a Σ^0_2 and a Π^0_2 formula because it is equivalent to $C \cap B_{\langle n,m\rangle} \neq \varnothing$ and

$$(\forall n', n'', m', m'')\big(d(\alpha(n), \alpha(n')) \leqslant q_m - q_{m'} \wedge d(\alpha(n), \alpha(n'')) \leqslant q_m - q_{m''} \wedge \\ \wedge\, d(\alpha(n'), \alpha(n'')) \geqslant q_{m'} + q_{m''} \implies B_{\langle n',m'\rangle} \cap C = \varnothing \vee B_{\langle n'',m''\rangle} \cap C = \varnothing\big).$$

Lemma II.3.25. *For every rich computable Polish computably K_σ space $\mathcal{X}$,* $\mathsf{CB}_{\mathcal{X}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$.

Proof. The right-to-left direction follows from the facts that $\mathsf{PK}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{CB}_{\mathcal{X}}$ and that, by Theorem *II*.3.18, $\mathsf{PK}_{\mathcal{X}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$.

For the opposite direction, notice that $\mathsf{CB}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{wCB}_{\mathcal{X}} \times \mathsf{ScCount}_{\mathcal{X}}$. Since $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$ is parallelizable (Proposition *II*.2.6) and $\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{wCB}_{\mathcal{X}}$ (Theorem *II*.3.21), it suffices to show that $\mathsf{ScCount}_{\mathcal{X}} \leqslant_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$. To do so, we now adapt the proof of Theorem *II*.2.32(*ii*) to show that $\mathsf{ScCount}_{\mathcal{X}} \leqslant_{\mathrm{W}} \boldsymbol{\Pi}^0_4\text{-}\mathsf{C}_{\mathbb{N}}$: this concludes the proof as $\boldsymbol{\Pi}^0_4\text{-}\mathsf{C}_{\mathbb{N}} \leqslant_{\mathrm{W}} \boldsymbol{\Pi}^1_1\text{-}\mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} {}^1\mathsf{PK}_{\mathbb{N}^{\mathbb{N}}}$ (Proposition $II.2.10$). Given in input $C \in \mathcal{A}_-(\mathcal{X})$, let

$$A := \{k : (k > 0 \implies \varphi(k-1, C) \wedge \neg\varphi(k, C)) \wedge (k = 0 \implies (\forall m)(\varphi(m, C)))\},$$

where $\varphi(k, C)$ says that there exists a finite string $\sigma = (\langle n_0, q_0\rangle, \dots, \langle n_{k-1}, q_{k-1}\rangle) \in \mathbb{N}^k$ such that for every $i \neq j < k$,

$$d(\alpha(n_i), \alpha(n_j)) \geqslant q_i + q_j \wedge \left|C \cap B_{\langle n_i, q_i\rangle}\right| = 1.$$

By Remark $II.3.24$, it is easy to check that each φ is Σ^0_3 and hence A is Π^0_4. By Remark $II.2.16$ the unique $k \in A$ is the correct answer for $\mathsf{ScCount}_{\mathcal{X}}(C)$. □

The final part of this section is devoted to spaces that are not K_σ.
Recall the following consequence of Hurewicz's theorem from classical descriptive set theory.

Theorem II.3.26 ([Kec12, Theorem 7.10]). *Let $\mathcal{X}$ be a Polish space. Then there is an embedding $\iota : \mathbb{N}^\mathbb{N} \to \mathcal{X}$ such that $\mathsf{range}(\iota)$ is closed if and only if X is not K_σ.*

Definition II.3.27. We say that a computable Polish space is computably non-K_σ if there exists a computable embedding $\iota : \mathbb{N}^\mathbb{N} \to \mathcal{X}$ such that $\mathsf{range}(\iota) \in \Pi^0_1(\mathcal{X})$.

The following theorem is a corollary of Lemma $II.3.4$.

Theorem II.3.28. *For any rich computable Polish space $\mathcal{X}$ which is computably non-K_σ, $\mathsf{CB}_{\mathbb{N}^\mathbb{N}} \leqslant_\mathrm{W} \mathsf{CB}_{\mathcal{X}}$.*

We leave open the question of whether there is a rich computable Polish space $\mathcal{X}$ such that $\mathsf{CB}_{\mathbb{N}^\mathbb{N}} <_\mathrm{W} \mathsf{CB}_{\mathcal{X}}$ (see Question $II.4.7$).

II.4 Conclusions and open questions

In this chapter, we studied problems related to the Cantor-Bendixson theorem. In contrast with reverse mathematics, we showed that many such problems lie in different Weihrauch degrees; some of these problems still lack a complete classification.

In Theorem $II.2.11$, we showed that $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} <_\mathrm{W} \mathsf{PK} \leqslant_\mathrm{W} \mathsf{lim} * \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. Upon hearing about this result, Linda Westrick asked the following question.

Question II.4.1. Is it true that $\mathsf{PK} \equiv_\mathrm{W} \mathsf{lim} * \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$?

By Theorems $II.2.27$, $II.2.29$, and $I.6.10$, and the fact that $\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$ is a first-order problem, we obtain $\mathsf{WF}^* \leqslant_\mathrm{W} \mathsf{ScCount}_{\mathbb{N}^\mathbb{N}} \leqslant_\mathrm{W} {}^1\widehat{\mathsf{WF}} \equiv_\mathrm{W} \mathsf{WF}^{u*}$. By [SV22, Corollary 7.8], $\mathsf{WF}^* <_\mathrm{W} \mathsf{WF}^{u*}$ and therefore at least one of the inequalities is strict.

Question II.4.2. Characterize the Weihrauch degree of $\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$.

In particular, if $\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}} \equiv_\mathrm{W} {}^1\widehat{\mathsf{WF}}$ we would obtain a nice characterization of the first-order part of $\widehat{\mathsf{WF}}$.

A related question is the following.

Question II.4.3. Is it true that $\mathsf{CB}_{\mathbb{N}^\mathbb{N}} \leqslant_\mathrm{W} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$ (and hence $\mathsf{CB}_{\mathbb{N}^\mathbb{N}} \equiv_\mathrm{W} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$)?

In Theorems $II.2.35$ and $II.2.32(iii)$ we proved that $\mathsf{ScList}_{2^\mathbb{N}} <_\mathrm{W} \mathsf{CB}_{2^\mathbb{N}}$ and $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} <_\mathrm{W} \mathsf{CB}$. A negative answer to Question $II.4.3$ would confirm this pattern. However, we have $\mathsf{PK}_{\mathbb{N}^\mathbb{N}} <_\mathrm{W} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$, while $\mathsf{ScList}_{2^\mathbb{N}} <_\mathrm{W} \mathsf{PK}_{2^\mathbb{N}} \equiv_\mathrm{W} \mathsf{CB}_{2^\mathbb{N}}$ and $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} <_\mathrm{W} \mathsf{PK} \equiv_\mathrm{W} \mathsf{CB}$ (Theorem $II.2.35$, Proposition $II.2.34$, and Theorem $II.2.32(iii)$): therefore the situation in $\mathbb{N}^\mathbb{N}$ differs from those in $2^{\mathbb{N}}$ and **Tr** and a positive answer is possible. In this case, we would obtain an unexpected result: namely, that the gap between $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ and $\mathsf{CB}_{\mathbb{N}^\mathbb{N}}$ is due entirely to the scattered part and its cardinality, rather than to the perfect kernel. If this is the case, the cardinality of the scattered part (i.e. $\mathsf{ScCount}_{\mathbb{N}^\mathbb{N}}$) would be of crucial importance because the scattered part on its own is not enough as witnessed by the fact that $\mathsf{wScList}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^\mathbb{N}}$ (Theorems *II*.2.25 and *II*.2.35).

The following questions are strictly related and concern the relationship of two of our problems with $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$, which plays a major role in the Weihrauch lattice. Choice principles have a convenient definition and hence, it is quite natural to compare any problem with them. In particular, $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ plays a pivotal role among the problems that, from the point of view of reverse mathematics, are equivalent to ATR_0.

Question II.4.4. Is it true that $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^\mathbb{N}}$? Even more, does $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{ScList}_{\mathbb{N}^\mathbb{N}}$ hold?

By Propositions *II*.2.12 and *II*.2.34 and Theorem *II*.2.36 we obtain $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \nleqslant_{\mathrm{W}} \mathsf{PK}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} * \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. Since $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{CB}_{\mathbb{N}^\mathbb{N}}$ this implies that to answer negatively both questions it suffices to show that $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \nleqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} * \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$. By [BDP12, Theorem 7.11] we know that $f \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$ if and only if f is computable with finitely many mind changes. In other words, $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} * \mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ if and only if $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ can be reduced to $\mathsf{PK}_{\mathbb{N}^\mathbb{N}}$ employing a backward functional which is computable with finitely many mind changes: intuitively, this seems unlikely to hold.

The last section left open some interesting questions. First of all, by Lemma *II*.3.5 we have that $\mathsf{PST}_{\mathbb{N}^\mathbb{N}}$ is a lower bound for $\mathsf{PST}_{\mathcal{X}}$ whenever $\mathcal{X}$ is a rich computable Polish space. In Theorem *II*.3.6 we showed that equivalence holds when $\mathcal{X} = [0,1]$ or $\mathcal{X} = \mathbb{R}$.

Question II.4.5. Is there a rich computable Polish space $\mathcal{X}$ such that $\mathsf{PST}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \mathsf{PST}_{\mathcal{X}}$?

Concerning the listing problems for countable closed sets, the situation for the so-called weak lists is quite clear, while we do not have a satisfactory result for problems requiring also the cardinality of the set. An open question is the following:

Question II.4.6. Does $\mathsf{List}_{2^\mathbb{N}} <_{\mathrm{W}} \mathsf{List}_{\mathbb{R}}$?

By Theorems *II*.3.18 and *II*.3.21 all problems of the form $\mathsf{PK}_{\mathcal{X}}$ and $\mathsf{wCB}_{\mathcal{X}}$ belong to the same Weihrauch degree as long as $\mathcal{X}$ is a rich computable Polish space. In contrast, we do not know if the same happens with $\mathsf{CB}_{\mathcal{X}}$.

Question II.4.7. Is there a rich computable Polish space $\mathcal{X}$ such that $\mathsf{CB}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \mathsf{CB}_{\mathcal{X}}$? By Lemma *II*.3.25 if such $\mathcal{X}$ exists must be computably non-K_σ. This problem is strictly related to the existence of a rich computable Polish space $\mathcal{X}$ such that $\mathsf{ScList}_{\mathbb{N}^\mathbb{N}} <_{\mathrm{W}} \mathsf{ScList}_{\mathcal{X}}$.

The last problem concerns the weak form of listing the scattered part of a set.

Question II.4.8. Is there a rich computable Polish space $\mathcal{X}$ such that $\mathsf{wScList}_{2^\mathbb{N}} <_{\mathrm{W}} \mathsf{wScList}_{\mathcal{X}} <_{\mathrm{W}} \mathsf{wScList}_{\mathbb{N}^\mathbb{N}}$?

III

The (induced) subgraph problem

III.1 Introduction to the problem

All the results in this chapter are a joint work with Arno Pauly and are collected in [CP23, CP22]. We also thank Alberto Marcone for the many suggestions.

We study the *subgraph problem* and the *induced subgraph problem* by means of (effective) Wadge reducibility and Weihrauch reducibility. The corresponding decision problems for finite graphs, i.e. the one taking as input graphs G and H and answering whether G is an (induced) subgraph of H, are well studied in the context of finite complexity theory. The same problems for infinite graphs have been less studied, but still in the literature there are many interesting results, usually involving computable graphs, i.e. graphs having a vertex set which is computably isomorphic to $\mathbb{N}$ and computable edge relation. An intriguing research direction in this field is to understand why some NP-complete problems on finite graphs become relatively "easy" in the infinite case, and others more complex. For example, in [BG89], the authors exhibit problems that for finite graphs are NP-complete complete (e.g. 3-colorability), for infinite graphs are at the low levels of the arithmetical hierarchy. On the other hand, in [Har91] it was shown that the decision problem asking whether a computable graph has a Hamiltonian path (i.e. a path visiting every vertex of the graph), which is NP-complete in the finite case, is Σ^1_1-complete, where completeness here is defined with respect to effective Wadge reducibility. See also [MR72, Bea76, AMS92] for other related works.

In the same spirit, Hirst and Lempp in [HL96] studied the interplay between finite complexity theory and reverse mathematics starting from the following analogy: deciding whether a finite graph has a Hamiltonian path is NP complete, but the same problem for Eulerian paths (i.e. paths visiting every edge in the graph) is in P. In reverse mathematics, the problem that, given as input an infinite sequence of infinite graphs, decides which ones have Hamiltonian paths is equivalent to $\mathbf{\Pi}^1_1-\mathsf{CA}_0$, while the corresponding problem for Eulerian graphs is equivalent to the weaker subsystem ACA_0. Unfortunately, in the same paper, the authors show that this parallelism does not hold in general: for example, in the third section of the same paper, they show that different formulations of the induced subgraph problem for infinite graphs are equivalent to $\mathbf{\Pi}^1_1-\mathsf{CA}_0$ while the same problems for finite graphs have a very different strength in finite complexity theory.

In [BHW21, §2] BeMent, Hirst and Wallace continued in this direction studying decision problems related to the ones discussed above in the Weihrauch lattice: for example, the ones that fixed some computable graph G and given in input a graph H decide whether G is an induced subgraph of H. Starting from these problems, we obtain several results about the complexity of (induced) subgraph related problems, not only in terms of Weihrauch reducibility, but also via (effective) Wadge reducibility: in particular, the results obtained in the latter solve questions related to the former.

The present chapter is organized as follows: in §III.2 we discuss different representations of graphs and, in §III.3, we give results on the Γ-completeness, with Γ being a lightface class, of sets of (names of) graphs of the form

$$\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{(\mathbf{i})\mathbf{s}} H\} := \{p \in \mathrm{dom}(\delta_{(E)Gr}) : G \subseteq_{(\mathbf{i})\mathbf{s}} \delta_{(E)Gr}(p)\}. \tag{1}$$

where G is a fixed graph.

In §III.3.1 we turn our attention to Weihrauch reducibility and decision problems: in particular, we solve a question left open in [BHW21, §2]. We also seize the opportunity to discuss again the interplay between Weihrauch reducibility and reverse mathematics mentioned at the beginning of Part 1, exhibiting graph-theoretic representatives of $\mathbf{\Pi}^1_1-\mathsf{CA}_0$ in the Weihrauch lattice. Table III.1 and Figure III.1 summarize the results in §III.3 and §III.3.1: the precise definitions and notations involved are given in due time (same for the next tables and figures).

G finite	$\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{s}} H\}$	Σ^0_1-complete
G finite	$\{H \in \mathbf{Gr} : G \subseteq_{\mathbf{is}} H\}$	Σ^0_1-complete
K_n	$\{H \in \mathbf{EGr} : G \subseteq_{\mathbf{is}} H\}$	Σ^0_1-complete
G finite and $G \not\cong K_n$	$\{H \in \mathbf{EGr} : G \subseteq_{\mathbf{is}} H\}$	Σ^0_2-complete
G c.e. and $K_\omega \nsubseteq_{\mathbf{is}} G$	$\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{is}} H\}$	Σ^1_1-complete
G c.e. and $R_\omega \subseteq_{\mathbf{s}} G$	$\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\}$	Σ^1_1-complete
T_{2k+1} (F_{2k+2})	$\{H \in (\mathbf{E})\mathbf{Gr} : \mathsf{T}_{2k+1}(\mathsf{F}_{2k+2}) \subseteq_{\mathbf{s}} H\}$	Σ^0_{2k+1}-complete (Π^0_{2k+2}-complete)

Table III.1: A summary of the results in §III.3.

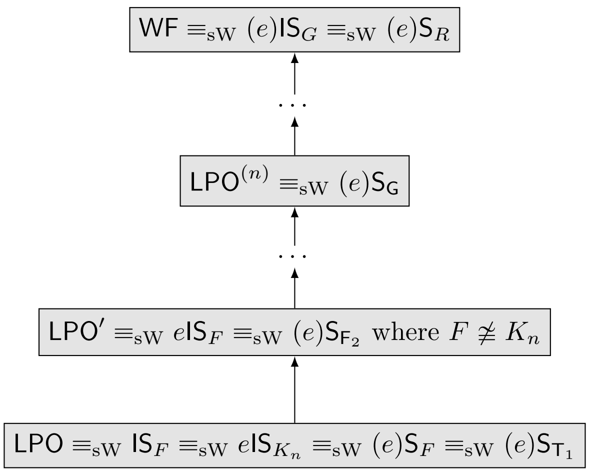

Figure III.1: Some of the multi-valued functions studied in this chapter. Black arrows represent Weihrauch reducibility in the direction of the arrow. Here F represent a finite graph, G an infinite c.e. graph, R a c.e. graph such that $R_\omega \subseteq_{\mathbf{s}} R$ and $\mathsf{G} \in \{\mathsf{T}_{2k+1}, \mathsf{F}_{2k+2}\}$.

The results in §III.3.2 are still preliminary ones, and they concern the (effective) Wadge complexity of sets of (names of) graphs and the Weihrauch complexity of decision problems, that are in some sense "opposite" to the ones considered in §III.3 and §III.3.1: in particular, we partially solve another question left open in [BHW21, §2]. Table III.2 summarizes the results in §III.3.2.

G finite	$\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{s}} G\}$	Π^0_1-complete
G finite	$\{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} G\}$	Π^0_1-complete
$\bigotimes^{\infty} G$ with G finite	$\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{s}} \bigotimes^{\infty} G\}$	Π^0_1-complete
$\bigotimes^{\infty} G$ with G finite	$\{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} \bigotimes^{\infty} G\}$	Π^0_1-complete
K_ω	$\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{s}} K_\omega\}$	computable
K_ω	$\{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} K_\omega\}$	Π^0_1-complete
K_ω	$\{H \in \mathbf{EGr} : H \subseteq_{\mathbf{is}} K_\omega\}$	Π^0_2-complete
$\bigotimes_{i \geqslant 1} R_i$	$\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{(\mathbf{i})\mathbf{s}} \bigotimes_{i \geqslant 1} R_i\}$	Π^0_3-complete
$\bigotimes_{i \geqslant 1} K_i$	$\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{(\mathbf{i})\mathbf{s}} \bigotimes_{i \geqslant 1} K_i\}$	Π^0_3-complete
$\mathcal{S}$	$\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{(\mathbf{i})\mathbf{s}} \mathcal{S}\}$	Π^0_5-hard

Table III.2: A summary of the results in §III.3.2.

Notice that the results in Table III.1 and III.2 allow us to determine the complexity (with respect to Wadge reducibility) of the corresponding sets of graphs in the boldface hierarchy.

In §III.4 we introduce "search problems" that, fixed a graph G and given in input a graph H such that $G \subseteq_{(\mathbf{i})\mathbf{s}} H$ output an (induced) subgraph of H that is isomorphic to G. In particular, we show that the situation for the induced subgraph relation is more "tidy" (i.e. the problems we consider are all Weihrauch equivalent to $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$, sometimes relatively to some oracle, see Theorem $III.4.5$), while the situation for the subgraph relation is more intricate. Indeed, we have different infinite graphs such that the corresponding problems for the subgraph relation are Weihrauch equivalent to $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ (see Proposition $III.4.6$ and Theorem $III.4.40$), others that are computable (see Theorem $III.4.39$) and others that are Weihrauch equivalent to (jumps of) lim (see Theorem $III.4.44$).

Of particular interest is the case when G is R_ω: we show that, restricting the domain of the corresponding problem to connected graph, we obtain examples of natural problems having computable finitary part (see Definition $I.6.15$) but noncomputable first-order part (see Definition $I.6.8$). Furthermore, even without the domain restriction such problems have the peculiar property of being hard to compute, but weak when they have to compute a problem on their own (see Figure III.2). The final section discusses the results and some open questions.

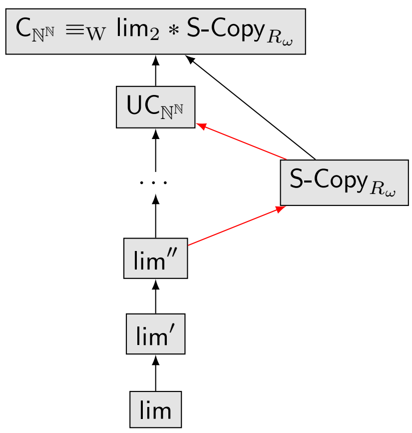

Figure III.2: A summary of the Weihrauch reductions between problems considered in §III.4.3. Black arrows represent Weihrauch reducibility in the direction of the arrow. Red arrows, (differently from Figure II.1 and II.2) represent the absence of a Weihrauch reduction in the direction of the arrow.

III.2 The represented spaces of graphs

We represent graphs in two ways: via their characteristic function or via an enumeration of the vertices and the edges.

- We denote by **Gr** the represented space of graphs represented via characteristic functions, where the representation map δ_{Gr} has domain

  $$\{p \in 2^{\mathbb{N}} : (\forall i, j \in \mathbb{N})(p(\langle i, j \rangle) = 1 \implies p(\langle j, i \rangle) = p(\langle i, i \rangle) = p(\langle j, j \rangle) = 1)\}.$$

  Any graph G has a unique δ_{Gr}-name $p \in 2^{\mathbb{N}}$ such that $i \in V(G) \iff p(\langle i, i \rangle) = 1$ and for $i \neq j$, $(i, j) \in E(G) \iff p(\langle i, j \rangle) = 1$.

- We denote by **EGr** the represented space of graphs represented via enumerations, where the representation map δ_{EGr} has domain

  $$\{p \in \mathbb{N}^{\mathbb{N}} : (\forall i \neq j \in \mathbb{N})(\exists k \in \mathbb{N})(p(k) = \langle i, j \rangle \implies (\exists \ell_0, \ell_1 < k)(p(\ell_0) = \langle i, i \rangle \wedge p(\ell_1) = \langle j, j \rangle))\}.$$

  A graph G instead has multiple δ_{EGr}-names namely $\{p \in \mathrm{dom}(\delta_{EGr}) : (i \in V(G) \iff (\exists k)(p(k) = i)) \wedge ((i, j) \in E(G) \iff (\exists \ell)(p(\ell) = \langle i, j \rangle))\}$.

Regarding the space **EGr**, the extra requirement of enumerating edges only after the involved vertices are enumerated is just to simplify some proofs in this chapter.

It is trivial that given a δ_{Gr}-name for a graph G we can compute a δ_{EGr}-name for G: the converse is false, but the next lemma tells us that from a δ_{EGr}-name for G we can compute a δ_{Gr}-name for a graph H "close" to G, i.e. a graph H containing a copy of G plus vertices of finite degree.

Lemma III.2.1. *There exists a computable* $\mathbf{F} :\subseteq \mathbf{EGr} \rightrightarrows \mathbf{Gr}$ *such that*

$$(\forall G \in \mathbf{EGr})(\forall H \in \mathbf{F}(G))(\exists V \subseteq V(H))(\forall v \in V(H) \setminus V)(G' \upharpoonright_V \cong G \wedge \deg^H(v) < \aleph_0).$$

Proof. Let q be a name for $G \in \mathbf{EGr}$. The function $\mathbf{F}$ computes from q a name p for a graph $H \in \mathbf{Gr}$ in stages. We write $p(\langle i, j \rangle) \uparrow_s$ to denote that, at stage s, we do not have decided yet if $(i, j) \in E(H)$ or, in case $i = j$, if $i \in V(H)$. For any $s \in \mathbb{N}$, we define the auxiliary maps $\iota_s : \mathbb{N} \to \mathbb{N}$ and $\iota : \mathbb{N} \to \mathbb{N}$ with the following properties:

(*i*) for every s, $\mathrm{dom}(\iota_s) := \{v : (\exists i < s)(q(i) = \langle v, v \rangle)\}$ and $\mathrm{dom}(\iota_s) \subseteq \mathrm{dom}(\iota_{s+1})$;

(*ii*) for every $v \in V(G)$, $\iota(v) := \lim_{s \to \infty} \iota_s(v)$ exists.

Construction.
We now describe how to compute p, but before doing so we introduce the following notation: we say that we "erase p up to stage s" if for all $i, j \leqslant s$, if $p(\langle i, j \rangle) \uparrow_s$, we set $p(\langle i, j \rangle) = 0$. We perform this procedure whenever we need to decide if $p(i) = 1$ or $p(i) = 0$ for some $i \in \mathbb{N}$, but we do not have received the information we wanted. This procedure does not affect our construction as, when the information arrives, if it tells us that the vertex/edge coded by i is not in H we have already settled $p(i)$ correctly, otherwise we can choose another vertex/edge coded by some $t > s$ that will play the role of vertex/edge coded by i. This informal description will be clearer during the construction.

At stage 0, do nothing. At stage $s + 1$, suppose $q(s) = \langle u, v \rangle$. First notice that, since an edge (u, v) is enumerated only after u and v have already been enumerated, in this construction it is never the case that $u \neq v \wedge (u \notin \mathrm{dom}(\iota_s) \vee v \notin \mathrm{dom}(\iota_s))$. We have the following cases

- *Case* 1: $u, v \notin \mathrm{dom}(\iota_s)$ and $u = v$. Let $m := \min\{k \notin \mathsf{range}(\iota_s) : k > s\}$. Choosing m as such, we ensure that $p(\langle m, m \rangle) \uparrow_s$ and hence we can set $p(\langle m, m \rangle) = 1$ and $\iota_{s+1}(u) = m$. Then, erase p up to stage s.

- *Case* 2: $u, v \in \mathrm{dom}(\iota_s)$ and $u = v$. By *Case* 1 $p(\langle \iota_s(u), \iota_s(u) \rangle) = 1$. Then, erase p up to stage s.

- *Case* 3: $u, v \in \mathrm{dom}(\iota_s)$ and $u \neq v$,

  (a) if $p(\langle \iota_s(u), \iota_s(v) \rangle) = p(\langle \iota_s(v), \iota_s(u) \rangle) = 1$, then erase p up to stage s;

  (b) if $p(\langle \iota_s(u), \iota_s(v) \rangle) \uparrow_s$ and $p(\langle \iota_s(v), \iota_s(u) \rangle) \uparrow_s$, let
  $$p(\langle \iota_s(u), \iota_s(v) \rangle) = p(\langle \iota_s(v), \iota_s(u) \rangle) = 1.$$
  Then, erase p up to stage s;

  (c) if $p(\langle \iota_s(u), \iota_s(v) \rangle) = 1$ and $p(\langle \iota_s(v), \iota_s(u) \rangle) = \uparrow_s$, let $p(\langle \iota_s(v), \iota_s(u) \rangle) = 1$. Then, erase p up to stage s.;

  (d) if $p(\langle \iota_s(u), \iota_s(v) \rangle) \vee p(\langle \iota_s(v), \iota_s(u) \rangle) = 0$, let $k_u := \min\{k : q(k) = \langle u, u \rangle\}$ and $k_v := \min\{k : q(k) = \langle v, v \rangle\}$: if $k_u < k_v$ declare v injured, u otherwise. Suppose v is injured (the case for u is the same). Then let $m := \min\{k \notin \mathsf{range}(\iota_s) : k > s\}$, let $p(\langle m, m \rangle) = 1$, $\iota_{s+1}(v) = m$ and for every $i \leqslant s$, if $q(i) = \langle v, w \rangle \vee q(i) = \langle w, v \rangle$ let $p(\langle m, \iota_s(w) \rangle) = p(\langle \iota_s(w), m \rangle) = 1$. Then, erase p up to stage s.;

Verification.
We first verify that ι and, for all s, ι_s have the desired properties. To verify the desired properties of ι, notice that (i) is straightforward: to prove (ii), notice that

$$(\forall v \in V(G))(|\{s : v \text{ is injured at stage } s\}| \leqslant |\{w : (v, w) \in E(G) \wedge w < v\}|).$$

Let $s_v := \max\{t : v \text{ is injured at stage } t\}$: then, since the only case in which $\iota_s(v) \neq \iota_{s+1}(v)$ is in *Case* 3(d), we have that for all $(\forall s \geqslant s_v)(\iota_{s_v}(v) = \iota_s(v))$ and hence $\iota(v)$ exists.

We now prove that $(\forall G \in \mathbf{EGr})(\forall H \in \mathbf{F}(G))(\exists V \subseteq V(H))(H\!\upharpoonright_V \cong G)$. Given a name p for $H \in \mathbf{Gr}$, let $V := \{\iota(v) : v \in V(G)\}$. We have to prove that for any $v, u \in V(G)$, $(v, u) \in E(G) \iff p(\langle \iota(v), \iota(u) \rangle) = 1$. For the left-to-right direction, suppose $(v, u) \in E(G)$ and notice that, by (ii), $\iota(v)$ and $\iota(u)$ exists. Let $s_0 := \max\{t : u \text{ or } v \text{ has been injured at stage } t\}$. By (ii) we get that $\iota_{s_0}(v) = \iota(v)$ and $\iota_{s_0}(u) = \iota(u)$ and by construction, $p(\langle \iota(u), \iota(v) \rangle) = 1$. For the opposite direction, notice that $p(\langle \iota(v), \iota(u) \rangle) = 1$ holds only if (u, v) is enumerated in $E(G)$. To conclude the proof, we need to show that $(\forall v \in V(H) \backslash V)(\deg^H(v) < \aleph_0)$. Notice that $v \in V(H) \backslash V$ if and only if $(\exists s, w)(v = \iota_s(w) \neq \iota(w))$ and, by construction, $(\forall t \geqslant s)(p(\langle v, t \rangle) = 0)$, i.e. $\deg^H(v) < \aleph_0$. □

III.3 Effective Wadge complexity of sets of graphs

In this chapter we heavily use the operations on graphs and trees defined in Chapter I. We just recap here their names: the symbols $\bigotimes$ and $\bigodot$ denote respectively the disconnected and connected union of graphs. For a graph G, $\overset{\infty}{\bigotimes} G$ denotes the graph having infinitely many disconnected copies of G and, $\bigsqcup$ denotes the disjoint union of trees.

In this section, we provide examples of Γ-complete sets of (names of) graphs where Γ is a lightface class. For a fixed graph G, we consider sets of the form

$$\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\} := \{p \in \mathrm{dom}(\delta_{(E)Gr}) : G \subseteq_{\mathbf{(i)s}} \delta_{(E)Gr}(p)\}. \tag{2}$$

The next proposition, whose easy proof is omitted, is a direct consequence of **Gr** and **EGr** definitions.

Proposition III.3.1. *For any graph G, $\{H \in \mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\} \leqslant_{\mathsf{EW}} \{H \in \mathbf{EGr} : G \subseteq_{\mathbf{(i)s}} H\}$.*

The following two operations on graphs and trees are fundamental in many constructions of this chapter. Given a tree T and a graph G where $V(G) = \{v_i : i \in \mathbb{N}\}$, let $\mathcal{A}^{\sqsubseteq}(T,G)$ and $\mathcal{A}^{|}(T,G)$ be such that $V(\mathcal{A}^{\sqsubseteq}(T,G)) = V(\mathcal{A}^{|}(T,G)) := T$ (i.e. any node in T is a vertex of the resulting graph) and

- $E(\mathcal{A}^{\sqsubseteq}(T,G)) := \{(\sigma,\tau) : (v_{|\sigma|}, v_{|\tau|}) \in E(G) \wedge (\sigma \sqsubset \tau \vee \tau \sqsubset \sigma)\}$.
- $E(\mathcal{A}^{|}(T,G)) := \{(\sigma,\tau) : (v_{|\sigma|}, v_{|\tau|}) \in E(G) \vee (\sigma \mid \tau)\}$.

Notice that both $\mathcal{A}^{\sqsubseteq}(T,G)$ and $\mathcal{A}^{|}(T,G)$ are computable relative to T and G.

Proposition III.3.2. *Let $T \in \mathcal{IF}$ and let G be an infinite graph. Then $G \subseteq_{\mathbf{is}} \mathcal{A}^{\sqsubseteq}(T,G)$ and $G \subseteq_{\mathbf{is}} \mathcal{A}^{|}(T,G)$.*

Proof. In both cases, suppose that $V(G) = \{v_i : i \in \mathbb{N}\}$, let $p \in [T]$ and consider $V := \{p[n] : n \in \mathbb{N}\}$. By definitions of $E(\mathcal{A}^{\sqsubseteq}(T,G))$ and $E(\mathcal{A}^{|}(T,G))$, it is easy to check that $\mathcal{A}^{\sqsubseteq}(T,G)_{\restriction V} \cong G$ and $\mathcal{A}^{|}(T,G)_{\restriction V} \cong G$ as well, and this concludes the proof. □

Remark III.3.3. In this chapter, most of the results are stated for graphs having a computable/c.e./hyperarithmetical copy. In §III.1, we said that a graph is computable if its vertex set is computably isomorphic to $\mathbb{N}$ and it has a computable edge relation: similar definitions hold for c.e. and hyperarithmetical graphs. From now on, for notational simplicity, we modify these definitions and we say that a graph is computable if it has a copy with vertex set computably isomorphic to $\mathbb{N}$ and computable edge relation: in all proofs, when we consider a computable graph, we always consider a computable copy of it (similarly for c.e. and hyperarithmetical graphs). For the computable case, this is equivalent to say that a computable graph has a computable δ_{Gr}-name and, for the c.e. case, that a c.e. graph has a computable δ_{EGr}-name.

In this section, we show that if we restrict to a computable or c.e. graph G, except when G is finite but not isomorphic to any K_n, we obtain that both $\{H \in \mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\}$ and $\{H \in \mathbf{EGr} : G \subseteq_{\mathbf{(i)s}} H\}$ are Γ-complete for some lightface class Γ, and hence, in these cases, the conclusion of Proposition $III.3.1$ becomes an equivalence. To prove results of this kind, we usually perform the following steps:

- show that $\{H \in \mathbf{EGr} : G \subseteq_{\mathbf{(i)s}} H\}$ is in Γ;
- prove that for any Γ-complete set A, $A \leqslant_{\mathsf{EW}} \{H \in \mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\}$;
- apply Proposition $III.3.1$ to obtain that both $\{H \in \mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\}$ and $\{H \in \mathbf{EGr} : G \subseteq_{\mathbf{(i)s}} H\}$ are Γ-complete (and hence effectively Wadge equivalent).

The following is a technical lemma about the complexity of some graph related formulas.

Lemma III.3.4. *Let $G \in (\mathbf{E})\mathbf{Gr}$, let $v, w \in V(G)$ and $d \in \mathbb{N}$. Then $v \leftrightsquigarrow^G w$ is a Σ^0_1 formula and $\deg^G(v) \leqslant d$ is a Π^0_1 formula.*

Proof. Given a name p for $G \in \mathbf{Gr}$ we have $v \leftrightsquigarrow^G w$ if and only if

$$(\exists n > 0)(\exists \sigma \in \mathbb{N}^n)(p(\langle \sigma(i), \sigma(i+1)\rangle) = 1 \wedge \sigma(0) = v \wedge \sigma(|\sigma|-1) = w \wedge \\ (n > 1 \implies (\forall i \neq j < |\sigma|-1)(\sigma(i) \neq \sigma(j)))),$$

and $\deg^G(v) \leqslant d$ if and only if

$$(\forall v_0, \dots, v_d \in V(G))(\forall i \leqslant d)(p(\langle v, v_i\rangle) = 1 \implies (\exists i, j \leqslant d)(i = j)).$$

Given a name q for $G \in \mathbf{EGr}$ we have $v \leftrightsquigarrow^G w$ if and only if

$$(\exists n > 0)(\exists \sigma \in \mathbb{N}^n)(\forall m < |\sigma|)(\exists k_m)(q(k_m) = \langle \sigma(i), \sigma(i+1)\rangle \wedge \sigma(0) = v \wedge \sigma(|\sigma|-1) = w \wedge \\ (n > 1 \implies (\forall i \neq j < |\sigma|-1)(\sigma(i) \neq \sigma(j)))),$$

and $\deg^G(v) \leqslant d$ if and only if

$$(\forall (k_0, v_0), \ldots, (k_d, v_d) \in \mathbb{N} \times V(G))(\forall i \leqslant n)(q(k_i) = \langle v, v_i\rangle \implies (\exists i, j \leqslant n)(i = j)).$$

□

We start our investigation on the (effective) Wadge complexity of sets of graphs line in (2).

Proposition III.3.5. *For any hyperarithmetical graph G, the following sets are Σ^1_1:*

$$\textit{(i)}\ \{H \in \mathbf{EGr} : G \subseteq_{\mathsf{s}} H\},\ \textit{(ii)}\ \{H \in \mathbf{Gr} : G \subseteq_{\mathsf{s}} H\},$$

$$\textit{(iii)}\ \{H \in \mathbf{Gr} : G \subseteq_{\mathsf{is}} H\}\ \textit{(iv)}\ \{H \in \mathbf{EGr} : G \subseteq_{\mathsf{is}} H\}.$$

If G is finite, the first three sets are Σ^0_1-complete. For any $n \in \mathbb{N}$, $\{H \in \mathbf{EGr} : K_n \subseteq_{\mathsf{is}} H\}$ is Σ^0_1-complete, otherwise, if $G \not\cong K_n$, the set $\{H \in \mathbf{EGr} : G \subseteq_{\mathsf{is}} H\}$ is Σ^0_2-complete.

Proof. We start showing that all sets in $(i) - (iv)$ are Σ^1_1.

(i) Let p be a name for $H \in \mathbf{EGr}$. Then $G \subseteq_{\mathsf{s}} H$ if and only if

$$(\exists f)(\forall i, j \in V(G))((i = j \vee (i,j) \in E(G)) \implies (\exists k)(p(k) = \langle f(i), f(j)\rangle)).$$

(ii) Let p be a name for $H \in \mathbf{Gr}$. Then $G \subseteq_{\mathsf{s}} H$ if and only if

$$(\exists f)(\forall i, j \in V(G))(((i = j \vee (i,j) \in E(G)) \implies p(\langle f(i), f(j)\rangle) = 1)).$$

(iii) Let p be a name for $H \in \mathbf{Gr}$. Then $G \subseteq_{\mathbf{is}} H$ if and only if

$$(\exists f)(\forall i, j \in V(G))(((i = j \vee (i,j) \in E(G)) \iff p(\langle f(i), f(j)\rangle) = 1)).$$

(iv) Let p be a name for $H \in \mathbf{EGr}$. Then $G \subseteq_{\mathbf{is}} H$ if and only if

$$(\exists f)(\forall i, j \in V(G))((i = j \vee (i,j) \in E(G)) \iff (\exists k)(p(k) = \langle f(i), f(j)\rangle)).$$

All the formulas are Σ^1_1 and this concludes the first part of the proof.

In case G is finite, the first existential quantification ranges over $\mathbb{N}$ while the quantification over $V(G)$ ranges over a finite set. This shows that the sets (i), (ii), and (iii) are Σ^0_1 sets, and (iv) is Σ^0_2. On the other hand, for any $n \in \mathbb{N}$, given a name p for $H \in \mathbf{EGr}$, $K_n \subseteq_{\mathbf{is}} H$ if and only if

$$(\exists \sigma \in \mathbb{N}^{|E(K_n)|+|V(K_n)|})(\forall i, j \in V(K_n))(\exists k)(p(k) = \langle \sigma(i), \sigma(j)\rangle),$$

and hence the formula defining the set in (iv) is Σ^0_1.

We now show that, for finite graphs, the sets above are complete in the corresponding lightface classes. We prove that the set in (iii) is Σ^0_1-complete and the set in (iv) is Σ^0_2-complete, and we skip the proofs for the sets in (i) and (ii) as the proof follows the same pattern. To do so, we prove that $\{p \in 2^{\mathbb{N}} : (\exists i)(p(i) = 1)\} \leqslant_{\mathsf{EW}} \{H \in \mathbf{Gr} : G \subseteq_{\mathbf{s}} H\}$ (recall that, by Lemma I.3.35, the left-hand-side set is Σ^0_1-complete). Given $p \in 2^{\mathbb{N}}$ we define a computable $f : 2^{\mathbb{N}} \to \mathbf{Gr}$ such that

- if $p(s) = 0$, then let $f(p[s])$ be the empty graph;
- if $p(s) = 1$, then let $f(p) \cong G$, i.e. add a fresh copy of G to the graph we are computing and stop the construction.

To show that f is an effective Wadge reduction it suffices to notice that, if $(\exists i)(p(i) = 1)$ then clearly $G \subseteq_{\mathbf{is}} f(p) \cong G$ while if $(\forall i)(p(i) = 1)$ then $f(p)$ is the empty graph and no nonempty graph is an induced subgraph of the empty graph.

To conclude the proof, it remains to prove that $\{H \in \mathbf{EGr} : G \subseteq_{\mathsf{is}} H\}$ for $G \not\cong K_n$ for some $n \in \mathbb{N}$ is Σ^0_2-complete and to do so we show that $\{p : (\forall^\infty i)(p(i) = 0)\} \leqslant_{\mathsf{EW}} \{H \in \mathbf{EGr} : G \subseteq_{\mathsf{is}} H\}$ (recall that, by Lemma $I.3.35$, the left-hand-side set is Σ^0_2-complete). Given $p \in 2^{\mathbb{N}}$, we define a computable $f : 2^{\mathbb{N}} \to \mathbf{EGr}$ such that

- if $p(s) = 0$, then let $f(p[s]) := f(p[s-1]) \bigotimes G'$, where G' is a fresh copy of G;
- if $p(s) = 1$, then let $f(p[s]) \cong K_{n_s}$ where $n_s := |\{i : i \in V(f(p[s-1]))\}|$, i.e. enumerate enough edges in the graph we are computing to make it isomorphic to a complete finite graph.

Finally, if $(\forall^\infty i)(p(i) = 0)$ then $f(p) \cong K_n \bigotimes \bigotimes^{\infty} G$ for some $n \in \mathbb{N}$, and clearly $G \subseteq_{\mathbf{is}} f(p)$. Otherwise, if $(\exists^\infty i)(p(i) = 1)$ then $f(p) \cong K_\omega$. Combining the fact that $f(p)$ is an induced subgraph of K_ω if and only if $f(p) \cong K_n$ for some $n \in \mathbb{N}$ or $f(p) \cong K_\omega$ and the fact that $G \not\cong K_\omega$ and $G \not\cong K_n$ for any $n \in \mathbb{N}$, we conclude that $G \nsubseteq_{\mathbf{is}} f(p)$. □

Before moving to the case when G is infinite, we give the following lemma, which is a particular case of the chain antichain principle (CAC). This principle asserts that each partial order on $\mathbb{N}$ contains an infinite chain or an infinite antichain (notice that a tree, in particular, is a partial order, where the order is given by the prefix order of the tree). This principle is well-studied in reverse mathematics and its proof is an easy application of Ramsey theorem for pairs.

Lemma III.3.6. *Let T be a well-founded tree and S an infinite subset of T. Then S contains an infinite anti-chain, i.e. a set of nodes that are pairwise incomparable with respect to the prefix order of T.*

In the next three proofs, the function $\mathbf{F}$ is the one defined in Lemma $III.2.1$: recall by Remark $III.3.3$, that for a graph G, being c.e. and having a computable δ_{EGr}-name are equivalent.

Theorem III.3.7. *Let G be a c.e. graph such that $K_\omega \nsubseteq_{\mathbf{is}} G$. Then $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathsf{is}} H\}$ are Σ^1_1-complete.*

Proof. By Proposition $III.3.5$ both sets are Σ^1_1 and by Proposition $III.3.1$, it suffices to prove that $\{H \in \mathbf{Gr} : G \subseteq_{\mathbf{is}} H\}$ is Σ^1_1-complete. To do so, we show that $\mathcal{IF} \leqslant_{\mathsf{EW}} \{H \in \mathbf{Gr} : G \subseteq_{\mathbf{is}} H\}$ (recall that, by Theorem $I.3.39$, $\mathcal{IF}$ is Σ^1_1-complete). Let $T \in \mathbf{Tr}$, consider $G' \in \mathbf{F}(G)$ and, given $V(G') = \{v_i : i \in \mathbb{N}\}$, compute $\mathcal{A}^{|}(T, G)$. Notice that

- if $T \in \mathcal{IF}$, by Proposition $III.3.2$, $G' \subseteq_{\mathbf{is}} \mathcal{A}^{|}(T, G')$ and hence, since $G \subseteq_{\mathbf{is}} G'$, we obtain that $G \subseteq_{\mathbf{is}} \mathcal{A}^{|}(T, G')$;
- if $T \in \mathcal{WF}$, we claim that $G \nsubseteq_{\mathbf{is}} \mathcal{A}^{|}(T, G')$. If T is finite (i.e. T has finitely many nodes), since G is infinite, the claim follows trivially. If T is infinite, by Lemma $III.3.6$, for any infinite H that is an induced subgraph of $\mathcal{A}^{|}(T, G')$ there exists H' that is an induced subgraph of H such that $V(H')$ is an anti-chain in T. By definition of $E(\mathcal{A}^{|}(T, G'))$, all incomparable nodes in T are connected by an edge, and hence $H' \cong K_\omega$ is a subgraph of H. Since $K_\omega \nsubseteq_{\mathbf{s}} G$ this concludes the proof of the claim.

Hence, T is ill-founded if and only if $G \subseteq_{\mathbf{is}} \mathcal{A}^{\restriction}(T, G')$, and this concludes the proof. □

While the theorem above holds only for the induced subgraph relation, the following holds also for the subgraph one.

Theorem III.3.8. *Let G be a c.e. graph such that $R_\omega \subseteq_{\mathbf{s}} G$. Then the sets $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{(\mathbf{i})\mathbf{s}} H\}$ are Σ^1_1-complete.*

Proof. By Proposition $III.3.5$ all four sets are Σ^1_1 and by Proposition $III.3.1$, it suffices to show that the sets $\{H \in \mathbf{Gr} : G \subseteq_{(\mathbf{i})\mathbf{s}} H\}$ are Σ^1_1-complete. We only show it for the subgraph relation, as the same proof works also for the induced one. As in Theorem $III.3.7$, we show that $\mathcal{IF} \leqslant_{\mathsf{EW}} \{H \in \textbf{repspacegraphs} : G \subseteq_{\mathbf{s}} H\}$. Let $T \in \mathbf{Tr}$, consider $G' \in \mathbf{F}(G)$ and, given $V(G') = \{v_i : i \in \mathbb{N}\}$, compute $\mathcal{A}^{\sqsubseteq}(T, G)$. Notice that:

- if $T \in \mathcal{IF}$, by Proposition $III.3.2$, $G' \subseteq_{\mathbf{is}} \mathcal{A}^{\sqsubseteq}(T, G')$ and hence, since $G \subseteq_{\mathbf{is}} G'$, $G \subseteq_{\mathbf{is}} \mathcal{A}^{\sqsubseteq}(T, G')$;
- if $T \in \mathcal{WF}$, we claim that $G \nsubseteq_{\mathbf{s}} \mathcal{A}^{\sqsubseteq}(T, G')$. Since, $R_\omega \subseteq_{\mathbf{s}} G$ by hypothesis, it suffices to show that $R_\omega \nsubseteq_{\mathbf{s}} \mathcal{A}^{\sqsubseteq}(T, G')$. We show the contrapositive: suppose that $R_\omega \subseteq_{\mathbf{s}} \mathcal{A}^{\sqsubseteq}(T, G')$ and let $\{\sigma_i : i \in \mathbb{N}\}$ such that for every i, $(\sigma_i, \sigma_{i+1}) \in E(\mathcal{A}^{\sqsubseteq}(T, G'))$ be the vertices of the copy of R_ω in $\mathcal{A}^{\sqsubseteq}(T, G')$. By definition of $E(\mathcal{A}^{\sqsubseteq}(T, G'))$, $(\sigma, \tau) \in E(\mathcal{A}^{\sqsubseteq}(T, G')) \implies \sigma \sqsubset \tau \vee \tau \sqsubset \sigma$. Hence, T contains infinitely many nodes comparable to each other, i.e. $T \in \mathcal{IF}$ and this proves the claim.

To conclude the proof, notice that $T \in \mathcal{IF}$ if and only if $G \subseteq_{\mathbf{is}} \mathcal{A}^{\sqsubseteq}(T, G')$. □

The next proposition gives us a lower bound to the complexity of sets of the form $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{s}} H\}$ where G is c.e. and infinite.

Proposition III.3.9. *For any infinite c.e. graph G, the sets $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{s}} H\}$ are Π^0_2-hard.*

Proof. Recall that $\{p \in 2^{\mathbb{N}} : (\exists^\infty n)(p(n) = 1)\}$ is Π^0_2-complete (Lemma $I.3.35$) and, by Proposition $III.3.1$, it suffices to show that $\{p \in 2^{\mathbb{N}} : (\exists^\infty n)(p(n) = 1)\} \leqslant_{\mathsf{EW}} \{H \in \mathbf{Gr} : G \subseteq_{\mathbf{s}} H\}$. Let $p \in 2^{\mathbb{N}}$ and consider $G' \in \mathbf{F}(G)$. We compute H letting $V(H) := \{n : p(n) = 1\}$ and $(n, m) \in E(H) \iff p(n) = p(m) = 1$. Notice that:

- if $(\exists^\infty i)(p(i) = 1)$ then $K_\omega \subseteq_{\mathbf{s}} H$, and since any graph is a subgraph of K_ω we obtain that $G \subseteq_{\mathbf{is}} G' \subseteq_{\mathbf{s}} H$;
- if $(\forall^\infty i)(p(i) = 0)$ then H is finite and since G is infinite, we obtain that $G \nsubseteq_{\mathbf{s}} H$.

Hence, $(\exists^\infty i)(p(i) = 1)$ if and only if $G \subseteq_{\mathbf{s}} H$ and this concludes the proof. □

So far, the sets of graphs of the form $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{(\mathbf{i})\mathbf{s}} H\}$ for a fixed graph G we considered are Γ-complete for $\Gamma \in \{\Sigma^0_1, \Sigma^0_2, \Sigma^1_1\}$: by the end of this section, we define sets of the same form that are Γ-complete with Γ varying along the lightface arithmetical hierarchy. To do so, for $k \in \mathbb{N}$, consider the graphs T_{2k+1} and F_{2k+2} where:

$$V(\mathsf{T}_{2k+1}) := \{\sigma \in \mathbb{N}^{<\mathbb{N}} : |\sigma| \leqslant k\},\ V(\mathsf{F}_{2k+2}) := \{\sigma \in \mathbb{N}^{<\mathbb{N}} : 0 < |\sigma| \leqslant k+1\} \text{ and,}$$

for $G \in \{\mathsf{T}_{2k+1}, \mathsf{F}_{2k+2}\}$,

$$E(G) := \{(\sigma, \tau) \in E(G) : \sigma, \tau \in V(G) \wedge (\sigma = \tau[|\tau| - 1] \vee \tau = \sigma[|\sigma| - 1])\}.$$

To have an intuitive idea of the graphs' definition above, notice that T and F stand respectively for "tree" and "forest" where a forest is the disconnected union of countably many trees. Starting from $\mathsf{T}_1 \cong (\{\langle\rangle\}, \varnothing)$ (the graph/tree having a unique vertex), the following relations hold:

$$\mathsf{F}_{2k+2} \cong \bigotimes^{\infty} \mathsf{T}_{2k+1} \text{ and } \mathsf{T}_{2k+3} \cong \bigsqcup_{i \in \mathbb{N}} T_i \text{ where } T_i \text{ is a connected component of } \mathsf{F}_{2k+2}.$$

Notice that all T_i's for $i \in \mathbb{N}$ are well-founded, and for a well-founded tree T we can define its *height* as $\max\{|\sigma| : \sigma \in T\}$: notice that the k in T_{2k+1} (respectively F_{2k+2}) corresponds to the height of T_{2k+1} (the connected components of F_{2k+2}). For example, $\mathsf{T}_1 = \{\langle\rangle\}$ is a tree of height 0, F_2 is the forest having infinitely many trees of height 0, T_3 is a tree of height 1 with infinitely many children, F_4 is the forest made by infinitely many copies of T_3 and so on.

Theorem III.3.10.

- *The sets* $\{H \in (\mathbf{E})\mathbf{Gr} : \mathsf{T}_{2k+1} \subseteq_{\mathbf{s}} H\}$ *are* Σ^0_{2k+1}*-complete;*
- *The sets* $\{H \in (\mathbf{E})\mathbf{Gr} : \mathsf{F}_{2k+2} \subseteq_{\mathbf{s}} H\}$ *are* Π^0_{2k+2}*-complete.*

The proof of this theorem is obtained by combining Lemmas *III*.3.12 and *III*.3.13: the first proves that the sets involved are respectively Σ^0_{2k+1} and Π^0_{2k+2}. The second one shows that (subspaces of) $(\mathbf{E})\mathbf{Gr}$ are hard (and hence by Lemma *III*.3.12 complete) for the corresponding lightface classes.

Remark III.3.11. Lemma *III*.3.13 shows that the complexity of such sets remains the same even if we restrict the graphs H to trees/forests of length at most k: this comes in handy in §III.4.4.

Lemma III.3.12. *Given a graph* H,

- $\mathsf{T}_{2k+1} \subseteq_{\mathbf{s}} H$ *if and only if* $(\exists v_0 \in V(H))(\exists^\infty v_1 \in V(H)) \dots (\exists^\infty v_k \in V(H))(\forall j < k)((v_j, v_{j+1}) \in E(H))$.
- $\mathsf{F}_{2k+2} \subseteq_{\mathbf{s}} H$ *if and only if* $(\exists^\infty v_0 \in V(H))(\exists^\infty v_1 \in V(H)) \dots (\exists^\infty v_k \in V(H))(\forall j < k)((v_j, v_{j+1}) \in E(H))$.

Hence, $\{H \in (\mathbf{E})\mathbf{Gr} : \mathsf{T}_{2k+1} \subseteq_{\mathbf{s}} H\}$ *and* $\{H \in (\mathbf{E})\mathbf{Gr} : \mathsf{F}_{2k+2} \subseteq_{\mathbf{s}} H\}$ *are respectively* Σ^0_{2k+1} *and* Π^0_{2k+2}.

Proof. In both cases, the left-to-right direction is trivial. For the opposite direction, we just prove the second item, as the proof of the first one is similar. Given a graph H, fix $k \in \mathbb{N}$ and, for every $j < k$, let $(v_j^m)_{m\in\mathbb{N}}$ be an enumeration of the v_j's in $V(H)$: notice that, for $i \neq l \leqslant j$ vertices that appear in $(v_i^m)_{m\in\mathbb{N}}$ may appear also in $(v_l^m)_{m\in\mathbb{N}}$. We define a function ι from $V(\mathsf{F}_{2k+2}) \subseteq \mathbb{N}^{<\mathbb{N}}$ to $V(H)$ letting $\iota(\sigma) := v^k_{|\sigma|}$ if and only if

$$(\forall \tau < \sigma)(\iota(\tau) \neq v^k_{|\sigma|}) \text{ and } (\exists l)(\iota(\sigma[|\sigma| - 2]) = v^l_{|\sigma|-1} \wedge (v^k_{|\sigma|}, v^l_{|\sigma|-1}) \in E(H))$$

Notice that, by definition of H, for any σ, $v^k_{|\sigma|}$ as above always exists. Indeed, for $\sigma = \langle\rangle$ this is immediate. For $\sigma \neq \langle\rangle$, it suffices to notice that at any stage we have defined only finitely many vertices in the range of ι while, by hypothesis, $\deg^H(\iota(\sigma[|\sigma| - 2])) = \aleph_0$ (and hence we can always find v^k_j satisfying the conditions above).

Let $\iota(V(\mathsf{F}_{2k+2})) := \{\iota(\sigma) : \sigma \in V(\mathsf{F}_{2k+2})\}$ and consider the graph $H{\restriction_V}$: clearly, $H{\restriction_V} \cong \mathsf{F}_{2k+2}$

and this concludes the proof of the lemma. □

Given a subspace $\mathcal{G} \subseteq (\mathbf{E})\mathbf{Gr}$, we define $\bigotimes \mathcal{G} := \{\bigotimes_{n\in\mathbb{N}} G_n : (\forall n)(G_n \in \mathcal{G})\}$. We denote by $\mathbf{Tr}_{\leqslant k}$ the represented spaces of trees of height at most k: that is,

$$\mathbf{Tr}_{\leqslant k} := \{T \in \mathbf{Tr} : (\forall \sigma \in T)(|\sigma| \leqslant k)\}.$$

Clearly $\mathbf{Tr}_{\leqslant k}$ is a subspace of $\mathbf{Gr}$, and in the next lemma we consider the space $\bigotimes \mathbf{Tr}_{\leqslant_k}$.

Lemma III.3.13. *For $k \in \mathbb{N}$:*

- $Trees_{2k+1} := \{G \in \bigotimes \mathbf{Tr}_{\leqslant_k} : \mathsf{T}_{2k+1} \subseteq_{\mathbf{s}} G\}$ *is* Σ^0_{2k+1}*-hard;*
- $Forests_{2k+2} := \{G \in \bigotimes \mathbf{Tr}_{\leqslant_k} : \mathsf{F}_{2k+2} \subseteq_{\mathbf{s}} G\}$ *is* Π^0_{2k+2}*-hard.*

Proof. By Lemma $III.3.12$, $Trees_{2k+1}$ and $Forests_{2k+2}$ are respectively Σ^0_{2k+1} and Π^0_{2k+2} sets.

To show that the sets above are complete for the corresponding lightface classes, we first prove by induction on k that $Forests_{2k+2}$ is Π^0_{2k+2}-complete. For the base case, since $\{p \in 2^{\mathbb{N}} : (\forall^\infty i)(p(i) = 0)\}$ is Π^0_2-complete (Lemma $I.3.35$), it suffices to show that $\{p \in 2^{\mathbb{N}} : (\forall^\infty i)(p(i) = 0)\} \leqslant_{\mathsf{EW}} Forests_2$. Given $p \in 2^{\mathbb{N}}$, compute $T := \{\langle\rangle\} \cup \{\langle n\rangle : p(n) = 0\}$ and notice that

$$p \in \{p \in 2^{\mathbb{N}} : (\exists^\infty i)(p(i) = 0)\} \iff \deg^T(\langle\rangle) = \aleph_0 \iff \mathsf{F}_2 \subseteq_{\mathbf{s}} T.$$

Assuming that $Forests_{2k+2}$ is Π^0_{2k+2}-complete we aim to show that $Forests_{2(k+1)+2}$ is Π^0_{2k+4}-complete. By Lemma $I.3.38$,

$$Forests^0_{2k+2} := \{(G_n)_{n\in\mathbb{N}} \in (\textstyle\bigotimes \mathbf{Tr}_{\leqslant_k})^{\mathbb{N}} : (\exists^\infty n)(G_n \in Forests_{2k+2})\} \text{ is } \Pi^0_{2k+4}\text{-complete.}$$

Notice that, given $(G_n)_{n\in\mathbb{N}} \in Forests^0_{2k+2}$ for every n, $G_n = \bigotimes_{i\in\mathbb{N}} (T^{n,i})_{i\in\mathbb{N}}$ where $T^{n,i} \in \mathbf{Tr}_{\leqslant k}$: given $(G_n)_{n\in\mathbb{N}} \in (\bigotimes \mathbf{Tr}_{\leqslant_k})^{\mathbb{N}}$, compute $F := \bigotimes_{n\in\mathbb{N}} \left(\bigsqcup_{i\in\mathbb{N}} T^{n,i}\right)$. Informally, F^n is obtained first connecting, for every $n \in \mathbb{N}$, to a new root the roots of the $\{T^{n,i} : i \in \mathbb{N}\}$ and then considering the disconnected union of the resulting trees: notice that by hypothesis all $T^{n,i}$ are in $\mathbf{Tr}_{\leqslant k}$ and, by construction all connected components of F are in $\mathbf{Tr}_{\leqslant k+1}$, and hence $F \in \bigotimes \mathbf{Tr}_{\leqslant k+1}$. It is easy to check that $(G_n)_{n\in\mathbb{N}} \in Forests^0_{2k+2} \iff F \in Forests_{2k+4}$, and hence $Forests_{2k+4}$ is Π^0_{2k+4}-complete.

To conclude the proof, it suffices to prove for any $k \in \mathbb{N}$, $Trees_{2k+1}$ is Σ^0_{2k+1}-complete. For $k = 0$ we can prove the statement showing that $P_1 \leqslant_{\mathsf{EW}} Trees_1$: the proof is an easy adaptation of the one given to prove $P_2 \leqslant_{\mathsf{EW}} Forests_2$. For $k > 0$, the proof is almost the same of the one showing that $Forests_{2k+2}$ is Π^0_{2k+2}-complete. Indeed, by Lemma $I.3.38$

$$Trees^0_{2k+2} := \{(G_n)_{n\in\mathbb{N}} \in (\textstyle\bigotimes \mathbf{Tr}_{\leqslant_k})^{\mathbb{N}} : (\exists n)(G_n \in Forests_{2k+2})\} \text{ is } \Sigma^0_{2k+3}\text{-complete,}$$

and we compute F as above. It is easy to check that $F \in Forests^1_{2k+2} \iff F \in Tree_{2k+3}$, and hence $Tree_{2k+3}$ is Σ^0_{2k+3}-complete and this concludes the proof. □

Due to the fact that we can always think an element of (products) of $\mathbf{Tr}_{\leqslant k}$ as an element of $\mathbf{Gr}$, the previous lemma, together with Lemma $III.3.12$, proves Theorem $III.3.10$.

We leave open whether there exists a computable/c.e. graph G such that $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{s}} H\} \in \Gamma$ for $\Gamma \notin \{\Sigma^0_{2k+1}, \Pi^0_{2k+2}, \Sigma^1_1\}$ for $k \in \mathbb{N}$.

Remark III.3.14. Notice that from the results presented in this section (and from the ones we are presenting in §III.3.2) we also obtain the complexity of the same sets of graphs in the boldface hierarchy: in this context, graphs are given via their characteristic functions, i.e. as an element of $\mathbf{Gr}$, and the completeness is defined with respect to Wadge reducibility. The Wadge reductions we use are the effective ones we use to prove the statements for the lightface case. We do not restate all the results for the boldface case, except for the next one that gives a sort of dichotomy for $\{H \in \mathbf{Gr} : G \subseteq_{\mathbf{is}} H\}$: this kind of result can be found for different structures for example in [Cam05].

Theorem III.3.15. *For any finite graph G, $\{H \in \mathbf{Gr} : G \subseteq_{\mathbf{is}} H\}$ is $\mathbf{\Sigma}^0_1$-complete, otherwise it is $\mathbf{\Sigma}^1_1$-complete.*

Proof. For finite G's the proof is the same as the one in Proposition $III.3.5(iii)$.

For infinite G's, the proof that $\{H \in \mathbf{Gr} : G \subseteq_{\mathbf{is}} H\}$ is $\mathbf{\Sigma}^1_1$ is the same of Proposition $III.3.5$. For completeness, we show that $\mathcal{IF} \leqslant_{\mathbf{W}} \{H \in \mathbf{Gr} : G \subseteq_{\mathbf{is}} H\}$. To do so, we combine the effective Wadge reductions used in the proofs of Theorems $III.3.7$ and $III.3.8$, noticing that if $K_\omega \subseteq_{\mathbf{is}} G$ then $R_\omega \subseteq_{\mathbf{s}} G$. Then it is easy to check that the following function is the desired Wadge reduction

$$f_G(T) := \begin{cases} \mathcal{A}^{\sqsubseteq}(T, G) & \text{if } K_\omega \subseteq_{\mathbf{is}} G, \\ \mathcal{A}^{|}(T, G) & \text{if } K_\omega \nsubseteq_{\mathbf{is}} G, \end{cases}$$

i.e., $T \in \mathcal{IF} \iff G \subseteq_{\mathbf{is}} f_G(T)$. □

III.3.1 Deciding (induced) subgraphs problems in the Weihrauch lattice

In this section, we consider problems of the following form:

Definition III.3.16. For a computable graph G, we define the functions $\mathsf{IS}_G : \mathbf{Gr} \to 2$ and $\mathsf{S}_G : \mathbf{Gr} \to 2$ by

$$\mathsf{IS}_G(H) = 1 \iff G \subseteq_{\mathbf{is}} H \text{ and } \mathsf{S}_G(H) = 1 \iff G \subseteq_{\mathbf{s}} H.$$

The same functions having domain **EGr** are denoted respectively with $e\mathsf{IS}_G$ and $e\mathsf{S}_G$.

Notice that in [BHW21], the authors considered only S_G and IS_G, and they denote these problems respectively with SE_G and S_G.

All the results in this section, are immediate consequences of §III.3: indeed, in the following proofs, the forward and backward functionals witnessing the Weihrauch reduction are respectively the effective Wadge reduction coming from the corresponding result in §III.3 and the identity.

Proposition III.3.17. *For any hyperarithmetical graph G,*

$$\mathsf{IS}_G \leqslant_{\mathrm{sW}} e\mathsf{IS}_G \leqslant_{\mathrm{sW}} \mathsf{WF} \text{ and } \mathsf{S}_G \leqslant_{\mathrm{sW}} e\mathsf{S}_G \leqslant_{\mathrm{sW}} \mathsf{WF}.$$

Proof. It follows by Propositions $III.3.1$ and $III.3.5$ and the fact that WF answers any Σ^1_1-complete question relative to the input (see §I.6) □

The following theorem discusses the problems IS_G and $e\mathsf{IS}_G$ for finite nonempty graphs.

Theorem III.3.18. *For any finite nonempty graph G, $\mathsf{LPO} \equiv_{\mathrm{sW}} (e)\mathsf{S}_G \equiv_{\mathrm{sW}} \mathsf{IS}_G$. If $G \cong K_n$ for some $n > 0$, then $\mathsf{LPO} \equiv_{\mathrm{sW}} e\mathsf{IS}_G$, otherwise $\mathsf{LPO}' \equiv_{\mathrm{sW}} e\mathsf{IS}_G$.*

Proof. The fact that $\mathsf{LPO} \equiv_{\mathrm{sW}} \mathsf{IS}_G$ is from [BHW21, Theorem 27]. Essentially the same proof also gives us that $\mathsf{LPO} \equiv_{\mathrm{sW}} \mathsf{S}_G$. Anyway, all results are direct consequences of Proposition *III*.3.5 and the fact that LPO and LPO' answer respectively any Σ^0_1.complete and Σ^0_2-complete questions relative to the input. □

In [BHW21, Theorem 24] the authors showed that $\mathsf{WF} \equiv_{\mathrm{sW}} (\mathsf{i})\mathsf{S}_{R_\omega} \equiv_{\mathrm{sW}} (\mathsf{i})\mathsf{S}_{R_\omega \odot C_n}$ for some $n > 2$ and left open the following question:

$$\text{is there a computable graph } G \text{ such that } \mathsf{LPO} <_{\mathrm{W}} \mathsf{IS}_G <_{\mathrm{W}} \mathsf{WF}? \qquad (3)$$

By Theorem *III*.3.18 we know that if such a G exists, it must be infinite.

Lemma III.3.19. *Let G be a c.e. graph such that $R_\omega \subseteq_{\mathbf{s}} G$. Then,* $\mathsf{WF} \equiv_{\mathrm{sW}} (e)\mathsf{S}_G \equiv_{\mathrm{sW}} (e)\mathsf{IS}_G$.

Proof. It follows immediately from Theorem *III*.3.8 and Proposition *III*.3.17. □

Lemma III.3.20. *Let G be a c.e. graph such that $K_\omega \nsubseteq_{\mathbf{is}} G$. Then* $\mathsf{WF} \equiv_{\mathrm{sW}} (e)\mathsf{IS}_G$.

Proof. It follows immediately from Theorem *III*.3.7 and Proposition *III*.3.17. □

We finally obtain a negative answer to question (3) showing that the conjecture does not hold for even a larger class of graphs and problems.

Theorem III.3.21. *There is no c.e. graph G such that* $\mathsf{LPO} <_{\mathrm{W}} (e)\mathsf{IS}_G <_{\mathrm{W}} \mathsf{WF}$.

Proof. If G is finite then $\mathsf{IS}_G \equiv_{\mathrm{sW}} \mathsf{LPO}$ (Theorem *III*.3.18). If G is an infinite graph then either $K_\omega \nsubseteq_{\mathbf{is}} G$ or $K_\omega \subseteq_{\mathbf{is}} G$ (and hence $R_\omega \subseteq_{\mathbf{s}} G$): by Proposition *III*.3.17, in both cases we obtain $\mathsf{IS}_G \equiv_{\mathrm{sW}} \mathsf{WF}$. □

As mentioned in the introduction of Part 1, a main theme of this part is to identify representatives of $\mathbf{\Pi}^1_1\text{–}\mathsf{CA}_0$ in the Weihrauch lattice: the next corollary gives us some "graph-theoretic" ones.

Corollary III.3.22. *Let G and H be infinite c.e. graphs such that $R_\omega \subseteq_{\mathbf{s}} G$ and $K_\omega \nsubseteq_{\mathbf{is}} H$. Then* $\widehat{\mathsf{WF}} \equiv_{\mathrm{sW}} \widehat{(e)\mathsf{S}_G} \equiv_{\mathrm{sW}} \widehat{(e)\mathsf{IS}_G} \equiv_{\mathrm{sW}} \widehat{(e)\mathsf{IS}_H}$.

The following theorem shows that the situation for S_G is quite different.

Theorem III.3.23. *For every $k \in \mathbb{N}$,* $(e)\mathsf{S}_{\mathsf{T}_{2k+1}} \equiv_{\mathrm{sW}} \mathsf{LPO}^{(2k)}$ *and* $(e)\mathsf{S}_{\mathsf{F}_{2k+2}} \equiv_{\mathrm{sW}} \mathsf{LPO}^{(2k+1)}$

Proof. It follows from Theorem *III*.3.10 and the fact that $\mathsf{LPO}^{(n)}$ answers a binary Σ^0_{n+1}-complete or Π^0_{n+1}-complete question (see §I.6). □

We do not know whether there exists a graph G such that $\mathsf{LPO}^{(n)} <_{\mathrm{W}} \mathsf{S}_G <_{\mathrm{W}} \mathsf{LPO}^{(n+1)}$: this question is essentially the same question left open at the end of §III.3 in terms of Weihrauch reducibility.

III.3.2 The "opposite" problem

In §III.3 we considered the (effective) Wadge complexity of sets of the form $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\}$. Now we consider sets having a sort of "opposite" definition, namely,

$$\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{(i)s}} G\} := \{p \in \mathrm{dom}(\delta_{(E)Gr}) : \delta_{(E)Gr}(p) \subseteq_{\mathbf{(i)s}} G\}.$$

We show that while in §III.3 we have many natural examples of graphs G such that the sets $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\}$ are Σ^1_1-complete, the "opposite" sets, for the same G, have a considerably lower complexity: the best we achieve is showing that there exists a graph G such that the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{is}} G\}$ are Π^0_5-hard, and we leave open whether there exists a graph G such that the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{is}} G\}$ are Γ-hard (or Γ-complete) for more complex Γ.

As in §III.3, we start from finite graphs. Notice that the analogue of Proposition $III.3.1$ holds also for this kind of sets: namely $\{H \in \mathbf{Gr} : H \subseteq_{\mathbf{(i)s}} G\} \leqslant_{\mathsf{EW}} \{H \in \mathbf{EGr} : H \subseteq_{\mathbf{(i)s}} G\}$.

Proposition III.3.24. *For any finite graph G, the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{s}} G\}$ and $\{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} G\}$ are Π^0_1-complete*

Proof. To show that the first two sets are Π^0_1, by the analogue of Proposition $III.3.1$, it suffices to show that, given a name q for $H \in \mathbf{Gr}$, since G is finite, $H \subseteq_{\mathbf{s}} G \iff (\forall s)(\delta_{Gr}(q[s]0^{\mathbb{N}}) \subseteq_{\mathbf{s}} G)$.

To show that the third set is Π^0_1 it suffices to notice that given a name q for $H \in \mathbf{Gr}$, $H \subseteq_{\mathbf{is}} G \iff (\forall s)(H[s] \subseteq_{\mathbf{is}} G)$, where $H[s]$ is the graph having $\{i < s : q(\langle i,i\rangle) = 1\}$ as vertex set and $\{i,j < s : q(\langle i,j\rangle) = 1\}$.

We only give the proof that the third set is complete for the class Π^0_1, as the same proof shows also the completeness for the first two sets. By the analogue of Proposition $III.3.1$ and Theorem $I.3.36$, it suffices to show that $\{p : (\forall i)(p(i) = 0)\} \leqslant_{\mathsf{EW}} \{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} G\}$. Let $G \odot v$ be the graph consisting of G and a disconnected vertex. Then, given $p \in 2^{\mathbb{N}}$, let H be such that

$$H \cong \begin{cases} G \odot v & \text{if } (\exists i)(p(i) = 1), \\ \varnothing & \text{otherwise.} \end{cases}$$

The graph H it is easy to compute: just add in H a copy of $G \odot v$ if a 1 appears in p, otherwise keep computing a name for the empty graph. Since $\varnothing \subseteq_{\mathbf{is}} G$ while $G \odot v \nsubseteq_{\mathbf{s}} G$, it is clear that $H \subseteq_{\mathbf{is}} G \iff (\forall i)(p(i) = 0)$. The same proof works also if we consider the subgraph relation. □

The proof of the proposition above can be easily adapted to prove the following result.

Proposition III.3.25. *For any finite graph G, the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{s}} \overset{\infty}{\bigotimes} G\}$ and $\{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} \overset{\infty}{\bigotimes} G\}$ are Π^0_1-complete*

We leave open whether $\{H \in \mathbf{EGr} : H \subseteq_{\mathbf{is}} G\}$ and $\{H \in \mathbf{EGr} : H \subseteq_{\mathbf{is}} \overset{\infty}{\bigotimes} G\}$ are complete for some class Γ, but we conjecture that they are complete for some class in the *lightface difference hierarchy* (not introduced in this thesis, see e.g. [Sac17, §I.2.3] for its definition).

The following proposition shows a big difference with sets of the form $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\}$. Indeed, for any c.e. graph G such that $R_\omega \subseteq_{\mathbf{s}} G$, the sets $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{(i)s}} H\}$ are Σ^1_1-complete (Theorem $III.3.8$). Clearly $R_\omega \subseteq_{\mathbf{s}} K_\omega$, but the following proposition shows that the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{(i)s}} K_\omega\}$ have a significantly lower complexity.

Proposition III.3.26. *The sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{s}} K_\omega\}$ are computable, $\{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} K_\omega\}$ is Π^0_1-complete and $\{H \in \mathbf{EGr} : H \subseteq_{\mathbf{is}} K_\omega\}$ is Π^0_2-complete.*

Proof. The fact that the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{s}} K_\omega\}$ are computable follows from the fact that any graph is a subgraph of K_ω.

To show that $\{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} K_\omega\}$ is Π^0_1 notice that given a name p for $H \in \mathbf{Gr}$, $H \subseteq_{\mathbf{is}} K_\omega \iff (\forall v \neq w)(p(\langle v, v\rangle) = p(\langle w, w\rangle) = 1 \implies p(\langle v, w\rangle) = 1)$. To prove completeness, we show $\{p : (\forall i)(p(i) = 0)\} \leqslant_{\mathsf{EW}} \{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} K_\omega\}$ (the left-hand-side is Π^0_1-complete by Theorem *I*.3.36). To do so, given $p \in 2^{\mathbb{N}}$, and $v \neq w$ it is immediate that we can compute H as

$$H \cong \begin{cases} (\{v, w\}, \varnothing) & \text{if } (\exists i)(p(i) = 1), \\ \varnothing & \text{otherwise.} \end{cases}$$

Clearly, $H \subseteq_{\mathbf{is}} K_\omega \iff (\forall i)(p(i) = 0)$ and this proves the reduction.

To show that $\{H \in \mathbf{EGr} : H \subseteq_{\mathbf{is}} K_\omega\}$ is Π^0_2, notice that given a name p for $H \in \mathbf{EGr}$, $H \subseteq_{\mathbf{is}} K_\omega \iff (\forall v \neq w)(\exists k_0, k_1, k_2)(k_2 > k_0, k_1 \wedge p(k_0) = \langle v, v\rangle \wedge p(k_1) = \langle w, w\rangle \implies p(k_2) = \langle v, w\rangle)$. To prove completeness, we show that $\{p : (\exists^\infty i)(p(i) = 1)\} \leqslant_{\mathsf{EW}} \{H \in \mathbf{EGr} : H \subseteq_{\mathbf{is}} K_\omega\}$ (the left-hand-side is Π^0_2-complete by Theorem *I*.3.36). To do so, we compute a δ_{EGr}-name q as follows. At stage 0 do nothing and at stage $s + 1$.

- if $p(s + 1) = 0$, then enumerate in q an isolated vertex to the graph computed up to stage s;
- if $p(s + 1) = 1$, then enumerate in q enough edges in the graph computed up to stage s to make it isomorphic to a complete finite graph.

It is clear that if $(\exists^\infty i)(p(i) = 1)$, then $\delta_{EGr}(q) \cong K_\omega$ (and clearly K_ω is an induced subgraph of itself). Otherwise, $\delta_{EGr}(q)$ contains cofinitely many isolated vertices and hence $\delta_{EGr}(q) \nsubseteq_{\mathbf{is}} K_\omega$. □

Theorem III.3.27. *The sets* $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{(\mathbf{i})\mathbf{s}} \bigotimes_{i \geqslant 1} R_i\}$ *and* $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{(\mathbf{i})\mathbf{s}} \bigotimes_{i \geqslant 1} K_i\}$ *are* Π^0_3*-complete.*

Proof. We first prove the sets above are Π^0_3. It is easy to check that $H \subseteq_{\mathbf{s}} \bigotimes_{i \geqslant 1} R_i \iff H \subseteq_{\mathbf{is}} \bigotimes_{i \geqslant 1} R_i$. Then $H \subseteq_{\mathbf{is}} \bigotimes_{i \geqslant 1} R_i$ if and only if the following hold:

- $(\forall v \in V(H))(\deg^H(v) \leqslant 2 \wedge v \not\leftrightsquigarrow^H v)$, i.e. every node in H has degree at most two and H contains no cyclic graph, and
- $(\forall v \in V(H))(\exists k)(\forall w \in V(H))(v \leftrightsquigarrow^H w \implies w < k)$, i.e. every connected component of H is finite.

First notice that, by Lemma *III*.3.4, regardless of whether $H \in \mathbf{EGr}$ or $H \in \mathbf{Gr}$, the first formula is Π^0_1, while the second one is Π^0_3.

The left-to-right direction of the "if and only if" is trivial. For the opposite direction, suppose that H satisfies the right-hand-side formulas, and notice that $H = \bigotimes_{n \in \mathbb{N}} D_n$ where either $D_n \cong R_m$ for some m or D_n is an isolated vertex. Then, the map $f : \mathbb{N} \to \mathbb{N}$ such that $f(0) := \min\{i : D_0 \subseteq_{\mathbf{is}} R_i\}$ and, for $n > 0$, $f(n) := \min\{i > f(n-1) : D_n \subseteq_{\mathbf{is}} R_i\}$ witnesses that $H \subseteq_{\mathbf{is}} \bigotimes_{i \geqslant 1} R_i$.

For the other case, we have that $H \subseteq_{\mathbf{s}} \bigotimes_{i \geqslant 1} K_i$ if and only if every connected component of H is finite, while $H \subseteq_{\mathbf{is}} \bigotimes_{i \geqslant 1} K_i$ if and only if $H \subseteq_{\mathbf{s}} \bigotimes_{i \geqslant 1} K_i$ and

- $(\forall v, w \in V(H))(v \leftrightsquigarrow^H w \implies (v, w) \in E(H))$.

We have already discussed that for $H \in (\mathbf{E})\mathbf{Gr}$ the formula "every connected component of H is finite" is Π^0_3; then, if $H \in \mathbf{EGr}$, by Lemma *III*.3.4, the formula in the item above is Π^0_2, otherwise, if $H \in \mathbf{Gr}$, it is Π^0_1. We skip the proofs of the equivalences as, once one notices that any finite graph is a subgraph of K_n for some $n \in \mathbb{N}$ and that $K_n \subseteq_{\mathbf{is}} K_{n+1}$, they are similar to the one given for $\bigotimes_{i \geqslant 1} R_i$.

For completeness, by Theorem $I.3.36$ we know that $\{p \in 2^{\mathbb{N}\times\mathbb{N}} : (\forall n)(\forall^\infty i)(p(m,i) = 1)\}$ is Π^0_3-complete, and hence we show that $\{p \in 2^{\mathbb{N}\times\mathbb{N}} : (\forall n)(\forall^\infty i)(p(n,i) = 1)\} \leqslant_{\mathsf{EW}} \{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} \bigotimes_{i\geqslant 1} R_i\}$. For every n we compute a name q_n for a graph H_n as follows. At stage 0, let $q_n(\langle\langle n,0\rangle,\langle n,0\rangle\rangle) = 1$. At stage $s+1$, let $k_0 := \max\{i \leqslant s : q_n(\langle\langle n,i\rangle,\langle n,i\rangle\rangle) = 1\}$:

- if $p(n,s+1) = 0$, let $q_n(\langle\langle n,s+1\rangle,\langle n,s+1\rangle\rangle) = 1$, and $q_n(\langle\langle n,k_0\rangle,\langle n,s+1\rangle\rangle) = 1$;
- if $p(n,s+1) = 1$ let $q_n(\langle\langle n,s+1\rangle,\langle n,s+1\rangle\rangle) = 0$, and $q_n(\langle\langle n,k_0\rangle,\langle n,s+1\rangle\rangle) = 0$.

Informally, at any stage, $H_n \cong R_m$ for some $m > 0$: in the first item we are extending such a ray, while in the second one we are leaving it as it is. Let $H := \bigotimes_{n\in\mathbb{N}} H_n$ and observe that if $(\exists n)(\exists^\infty i)(p(n,i) = 0)$, then $H_n \cong R_\omega$, and hence $H \nsubseteq_{\mathbf{is}} \bigotimes_{i\geqslant 1} R_i$. Otherwise, if $(\forall n)(\forall^\infty i)(p(n,i) = 1)$, then for every n, $H_n \cong R_m$ for some $m > 0$ and hence $H \subseteq_{\mathbf{is}} \bigotimes_{i\geqslant 1} R_i$ and hence the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{(i)s}} \bigotimes_{i\geqslant 1} R_i\}$ are Π^0_3-complete.

The same reduction described above also shows that the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{s}} \bigotimes_{i\geqslant 1} K_i\}$ are Π^0_3-complete. To show that $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{is}} \bigotimes_{i\geqslant 1} K_i\}$ are Π^0_3-complete it suffices to slightly modify the construction, so that the H_n's are not rays but complete graphs. This concludes the proof. □

We conclude the results about the effective Wadge complexity of subsets of (names of) graphs discussing $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{(i)s}} \mathcal{S}\}$, where $\mathcal{S}$ is defined as follows. For any $i \in \mathbb{N}$, let $S^i := \{j^n : n \in \mathbb{N} \wedge j < i+1\} \cup \{m^n : m \geqslant i+1 \wedge n < m-i\}$, i.e. a tree where $|[S^i]| = i+1$ and such that, for every m, it has a finite path of length m; the only common initial segment of paths in S^i, of finite or infinite length, is $\langle\rangle$. Finally, let $\mathcal{S} := \bigotimes_{i\in\mathbb{N}} S^i$.

Proposition III.3.28. *Let $\mathcal{S}$ as above: then the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{(i)s}} \mathcal{S}\}$ are Π^0_5-hard.*

Proof. We prove the proposition only for $\{H \in \mathbf{Gr} : H \subseteq_{\mathbf{is}} \mathcal{S}\}$, but the same proof works for the rest of the cases.

Consider the set $\{(p \in 2^{\mathbb{N}\times\mathbb{N}\times\mathbb{N}}) : (\forall i)(\forall^\infty j)(\forall^\infty n)(p(i,j,n) = 0)\}$ and notice that by Theorem $I.3.36$ such a set is Π^0_5-complete. For any $i \in \mathbb{N}$, we define T^i as

$$\{\langle\rangle\} \cup \{\langle j\rangle^\frown\sigma : j \in \mathbb{N} \wedge \sigma \in \mathbb{N}^{<\mathbb{N}} \wedge (\forall m < |\sigma|)(p(i,j,\sigma(m)) = 1) \wedge (\forall k < |\sigma|-1)(\sigma(k) < \sigma(k+1))\},$$

and let $T := \bigotimes_{i\in\mathbb{N}} T^i$. Notice that for every i, the only common segment of (in)finite sequences in T^i is $\langle\rangle$. It is easy to check that $|[T^i]| = |\{j : (\exists^\infty n)(p(i,j,n) = 1)\}|$.

- if $(\exists i)(\exists^\infty j)(\exists^\infty n)(p(i,j,n) = 1)$. Notice that $|[T^i]| = \aleph_0$: since there exists no i such that $|[S^i]| = \aleph_0$ this immediately implies that $T \nsubseteq_{\mathbf{is}} S$.
- Otherwise, $(\forall i)(\forall^\infty j)(\forall^\infty n)(p(i,j,n) = 0)$. It is immediate that, for every i, $|[T^i]| = |\{j : (\exists^\infty n)(p(i,j,n) = 1)\}| = k$ for some $k \in \mathbb{N}$. Notice that $T^i \subseteq_{\mathbf{is}} S^k$. Indeed, the only common initial segment of (in)finite sequences in T^i is $\langle\rangle$ and $|[T^i]| = |[S^k]|$. Furthermore, since for every $m \in \mathbb{N}$ there exists some $n \in \mathbb{N}$ such that $m^n \in S^k$, every $\sigma \in T^i$ such that $(\forall t)(\sigma^\frown t \notin T^i)$ can be mapped to some m^n where $n > |\sigma|$. We now define a function $f : \mathbb{N} \to \mathbb{N}$ witnessing that $T \subseteq_{\mathbf{is}} \mathcal{S}$. Let $f(0) = k_0$ where $k_0 = |[T^0]|$ and at stage $s+1$ let $f(s+1) = k_{s+1}$ where $k_{s+1} := \min\{k \geqslant k_s : |[T^{s+1}]| \leqslant k\}$. Clearly, for any s, k_s exists as at stage s we have only defined $f(0), \dots f(s-1)$ and $(\forall k > |[T^i]|)(T^i \subseteq_{\mathbf{is}} S^k)$. Finally, $(\forall i)(T^i \subseteq_{\mathbf{is}} S^{f(i)})$ and hence $T \subseteq_{\mathbf{is}} \mathcal{S}$. This concludes the proof. □

Now we move to Weihrauch reducibility and consider the decision problems that are "opposite" to the ones defined in Definition $III.3.16$

Definition III.3.29. For a computable graph G, we define the functions $\mathsf{IS}^G : \mathbf{Gr} \to 2$ and $\mathsf{S}^G : \mathbf{Gr} \to 2$ by

$$\mathsf{IS}^G(H) = 1 \iff H \subseteq_{\mathbf{is}} G \text{ and } \mathsf{S}^G(H) = 1 \iff H \subseteq_{\mathbf{s}} G.$$

The same functions having domain **EGr** are denoted respectively with $e\mathsf{IS}^G$ and $e\mathsf{S}^G$.

In the next proofs, recall from I.6 that $\mathsf{LPO}^{(n)}$ answers a Σ^0_{n+1} or Π^0_{n+1} question relative to the input.

Proposition III.3.30. *For any finite graph G, $(e)\mathsf{S}^G \equiv_{\mathrm{sW}} (e)\mathsf{S}^{\overset{\infty}{\bigotimes} G} \equiv_{\mathrm{sW}} \mathsf{LPO}$.*

Proof. All the results follow from Propositions III.3.24 and III.3.25. □

Proposition III.3.31. *The problems $(e)\mathsf{S}^{K_\omega}$ are the constant functions taking value 1 and hence are computable. Instead, $\mathsf{IS}^{K_\omega} \equiv_{\mathrm{sW}} \mathsf{LPO}$ and $e\mathsf{IS}^{K_\omega} \equiv_{\mathrm{sW}} \mathsf{LPO}'$.*

Proof. The fact that $\mathsf{IS}^{K_\omega} \equiv_{\mathrm{sW}} \mathsf{LPO}$ was already mentioned in [BHW21], but all the results follow from Proposition III.3.26. □

Notice that in [BHW21], the authors left open whether there is a graph G such that $\mathsf{LPO} <_{\mathrm{W}} \mathsf{IS}^G$. The following proposition gives a positive answer to this question.

Proposition III.3.32. $(e)\mathsf{S}^{\bigotimes_{i\geqslant 1} R_i} \equiv_{\mathrm{sW}} (e)\mathsf{IS}^{\bigotimes_{i\geqslant 1} R_i} \equiv_{\mathrm{sW}} (e)\mathsf{S}^{\bigotimes_{i\geqslant 1} K_i} \equiv_{\mathrm{sW}} (e)\mathsf{IS}^{\bigotimes_{i\geqslant 1} K_i} \equiv_{\mathrm{sW}} \mathsf{LPO}''$.

Proof. All the results follow from Theorem III.3.27. □

It is open whether there exists a graph G such that $\mathsf{IS}^G \equiv_{\mathrm{sW}} \mathsf{WF}$: indeed, $\mathsf{IS}^{\mathcal{S}}$ is the strongest problem of this form we have found. Again the answer to this question is strictly related to the effective Wadge complexity of sets of (names) of graphs considered in this section.

Proposition III.3.33. $\mathsf{LPO}^{(4)} \leqslant_{\mathrm{sW}} (e)\mathsf{S}^{\mathcal{S}}, (e)\mathsf{IS}^{\mathcal{S}}$.

Proof. It follows immediately from Proposition III.3.28. □

III.4 Searching the (induced) subgraph

In this section, we focus entirely on Weihrauch reducibility: notice that some results are still consequences of §III.3, but most of them require new proof techniques.

Definition III.4.1. Given a graph G, we define four multi-valued functions:

$$\mathsf{IS\text{-}Copy}^{\chi\chi}_G :\subseteq \mathbf{Gr} \rightrightarrows \mathbf{Gr},\ \mathsf{IS\text{-}Copy}^{\chi e}_G :\subseteq \mathbf{Gr} \rightrightarrows \mathbf{EGr},$$

$$\mathsf{IS\text{-}Copy}^{ee}_G :\subseteq \mathbf{EGr} \rightrightarrows \mathbf{EGr},\ \mathsf{IS\text{-}Copy}^{e\chi}_G :\subseteq \mathbf{EGr} \rightrightarrows \mathbf{Gr},$$

with domains $\{H : G \subseteq_{\mathbf{is}} H\}$ and ranges $\{G' : G' \cong G \land G' \text{ is an induced subgraph of } H\}$. The corresponding functions for the subgraph case are denoted replacing $\mathsf{IS\text{-}Copy}$ with $\mathsf{S\text{-}Copy}$ and have domains $\{H : G \subseteq_{\mathbf{s}} H\}$ and ranges $\{G' : G' \cong G \wedge G' \text{ is a subgraph of } H\}$.

Informally, in the definition above, the χ (respectively e) at the first (second) position indicates that the domain (range) of the corresponding function is a subset of **Gr** (**EGr**). We adopt the following convention: whenever we write $\mathsf{IS\text{-}Copy}_G$ we are referring to all four functions in Definition III.4.1, similarly for $\mathsf{S\text{-}Copy}_G$.

The following relations hold, also replacing $\mathsf{IS\text{-}Copy}$ with $\mathsf{S\text{-}Copy}$.

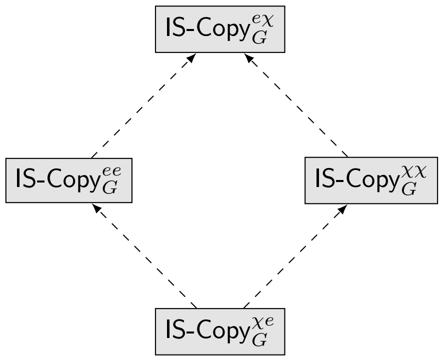

Figure III.3: Dashed arrows represent Weihrauch reducibility in the direction of the arrow, leaving open whether the reduction is strict.

The next proposition shows that the main functions we consider in this section are cylinders, which implies that most reductions we obtain in this section are actually strong ones.

Proposition III.4.2. *For any infinite graph G, $\mathsf{S\text{-}Copy}_G$ and $\mathsf{IS\text{-}Copy}_G$ are cylinders*

Proof. We only prove the statement for $\mathsf{IS\text{-}Copy}_G$ as the same proof shows that $\mathsf{S\text{-}Copy}_G$ are cylinders as well. Let $p \in \mathbb{N}^\mathbb{N}$ and let $H \in \operatorname{dom}(\mathsf{S\text{-}Copy}_G)$: we compute the graph H' such that $V(H') := \{\langle v, p[v]\rangle : v \in V(H)\}$ and $E(H') := \{(\langle v, p[v]\rangle, \langle w, p[w]\rangle) : (v, w) \in E(H)\}$. Clearly, $H' \cong H$ and the isomorphism between the two graphs is computable. From $G' \in \mathsf{IS\text{-}Copy}_G(H')$ compute the graph S where $V(S) = \{v : \langle v, p[v]\rangle \in V(G')\}$ and $E(S) = \{(v, w) : (\langle v, p[v]\rangle, \langle w, p[w]\rangle) \in E(G')\}$. Clearly $S \in \mathsf{IS\text{-}Copy}_G(H)$, and from $\{p[v] : \langle v, p[v]\rangle \in V(S)\}$ (that by hypothesis is infinite) we can recover longer and longer prefixes of p. This concludes the proof □

The following proposition, together with Figure III.3, shows that $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ is an upper bound for all of them (notice that $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ is a cylinder as well).

Proposition III.4.3. *For any infinite hyperarithmetical graph G, $\mathsf{IS\text{-}Copy}_G, \mathsf{S\text{-}Copy}_G \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$.*

Proof. By Figure III.3, it suffices to show that $\mathsf{IS\text{-}Copy}_G^{e\chi}, \mathsf{S\text{-}Copy}_G^{e\chi} \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$. We only show that $\mathsf{IS\text{-}Copy}_G^{e\chi} \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ as the same proof works also for showing that $\mathsf{S\text{-}Copy}_G^{\chi e} \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$. Since $\boldsymbol{\Sigma}_1^1\text{-}\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ (see §I.6) we show that $\mathsf{IS\text{-}Copy}_G^{e\chi} \leqslant_{\mathrm{W}} \boldsymbol{\Sigma}_1^1\text{-}\mathsf{C}_{\mathbb{N}^\mathbb{N}}$. Given in input H, by Proposition III.3.5, notice that $\{G' \in \mathbf{EGr} : G' \cong G \wedge G' \text{ is an induced subgraph of } H\}$ is a nonempty $\Sigma_1^{1,H}$ subset of $\mathbb{N}^\mathbb{N}$ and hence a suitable input for $\boldsymbol{\Sigma}_1^1\text{-}\mathsf{C}_{\mathbb{N}^\mathbb{N}}$: clearly any $G \in \boldsymbol{\Sigma}_1^1\text{-}\mathsf{C}_{\mathbb{N}^\mathbb{N}}(H)$ is such that $G \in \mathsf{IS\text{-}Copy}_G^{e\chi}(H)$. □

We first consider the case of G being finite, and notice that the case distinction, in some sense, is close to the one in Proposition III.3.5.

Theorem III.4.4. *For any finite graph G the following holds:*

(i) the problems $\mathsf{S\text{-}Copy}_G$ *are computable;*

(ii) $\mathsf{IS\text{-}Copy}_G^{\chi e}$ *and* $\mathsf{IS\text{-}Copy}_G^{\chi\chi}$ *are computable;*

(iii) $\mathsf{IS\text{-}Copy}_{K_n}^{ee}$ *and* $\mathsf{IS\text{-}Copy}_{K_n}^{e\chi}$ *are computable;*

(iv) if $G \not\cong K_n$, $\mathsf{IS\text{-}Copy}_G^{ee} \equiv_{\mathrm{sW}} \mathsf{IS\text{-}Copy}_G^{e\chi} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathbb{N}}$.

Proof. To prove (i), by Figure III.3 it suffices to show that $\mathsf{S\text{-}Copy}_G^{e\chi}$ is computable. Given a δ_{EGr}-name h for an input of $\mathsf{S\text{-}Copy}_G^{e\chi}$ we compute a δ_{Gr}-name p for a solution of $\mathsf{S\text{-}Copy}_G^{e\chi}$ as follows. At any stage s, we check whether there exists a subgraph isomorphic to G in the finite graph determined by $h[s]$. Formally, for every s, we check whether there exists an injective $f : V(G) \to V(\delta_{EGr}(h[s]0^{\mathbb{N}}))$ such that

- $i \in V(G) \implies (\exists k_i < s)(h(k_i) = \langle f(i), f(i)\rangle)$ and
- $(i,j) \in E(G) \implies (\exists \ell_i < s)(h(\ell_i) = \langle f(i), f(j)\rangle)$.

This can be done computably as both $\delta_{EGr}(h[s]0^{\mathbb{N}})$ and G are finite and hence, at any stage, there are only finitely many f's to check. If $G \nsubseteq_{\mathbf{s}} \delta_{EGr}(h[s]0^{\mathbb{N}})$ do nothing. Otherwise, let $f_s : V(G) \to V(\delta_{EGr}(h[s]0^{\mathbb{N}}))$ be a function witnessing $G \subseteq_{\mathbf{s}} \delta_{EGr}(h[s]0^{\mathbb{N}})$ and let p be such that

$$p(\langle i,j\rangle) := \begin{cases} 1 & \text{if } (i = j \wedge f_s^{-1}(i) \in V(G)) \vee ((f_s^{-1}(i), f^{-1}(j)) \in E(G)), \\ 0 & \text{otherwise.} \end{cases}$$

It is straightforward to check that p is a δ_{Gr}-name for a solution of $\mathsf{S\text{-}Copy}_G^{e\chi}$.

For (ii), by Figure III.3, it suffices to prove that $\mathsf{IS\text{-}Copy}_G^{\chi\chi}$ is computable. The proof is the same of (i), except for the fact that the injective f is such that $i \in V(G)$ implies $h(\langle f(i), f(i)\rangle) = 1$ and $(i,j) \in E(G)$ if and only if $h(\langle f(i), f(j)\rangle) = 1$.

The claim in (iii) follows from (i) as, for any graph H, $K_n \subseteq_{\mathbf{s}} H$ if and only if $K_n \subseteq_{\mathbf{is}} H$.

To prove (iv), by Figure III.3, it suffices to show that $\mathsf{IS\text{-}Copy}_G^{e\chi} \leqslant_{\mathrm{sW}} \mathsf{C}_{\mathbb{N}}$ and $\mathsf{C}_{\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{IS\text{-}Copy}_G^{ee}$. For the first reduction, given a name p for an input H of $\mathsf{IS\text{-}Copy}_G^{e\chi}$, let A be the set of $(\sigma,\tau) \in (\mathbb{N} \times \mathbb{N})^{V(G)}$ such that $(\forall i,j \in V(G))(\forall z \in \mathbb{N})$

- $p(\langle \tau(i), \tau(i)\rangle) = \langle \sigma(i), \sigma(i)\rangle$;
- $(i,j) \in E(G) \implies p(\langle \tau(i), \tau(j)\rangle) = \langle \sigma(i), \sigma(j)\rangle$ and
- $(i,j) \notin E(G) \implies p(z) \neq \langle \sigma(i), \sigma(j)\rangle$.

Clearly, A is nonempty, and since any (σ,τ) can be coded as a natural number, A is a suitable input for $\mathsf{C}_{\mathbb{N}}$. Let $n \in \mathsf{C}_{\mathbb{N}}(A)$ be the code of some (σ,τ). We compute a name q for a copy of $G' \in \mathsf{IS\text{-}Copy}_G^{e\chi}(H)$ letting $q(\langle n,m\rangle) = 1$ if and only if $(\exists i,j < |\sigma|)(n = \sigma(i) \wedge m = \sigma(j))$.

To prove that $\mathsf{C}_{\mathbb{N}} \leqslant_{\mathrm{sW}} \mathsf{IS\text{-}Copy}_G^{ee}$, let $A \in \mathcal{A}(\mathbb{N})$ be an input of $\mathsf{C}_{\mathbb{N}}$. We denote by $A^c[s]$ the enumeration of the complement of A up to stage s and notice that, since by hypothesis, $G \not\cong K_n$, we have that $V(G)^2 \backslash E(G) \neq \emptyset$. We compute an input H for $\mathsf{IS\text{-}Copy}_G^{ee}$ as follows. At stage 0, for every $v \in V(G)$ and for every $(v,w) \in E(G)$, enumerate $\langle v, 0\rangle$ in $V(H)$ and $(\langle v,0\rangle, \langle w,0\rangle)$ in $E(H)$ and let $n_0 := 0$. At stage $s+1$, let $n_{s+1} := \min\{n : n \notin A^c[s+1]\}$. If $n_{s+1} = n_s$, do nothing, otherwise, for every $v,w \in V(H)$ at stage $s+1$, enumerate (v,w) in $E(H)$, i.e. modify H so that is isomorphic to a complete graph. Then for every $v \in V(G)$ and for every $(v,w) \in E(G)$, enumerate $\langle v, n_{s+1}\rangle$ in $V(H)$ and $(\langle v, n_{s+1}\rangle, \langle w, n_{s+1}\rangle)$ in $E(H)$. In the limit, we obtain that either H is such that $V(H) = \{\langle v,0\rangle : v \in V(G)\}$ and $E(H) = \{(\langle v,0\rangle, \langle w,0\rangle) : (v,w) \in E(G)\}$ (in case $0 \in A$), or $H \cong K_m \bigotimes G'$ for some $m \in \mathbb{N}$ and $V(G') = \{\langle v, n_s\rangle : v \in V(G)\}$ and $E(G') = \{(\langle v,n_s\rangle, \langle w,n_s\rangle) : (v,w) \in E(G)\}$ where $n_s = \min\{n : n \in A\}$. Since $G \not\cong K_n$, we know that any $H' \in \mathsf{IS\text{-}Copy}_G^{ee}(H)$ is not contained in the copy of K_m in H and hence, given $\langle v, n_s \rangle \in V(H')$, it is easy to check that $n_s \in A$ and this concludes the proof. □

III.4.1 The induced subgraph problem for infinite graphs

We now move our attention to the case when G is infinite. The results are summarized in the next two theorems: the first concerns the problems $\mathsf{IS\text{-}Copy}_G$, while the second the problems $\mathsf{S\text{-}Copy}_G$. The first theorem analyzes $\mathsf{IS\text{-}Copy}_G$ for all infinite G's, while the second one does not include certain graphs some of which are discussed in §III.4.2. We highlight that some results are stated for c.e. graphs, others just for computable graphs.

Theorem III.4.5. *For any infinite graph G, $\mathsf{IS\text{-}Copy}_G \equiv_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ relative to some oracle. In particular, we have the following cases:*

(i) if $|\{v \in V(G) : \deg^G(v) < \aleph_0\}| < \aleph_0$ we have two cases:

- *(a) if G is computable then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{IS\text{-}Copy}_G$;*
- *(b) if G is c.e., then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{IS\text{-}Copy}_G^{ee} \equiv_W \mathsf{IS\text{-}Copy}_G^{e\chi}$.*

(ii) If G is c.e. and $|\{v \in V(G) : \deg^G(v) = \aleph_0\}| < \aleph_0$, then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{IS\text{-}Copy}_G$.

(iii) Let $\lambda : \mathbb{N} \to \mathbb{N}$ be such that $\lambda(n) := \min\{k : |v \in V(G) : \deg^G(v) \leqslant k\}| \geqslant k$. If $\{v \in V(G) : \deg^G(v) < \aleph_0\} = \aleph_0$ and $\{v \in V(G) : \deg^G(v) = \aleph_0\} = \aleph_0$ we have two cases:

- *(a) if G is computable then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{IS\text{-}Copy}_G$ relative to λ;*
- *(b) if G is c.e., then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{IS\text{-}Copy}_G^{ee} \equiv_W \mathsf{IS\text{-}Copy}_G^{e\chi}$ relative to λ.*

Proposition III.4.6. *Let G be an infinite graph such that $|\{v \in V(G) : \deg^G(v) < \aleph_0\}| < \aleph_0$. If G is computable then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{S\text{-}Copy}_G$ while if G is c.e., then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{S\text{-}Copy}_G^{ee} \equiv_W \mathsf{S\text{-}Copy}_G^{e\chi}$.*

The next part of this subsection is devoted to prove Theorem *III*.4.5 and Proposition *III*.4.6: Proposition *III*.4.3 implies all the reductions from $\mathsf{IS\text{-}Copy}_G$ to $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$: Figure III.3 and the remaining lemmas of this subsection prove the converse directions.

The following lemma proves Theorem *III*.4.5(*i*) and Proposition *III*.4.6.

Lemma III.4.7. *Let G be an infinite graph such that $|\{v \in V(G) : \deg^G(v) < \aleph_0\}| < \aleph_0$. If G is computable then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{S\text{-}Copy}_G, \mathsf{IS\text{-}Copy}_G$. If G is c.e. then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{IS\text{-}Copy}_G^{ee}, \mathsf{S\text{-}Copy}_G^{ee}$.*

Proof. For the first part, by Figure III.3, it suffices to show that $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{S\text{-}Copy}_G^{\chi e}, \mathsf{IS\text{-}Copy}_G^{\chi e}$. We only prove the statement for $\mathsf{S\text{-}Copy}_G^{\chi e}$ as the proof for $\mathsf{IS\text{-}Copy}_G^{\chi e}$ is the same. Let $T \in$ **Tr** be an input for $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$, let $V(G) = \{v_i : i \in \mathbb{N}\}$ and compute $\mathcal{A}^{\sqsubseteq}(T, G)$ (see §III.3 for its definition). Notice that, since G is computable, $\mathcal{A}^{\sqsubseteq}(T, G)$ is computable with respect to T and by Proposition *III*.3.2, $G \subseteq_{\mathbf{s}} \mathcal{A}^{\sqsubseteq}(T, G)$: hence $\mathcal{A}^{\sqsubseteq}(T, G)$ is a suitable input for $\mathsf{S\text{-}Copy}_G^{\chi e}$. Let $G' \in \mathsf{S\text{-}Copy}_G^{\chi e}(\mathcal{A}^{\sqsubseteq}(T, G))$. We claim that

$$(\forall \sigma \in V(G'))(\deg^{G'}(\sigma) = \aleph_0 \implies (\exists \tau \in V(G'))(\deg^{G'}(\tau) = \aleph_0 \land \sigma \sqsubset \tau)).$$

To prove the claim notice that, by hypothesis, $(\forall^\infty \sigma \in V(G'))(\deg^{G'}(\sigma) = \aleph_0)$. Hence, given σ such that $\deg^{G'}(\sigma) = \aleph_0$, there exists another vertex $\tau \in V(G')$ such that $(\sigma, \tau) \in E(G')$ and $\deg^{G'}(\tau) = \aleph_0$: the definition of $E(\mathcal{A}^{\sqsubseteq}(T, G))$ implies that $\sigma \sqsubset \tau$. This concludes the proof of the claim.

Let $N := \max\{\deg^G(v) : v \in A\}$. We compute a sequence $\{\sigma_s : s \in \mathbb{N}\} \subseteq V(G')$ such that $\bigcup_s \sigma_s \in [T]$. At stage 0, let σ_0 be the first vertex enumerated by the name of G' satisfying $\deg^{G'}(\sigma_0) = \aleph_0$ (which exists by the previous claim): this is a computable process as it suffices to verify $\deg^{G'}(\sigma_0) > N$. Suppose we have computed the sequence up to σ_s. At stage $s+1$, let σ_{s+1} be the first vertex enumerated by the name of G', satisfying $\sigma_s \sqsubset \sigma_{s+1} \wedge \deg^{G'}(\sigma_{s+1}) > N$ (the existence of σ_{s+1} is guaranteed by the previous claim). This proves when G is computable.

To show that if G is c.e. then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{IS\text{-}Copy}^{ee}_G, \mathsf{S\text{-}Copy}^{ee}_G$ notice that the proof is exactly the same. In this case, $\mathcal{A}^{\sqsubseteq}(T,G)$ is c.e. with respect to T, but this is fine as the input for $\mathsf{IS\text{-}Copy}^{ee}_G$ and $\mathsf{IS\text{-}Copy}^{e\chi}_G$ is in $\mathbf{EGr}$. ☐

The following lemma proves Theorem *III*.4.5(*ii*).

Lemma III.4.8. *Let G be an infinite graph such that $\{v \in V(G) : \deg^G(v) = \aleph_0\}| < \aleph_0$. If G is c.e., then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{IS\text{-}Copy}_G$.*

Proof. By Figure III.3, it suffices to show that $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{IS\text{-}Copy}^{\chi e}_G$. Let $T \in \mathbf{Tr}$ be an input for $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$, and compute $G' \in \mathbf{F}(G)$ where $\mathbf{F}$ is the function of Lemma *III*.2.1. Let $\mathcal{A}^{|}(T,G')$ (see §III.3 for its definition): since $G' \in \mathbf{Gr}$, $T \in \mathcal{IF}$ and $G \subseteq_{\mathbf{is}} G'$ (Lemma *III*.2.1), by Proposition *III*.3.2, $G \subseteq_{\mathbf{is}} \mathcal{A}^{|}(T,G')$ (i.e. $\mathcal{A}^{|}(T,G')$ is a suitable input for $\mathsf{IS\text{-}Copy}^{\chi e}_G$). Let $H \in \mathsf{IS\text{-}Copy}^{\chi e}_G(\mathcal{A}^{|}(T,G'))$.

Notice that $(\forall^\infty \sigma, \tau \in V(H))(\sigma \sqsubset \tau \vee \tau \sqsubset \sigma)$. If not then there exists $\{\sigma_i : i \in \mathbb{N}\} \subseteq V(H)$ such that $(\forall i \neq j)(\sigma_i \mid \sigma_j)$. By definition of $\mathcal{A}^{|}(T,G')$ and the fact that H is an induced subgraph of $\mathcal{A}^{|}(T,G')$, we obtain that $(\forall i \neq j)((\sigma_i,\sigma_j) \in E(H))$, and hence $(\forall i)(\deg^H(\sigma_i) = \aleph_0)$. Since $H \cong G$, this contradicts the hypothesis that $|\{v \in V(G) : \deg^G(v) = \aleph_0\}| < \aleph_0$. In other words, we have just showed that $(\exists f \in [T])(\forall^\infty \sigma \in V(H))(\sigma \sqsubset f)$. We now show that:

$$(\forall \sigma \in V(H))(\deg^H(\sigma) < \aleph_0 \implies \sigma \sqsubset f). \tag{4}$$

Otherwise, $(\exists \tau \in V(H))(\deg^H(\tau) < \aleph_0 \wedge \tau \not\sqsubset f)$ but we have just shown that $(\forall^\infty \sigma \in V(H))(\sigma \sqsubset f)$ and hence $(\forall^\infty \sigma \in V(H))(\sigma \mid \tau)$. By definition of $\mathcal{A}^{\sqsubseteq}(T,G')$ and the fact that H is an induced subgraph of $\mathcal{A}^{\sqsubseteq}(T,G')$ we obtain that $\deg^H(\tau) = \aleph_0$, getting the desired contradiction.

Now we compute a sequence of vertices $\{\sigma_i : i \in \mathbb{N}\} \subseteq V(H)$ such that $\bigcup_i \sigma_i = f$. Let $N := |\{\sigma \in V(G) : \deg^G(\sigma) = \aleph_0\}|$. For every s, let $\sigma_s \in V(H)$ be such that $(\exists \tau_0, \dots, \tau_N \in V(H))(\forall i \leqslant N)(\sigma_s \sqsubset \tau_i)$ and $(\exists! \tau'_0, \dots, \tau'_{s-1} \in V(H))(\forall i < s)(\tau'_i \sqsubset \sigma_s)$ (the second condition ensures that $|\sigma_s| \geqslant s$). For any s the existence of σ_s is guaranteed by (4): just let $\sigma_s \in \{\sigma \in V(H) : \deg^H(\sigma) < \aleph_0 \wedge |\sigma| \geqslant s\}$. It remains to show that for every s, $\sigma_s \sqsubset f$. To do so, notice that any σ_s has (at least) $N+1$ many extensions $\tau_0, \dots, \tau_N$ in T. By hypothesis there are only N many vertices of infinite degree in $V(H)$, hence there exists an $i < N$ such that $\deg^H(\tau_i) < \aleph_0$, i.e. $\sigma_s \sqsubset \tau_i \sqsubset f$, and this concludes the proof. ☐

Notice that Lemma *III*.4.7 and Lemma *III*.4.8 do not exhaust all the possible cases: it may be the case that G is such that $\{v \in V(G) : \deg^G(v) < \aleph_0\} = \aleph_0$ (as in Lemma *III*.4.8) but $\{v \in V(G) : \deg^G(v) = \aleph_0\} = \aleph_0$ too. The following lemma shows Theorem *III*.4.5(*iii*) and concludes the proof of Theorem *III*.4.5. The proof of the next lemma is similar to the one of Lemma *III*.4.8: the main difference is that here $|\{\sigma \in V(G) : \deg^G(\sigma) = \aleph_0\}| \notin \mathbb{N}$ and hence the reduction is given relative to an oracle.

Lemma III.4.9. *Let $\lambda : \mathbb{N} \to \mathbb{N}$ be such that $\lambda(n) := \min\{k : |v \in V(G) : \deg^G(v) \leqslant k|\} \geqslant k$ and let G be an infinite graph such that*

$$\{v \in V(G) : \deg^G(v) < \aleph_0\} = \aleph_0 \textit{ and } \{v \in V(G) : \deg^G(v) = \aleph_0\} = \aleph_0.$$

If G is computable then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_\mathrm{W} \mathsf{S\text{-}Copy}_G$, $\mathsf{IS\text{-}Copy}_G$ *relative to* λ. *If G is c.e. then* $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_\mathrm{W} \mathsf{IS\text{-}Copy}^{ee}_G, \mathsf{S\text{-}Copy}^{ee}_G$ *relative to* λ.

Proof. By Figure III.3, it suffices to show that $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_\mathrm{W} \mathsf{IS\text{-}Copy}^{\chi e}_G$. Let $T \in \mathbf{Tr}$ be an input for $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$, assume $V(G) = \{\sigma_i : i \in \mathbb{N}\}$ and compute $\mathcal{A}^{\mid}(T,G)$. Let $G' \in \mathsf{IS\text{-}Copy}^{\chi e}_G(\mathcal{A}^{\mid}(T,G))$ and notice that the same proof of Lemma $III.4.8$ gives us that cofinitely vertices in G' belong to the same path $f \in [T]$ and

$$(\forall \sigma \in V(G'))(\deg^{G'}(\sigma) < \aleph_0 \implies \sigma \sqsubset f). \tag{5}$$

Now we compute a sequence of vertices $\{\sigma_s : s \in \mathbb{N}\} \subseteq V(G')$ such that $\bigcup_s \sigma_s = f$. For any s, let $\sigma_s \in V(G')$ be such that:

(*i*) $(\exists \tau_0, \dots, \tau_{\lambda(s+1)} \in V(G'))(\forall i \leqslant \lambda(s+1))(\sigma_s \sqsubset \tau_i)$;

(*ii*) if $s > 0$, $(\exists! \tau'_0, \dots, \tau'_{s-1} \in V(G'))(\forall i < s-1)(\tau'_i \sqsubset \tau'_{i+1} \sqsubset \sigma_s)$.

Condition (ii) ensures that $|\sigma_s| \geqslant s$ and for any s the existence of σ_s is guaranteed by (5): just let $\sigma_s \in \{\sigma \in V(G') : \deg^{G'}(\sigma) < \aleph_0 \wedge |\sigma| \geqslant s\}$. It remains to show that for every s, $\sigma_s \sqsubset f$. To do so, notice that by hypothesis $(\exists v_0, \dots, v_s \in V(G))(\deg^G(v_s) \leqslant \lambda(s+1))$. Any isomorphism from G to G', for any $i \leqslant s$, must map v_i to some $\tau \in V(G')$ such that $\tau \sqsubseteq \sigma_s \vee \sigma_s \sqsubseteq \tau$. Suppose it is not the case: if $\tau \mid \sigma_s$ then, since $(\forall i < \lambda(s+1))(\sigma_s \sqsubset \tau_i)$, we have that $(\forall i < \lambda(s+1))(\tau \mid \tau_i)$ and so $\deg^{G'}(\tau) \geqslant \lambda(s+1)+1$. Hence, any isomorphism from G to G' maps one between $v_0, \dots, v_s$ in some $\tau \in V(G')$ such that $\tau \sqsubseteq \sigma_s$ (this is guaranteed by (ii)): indeed, there are only s many vertices in G' that are prefixes of σ_s, hence $\sigma_s \sqsubset f$ and this concludes the proof.

To show that if G is c.e. then $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_\mathrm{W} \mathsf{IS\text{-}Copy}^{ee}_G, \mathsf{S\text{-}Copy}^{ee}_G$ notice that the proof is exactly the same. In this case, $\mathcal{A}^{\mid}(T,G)$ is c.e. with respect to T, but this is fine as the input for $\mathsf{IS\text{-}Copy}^{ee}_G$ and $\mathsf{IS\text{-}Copy}^{e\chi}_G$ is in $\mathbf{EGr}$. □

Notice that for some graphs, the λ defined in the previous theorem is computable. For example, this is the case for *highly recursive* graphs, i.e. graphs in which for every $v \in V(G)$, we can compute $\deg^G(v)$. These particular graphs have been considered, for different problems, for example in [MR72]. We also mention that Lemma $III.4.9$ holds even if we replace λ with any γ bounding λ.

We leave open whether it is possible to "get rid of" the oracle in Lemma $III.4.9$, obtaining a Weihrauch reduction like in Lemmas $III.4.7$ and $III.4.8$ that would give us the result that for any computable/c.e. graph G, $\mathsf{IS\text{-}Copy}_G \equiv_\mathrm{W} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$.

III.4.2 The subgraph problem: when $R_\omega \subseteq_\mathbf{s} G$

Proposition $III.4.6$ shows that $\mathsf{S\text{-}Copy}_G \equiv_\mathrm{W} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ where G is an infinite c.e. graph such that $|\{v \in V(G) : \deg^G(v) < \aleph_0\}| < \aleph_0$. To study the Weihrauch degree of $\mathsf{S\text{-}Copy}_G$ for c.e. graphs not satisfying such a condition, and in particular to understand which of them satisfy $\mathsf{S\text{-}Copy}_G \equiv_\mathrm{W} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$, we start from those graphs that, intuitively, are "ill-founded", i.e. those graphs G such that $R_\omega \subseteq_\mathbf{s} G$. A piece of evidence supporting the claim that $\mathsf{S\text{-}Copy}_G \equiv_\mathrm{W} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ for any G such that $R_\omega \subseteq_\mathbf{s} G$ is Lemma $III.3.19$: in that case, the fact that $R_\omega \subseteq_\mathbf{s} G$ implied that $(e)\mathsf{S}_G \equiv_\mathrm{W} \mathsf{WF}$. In this section we show this intuition is not entirely true: before doing so, we define the following multi-valued functions.

Definition III.4.10. Let G be a graph such that $R_\omega \subseteq_\mathbf{s} G$. The multi-valued functions $R_\omega\text{-}\mathsf{Emb}_G :\subseteq \mathbf{Gr} \rightrightarrows \mathbb{N}^\mathbb{N}$ and $eR_\omega\text{-}\mathsf{Emb}_G :\subseteq \mathbf{EGr} \rightrightarrows \mathbb{N}^\mathbb{N}$ with domains respectively $\{H \in \mathbf{Gr} : G \cong H\}$ and $\{H \in \mathbf{EGr} : G \cong H\}$ are defined as

$$(e)R_\omega\text{-}\mathsf{Emb}_G(H) := \{p : (\forall i)((p(i), p(i+1)) \in E(H))\}.$$

The next proposition shows that R_ω-Emb_G and eR_ω-Emb_G are cylinders, which implies that most reductions we obtain in this section are actually strong ones: the proof is similar to the one proving Proposition *III*.4.2.

Proposition III.4.11. *For any graph G, R_ω-Emb_G and eR_ω-Emb_G are cylinders.*

Lemma III.4.12. *Given a hyperarithmetical graph G such that $R_\omega \subseteq_{\mathbf{s}} G$, R_ω-$\mathsf{Emb}_G \leqslant_{\mathrm{W}} eR_\omega$-$\mathsf{Emb}_G \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$.*

Proof. The first reduction directly follows from the fact that from the characteristic function of a graph we can compute an enumeration of it, while for the second one it suffices to notice that if $H \in \mathrm{dom}(eR_\omega\text{-}\mathsf{Emb}_G)$ then $A := \{p : (\forall i)((p(i), p(i+1)) \in E(H))\} \in \mathrm{dom}(\mathsf{C}_{\mathbb{N}^\mathbb{N}})$ and so any $p \in \mathsf{C}_{\mathbb{N}^\mathbb{N}}(A)$ is a solution for eR_ω-$\mathsf{Emb}_G(H)$. ☐

The following proposition tells us that $\mathsf{S\text{-}Copy}_G$ composed with $(e)R_\omega$-Emb_G computes $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$.

Proposition III.4.13. *Given a c.e. graph G such that $R_\omega \subseteq_{\mathbf{s}} G$,*

$$eR_\omega\text{-}\mathsf{Emb}_G * \mathsf{S\text{-}Copy}^{ee}_{R_\omega} \equiv_{\mathrm{W}} R_\omega\text{-}\mathsf{Emb}_G * \mathsf{S\text{-}Copy}^{e\chi}_{G} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}.$$

Given a computable graph G such that $R_\omega \subseteq_{\mathbf{s}} G$,

$$eR_\omega\text{-}\mathsf{Emb}_G * \mathsf{S\text{-}Copy}^{\chi e}_{R_\omega} \equiv_{\mathrm{W}} R_\omega\text{-}\mathsf{Emb}_G * \mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}.$$

Proof. We only prove that eR_ω-$\mathsf{Emb}_G * \mathsf{S\text{-}Copy}^{\chi e}_{R_\omega} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ as the same proof works also for the other equivalences. Proposition *III*.4.3 and Lemma *III*.4.12 and the fact that $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ is closed under compositional product imply that eR_ω-$\mathsf{Emb}_G * \mathsf{IS\text{-}Copy}^{\chi e}_G \leqslant_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$.

For the opposite direction, given an input T for $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$, compute $\mathcal{A}^{\sqsubseteq}(T, G)$ and notice that since $T \in \mathcal{IF}$, $\mathcal{A}^{\sqsubseteq}(T, G) \in \mathrm{dom}(\mathsf{IS\text{-}Copy}^{\chi e}_G)$ (Proposition *III*.3.2). Let $G' \in \mathsf{IS\text{-}Copy}^{\chi e}_G(\mathcal{A}^{\sqsubseteq}(T, G))$ and let $p \in eR_\omega$-$\mathsf{Emb}_G(G')$. Notice that $i_0 := \min\{i : (\forall j \geqslant i)(p[j] \sqsubset p[j+1])\}$ exists and it is computable: clearly $\bigcup_{i > i_0} p[i] \in [T]$. ☐

By Proposition *III*.4.13, in order to show that a graph G is such that $\mathsf{S\text{-}Copy}_G \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$, it suffices to show that $(e)R_\omega$-Emb_G is computable. The next proposition gives some examples of graphs satisfying this condition. In the next proposition, $2^{<\mathbb{N}}$ denotes the full binary tree $\{\sigma : \sigma \in 2^{<\mathbb{N}}\}$.

Proposition III.4.14. *Let $n > 2$ and $m > 0$: if $G \in \{\mathsf{L}, C_n \odot R_\omega, K_m \odot R_\omega, 2^{<\mathbb{N}}\}$ then $\mathsf{S\text{-}Copy}_G \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$.*

Proof. By Lemma *III*.4.12 and Proposition *III*.4.13, it suffices to show that eR_ω-Emb_G is computable. Let H be an in input for eR_ω-$\mathsf{Emb}_{\mathsf{L}}$, and notice that for any $v \in V(H)$, $\deg^H(v) = 2$. To compute $p \in eR_\omega$-$\mathsf{Emb}_{\mathsf{L}}(L)$ let $p(0)$ be such that $p(0) \in V(H)$ and for every $i > 0$ choose $p(i)$ such $(p(i), p(i+1)) \in E(H)$ since, at each stage, $p(i)$ exists and is unique, this shows that eR_ω-$\mathsf{Emb}_{\mathsf{L}}$ is computable.

Let H be an in input for eR_ω-$\mathsf{Emb}_{(C_n \odot R_\omega)}$. We compute $p \in eR_\omega$-$\mathsf{Emb}_{(C_n \odot R_\omega)}$ as follows. Wait for some finite stage witnessing that $C_n \odot R_1 \subseteq_{\mathbf{s}} H$ and denote by H' the copy of $C_n \odot R_1$ in H. Then let $p(0)$ be the unique $v \in V(H')$ such that $\deg^{H'}(v) = 1$: clearly, all vertices in $H \backslash H'$ "continue" as a copy of R_ω, hence for $s > 0$ just let $p(s)$ be such that $(p(s-1), p(s)) \in E(H \backslash H')$. A similar proof holds for $K_m \odot R_\omega$.

Let H be an input for eR_ω-$\mathsf{Emb}_{2^{<\mathbb{N}}}$. We compute $p \in eR_\omega$-$\mathsf{Emb}_{2^{<\mathbb{N}}}(H)$ as follows. Let $p(0)$ be any node in $V(H)$ and for any s let $p(s+1) := \sigma$ where $|\sigma| = s$ and $\sigma \sqsupset p[s-1]$: by definition of $2^{<\mathbb{N}}$, we can always find σ as such and this concludes the proof. □

III.4.3 A particular case: $\mathsf{S\text{-}Copy}_{R_\omega}$

It is natural to ask whether Proposition *III*.4.14 holds for R_ω: the following proposition shows that $(e)R_\omega\text{-}\mathsf{Emb}_{R_\omega}$ is not computable, and hence we cannot apply the same strategy used in the proof of Proposition *III*.4.14.

Proposition III.4.15. $\mathsf{lim}_2 \equiv_{\mathrm{W}} R_\omega\text{-}\mathsf{Emb}_{R_\omega} \equiv_{\mathrm{W}} eR_\omega\text{-}\mathsf{Emb}_{R_\omega}$.

Proof. Lemma *III*.4.12 implies that $R_\omega\text{-}\mathsf{Emb}_{R_\omega} \leqslant_{\mathrm{W}} eR_\omega\text{-}\mathsf{Emb}_{R_\omega}$.

We now show that $eR_\omega\text{-}\mathsf{Emb}_{R_\omega} \leqslant_{\mathrm{W}} \mathsf{lim}_2$. Let $G \cong R_\omega$ be an input for $eR_\omega\text{-}\mathsf{Emb}_{R_\omega}$: we compute an input $q \in 2^{\mathbb{N}}$ for lim_2 in stages as follows. At stage 0, let $v, w \in V(G)$ be such that $(v, w) \in E(G)$ and let $q(0) := 0$. At stage $s + 1$, take some fresh $u_{s+1} \in V(G)$ (i.e. one that has not been considered yet) and let

$$q(s+1) := \begin{cases} 0 & \text{if } v \leftrightsquigarrow^G_w u_{s+1}, \\ 1 & \text{if } v \leftrightsquigarrow^G_{\neg w} u_{s+1}. \end{cases}$$

Informally, we are computing q considering a copy R of R_ω starting from v and checking whether R continues in the direction of w (in which case $\mathsf{lim}(q) = 0$) or not (in which case $\mathsf{lim}(q) = 1$). If $\mathsf{lim}_2(q) = 0$, we compute $p \in eR_\omega\text{-}\mathsf{Emb}_{R_\omega}(G)$ letting $p(0) := v$ and, for $i > 0$, $p(i) := u_i$ where u_i is the unique vertex connected to v, by a ray of length i passing via w: the fact that $\mathsf{lim}_2(q) = 0$ implies that $(\exists^\infty s)(v \leftrightsquigarrow^G_w u_s)$, and hence we can always find u_i. The case in which $\mathsf{lim}(q) = 1$ is held similarly letting $p(0) := v$ and, for $i > 0$, $p(i) := u_i$ where u_i is the unique vertex connected to v, by a ray of length i not passing via w.

To conclude the proof it suffices to show that $\mathsf{lim}_2 \leqslant_{\mathrm{W}} R_\omega\text{-}\mathsf{Emb}_{R_\omega}$. Let $q \in 2^{\mathbb{N}}$ be an input for lim_2. We compute an input for $R_\omega\text{-}\mathsf{Emb}_{R_\omega}$ in stages: if at stage s $q(s) = 0$, we extend the ray computed so far to the left, otherwise to the right. More formally, we compute a name g for an input of $R_\omega\text{-}\mathsf{Emb}_{R_\omega}$ as follows: at stage 0 let $g(\langle 0,0\rangle) = 1$ and if $q(0) = 0$ let $g(\langle 2,2\rangle) = g(\langle 0,2\rangle) = 1$, otherwise let $g(\langle 1,1\rangle) = g(\langle 0,1\rangle) = 1$. At stage $s+1$, if $q(s+1) = 0$ let $g(\langle 2s+2, 2s+2\rangle) = 1$ and $g(\langle 2s+2, x\rangle) = 1$ where $x := \max\{n = 2t+2 : t < s \wedge g(\langle n,n\rangle) = 1\}$. Similarly, if $q(s+1) = 1$ let $g(\langle 2s+1, 2s+1\rangle) = 1$ and $g(\langle 2s+1, x\rangle) = 1 \in E(G)$ where $x := \max\{n = 2t+1 : t < s \wedge g(\langle n,n\rangle) = 1\}$. Since q converges either to 0 or 1, $\delta_{Gr}(g) \cong R_\omega$ and so it is a suitable input for $R_\omega\text{-}\mathsf{Emb}_{R_\omega}$. Let $p \in R_\omega\text{-}\mathsf{Emb}_{R_\omega}(\delta_{\mathbf{Gr}}(g))$:

$$\mathsf{lim}_2(p) = \begin{cases} 0 & \text{if } (p(0) < p(1) \wedge p(1) \text{ is even}) \vee (p(0) > p(1) \wedge (p(1) \text{ is odd} \vee p(1) = 0), \\ 1 & \text{if } (p(0) < p(1) \wedge p(1) \text{ is odd}) \vee (p(0) > p(1) \wedge p(1) \text{ is even}), \end{cases}$$

and this concludes the proof. □

Combining the proposition above with Propositions *III*.4.13 and *III*.4.15 we get the following corollary.

Corollary III.4.16. $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{lim}_2 * \mathsf{S\text{-}Copy}_{R_\omega}$.

It is natural to ask whether we really need lim_2, i.e. does $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{S\text{-}Copy}_{R_\omega}$? The next proposition tells us that the first-order parts of the two problems coincide.

Proposition III.4.17. ${}^1\mathsf{S\text{-}Copy}_{R_\omega} \equiv_{\mathrm{W}} {}^1\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_{\mathrm{W}} \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_{\mathbb{N}}$.

Proof. Propositions $III.4.3$ and $I.6.29$ imply that ${}^1\mathsf{S\text{-}Copy}_{R_\omega} \leqslant_W {}^1\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N}$. To conclude the proof, by Figure III.3, it suffices to show that $\mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N} \leqslant_W \mathsf{S\text{-}Copy}^{\chi e}_{R_\omega}$. We can think of an input for $\mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N}$ as a sequence $(T^i)_{i\in\mathbb{N}} \in \mathbf{Tr}^\mathbb{N}$ such that $(\exists i)(T^i \in \mathcal{IF})$. Let $T := \bigotimes_{i\in\mathbb{N}} T^i$ (notice that $\otimes$ is the disconnected union of trees): since at least one $T^i \in \mathcal{IF}$, $S \in \operatorname{dom}(\mathsf{S\text{-}Copy}^{\chi e}_{R_\omega})$. Given $G \in \mathsf{S\text{-}Copy}^{\chi e}_{R_\omega}(S)$, we have that G is a subgraph of T^i where $T^i \in \mathcal{IF}$ and since any vertex in $V(G)$ is of the form $\langle i, \sigma\rangle$, we can easily compute i. □

The following theorem surprisingly shows that the problems $\mathsf{S\text{-}Copy}_{R_\omega}$ are significantly weaker than $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$.

Theorem III.4.18. $\mathsf{S\text{-}Copy}_{R_\omega} <_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$. *In particular,*

$$(i)\ \mathsf{S\text{-}Copy}^{\chi e}_{R_\omega}, \mathsf{S\text{-}Copy}^{ee}_{R_\omega} \mid_W \mathsf{lim},\ (ii)\ \mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega} \mid_W \mathsf{lim}' \text{ and } (iii)\ \mathsf{S\text{-}Copy}^{e\chi}_{R_\omega} \mid_W \mathsf{lim}''.$$

The proof of the theorem above is given throughout the remaining part of this subsection: notice that the right-to-left nonreductions are almost immediate. Indeed, it suffices to notice that $\mathsf{S\text{-}Copy}_{R_\omega} \equiv_W \mathsf{lim}_2 * \mathsf{C}_{\mathbb{N}^\mathbb{N}}$ (Corollary $III.4.16$), while, since for every n, $\mathsf{lim}^{(n)}$ is closed under compositional product and $\mathsf{lim}_2 <_W \mathsf{lim}^{(n)}$ we have that $\mathsf{lim}^{(n)} * \mathsf{lim}_2 \equiv_W \mathsf{lim}^{(n)} <_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$. Hence, the remaining part of this subsection is devoted to prove the left-to-right nonreductions.

Step 1 of Theorem $III.4.18$' proof: restricting to connected graphs

To prove the theorem above, it is convenient to consider $\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}_{R_\omega}$, which is the same problem as $\mathsf{S\text{-}Copy}_{R_\omega}$ with the domain restricted to connected graphs. The reductions in Figure III.4.1, Corollary $III.4.16$ and Proposition $III.4.2$ hold also for this restricted version as summarized in the next proposition.

Proposition III.4.19.

- *For any graph G,* $\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{\chi e}_G \leqslant_W \mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{\chi\chi}_G, \mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{ee}_G \leqslant_W \mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{e\chi}_G$;
- $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \equiv_W \mathsf{lim}_2 * \mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{\chi e}_{R_\omega}$;
- *for any infinite graph G,* $\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}_G$ *are cylinders.*

Regarding the connectedness of graphs, we define the multi-valued function D introduced in [GHM15, §6].

Definition III.4.20. We define the multi-valued function $\mathsf{D} : \mathbf{Gr} \rightrightarrows \mathbb{N}^\mathbb{N}$ defined as $\mathsf{D}(G) := f$ where $f : V(G) \to \mathbb{N}$ is such that $(\forall v_1 \neq v_2 \in V(G))(f(v_1) = f(v_2) \iff v_1 \leftrightsquigarrow^G v_2)$.

Lemma III.4.21 ([GHM15, Theorem 6.4])**.** $\mathsf{D} \equiv_W \mathsf{lim}$.

We first need the following technical lemma.

Lemma III.4.22. *If* $\mathsf{lim} \leqslant_W f * g$ *where* $g : \mathbf{X} \rightrightarrows \mathbb{N}$, *then* $\mathsf{LPO} \leqslant_W f$ *relative to some oracle.*

Proof. Since $\mathsf{lim} \equiv_W \mathsf{J}$, where J is the Turing jump operator (see §I.6), suppose that $\mathsf{J} \leqslant_W f * g$. Let $B_n \subseteq X$ be the set of inputs to g where $n \in \mathbb{N}$ is a valid output. Let $\Phi : 2^\mathbb{N} \to \mathbf{X}$ be the forward functional witnessing the reduction $\mathsf{J} \leqslant_W f * g$ and let $A_n := H^{-1}(B_n)$: notice that $2^\mathbb{N} = \bigcup_{n\in\mathbb{N}} A_n$. If we restrict J to A_n (denoted by $\mathsf{J}|_{A_n}$), we can safely replace g with the constant function n. As a consequence, we conclude that $J|_{A_n} \leqslant_W f$ for all $n \in \mathbb{N}$.

Let $C \subseteq \mathbb{N}$ be the set of all $n \in \mathbb{N}$ such that $\mathsf{J}|_{A_n}$ (that, J with domain restricted to A_n) is continuous. If $\mathsf{J}|_{A_n}$ is continuous, it is computable relative to some oracle, say p_n. Now consider $q := \oplus_{n \in C} p_n$. If $q \in A_n$ for some $n \in C$ were true, then $\mathsf{J}(q) \leqslant_{\mathrm{T}} q$ would follow, a contradiction. Thus, there exists some $d \in \mathbb{N} \setminus C$, i.e. some $\mathsf{J}|_{A_d}$ is discontinuous.

As $\mathsf{J}|_{A_d} :\subseteq 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ is a discontinuous function between admissible represented space, it follows that $\mathsf{LPO} \leqslant_{\mathrm{W}} \mathsf{J}|_{A_d}$ relative to some oracle. We already established $\mathsf{J}|_{A_d} \leqslant_{\mathrm{W}} f$, so $\mathsf{LPO} \leqslant_{\mathrm{W}} f$ relative to some oracle follows. □

Lemma III.4.23.

(*i*) $\mathsf{S\text{-}Copy}^{ee}_{R_\omega} \leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{ee}_{R_\omega} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_{\mathbb{N}}$;

(*ii*) $\mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega} \leqslant_{\mathrm{W}} (\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_{\mathbb{N}})'$.

Proof. Let q be a name for an input $H \in \mathbf{EGr}$ of $\mathsf{S\text{-}Copy}^{ee}_{R_\omega}$ and let $A := \{v : (\exists G_0 \cong G)(G_0 \text{ is a subgraph of } H \wedge v \in V(G_0))\}$. This is a valid input for $\mathbf{\Sigma}^1_1\text{-}\mathsf{C}_{\mathbb{N}}$ and, given $v \in \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_{\mathbb{N}}(A)$, we can compute a name for the graph $H_0 := H\restriction_{\{w \in V(H) : v \leftrightsquigarrow^H w\}} \in \mathbf{EGr}$ (notice that H_0 is connected and, by Lemma *III*.3.4, $v \leftrightsquigarrow^H w$ is a Σ^0_1 property). To do so, we enumerate a vertex w in $V(H_0)$ only when $(\exists s)(v \leftrightsquigarrow^{\delta_{EGr}(q[s]0^{\mathbb{N}})} w)$. Then we enumerate all the vertices/edges in the path from v to w. By definition of v, $R_\omega \subseteq_{\mathbf{s}} H_0$ and any $R \in \mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{ee}_{R_\omega}(H_0)$ is a valid solution for $\mathsf{S\text{-}Copy}^{ee}_{R_\omega}$.

To prove the second item, by the fact that for any multi-valued function f, $f * \lim \equiv_{\mathrm{W}} f'$ (see §I.6) and Lemma *III*.4.21, the right-hand-side of the reduction is equivalent to $\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_{\mathbb{N}} * \mathsf{D}$ and hence it suffices to show that $\mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega} \leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_{\mathbb{N}} * \mathsf{D}$. Let q be a name for an input $H \in \mathbf{Gr}$ of $\mathsf{S\text{-}Copy}^{\chi\chi}_{G}$, and let $f \in \mathsf{D}(H)$. Then, given A as above, let $v \in \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_{\mathbb{N}}(A)$. Now consider the graph $H\restriction_{\{w : f(w) = f(v)\}}$: this is a suitable input for $\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega}$ any $R \in \mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega}(H\restriction_{\{w : f(w) = f(v)\}})$ is a valid solution for $\mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega}(H)$. □

Step 2 of Theorem *III*.4.18' proof: proving the theorem for $\mathsf{S\text{-}Copy}^{\chi e}_{R_\omega}$ and $\mathsf{S\text{-}Copy}^{ee}_{R_\omega}$

The finitary part of a problem was defined in §I.6.

Theorem III.4.24. *For every $k \in \mathbb{N}$, $\mathrm{Fin}_k(\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{ee}_{R_\omega})$ is computable, and hence we obtain that $\mathrm{Fin}(\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{ee}_{R_\omega})$ is computable as well.*

Proof. Let $f :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows k$ for some $k \in \mathbb{N}$ and suppose that $f \leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{ee}_{R_\omega}$ as witnessed by the computable maps Φ and Ψ.

Let r be a name for $x \in \mathrm{dom}(f)$ and notice that R_ω is a subgraph of $\Phi(r)$. The proof strategy is the following:

(*i*) proving that $\Phi(r)$ contains $k+1$-many distinct rays of finite length $R_{\ell_0}, \dots, R_{\ell_k}$ with names p_{ℓ_i} such that the following hold

- $\bigodot_{i \leqslant k} R_{\ell_i} \cong R_N$ where $N := \sum_{i \leqslant k} R_{\ell_i}$,
- $\bigodot_{i \leqslant k} R_{\ell_i}$ is a subgraph of $\Phi(r)$ and
- for every $i \leqslant k$, $\Psi(r[|p_{\ell_i}|], p_{\ell_i}) \downarrow = j$ for $j < k$.

(*ii*) Once we have proven (*i*), the fact that there are only k-many solutions for $f(x)$ implies that

$$(\exists m)(\exists i \neq j)(\Psi(p[|q_{\ell_i}|], q_{\ell_i}) \downarrow = \Psi(r[|q_{\ell_j}|], q_{\ell_j}) \downarrow = m).$$

To conclude the proof, we show that finding m is a computable process, and that $m \in f(x)$.

To prove (i), let p_0 be a name for $S_0 \in \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}(r, \Phi(r))$ and, for readability, assume $V(S_0) = \{v_i : i \in \mathbb{N}\}$ and $E(S_0) = \{(v_i, v_{i+1}) : i \in \mathbb{N}\}$. Let $s_0 := \min\{t > 0 : \Psi(r[t], p_0[t]) \downarrow\}$ and let $\ell_0 := \max\{i : (\exists j \leqslant s_0)(v_i = p_0(\langle j, j\rangle))\}$. Without loss of generality, we can assume that $\delta_{EGr}(p_0[s_0]0^{\mathbb{N}}) \cong R_{\ell_0}$. Indeed, if $\delta_{EGr}(p_0[s_0]0^{\mathbb{N}}) \not\cong R_{\ell_0}$, since $\Psi(r[s_0], p_0[s_0]) \downarrow$ and $S_0 \cong R_\omega$ we can extend $p_0[s_0]$ to $p_0[s_0]^\frown\sigma$ where σ is the (finite string) having digits $\langle v_i, v_i\rangle$ for $i < \ell_0$ and $(v_j, v_j + 1)$ for $j < \ell_0 - 1$. Then, we obtain that $\delta_{EGr}(p_0[s_0]^\frown\sigma 0^{\mathbb{N}}) \cong R_{\ell_0}$ and $\Psi(r[s_0 + |\sigma|], p_0[s_0]^\frown\sigma) \downarrow$. For $0 < m \leqslant k$, let p_m be a name for S_m where $V(S_m) = V(S_0)\backslash\{v_i : i < \ell_{m-1}\}$ and $E(S_m) = E(S_0)\backslash\{(v_i, v_{i+1}) : i < \ell_{m-1} - 1\}$. Notice that $S_m \cong R_\omega$, hence $S_m \in \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi e}_{R_\omega}(p, \Phi(r))$. Then, given $s_m := \min\{t > 0 : \Psi(p[t], p_m[t]) \downarrow\}$, let $\ell_m := \max\{i : (\exists j \leqslant s_k)(v_i = p_m(\langle j, j\rangle))\}$. Again, without loss of generality, we can assume that $\delta_{EGr}(p_m[s_m]0^{\mathbb{N}}) \cong R_{\ell_m}$. Since for every $i < k$, $\max\{v : v \in V(\delta_{EGr}(p_{\ell_i}[s_{\ell_i}]0^{\mathbb{N}}))\} = \min\{v : v \in V(\delta_{EGr}(p_{\ell_{i-1}}[s_{\ell_{i-1}}]0^{\mathbb{N}}))\}$, we obtain that $\bigodot_{i \leqslant k} R_{\ell_i} \cong \bigodot_{i \leqslant k} \delta_{EGr}(p_{\ell_i}[s_{\ell_i}]0^{\mathbb{N}})$ and the proof of (i).

To prove (ii), it suffices to show that

$$(\exists S \in \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}(\Phi(r)))(R_{\ell_i} \text{ is a subgraph of } S \vee R_{\ell_j} \text{ is a subgraph of } S). \tag{6}$$

Indeed, suppose R_{ℓ_i} is a subgraph of S (the case for R_{ℓ_j} is analogous): then there is a name for S that begins with q_{ℓ_i}, i.e. an enumeration of R_{ℓ_i}. Since $\Psi(r[|q_{\ell_i}|], q_{\ell_i}) \downarrow = m$ by hypothesis, we are done. So let $S' \in \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}(\Phi(r))$. We have the following cases:

- if (6) holds there is nothing to prove;
- if $V(S') \cap (V(R_{\ell_i}) \cup V(R_{\ell_j})) = \varnothing$, notice that by hypothesis $\Phi(r)$ is connected, hence $(\exists v \in V(R_{\ell_i}))(\exists w \in V(S))(v \leftrightsquigarrow^{\Phi(r)} w)$. If $\deg^{R_{\ell_i}}(v) = 1$, let S be the infinite ray starting with R_{ℓ_i}, passing through w and continuing as S'. Otherwise, let S be the infinite ray starting with R_{ℓ_j}, passing through v and w and continuing as S'.
- if $V(S') \cap V(R_{\ell_i}) \sqsupseteq \{v\}$ let S be any infinite ray containing R_{ℓ_j}, passing through v and continuing as S'; the case $V(S) \cap V(R_{\ell_j}) \sqsupseteq \{v\}$ is analogous.

Since the procedure to search $\bigodot_{i \leqslant k} R_{\ell_i}$ is computable we have shown that f is computable and this proves the theorem. □

The following corollary is immediate combining the previous theorem and Proposition $III.4.19$.

Corollary III.4.25. $\mathsf{LPO} \not\leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi e}_{R_\omega}$, $\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}$ *(the same holds relative to any oracle).*

The next proposition proves Theorem $III.4.18(i)$ for $\mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}$ and $\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}$.

Proposition III.4.26. $\lim \not\leqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{\chi e}_{R_\omega}, \mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}$.

Proof. For the sake of contradiction, suppose that $\lim \leqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}$: by Lemma $III.4.23(i)$ we obtain that $\lim \leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_{\mathbb{N}}$. Lemma $III.4.22$ implies that $\mathsf{LPO} \leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}$ relative to some oracle, contradicting Corollary $III.4.25$. □

Before moving our attention to $\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$ and $\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}$, we prove that the fact that $\mathrm{Fin}(\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega})$ is computable (Theorem $III.4.24$) cannot be extended to first-order functions. Let $\mathrm{ACC}_{\mathbb{N}}$ be the restriction of $\mathsf{C}_{\mathbb{N}}$ to sets of the form $\mathbb{N}$ or $\mathbb{N}\backslash\{n\}$ for some $n \in \mathbb{N}$: notice that this problem is not computable.

Proposition III.4.27. $\mathrm{ACC}_{\mathbb{N}} <_{\mathrm{W}} {}^{1}\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}$.

Proof. Let A be an input for $\mathrm{ACC}_{\mathbb{N}}$, and let $A^c[s]$ denote the enumeration of the complement of A up to stage s. We compute a graph G as follows. At stage 0, let $1 \in V(G)$ (we deliberately leave 0 outside $V(G)$ for the moment). At stage $s+1$,

- if $A^c[s+1] = \varnothing$, let $s+2 \in V(G)$ and $(s+1, s+2) \in E(G)$.
- if $A^c[s+1] = \{n\}$
  - if $n \leqslant s+1$, let $0 \in V(G)$, $(n, 0) \in E(G)$ and add a copy of R_ω starting from 0 and end the construction.
  - if $n > s+1$, add a ray from $s+1$ to n having vertices $\{i : s+1 \leqslant i \leqslant n\}$ and let $(n, 0) \in E(G)$. Then add a copy of R_ω starting from 0 and end the construction.

Let $R' \in \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}(G)$. Exactly one of the following holds:

- $(n, n+1), (n+1, n+2) \in E(R')$, for $n > 0$. In this case, $n \in \mathrm{ACC}_{\mathbb{N}}(A)$;
- $(n, 0) \in E(R')$ for $n > 0$. Then any $m \neq n$ is in $\mathrm{ACC}_{\mathbb{N}}(A)$.

Strictness follows from Proposition $III.4.19$. □

Combining Theorem $III.4.24$ and Propositions $III.4.27$ and $I.6.16$ and Figure III.4.1 we obtain the following corollary.

Corollary III.4.28. $\mathrm{Fin}(\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}) <_{\mathrm{W}} {}^{1}\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}$.

In the light of the above corollary, it is natural to characterize ${}^{1}\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega}$ in terms of some well-known problem in the Weihrauch lattice.

Step 3: proving Theorem $III.4.18$ for $\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$ and $\mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}$

The next proposition, together with the next corollary, shows that $\mathrm{Fin}(\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi e}_{R_\omega})$ is not computable, and hence the same proof technique used to prove Theorem $III.4.18(i)$ does not work here.

Proposition III.4.29. $\mathsf{C}_{2^{\mathbb{N}}} <_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$.

Proof. Let $T \in \mathbf{Tr}_2$ be an input for $\mathsf{C}_{2^{\mathbb{N}}}$ and let $G \in \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}(T)$. Notice that, the fact that $T \in \mathbf{Tr}_2$ implies that for every n $|\{\sigma \in V(T) : |\sigma| = n\}| \leqslant 2^n$, and this combined with the fact that $G \in \mathbf{Gr}$ implies that we can compute $|\{\sigma \in V(G) : |\sigma| = n\}|$. Finally, it is easy to verify that $\bigcup_{n\in\mathbb{N}}\{\sigma : |\{\sigma \in V(G) : |\sigma| = n\}| = 1\} \in [T]$ and this proves the reduction.

Strictness follows from Proposition $III.4.19$ and the fact that $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}} \not\leqslant_{\mathrm{W}} \mathsf{lim}_2 * \mathsf{C}_{2^{\mathbb{N}}}$. □

Notice that $\mathsf{C}_2^* \equiv_{\mathrm{W}} {}^{1}\mathsf{C}_{2^{\mathbb{N}}}$ ([SV22, Corollary 7.6]).

Corollary III.4.30. $\mathsf{C}_2^* \equiv_{\mathrm{W}} {}^{1}\mathsf{C}_{2^{\mathbb{N}}} <_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$.

Proposition $III.4.29$ says that $\mathrm{Fin}(\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega})$ is not computable: despite this, we can prove that $\mathsf{LPO} \not\leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$. To do so, we introduce the notion of *promptly connected* graph. Recall that, for a graph G and $V \subseteq V(G)$ the graph induced by V on G, denoted by $G_{\restriction V}$ is such that $V(G_{\restriction V}) := V$ and $E(G_{\restriction V}) := E(G) \cap (V \times V)$.

Definition III.4.31. A graph G is *promptly connected* if for every n, the graph $G_{\upharpoonright V \cap \{0,\dots,n\}}$ is connected.

Proposition III.4.32. *Let G be a promptly connected graph and let $v_0 := \min\{v : v \in V(G)\}$. Then, for every $v \in V(G) \setminus \{v_0\}$, m and v are "increasingly connected", i.e.,*

$$(\exists \sigma)(\forall i < |\sigma| - 1)(\sigma(0) = v_0 \wedge \sigma(|\sigma| - 1) = v \wedge (\sigma(i), \sigma(i+1)) \in E(G) \wedge \sigma(i) < \sigma(i+1)).$$

Proof. Let $\{v_i : i \in \mathbb{N} \wedge (\forall i)(v_i < v_{i+1})\}$ be an increasing enumeration of $V(G)$. The proof goes by an easy induction on the v_i's. The case for v_1 holds trivially: indeed, $v_0 < v_1$ and, by definition of promptly connected graph, $(v_0, v_1) \in E(G)$. Suppose that v_0 and v_s for $s > 1$ are increasingly connected: we show that $v_0, \dots, v_{s+1}$ are increasingly connected as well. By definition of promptly connected graph, $G_{\upharpoonright V \cap \{0,\dots,v_{s+1}\}}$ is a connected graph, and hence $(\exists i < s+1)((v_i, v_{s+1}) \in E(G))$. By inductive hypothesis, v_0 and v_i are increasingly connected by some path σ and hence, since $v_i < v_{s+1}$, the path $\sigma^\frown v_{s+1}$ witnesses that v_0 and v_{s+1} are increasingly connected. ☐

Lemma III.4.33. *There exists a computable function $\mathbf{PC} :\subseteq \mathbf{Gr} \to \mathbf{Gr}$ that, given in input a connected graph $G \in \mathbf{Gr}$, is such that $\mathbf{PC}(G) \cong G$, $\mathbf{PC}(G)$ is promptly connected and $0 \in V(\mathbf{PC}(G))$.*

Proof. Given a name p for $G \in \mathbf{Gr}$, the function $\mathbf{PC}$ computes a name for a graph in $\mathbf{Gr}$ as follows. We define an auxiliary map $\iota : V(G) \to \mathbb{N}$ that arranges the vertices of G so that $\mathbf{PC}(G)$ satisfies the properties of the lemma. Let $(v_i)_{i \in \mathbb{N}}$ be an enumeration of $V(G)$. At stage 0, let $\iota_0(v_0) = 0$. At stage $s+1$, let $n_{s+1} := \max\{n : n \in \mathsf{range}(\iota) \text{ at stage } s\}$ and, if $(\exists i < s+1)((v_i, v_{s+1}) \in E(G))$, let $\iota(v_{s+1}) = n_{s+1} + 1$. Otherwise, wait for a stage $t_{s+1} := \min\{t : v_0 \leftrightsquigarrow^G v_{s+1} \text{ at stage } t\}$ and let σ_{s+1} be the path connecting v_0 and v_{s+1} in G at stage t_{s+1}. For every $i < |\sigma|$, if $\sigma(i) \notin \operatorname{dom}(\iota)$ at stage s let $\iota(\sigma(i)) := n_{s+1} + i$. This ends the construction.

Let $\mathbf{PC}(G)$ be the graph having $\{\iota(v_i) : v_i \in V(G)\}$ and $\{(\iota(v_i), \iota(v_j)) : (v_i, v_j) \in E(G)\}$ as vertex set and edge set respectively. It is straightforward to verify that $\mathbf{PC}(G)$ satisfies the properties of the lemma. ☐

The following lemma gives us a useful property of promptly connected graphs. First, given a connected graph G we define the *distance* between $v, w \in V(G)$ as

$$d^G(v, w) := \min\{n : (\exists \sigma \in \mathbb{N}^n)(\forall i < |\sigma| - 1)(\sigma(0) = v \wedge \sigma(|\sigma| - 1) = w \wedge (\sigma(i), \sigma(i+1)) \in E(G))\}.$$

Lemma III.4.34. *Suppose G is a promptly connected graph such that $(\forall v \in V(G))(\deg^G(v) < \aleph_0)$ and $R_\omega \subseteq_{\mathbf{s}} G$. then,*

$$(\exists R \cong R_\omega)(R \subseteq G \wedge (\forall v, w \in V(R))(d^G(0, v) < d^G(0, w) \implies v < w)),$$

Proof. Consider the tree:

$$T := \{\sigma : \sigma(0) = \min\{v : v \in V(G)\} \wedge (\forall i < |\sigma| - 1)((\sigma(i), \sigma(i+1)) \in E(G) \wedge \sigma(i) < \sigma(i+1))\}.$$

To prove the lemma, it suffices to show that $[T] \neq \emptyset$. Indeed, suppose $p \in [T]$: then the graph having $\{p(i) : i \in \mathbb{N}\}$ and $\{(p(i), p(i+1)) : i \in \mathbb{N}\}$ as vertex set and edge set respectively is the copy $R \cong R_\omega$ contained in G we were looking for.

To show that $[T] \neq \emptyset$, first notice that, by hypothesis, $(\forall v \in V(G))(\deg^G(v) < \aleph_0)$ and in particular $(\forall \sigma \in T)(\forall i < |\sigma|)(|\{v : (\sigma(i), v) \in E(G)\}| < \aleph_0)$, implying that T is finitely branching. Hence, by König lemma (i.e. every infinite finitely branching tree has an infinite path), it suffices to show that T is infinite. By Proposition $III.4.32$, for every $v \in V(G)\setminus \min\{v : v \in V(G)\}$, $\min\{v : v \in V(G)\}$ and v are increasingly connected, and hence, for any $v \in V(G)$, there exists $\sigma \in T$ such that $\sigma(|\sigma| - 1) = v$ and since G is infinite, T is infinite as well, and this concludes the proof. □

We are now ready to show that $\mathsf{LPO} \nleqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$.

Proposition III.4.35. $\mathsf{LPO} \nleqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$ *relative to any oracle* μ.

Proof. Let Φ, Ψ be the computable (relative to μ) forward and backward functionals witnessing that $\mathsf{LPO} \leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$. Without loss of generality, we can restrict the domain of LPO to $A := \{0^{\mathbb{N}}\} \cup \{0^i 10^{\mathbb{N}} : i \in \mathbb{N}\}$: doing so, we ensure that Φ produces only countably many graphs. Let $\mathbf{PC}$ be the function defined in Lemma $III.4.33$, and compute the graphs $G_\infty := \mathbf{PC}(\Phi(0^{\mathbb{N}}))$ and, for every $i \in \mathbb{N}$, $G_i := \mathbf{PC}(\Phi(0^i 10^{\mathbb{N}}))$. Notice that, by Proposition $III.4.32$ and Lemma $III.4.34$, for every $x \in \mathbb{N} \cup \{\infty\}$, $0 \in V(G_x)$ and, for any $v \in V(G_x)$, 0 and v are increasingly connected. The rest of the proof shows that there is a $\lambda : \mathbb{N} \to \mathbb{N}$ such that for every $x \in \mathbb{N} \cup \{\infty\}$ there exists $R \cong R_\omega$ in G_x (with $V(R) = \{v_i : i \in \mathbb{N}\}$, $E(R) := \{(v_i, v_{i+1}) : i \in \mathbb{N}\}$ and $v_0 = 0$) with the property that, for every $n \in \mathbb{N}$, $v_n \leqslant \lambda(v_{n+1})$. The existence of such a λ would witness that $\mathsf{LPO} \leqslant_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}}$ relative to $\mu \oplus \lambda$, giving rise to a contradiction. Indeed, given an input $p \in A$ for (the restricted version of) LPO, via the $\mathbf{PC}$ defined in Lemma $III.4.33$ we can compute $\mathbf{PC}(\Phi(p))$ being promptly connected. Then, consider the graph G such that $V(G) := V(\mathbf{PC}(\Phi(p)))$ and $E(G) := \{(v, w) : v, w \in V(\mathbf{PC}(\Phi(p))) \wedge \max\{v, w\} < \lambda(\min\{v, w\})\}$. Clearly, G satisfies the conditions of Lemma $III.4.34$. Hence, compute the tree T of the proof of Lemma $III.4.34$ (a valid input for $\mathsf{C}_{2^{\mathbb{N}}}$) and, given $f \in \mathsf{C}_{2^{\mathbb{N}}}([T])$ (as we have done in the same proof) we compute a δ_{Gr}-name q for some $R \cong R_\omega$ in G. Then, $\Psi(p, R)$ is a correct answer for LPO, showing that $\mathsf{LPO} \leqslant_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}}$ relative to the oracle $\lambda \oplus \mu$, obtaining the desired contradiction.

We define λ as follows. Let p_∞ be the name for $G_\infty \in \mathbf{Gr}$ and, for every $i \in \mathbb{N}$ let p_i be the name for $G_i \in \mathbf{Gr}$. Let $R \in \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}(G_\infty)$ with name r, and let v be such that $\deg^R(v) = 1$. Since R is a solution for $\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}(G_\infty)$, there exists a stage ℓ such that $v < \ell$ and $\Psi(0^\ell, \delta_{Gr}(r[\ell]0^{\mathbb{N}})) \downarrow = 1$. Without loss of generality, since Φ is computable and in particular continuous, we can assume that for every $k \geqslant \ell$, $\delta_{Gr}(p_k[\ell]0^{\mathbb{N}}) \cong \delta_{Gr}(p_\infty[\ell]0^{\mathbb{N}})$. Notice that, for the reduction to work correctly, it must be the case that,

$$(\forall k \geqslant \ell)(\forall S \cong R_\omega)(S \text{ is a subgraph of } G_k \implies \Psi(0^{\mathbb{N}}, R)\downarrow \neq \Psi(0^k 10^{\mathbb{N}}, S)\downarrow). \tag{7}$$

We claim that (7) implies

$$(\forall k \geqslant \ell)(\forall S \cong R_\omega)(\exists u \neq v < \ell)((S \text{ is a subgraph of } G_k \wedge \deg^S(v) = 1) \implies u \in V(S)).$$

Suppose not: then, for $k \geqslant \ell$, let s' be a name for $S' \in \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}(G_k)$ be such that $\deg^S(v) = 1$ and suppose that for every $u \neq v < \ell$, $u \notin V(R)$, i.e. vertices after v in $V(S')$ are greater than ℓ. Notice that $R[\ell] \odot S' \in \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}(G_k)$ (where the connection involves the vertex v) and let s'' be a δ_{Gr}-name for such a solution. Clearly, $\delta_{Gr}(s''[\ell]0^{\mathbb{N}}) \cong \delta_{Gr}(r[\ell]0^{\mathbb{N}})$ and so, by hypothesis, $\Psi(0^\ell, s''[l]) \downarrow = 1$, a contradiction. This concludes the proof of the claim.

Furthermore, since G_k is promptly connected, by Proposition $III.4.32$ 0 and v are increasingly connected. Then, given

$$a_k := \min\{w : (\forall i < k)(\exists R \cong R_\omega)(R \subseteq G_i \wedge 0, v, w \in V(R) \wedge d^R(0, v) < d^R(0, w))\}.$$

let $\lambda(v) := \max\{\ell, a_k\}$: the definition of $\lambda(v)$ ensures that in every G_x, for $x \in \mathbb{N} \cup \{\infty\}$, we can find a copy R of R_ω such that $\deg^R(0) = 1$, passes through v and "continues" with some $w \leqslant \lambda(v)$. This concludes the proof. □

So far we have shown that $\mathsf{LPO} \nleqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{a}_{R_\omega}$ for $a \in \{\chi e, ee, \chi\chi\}$. The next proposition shows that instead, $\mathsf{LPO} <_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}$.

Proposition III.4.36. $\mathsf{LPO} <_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}$.

Proof. For the reduction, let $p \in 2^{\mathbb{N}}$ be an input for LPO: we compute a tree $T \in \mathbf{Tr}_2$ as follows. At stage s, if $p(s) = 0$, let $0^s \in T$. If $p(s) = 1$, we stop inspecting p at later stages and for every t, let $1^t \in T$. It is clear that if $p = 0^{\mathbb{N}}$ then $[T] = \{0^{\mathbb{N}}\}$, while if $(\exists i)(p(i) = 1)$, $[T] = \{1^{\mathbb{N}}\}$. The fact that $T \in \mathcal{IF}_2$ implies that $T \in \mathrm{dom}(\mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega})$. Let $R \in \mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}(T)$, and notice that by definition of T there exists an s such that for all $t > s$, either $0^t \in V(R)$ and $1^t \notin V(R)$ or vice versa. Hence, it suffices to search for such an s, and in the first case $\mathsf{LPO}(p) = 1$ while in the second one $\mathsf{LPO}(p) = 0$.

The fact that $\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega} \nleqslant_{\mathrm{W}} \mathsf{LPO}$ is straightforward and this concludes the proof. □

Combining Theorem *III*.4.24 and Propositions *III*.4.35 and *III*.4.36 we obtain the following corollary.

Corollary III.4.37. $\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$, $\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{ee}_{R_\omega} <_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}$.

Despite Proposition *III*.4.36 we are still able to prove that $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \nleqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}$: indeed, the next theorem concludes the proof of Theorem *III*.4.18.

Proposition III.4.38. $\mathsf{lim}' \nleqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$ *and* $\mathsf{lim}'' \nleqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}$.

Proof. For the first nonreduction, assume for the sake of contradiction, that $\mathsf{lim}' \leqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$. By Lemma *III*.4.23(*ii*), $\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{G} \leqslant_{\mathrm{W}} (\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{G} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N})'$. By Theorem *I*.6.22, if $\mathsf{lim}' \leqslant_{\mathrm{W}} (\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{G} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N})'$ then $\mathsf{lim} \leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{G} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N}$ relative to $\varnothing'$. If so, Lemma *III*.4.22 implies that $\mathsf{LPO} \leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{G}$, contradicting Proposition *III*.4.35.

For the second nonreduction, assume again for the sake of contradiction, that $\mathsf{lim}'' \leqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega}$. By [BG09, Lemma 6.3], we get that computing the characteristic function of a set from its enumeration is Weihrauch equivalent to lim. Hence, $\mathsf{S}\text{-}\mathsf{Copy}^{e\chi}_{R_\omega} \leqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}{}'$ and from this we obtain that $\mathsf{lim}'' \leqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}{}'$. Theorem *I*.6.22 implies that $\mathsf{lim}' \leqslant_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$ relative to $\varnothing'$ and combined with the fact that $\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega} \leqslant_{\mathrm{W}} (\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{G} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N})'$ we would obtain $\mathsf{lim}' \leqslant_{\mathrm{W}} (\mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{G} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N})'$. Applying again Theorem *I*.6.22 we would finally get that $\mathsf{lim} \leqslant_{\mathrm{W}} \mathbf{Con}\text{-}\mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{G} * \mathbf{\Sigma}^1_1\text{-}\mathsf{C}_\mathbb{N}$ relative to $\varnothing''$, contradicting Proposition *III*.4.35. □

Notice that the strongest result we have for non first-order problems is $\mathsf{C}_{2^\mathbb{N}} <_{\mathrm{W}} \mathsf{S}\text{-}\mathsf{Copy}^{\chi\chi}_{R_\omega}$. It is not clear to us what $\mathsf{S}\text{-}\mathsf{Copy}_{R_\omega}$ compute.

The subgraph problem: when G has only finite components

Now we deal with the problems $\mathsf{S}\text{-}\mathsf{Copy}_G$ where $G := \bigotimes_{i \in \mathbb{N}} F_i$, with F_i a finite graph. For these problems, it seems to be harder to reach the equivalence with $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$: indeed, we have already discussed that $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$ can be stated as the task of finding a path through an ill-founded tree, and graphs of this kind are far from being intuitively "ill-founded", i.e. from having R_ω as a subgraph. This intuition is actually wrong: even if for many graphs G the problems $\mathsf{S}\text{-}\mathsf{Copy}_G$ are computable, for others we have that $\mathsf{S\text{-}Copy}_G \equiv_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$. The important distinction is the following. Given $G := \bigotimes_{i\in\mathbb{N}} F_i$ we distinguish whether

$$(\forall^\infty i)(\exists^\infty j)(F_i \subseteq_{\mathbf{s}} F_j). \tag{8}$$

Theorem III.4.39. *Let G be an infinite computable graph such that $G = \bigotimes_{i\in\mathbb{N}} F_i$, where F_i is a finite graph and $(\forall^\infty i)(\exists^\infty j)(F_i \subseteq_{\mathbf{s}} F_j)$. Then the problems $\mathsf{S\text{-}Copy}_G$ are computable.*

Proof. By Figure III.3, it suffices to show that $\mathsf{S\text{-}Copy}_G^{e\chi}$ is computable. By definition there exists $k \in \mathbb{N}$ such that there are k-many graphs $F_{n_0}, \dots, F_{n_{k-1}}$ that are subgraphs of just finitely many F_i's. Let $A := \{F_i : (\exists l < k)(F_{n_l} \subseteq_{\mathbf{s}} F_i)\}$ and notice that $|A| < \aleph_0$. Given $H \in \operatorname{dom}(\mathsf{IS\text{-}Copy}_G^{e\chi})$ in input, we compute G' being a subgraph of H such that $G' \cong G$ with the two following procedures that can be performed in parallel.

- The first procedure waits for a finite stage witnessing that $(\forall F \in A)(F \subseteq_{\mathbf{s}} H)$ and adds to G' the corresponding copy of F in H. Since all F's are finite and A is finite as well, such an s exists, and we can computably find it.
- The second procedure takes care of all the $F_s \in \{F_i : i \in \mathbb{N}\}\setminus A$. For every $s > k$, it adds to G' the first copy of F_s in H that it finds. We claim that this procedure eventually adds to G' a copy for every F_s. Recall that, by hypothesis, $(\forall s > k)(\exists^\infty j)(F_i \subseteq_{\mathbf{s}} F_j)$. Suppose that at stage s there exists an $m > k$ such that F_m has not been added to G' yet. Since $(\exists^\infty j)(F_m \subseteq_{\mathbf{s}} F_j)$, and we have seen only a finite portion of H, we can wait for a finite stage greater than s such that F'_m is a subgraph of H, $F'_m \cong F_m$ and F'_m has not been added to G' yet: hence we can add F'_m to G'. This concludes the proof of the claim.

This completes the proof. □

The graph $\bigotimes_{i\geqslant 3} C_i$ does not satisfy (8): the next theorem shows that $\mathsf{IS\text{-}Copy}^{\chi e}_{\bigotimes_{i\geqslant 3} C_i} \equiv_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$.

Theorem III.4.40. $\mathsf{IS\text{-}Copy}_{\bigotimes_{i\geqslant 3} C_i} \equiv_W \mathsf{C}_{\mathbb{N}^\mathbb{N}}$.

Proof. The left-to-right direction is Proposition $III.4.3$. For the opposite direction, by Figure III.3, it suffices to show that $\mathsf{C}_{\mathbb{N}^\mathbb{N}} \leqslant_W \mathsf{S\text{-}Copy}_G^{e\chi}$. We begin partitioning the C_i's in three infinite disjoint sets: i.e., $\{C_i : i \geqslant 3\} = \{P_n : n \in \mathbb{N}\} \cup \{F_n : n \in \mathbb{N}\} \cup \{G_n : n \in \mathbb{N}\}$. Let $T \in \mathbf{Tr}$ be an input for $\mathsf{C}_{\mathbb{N}^\mathbb{N}}$: we compute a graph $H \in \operatorname{dom}(\mathsf{IS\text{-}Copy}^{\chi e}_{\bigotimes_{i\geqslant 3} C_i})$ in stages as follows. First, for every $n \in \mathbb{N}$, we add to H

- infinitely many copies of P_n, that we denote by $\{P^i_n : i \in \mathbb{N} \wedge P^i_n \cong P_n\}$,
- two copies of G_n denoted by G^0_n, G^1_n, and
- a copy of F_n.

For every $i, n \in \mathbb{N}$, we associate to P^i_n a *box* containing the graphs

$$P^i_n \odot G^1_{\langle n,i,0\rangle} \text{ and, in a disjoint fashion, } \bigotimes_{k\in\mathbb{N}} \big(G^0_{\langle n,i,k\rangle} \odot G^1_{\langle n,i,k+1\rangle}\big).$$

Informally, P^i_n's *box* contains infinitely many disjoint graphs, each of which is obtained by the connected union of two cyclic graphs.

Notice that for any $k > 0$, $G^0_{\langle n,i,k\rangle}$ has a designed docking vertex (different from the ones involved in $G^1_{\langle n,i,k\rangle} \odot G^0_{\langle n,i,k+1\rangle}$) to which, in later stages, we may attach a graph from $\{F_n :$

$n \in \mathbb{N}\}$, i.e. $G^1_{\langle n,i,k\rangle} \bigodot G^0_{\langle n,i,k+1\rangle}$ may become $G^1_{\langle n,i,k\rangle} \bigodot G^0_{\langle n,i,k+1\rangle} \bigodot F_n$. We say that $G^0_{\langle n,i,k\rangle}$ is *free* if no F_n is attached to the designed docking vertex.

Without loss of generality, we assume that $|\{\sigma : \sigma \notin T\}| = \aleph_0$ is infinite: indeed, if such a set is finite, instead of considering T, we consider the tree $T' := \langle\rangle \cup \{1\tau : \tau \in T\}$: clearly, $\{\sigma : \sigma \notin T'\}$ is infinite and from T' we can easily compute T. Let $(\sigma_s)_{s\in\mathbb{N}}$ be a computable (with respect to T) enumeration of $\{\sigma : \sigma \notin T\}$. At stage σ_s, let $k := \min\{j : G^0_{\langle n,i,j\rangle}$ is free$\}$: for every $n < |\sigma_s|$ attach F_s to $G^0_{\langle n,i,k\rangle}$ if and only if $\sigma_s(n) = i$ (recall that $G^0_{\langle n,i,j\rangle}$ is only in P^i_n's box). This ends the construction.

We first claim that $G \subseteq_{\mathbf{s}} H$. Let $q \in [T]$, and consider the following copy of G in H. For every $n \in \mathbb{N}$, consider the graph G' containing,

- $P^{q(n)}_n$ and,
  - if $P^i_n \in G'$ then, for every $k \in \mathbb{N}$, we add to G' the graphs $G^0_{\langle n,i,k\rangle}$,
  - if $P^i_n \notin G'$ then, for every $k \in \mathbb{N}$, we add to G' the graphs $G^1_{\langle n,i,k\rangle}$ (this choice allow us to put in G', if needed, a copy of F_m contained in P^i_n'box).
- the copy of F_n belonging to $P^{\sigma_n(|\sigma_n|-1)}_{|\sigma_n|-1}$ box: since $\sigma_n \notin T$, $P^{\sigma_n(|\sigma_n|-1)}_{|\sigma_n|} \notin G'$, hence by the previous point we can choose the copy of F_n in this box.

Hence, for every n, we added in G' a copy of P_n, G_n, and F_n and this concludes the proof of the claim.

To conclude the proof we need to show that from any $G' \in \mathsf{IS\text{-}Copy}^{\chi_e}_G(H)$ we can compute some $q \in [T]$. First, notice the following useful fact. Suppose that $P^i_n \in G'$ and recall that, for every x, G_x has only two copies in H, namely G^0_x and G^1_x. Since $P^i_n \in G'$ and P^i_n shares a vertex with $G^1_{\langle n,i,0\rangle}$ we are forced to add in G' the copy $G^0_{\langle n,i,0\rangle}$. Similarly, for any $m \in \mathbb{N}$, $G^0_{\langle n,i,m\rangle}$ shares a vertex with $G^1_{\langle n,i,m+1\rangle}$: the fact that $G^0_{\langle n,i,0\rangle}$ in G' forces us, for every $m \in \mathbb{N}$, to add $G^0_{\langle n,i,m\rangle}$ in G'. Another important consequence of this observation is that if $P^i_n \in G'$ we cannot put in G' any copy of F_s from P^i_n's box, as F_s shares a vertex with $G^0_{\langle n,i,t\rangle}$ for some $t \in \mathbb{N}$, and we have just argued that $G^0_{\langle n,i,t\rangle}$ is in G. So let $(P^i_n)_{i,n\in\mathbb{N}}$ be the copies of P_n in G': we claim that there exists a $q \in [T]$ such that for every $n, i \in \mathbb{N}$, $q(n) = i$. Suppose not. This means that

$$(\exists \tau \in \mathbb{N}^{<\mathbb{N}})(\forall m < |\tau|)(\tau[|\tau| - 2] \in T \land \tau \notin T \land P^{\tau(m)}_m \in G').$$

In other words, $\tau = \sigma_s$ for some s (where $(\sigma_s)_{s\in\mathbb{N}}$ is the computable enumeration of $\{\sigma : \sigma \notin T\}$). By construction, the only copies of F_s are in $P^{\tau(m)}_m$'s box for $m < |\tau|$: since $P^{\tau(m)}_m$ are all in G', from the observation above, we cannot put any copy of F_s in G', contradicting that $G' \in \mathsf{IS\text{-}Copy}^{\chi_e}_G(H)$, and this concludes the proof. □

Notice that this construction heavily relies on the fact that the connected union of at most three connected and finite components of G is not isomorphic to any connected component of G. This means that the theorem above holds for any graph having this property.

III.4.4 Other subgraph problems

We have shown examples of graphs G such that $\mathsf{S\text{-}Copy}_G \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^\mathbb{N}}$, graphs G such that the problems $\mathsf{S\text{-}Copy}_G$ are computable and others that are difficult to compute but weak when they have to compute other problems. In this section, we show that there are also graphs G such that $\mathsf{S\text{-}Copy}_G$ occupy well known areas of the Weihrauch lattice. Before doing so, we give a convenient characterization of $\mathsf{lim}^{(n)}$ for every $n \in \mathbb{N}$. Recall that for a represented space $\mathbf{X}$, that $\mathcal{O}(\mathbf{X})$ is the final topology on X induced by δ_X (see Definition I.2.13)

Definition III.4.41. Let $\mathrm{EnumInf}_{\Pi_n} :\subseteq \mathbf{\Pi}^0_n(\mathbb{N}) \rightrightarrows \mathcal{O}(\mathbb{N})$ be defined by $U \in \mathrm{EnumInf}(A)$ if and only if $U \subseteq A \wedge |U| = \aleph_0$.

Lemma III.4.42. *For $n > 0$, $\mathrm{EnumInf}_{\Pi_n} \equiv_{\mathrm{W}} \mathsf{lim}^{(n-1)}$.*

Proof. Notice that $\widehat{\mathsf{lim}^{(n-1)}} \equiv_{\mathrm{W}} (\mathsf{id} : \mathbf{\Pi}^0_n(\mathbb{N}) \to 2^{\mathbb{N}})$, i.e. the function computing the characteristic function of a $\mathbf{\Pi}^0_n$ set. Hence, we show $\mathrm{EnumInf}_{\Pi_n} \equiv_{\mathrm{W}} (\mathsf{id} : \mathbf{\Pi}^0_n(\mathbb{N}) \to 2^{\mathbb{N}})$.

The left-to-right direction is trivial and if $n = 0$, both sides are trivially computable.

For the opposite direction, we need to show that from a $\mathbf{\Pi}^0_n$-name of a set $A \subseteq \mathbb{N}$ we can compute a $\mathbf{\Pi}^0_n$-name of an infinite set $B \subseteq \mathbb{N}$, such that any enumeration of an infinite subset of B allows us to recover the characteristic function of A.

The given $\mathbf{\Pi}^0_n$-name A brings with it a sequence $(C_i)_{i\in\mathbb{N}}$ of $\mathbf{\Pi}^0_{n-1}$-sets with $\mathbb{N}\backslash A = \bigcup_{i\in\mathbb{N}} C_i$. For $n \in \mathbb{N}$, we let $\lambda_A(n) = 0$ if $n \in A$, and $\lambda_A(n) = \min\{i \mid n \in C_i\} + 1$ if $n \notin A$. Let $p_0, p_1, \dots$ be the increasing enumeration of the prime numbers. We now define:

$$B = \{\prod_{i\leqslant k} p_i^{\lambda_A(i)} \mid k \in \mathbb{N}\}$$

The set B has the desired properties: we can obtain its $\mathbf{\Pi}^0_n$-name from the name of A; the set B is infinite, and an enumeration of any infinite subset of B allows us to recover B, and then subsequently A. □

The graphs we promised at the beginning of this section are T_{2k+1} and F_{2k+2}, defined in §III.3. Before stating the main theorem of this section, we give a preparatory result.

Lemma III.4.43. *The problems $\mathsf{S\text{-}Copy}_{\mathsf{T}_1}$ and $\mathsf{S\text{-}Copy}_{\mathsf{F}_2}$ are computable and $\mathsf{S\text{-}Copy}_{\mathsf{T}_3} \equiv_{\mathrm{W}} \mathbf{\Pi}^0_2\text{-}\mathsf{C}_{\mathbb{N}}$.*

Proof. The fact that $\mathsf{S\text{-}Copy}_{\mathsf{T}_1}$ and $\mathsf{S\text{-}Copy}_{\mathsf{F}_2}$ are computable is straightforward, regardless of whether graphs are given as input/output as elements of **Gr** or **EGr**: in the first case, given $H \in \mathrm{dom}(\mathsf{S\text{-}Copy}_{\mathsf{T}_1})$, it suffices to wait for a vertex v to appear in $V(H)$: it is clear that $(\{v\}, \{\varnothing\}) \in \mathsf{S\text{-}Copy}_{\mathsf{T}_1}(H)$. Similarly, given $H \in \mathrm{dom}(\mathsf{S\text{-}Copy}_{\mathsf{F}_2})$, we obtain that $(V(H), \{\varnothing\}) \in \mathsf{S\text{-}Copy}_{\mathsf{F}_2}(H)$.

To show that $\mathsf{S\text{-}Copy}_{\mathsf{T}_3} \leqslant_{\mathrm{W}} \mathbf{\Pi}^0_2\text{-}\mathsf{C}_{\mathbb{N}}$, by Figure III.3, it suffices to show that $\mathsf{S\text{-}Copy}^{e\chi}_{\mathsf{T}_3} \leqslant_{\mathrm{W}} \mathbf{\Pi}^0_2\text{-}\mathsf{C}_{\mathbb{N}}$. Given in input $H \in \mathrm{dom}(\mathsf{S\text{-}Copy}^{e\chi}_{\mathsf{T}_3})$, let $A := \{v \in V(H) : \deg^H(v) = \aleph_0\}$. It is clear that $A \in \mathbf{\Pi}^0_2(\mathbb{N})$ and A is nonempty. Then, given $v_0 \in \mathbf{\Pi}^0_2\text{-}\mathsf{C}_{\mathbb{N}}$ we compute the graph G' such that $V(G') = \{v_0\} \cup \{v : (v_0, v) \in E(H)\}$ and $E(G') := \{(v_0, v) : (v_0, v) \in E(H)\}$.

For the converse, by Figure III.3, it suffices to show that $\mathbf{\Pi}^0_2\text{-}\mathsf{C}_{\mathbb{N}} \leqslant_{\mathrm{W}} \mathsf{S\text{-}Copy}^{\chi e}_{\mathsf{T}_3}$. An input for $\mathbf{\Pi}^0_2\text{-}\mathsf{C}_{\mathbb{N}}$ is a nonempty set $A \in \mathbf{\Pi}^0_2(\mathbb{N})$. By Lemma $I.3.35$, we can associate to every $n \in \mathbb{N}$ an infinite sequence $p_n \in 2^{\mathbb{N}}$ such that $n \in \mathbb{N} \iff (\exists^\infty i)(p_n(i) = 1)$. Let H be the graph such that $V(H) := \mathbb{N}$ and $E(H) := \{(\langle n, 0\rangle, \langle n, i+1\rangle) : p_n(i) = 1\}$. Since $A \neq \varnothing$ by hypothesis, we have that $(\exists m)(\exists^\infty i)(p_m(i) = 1)$ and hence $\deg^H(\langle m, 0\rangle) = \aleph_0$, i.e. $\mathsf{T}_3 \subseteq_{\mathbf{s}} H$: this show that H is a suitable input for $\mathsf{S\text{-}Copy}^{\chi e}_{\mathsf{T}_3}$. It is also clear that given $G' \in \mathsf{S\text{-}Copy}^{\chi e}_{H'}$ we can compute $n \in A$ (just look at the projection on the first coordinate of a vertex in $V(G')$). □

We conclude this section with Theorem $III.4.44$.

Theorem III.4.44. *For $k > 0$,*

$$(i)\ \mathsf{S\text{-}Copy}_{\mathsf{T}_{2k+1}} \leqslant_{\mathrm{W}} \mathbf{\Pi}^0_{2k}\text{-}\mathsf{C}_{\mathbb{N}} \times \widehat{\mathsf{LPO}^{(2k-3)}} \textit{ and } (ii)\ \mathsf{S\text{-}Copy}_{\mathsf{F}_{2k+2}} \equiv_{\mathrm{W}} \widehat{\mathsf{LPO}^{(2k-1)}}.$$

Proof. We only prove (ii) as the proof of (i) is similar to the left-to-right direction of (ii).

For the left-to-right direction, by Figure III.3, it suffices to show that $\mathsf{S\text{-}Copy}^{e\chi}_{\mathsf{F}_{2k+2}} \leqslant_{\mathrm{W}} \widehat{\mathsf{LPO}^{(2k-1)}}$. Since, for any n, $\widehat{\mathsf{LPO}^{(n)}}$ is clearly closed under parallelization, it suffices to show that

$$\mathsf{S\text{-}Copy}^{e\chi}_{\mathsf{F}_{2k+2}} \leqslant_{\mathrm{W}} \bigtimes_{0<i\leqslant 2k-1} \widehat{\mathsf{LPO}^{(i)}}.$$

For every i such that $0 < i \leqslant 2k-1$, and for every $v \in V(H)$ we can uniformly compute a sequence $(p_v)_{v\in V(H)}$ such that $\mathsf{LPO}^{(i)}(p_v) = 1$ if and only if

$$(\exists^\infty v_0 \in V(H)) \dots (\exists^\infty v_j \in V(H))(\forall j < i)((v, v_0) \in E(H) \wedge (v_j, v_{j+1}) \in E(H)).$$

Now for every i such that $0 < i \leqslant 2k-1$, let $(v^m_i)_{m\in\mathbb{N}}$ be an enumeration of $\{v \in V(H) : \mathsf{LPO}^{(i)}(p_v) = 1\}$. Following the same ideas of Lemma *III*.3.12' proof we can compute a δ_{Gr}-name for a copy of F_{2k+2} in H.

For the converse direction, notice that, by Lemma *III*.4.42, $\widehat{\mathsf{LPO}^{(2k-1)}} \equiv_{\mathrm{W}} \mathsf{lim}^{(2k-1)} \equiv_{\mathrm{W}} \mathrm{EnumInf}_{\Pi_{2k}}$: hence, by Figure III.3, it suffices to show that $\mathrm{EnumInf}_{\Pi_{2k}} \leqslant_{\mathrm{W}} \mathsf{S\text{-}Copy}^{\chi e}_{\mathsf{F}_{2k+2}}$. Recall from §III.3 that given a subspace $\mathcal{G} \subseteq (\mathbf{E})\mathbf{Gr}$, we define $\bigotimes \mathbf{Tr}_{\leqslant k} := \{\bigotimes_{n\in\mathbb{N}} G_n : (\forall n)(G_n \in \mathbf{Tr}_{\leqslant k})\}$, where $\mathbf{Tr}_{\leqslant k}$ is the space of trees with height at most k. Lemma *III*.3.13 tells us that $Forests_{2k} := \{G \in \bigotimes \mathbf{Tr}_{\leqslant k} : \mathsf{F}_{2k} \subseteq_{\mathbf{s}} G\}$ is Π^0_{2k}-complete: this allows us to think of an input A for $\mathrm{EnumInf}_{\Pi_{2k}}$ as a sequence $(F_n)_{n\in\mathbb{N}}$ such that $n \in A \iff F_n \in Forests_{2k}$. Since by Definition *III*.4.41 A is infinite, $\bigotimes_{n\in\mathbb{N}} F_n$ is a suitable input for $\mathsf{S\text{-}Copy}_{\mathsf{F}_{2k+2}}$ and from any $G \in \mathsf{S\text{-}Copy}^{\chi e}_{\bigotimes_{n\in\mathbb{N}} F_n}$, checking the projection on the first coordinate of the nodes in G we can compute an infinite subset of A. □

We conjecture that the reduction in Theorem *III*.4.44(ii) is actually an equivalence, but the details of the proof still need to be adjusted. We leave open whether there exist a graph G such that $\mathsf{S\text{-}Copy}_G \equiv_{\mathrm{W}} f$ for some $f \notin \{\mathsf{C}_{\mathbb{N}^\mathbb{N}}, \mathsf{id}, \mathbf{\Pi}^0_{2k}\text{-}\mathsf{C}_{\mathbb{N}} \times \widehat{\mathsf{LPO}^{(k)}}, \widehat{\mathsf{LPO}^{(k)}}\}$ for $k \in \mathbb{N}$.

III.5 Conclusions and open problems

In this chapter, we investigated the subgraph problem and the induced subgraph problem using tools from (effective) Wadge reducibility and Weihrauch reducibility. We studied the (effective) Wadge complexity of certain subsets of graphs, and we located decision problems and "search" problems in the Weihrauch hierarchy. Regarding Weihrauch reducibility, we solved some questions left open in [BHW21], but the exact (effective) Wadge complexity of some subsets of graphs we studied and the Weihrauch degree of certain problems remains open.

For the induced subgraph relation, in §III.3 we showed that, at least when G is computable, for any graph G, $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{is}} H\} \in \Gamma$ for $\Gamma \in \{\Sigma^0_1, \Sigma^1_1\}$. Theorem *III*.3.10 witnesses that, for the subgraph case, there exists a graph G_k such that $\{H \in (\mathbf{E})\mathbf{Gr} : G_k \subseteq_{\mathbf{s}} H\} \in \Gamma$ for $\Gamma \in \{\Sigma^0_{2k+1}, \Pi^0_{2k+2}, \Sigma^1_1\}$ for $k \in \mathbb{N}$.

Question III.5.1. Is there a computable/c.e. graph G and some $k \in \mathbb{N}$ such that $\{H \in (\mathbf{E})\mathbf{Gr} : G \subseteq_{\mathbf{s}} H\} \in \Gamma$ for $\Gamma \notin \{\Sigma^0_{2k+1}, \Pi^0_{2k+2}, \Sigma^1_1\}$?

In terms of Weihrauch reducibility essentially the same question can be rephrased as follows.

Question III.5.2. Is there a computable/c.e. graph G and some $k > 0$ such that $\mathsf{LPO}^{(k)} <_{\mathrm{W}} \mathsf{S}_G <_{\mathrm{W}} \mathsf{LPO}^{(k+1)}$?

In §III.3.2 we considered the (effective) Wadge complexity of sets of the form $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{(i)s}} G\}$. Propositions *III*.3.24 and *III*.3.25 show that, when G is a finite graph, the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{s}} G\}$ and $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{is}} G\}$ are Π^0_1-complete, and the same holds replacing G with $\bigotimes^{\infty} G$.

Question III.5.3. Understand whether $\{H \in \mathbf{EGr} : H \subseteq_{\mathbf{is}} G\}$ and $\{H \in \mathbf{EGr} : H \subseteq_{\mathbf{is}} \bigotimes^{\infty} G\}$ are complete for some class Γ.

For the question above, we conjecture that they are complete for some class Γ in the lightface difference hierarchy, and this Γ depends on the particular graph G.

So far, $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{is}} \mathcal{S}\}$ are the most complex (in terms of effective Wadge reducibility) sets of graphs of this form we have found.

Question III.5.4. Is there a computable/c.e. graph G such that the sets $\{H \in (\mathbf{E})\mathbf{Gr} : H \subseteq_{\mathbf{(i)s}} G\}$ are Γ-hard (or Γ-complete) where Γ is more complex than Π^0_5?. Or, in terms of Weihrauch reducibility, is there a computable/c.e. graph G such that $\mathsf{LPO}^{(4)} <_{\mathrm{W}} \mathsf{IS}_G$?

Regarding "search" problems, Theorem *III*.4.5 shows that $\mathsf{IS\text{-}Copy}_G \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ but, in case G is such that

$$\{v \in V(G) : \deg^G(v) < \aleph_0\} = \aleph_0 \text{ and } \{v \in V(G) : \deg^G(v) = \aleph_0\} = \aleph_0,$$

the right-to-left reduction of the equivalence holds relative to an oracle.

Question III.5.5. Is it the case that for any computable/c.e. graph G, $\mathsf{IS\text{-}Copy}_G \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ (with no oracle involved)?

We have already mentioned in this chapter's introduction that the problems $\mathsf{S\text{-}Copy}_{R_\omega}$ have the unusual property of being hard to compute, but weak when they have to compute a problem on their own. Theorem *III*.4.18, in particular, shows what $\mathsf{S\text{-}Copy}_{R_\omega}$ cannot compute but, regarding what $\mathsf{S\text{-}Copy}_{R_\omega}$ compute, the only satisfactory result we have is that ${}^1\mathsf{S\text{-}Copy}_{R_\omega} \equiv_{\mathrm{W}} {}^1\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ (Proposition *III*.4.17). For non first-order problems, the best we were able to show is implied by Proposition *III*.4.29 and it shows that $\mathsf{C}_{2^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{S\text{-}Copy}^{\chi\chi}_{R_\omega}$.

Question III.5.6. What can $\mathsf{S\text{-}Copy}_{R_\omega}$ compute?

Corollary *III*.4.28 shows that $\mathrm{Fin}(\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{ee}_{R_\omega}) <_{\mathrm{W}} {}^1\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{ee}_{R_\omega}$: it is natural to ask the following.

Question III.5.7. Does ${}^1\mathbf{Con}\text{-}\mathsf{S\text{-}Copy}^{ee}_{R_\omega} \equiv_{\mathrm{W}} f$ for some well-known f in the Weihrauch lattice?

We conclude this section (and this chapter) with two open questions arising from §III.4.4. For the first one (asking whether the reduction in Theorem *III*.4.44(*ii*) is an equivalence), we conjecture a positive answer, but the details of the proof still need to be adjusted.

Question III.5.8. Does $\mathsf{S\text{-}Copy}_{\mathsf{T}_{2k+1}} \equiv_{\mathrm{W}} \mathbf{\Pi}^0_{2k}\text{-}\mathsf{C}_{\mathbb{N}} \times \widehat{\mathsf{LPO}^{(2k-3)}}$?

Question III.5.9. Is there a computable/c.e. a graph G such that $\mathsf{S\text{-}Copy}_G \equiv_{\mathrm{W}} f$ for some $f \notin \{\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}, \mathsf{id}, \mathbf{\Pi}^0_{2k}\text{-}\mathsf{C}_{\mathbb{N}} \times \widehat{\mathsf{LPO}^{(k)}}, \widehat{\mathsf{LPO}^{(k)}}\}$ for some $k \in \mathbb{N}$?

2

Algorithmic learning theory of algebraic structures

The purpose, the paradigm, and the learning criteria

Algorithmic learning theory, independently discovered by E.M. Gold and Hilary Putnam, is a research program modeling an empirical phenomenon in which a *learner*, given an increasing amount of data about some empirical inquiry, tries to converge to the correct conclusions about it. This subject has implications for the philosophy of science and for the empirical methodology in general but has also many interactions with various mathematical areas from computability and complexity theory, statistics, and combinatorics to more practical fields like machine learning. For a general treatment of algorithmic learning theory, we refer the reader to:

- Jain, Osherson, Royer and Sharma's book "Systems that learn: An introduction to learning theory" for classical results on algorithmic learning theory ([JORS99]), and to
- Harizanov, Goethe and Friend's Chapter "Introduction to the Philosophy and Mathematics of Algorithmic Learning Theory" in "Induction, Algorithmic Learning Theory, and Philosophy" for more topics on algorithmic learning theory, including algorithmic learning of algebraic structures.

We give an informal description of how algorithmic learning theory models the process of searching for an answer to a given empirical inquiry. In this setting, a learner (we can think of it as a Turing machine) is given a learning problem which consists of a set of possible "realities". Step by step the learner sees a *presentation* of a reality and tries to *converge* to the correct solution: depending on the particular case, there may be some *restriction* applied to the learner. The learner solves (i.e. *learns*) the learning problem if it converges to the correct answer whatever the input is. This scenario is often described, as in [JORS99] as a dialogue between *Nature*, that produces a presentation of a reality, and the learner that tries to guess the correct reality outputting at each stage some conjecture.

The framework we just introduced is quite general, and some aspects need to be discussed more in detail. In particular, we focus on:

(*i*) the way in which the set of possible realities is presented to the learner;

(*ii*) the restrictions applied to the learner. Some of them regard the computational strength of the learner while others forbid the learner some "patterns" (in a sense made more precise below) during the learning process;

(*iii*) the exact definition of convergence for a learner.

Algorithmic learning theory can model very different scenarios, but here we are interested in studying the learning process of algebraic structures. On the other hand, we briefly discuss two of the paradigms that have been studied the most by algorithmic learning theorists: learning of computable functions and learning of languages. With the informal presentation of these two paradigms, we also seize the chance to discuss items (*i*)-(*iii*). For computable functions, the learning problem is a collection of total computable functions C. We can formalize a learner as a function $\mathbf{M} : \mathbb{N}^{<\mathbb{N}} \to \mathbb{N}$ that takes as input a computable sequence of natural numbers $f(0), f(1), \ldots$ (the range of some $f \in C$) and outputs $e \in \mathbb{N}$, i.e. the code of φ_e, the e-th total computable function. Regarding item (*iii*), we discuss two different definitions of convergence for a learner, namely *explanatory learning*, denoted by **Ex**, and *behaviorally correct learning*, denoted by **Bc**. In the first one, a learner $\mathbf{M}$ succeeds in learning C, if, in the limit, it stabilizes to some e such that $\varphi_e = f$. In the second one, $\mathbf{M}$ succeeds in learning C if, for all but finitely many stages, it produces indices for correct programs (not necessarily the same index). The fact that, in this context, **Ex**-learnability implies **Bc**-learnability was shown in [CS83].

In learning of languages, a language is represented as a computable set of numbers coding finite strings of words, and the learner tries to output the grammar of the corresponding language. Regarding item (*i*), we introduce two different ways in which data are presented to the learner, namely from *text* (denoted by **Txt**) or from *informant* (denoted by **Inf**). In the presentation from text, the learner receives as input only positive information, i.e. a stream that only enumerates (codes of) the strings that are in the language to be learned. In the presentation from informant, the learner receives as input both positive and negative information, i.e. a list of all (codes of) strings, together with a label indicating whether the string belongs to the language.

We now turn our attention to learning of algebraic structures, firstly considered by Glymour ([Gly85]). More recently, other works highlighted the interplay between different types of learnability of computable or c.e. structures and their algebraic properties. For example, in [HS07], Harizanov and Stephan considered learning of algorithmically generated subspaces of computable vector spaces, in [MS04] Merkle and Stephan studied the learnability of isolated branches on uniformly computable sequences of trees, and in [SV01] Stephan and Ventsov explored learning properties of various algebraic structures. The framework we use in the next sections was introduced by Fokina, Kötzing, and San Mauro in [FKSM19] for learning families of equivalence structures. In [BFSM20] Bazhenov, Fokina, and San Mauro expanded the framework to algebraic structures in general, while in [BFSM20] Bazhenov and San Mauro provide the definition of the framework that is, essentially, the one we are using here and that we redefine below.

We have not discussed yet item (*ii*) in the list above: imposing restrictions on the learnability process allows us (again) to confirm the main theme of this thesis, that is the *comparison* of different learning *problems* with respect to that restriction. For example, if a family $\mathfrak{K}$ is learnable with respect to some restriction R and $\mathfrak{K}'$ is not, we derive that $\mathfrak{K}$ is easier to learn (in the sense of R) than $\mathfrak{K}'$. Learning problems can be compared in different ways: for example, imposing a limited amount of wrong conjectures that a learner can output, or considering its efficiency, measuring the time the learner needs before converging to the correct answers. Another natural restriction one may apply is to the number of *mind changes* that a learner is allowed to do before converging to the correct answer: this is actually the main topic of Chapter V.

The paradigm and the learning criteria

We are now ready to define our paradigm. We ignore how a given family is enumerated, and we just assume that any structure $\mathcal{A}$ gives rise to a corresponding *conjecture* $\ulcorner\mathcal{A}\urcorner$, to be understood as conveying the piece of information "this is $\mathcal{A}$".

Definition. Let $\mathfrak{K}$ be a countable family of nonisomorphic structures. The components of our framework are the following:

- The *learning domain* (LD($\mathfrak{K}$)) is the collection of all copies of the structures from $\mathfrak{K}$. That is,
$$\mathrm{LD}(\mathfrak{K}) := \bigcup_{\mathcal{A}\in\mathfrak{K}} \{\mathcal{S} : \mathcal{S} \cong \mathcal{A}\}.$$
As we identify each countable structure with an element of Cantor space, we obtain that $\mathrm{LD}(\mathfrak{K}) \subseteq 2^{\mathbb{N}}$.

- The *hypothesis space* (HS) contains, for each $\mathcal{A} \in \mathfrak{K}$, a formal symbol $\ulcorner\mathcal{A}\urcorner$ and a question mark symbol. That is,
$$\mathrm{HS}(\mathfrak{K}) := \{\ulcorner\mathcal{A}\urcorner : \mathcal{A} \in \mathfrak{K}\} \cup \{?\}.$$

- Recall that a countable structure can be identified via its atomic diagram that can be coded as an element of $2^{\mathbb{N}}$ (see §I.4.2). A *learner* **M** sees, by stages, finite sequences of increasing length of the atomic diagram of the given structure in the learning domain, and is required to output conjectures. This is formalized by saying that $\mathbf{M}$ is a function

$$\text{from } 2^{<\mathbb{N}} \text{ to } \mathrm{HS}(\mathfrak{K}).$$

We say that the learning is *successful* if, for each structure $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$, the learner eventually stabilizes to a correct conjecture about its isomorphism type. That is,

$$\lim_{n\to\infty} \mathbf{M}(\mathcal{S}\restriction_n) = \ulcorner\mathcal{A}\urcorner \iff \mathcal{S} \cong \mathcal{A}.$$

We say that $\mathfrak{K}$ is *learnable*, if some learner $\mathbf{M}$ successfully learns $\mathfrak{K}$.

From the previous definition, notice that we are only interested in the case of learning from informant: indeed, the learner has access to increasing fragments of the atomic diagram of a structure, and the atomic diagram of a structure contains both positive and negative information. We also point out that, since we make no assumption on the computational complexity of the learner, in this paradigm both **Ex**-learning and **Bc**-learning coincide. To see this, first notice that the notion of learnability defined above, in the paradigm we described, is the natural representation of **Ex**-learnability. Let's see what happens if we adapt our learning paradigm to mimic **Bc**-learnability. To do so, we ask that every structure $\mathcal{A} \in \mathfrak{K}$ is associated to countably many conjectures $\{\langle\ulcorner A\urcorner, i\rangle : i \in \mathbb{N}\}$. Then, one says that a learner $\mathbf{M}$ is **Bc**-*successful* if for any $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$,

$$(\exists i)(\lim_{n\to\infty} \mathbf{M}(\mathcal{S}\restriction_n) = \langle\ulcorner\mathcal{A}\urcorner, i\rangle) \iff \mathcal{S} \cong \mathcal{A}.$$

We say that $\mathfrak{K}$ is **Bc**-*learnable*, if some learner $\mathbf{M}$ **Bc**-successfully learns $\mathfrak{K}$. The following easy proposition shows that the two notions of learnability for structure defined so far coincide.

Proposition. *Let $\mathfrak{K}$ be a family of structures. Then $\mathfrak{K}$ is* **Bc**-*learnable if and only if $\mathfrak{K}$ is learnable.*

Proof. Clearly, if $\mathfrak{K}$ is learnable then it is also **Bc**-learnable

For the converse, suppose that there exists a learner $\mathbf{M}$ which **Bc**-learns $\mathfrak{K}$. For any $\mathcal{A} \in \mathfrak{K}$, choose a conjecture $\langle\ulcorner\mathcal{A}\urcorner, i\rangle$ for some $i \in \mathbb{N}$ and name it $\ulcorner\mathcal{A}\urcorner$: now every, $\mathcal{A}$ has a unique code $\ulcorner\mathcal{A}\urcorner$. Since there is no restriction on the computational complexity of learners, we can define a learner $\mathbf{M}'$ that learns $\mathfrak{K}$ as follows. Given $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$:

$$\mathbf{M}'(\mathcal{S}\restriction_s) := \begin{cases} \ulcorner\mathcal{A}\urcorner & \text{if } (\exists i)(\mathbf{M}(\mathcal{S}\restriction_s) = \langle\ulcorner\mathcal{A}\urcorner, i\rangle), \\ ? & \text{otherwise.} \end{cases} \qquad \square$$

The fact that both positive and negative information about any structure is provided to the learner, together with the proposition above, justifies the notational simplification of just writing that a family is "learnable" instead of "**InfEx**-learnable" or "**InfBc**-learnable".

Remark. From now on, when we use the word "learnable" without any prefix, we mean learnable in the sense of the previous definition.

The following definition allows us to restrict the behavior of the learner by counting the number of mind changes and it is needed in Chapter V. We say that $\mathbf{M}$ *changes its mind at* σ if

$$\mathbf{M}(\sigma) \neq \mathbf{M}(\sigma[|\sigma| - 2]) \text{ and } \mathbf{M}(\sigma[|\sigma| - 2]) \neq ?.$$

Definition. Let **M** be a learner, $\mathfrak{K}$ be a countable family of computable structures, and let $c : 2^{<\mathbb{N}} \rightarrow$ Ordinals. We say that c is a *mind change counter for* **M** *and* $\mathfrak{K}$ if

- $c(\sigma) \leqslant c(\sigma^-)$ for all $\sigma \neq \langle\rangle$, and
- $c(\sigma) < c(\sigma^-)$ if and only if **M** changes its mind at some $\sigma \in 2^{<\mathbb{N}}$.

Then, we say that $\mathfrak{K}$ is *α-learnable* if and only if there is a learner **M** that learns $\mathfrak{K}$ and there is a mind change counter c for **M** and $\mathfrak{K}$ such that $c(\langle\rangle) = \alpha$. We say that $\mathfrak{K}$ is *properly α-learnable* if $\mathfrak{K}$ is α-learnable but not β-learnable for all $\beta < \alpha$.

Notice that 0-learnability corresponds to what in classical algorithmic learning theory is called **InfFin**-learnability.

Remark. Notice that, in the first point of the definition, one could define a counter c with the property that, for $\sigma \in 2^{<\mathbb{N}}$, $c(\sigma) < c(\sigma[|\sigma| - 2])$ even if **M** does not change its mind at σ. In this case, **M** would have different counters that are in a certain sense not "optimal" with respect to **M**'s mind changes. Our choice allows us to associate a single counter c to a learner **M**, so that, given **M** and once we have set $c(\langle\rangle)$ we can easily recompute c at any stage: this makes our proofs smoother.

We conclude this subsection by giving a useful model-theoretic characterization of learnability in terms of infinitary formulas.

Theorem ([BFSM20, Theorem 3.1])**.** *Let $\mathfrak{K} := \{A_i : i \in \mathbb{N}\}$ be a countable family of pairwise nonisomorphic structures. Then, $\mathfrak{K}$ is learnable if and only if there are $\Sigma_2^{\inf}$ formulas $\phi_0, \ldots, \phi_n, \ldots$ such that*

$$\mathcal{A}_i \models \phi_j \Leftrightarrow i = j.$$

Such a characterization comes in handy in Theorem *IV*.2.12 where we provide a syntactic characterization of another learning criterion. In [BFSM20, Remark 3.2] the authors mention that the statement of this result, is similar to the one given in [MO98, Corollary (52)] by Martin and Osherson, but the proof is significantly different and uses Turing computable embeddings, (see §I.4.2 for the definition).

IV

Learning families of algebraic structures with the help of Borel equivalence relations

All the results in this chapter are a joint work with Nikolay Bazhenov and Luca San Mauro and are collected in [BCS23].

In Part 2's introduction we have presented algorithmic learning theory in general, and we have introduced the framework we intend to work with for learning of algebraic structures: it is now time to notice that the framework has a "defect". Indeed, at the current state, a family of structures is either learnable or not and there is no way of calibrating the complexity of nonlearnable families. To address this issue, we borrowed various ideas from descriptive set theory: this choice is justified by the fact that a primary theme of modern descriptive set theory is the study of classification problems (in particular of isomorphism problems) via different notions of reducibility between definable equivalence relations (see §I.4.1). Indeed, in the paradigm defined in Part 2's introduction, we immediately notice that isomorphism plays a central role: for every structure in a given family, the learner is required to guess its isomorphism type, implying that the nonlearnability of the family is, in some sense, rooted in the complexity of the associated isomorphism relation. Yet, two aspects shall be stressed:

1. The isomorphism relations customarily studied in descriptive set theory refer to *large* collections of countable structures (e.g., *all* graphs, Abelian groups, or metric spaces). On the contrary, here we focus on learning *small* families (i.e., countable families, and in fact often finite ones as in [BSM21]);

2. At any finite stage, the learner sees only a finite fragment of (the presentation of) a structure in the family to be learned, and each conjecture is formulated without knowing how exactly the observed structure will be extended. In topological terms, this coincides with asking that the learning must be a *continuous* process.

We mention that, for our purposes, the equivalence relations under consideration do not need to be Borel ones: on the other hand, these are considered important benchmarks both in invariant descriptive set theory, and also in other works regarding, for example, computable reducibility (see e.g. [Mil21]).

The starting point is Theorem IV.1.1, in which we show that a family of structures $\mathfrak{K}$ is learnable if and only if the isomorphism relation associated with $\mathfrak{K}$ is continuously reducible to the relation E_0. Since E_0 is a fundamental benchmark in the theory of Borel equivalence relations (e.g., recall the Glimm-Effros dichotomy, stated in Theorem I.4.2), such a new characterization of learnability for structures may serve as a piece of evidence that our paradigm is a natural one. Once we have this descriptive set theoretic characterization of learnability, it is natural to wonder what happens if we replace E_0 with other Borel equivalence relations, and this led us to the definition of *E-learnability.* That is, a family of structures $\mathfrak{K}$ is E-learnable, for a Borel equivalence relation E, if there is a continuous reduction from the isomorphism relation associated with $\mathfrak{K}$ to E. In a rough and more descriptive set-theoretic sense, E-learnability is the study of the complexity of isomorphism problems restricted to a countable setting, and in which the reduction is a continuous one. But if, on one hand, we are restricting the possible isomorphism problems and the complexity of the "more classical" Borel reduction, on the other, we are broadening our study to a finer classification of a smaller class of problems. We mention that the notion of E-learnability witnesses a connection between algorithmic learning of algebraic structures and invariant descriptive set theory, and this connection has a twofold interest that we discuss in the next two items.

- Replacing E_0 with Borel equivalence relations of lower/higher complexity, one immediately unlocks the promised hierarchy of learning problems. That is, given Borel equivalence relations E and F, we say that E is *countable-learning reducible* (respectively, *finite-learning reducible*) to F, if every countable (finite) E-learnable family is also F-learnable. In learning theoretic terms, this corresponds in weakening/strengthening the notion of learnability. More precisely, once Theorem $IV.1.1$ settles the correspondence between the latter and E_0-learnability, replacing E_0 with weaker (respectively, stronger), in the sense of continuous reducibility, equivalence relations we are restricting (relaxing) the convergence condition of the learner. We investigate the learning power of several benchmark Borel equivalence relations, offering both examples of relations that do not enlarge the scope of E_0-learnability (Theorems $IV.2.6$ and $IV.2.4$) and equivalence relations which do so (Theorems $IV.2.8$ and $IV.3.3$). Interestingly, we also show that the learning power of some equivalence relations is affected by whether we restrict the attention to families containing only finitely many isomorphism types, or we rather allow countably infinite families.

- It may seem odd that the definition of E-learnability, except for its name, does not mention any learnability-related notion. On the other hand, we already said that, by Theorem $IV.1.1$, E_0-learnability coincides with learnability. One could ask whether similar phenomena hold between different learnability notions (coming from classical algorithmic learning theory) different from the one considered here, and other E-learnabilities for E different from E_0. In Chapter V we show that the notion of α-learnability (defined in Part 2's introduction) is tightly connected to the one of Id-learnability. In another work, not mentioned in this thesis, we are exploring the relationships with other "classical" learning paradigms in the learning hierarchy mentioned in the previous item, such as partial learnability.

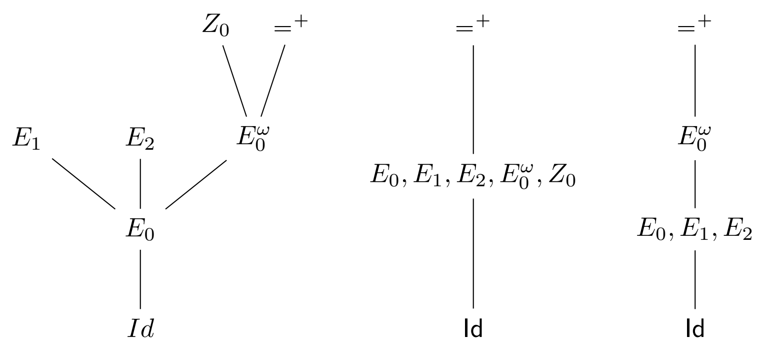

Figure IV.1: On the left side, we reported Figure I.2: the arrows represent continuous reducibility. At the center, we have a diagram summarizing the reductions up to countable-learning reducibility, and on the right side the diagram summarizing reductions up to finite-learning reducibility.

Figure IV.1, also justifies the introduction of the learning hierarchies: indeed, in both cases, these differ from the hierarchy given by continuous reducibility.

IV.1 InfEx-learnability and E_0

We start showing the promised descriptive set-theoretic interpretation of our learning framework. Remember that E_0 denotes the relation of eventual agreement on element of $2^{\mathbb{N}}$, i.e., given $p, q \in 2^{\mathbb{N}}$,

$$p \; E_0 \; q \iff (\exists m)(\forall n \geqslant m)(p(n) = q(n)).$$

Theorem IV.1.1. *A family of structures $\mathfrak{K}$ is learnable if and only if there is a continuous function $\Gamma : 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ such that, for all $\mathcal{A}, \mathcal{B} \in \mathrm{LD}(\mathfrak{K})$,*

$$\mathcal{A} \cong \mathcal{B} \iff \Gamma(\mathcal{A}) \; E_0 \; \Gamma(\mathcal{B}).$$

Proof. For the sake of exposition, we assume that $\mathfrak{K}$ is infinite (the other case being easier) and that $\mathfrak{K} = \{\mathcal{A}_i : i \in \mathbb{N}\}$: we denote $\Gamma(\mathcal{A}_i)$ by q_i.

For the left-to-right direction, let Γ be a function that induces a continuous reduction from $\mathrm{LD}(\mathfrak{K})$ to E_0. We need to show that $\mathfrak{K}$ is learnable. Certainly, $(q_i \not\!\!E_0 \; q_j)$, for all $i \neq j$. Since Γ is continuous (see Theorem $I.2.2$) there exists an oracle $X \in 2^{\mathbb{N}}$ and a Turing operator Φ so that

$$\Gamma(p) = \Phi^{X \oplus p}, \text{ for every } p \in 2^{\mathbb{N}}.$$

Let $p \in 2^{\mathbb{N}}$ and consider $Y := X \oplus \bigoplus_{i \in \mathbb{N}} q_i$. We define a Y-computable auxiliary function $f_{sim}(p; i, s)$. Informally speaking, $f_{sim}(p; i, s)$ is a measure of similarity (at the stage s) between $\Gamma(p)$ and q_i. Let

$$\ell_s := \max\{n : (\forall x \leqslant n)(\Phi^{(X \oplus p) \upharpoonright s}(x)[s] \text{ is defined})\}.$$

If there is no such ℓ_s, then set $f_{sim}(p; i, s) := -1$ for all $i \in \mathbb{N}$. Otherwise, for $i \in \mathbb{N}$, let

$$f_{sim}(p; i, s) := \begin{cases} \max\{k : \ell_s - k \geqslant i \wedge (\forall j \leqslant k) \\ (\Phi^{(X \oplus p) \upharpoonright s}(\ell_s - j) = q_i(\ell_s - j))\} & \text{if } \Phi^{(X \oplus p) \upharpoonright s}(\ell_s) = q_i(\ell_s) \text{ and } \ell_s \geqslant i; \\ -1, & \text{otherwise.} \end{cases}$$

Without loss of generality, we assume that $\ell_{s+1} \in \{\ell_s, \ell_s + 1\}$. It is not hard to show that the function f_{sim} satisfies the following properties. Suppose that $p \in 2^{\mathbb{N}}$ encodes a copy of the structure $\mathcal{A}_{i_0}$, for some $i_0 \in \mathbb{N}$.

- Note that there is $m_0 \in \mathbb{N}$ such that for all $x \geqslant m_0$, we have $\Gamma(p)(x) = q_{i_0}(x)$. This implies that there exists a stage s_0 such that every $s \geqslant s_0$ satisfies $f_{sim}(p; i_0, s+1) \geqslant f_{sim}(p; i_0, s) > -1$. In addition,

$$\lim_s f_{sim}(p; i_0, s) = \infty.$$

- Let $i \neq i_0$. Since $(\Gamma(p) \not\!\!E_0 \; q_i)$ and $\ell_{s+1} \leqslant \ell_s + 1$ for all s, there are infinitely many stages s such that $\Phi^{(X \oplus p) \upharpoonright s}(\ell_s) \neq q_i(\ell_s)$ and $f_{sim}(p; i, s) = -1$. Therefore,

$$\liminf_s f_{sim}(p; i, s) = -1.$$

Construction.
We define our desired learner $\mathbf{M}$. First let, $B_s := \{j \leqslant s : f_{sim}(p; j, s) > -1\}$ and

$$j^* := \min\big\{j \in B_s : f_{sim}(p; j, s) = \max\{f_{sim}(p; m, s) : m \in B_s\}\big\}.$$

Given $p \in 2^{\mathbb{N}}$, for any s, let

$$\mathbf{M}(p \upharpoonright s) := \begin{cases} ? & \text{if } p \upharpoonright s = \langle\rangle \\ \mathbf{M}(p \upharpoonright s-1) & \text{if } B_s = \varnothing \wedge s > 0 \\ \ulcorner \mathcal{A}_{j*} \urcorner & \text{otherwise.} \end{cases}$$

Verification.
We show that $\mathbf{M}$ learns our family $\mathfrak{K}$. Suppose that $p \in 2^{\mathbb{N}}$ encodes a copy $\mathcal{S}$ of some $\mathcal{A}_{i_0}$. Let s_0 be a stage such that $f_{sim}(p; i_0, s) \neq -1$ for all $s \geqslant s_0$. Set $\ell^* := \ell_{s_0}$. Observe the following:

- By the definition of the function f_{sim}, we have $\ell^* \geqslant i_0$. In addition, at each stage $s \geqslant s_0$, the value $\mathbf{M}(p \upharpoonright s)$ is defined according to j^*.
- Suppose that $j > \ell^*$ and $s > s_0$. If $j > \ell_s$, then $f_{sim}(p; j, s) = -1$. If $j \leqslant \ell_s$, then

$$f_{sim}(p; j, s) \leqslant \ell_s - j < \ell_s - \ell^* \leqslant (\ell_s - \ell^*) + f_{sim}(p; i_0, s_0) = f_{sim}(p; i_0, s)$$

  Hence, the definition of j^* implies that $\mathbf{M}(p \upharpoonright s) \neq \ulcorner \mathcal{A}_j \urcorner$.

We deduce that for all $s > s_0$, we have $\mathbf{M}(p \upharpoonright s) \in \{\ulcorner \mathcal{A}_i \urcorner : 0 \leqslant i \leqslant \ell^*\}$. Choose a stage $s_1 > s_0$ such that for each $j \in \{0, 1, \ldots, \ell^*\} \backslash \{i_0\}$, there is another stage t_j satisfying $\max(s_0, \ell^*) < t_j < s_1$ and $f_{sim}(p; j, t_j) = -1$. For every $j \in \{0, 1, \ldots, \ell^*\} \backslash \{i_0\}$ and $s > s_1$, observe the following: if $t_j \leqslant \ell_s$, then we have

$$f_{sim}(p; j, s) \leqslant \ell_s - t_j < \ell_s - \ell^* \leqslant f_{sim}(p; i_0, s).$$

Choose $s_2 > s_1$ such that $\ell_{s_2} \geqslant \max\{t_j :\in \{0, 1, \ldots, \ell^*\} \backslash \{i_0\}\}$. Then the definition of j^* implies that $\mathbf{M}(p \upharpoonright s) = \ulcorner \mathcal{A}_{i_0} \urcorner$ for all $s > s_2$. Therefore, in the limit, $\mathbf{M}$ says that "$\mathcal{S}$ is a copy of $\mathcal{A}_{i_0}$", and the family $\mathfrak{K}$ is learnable by $\mathbf{M}$.

For the right-to-left direction, let $\mathbf{M}$ be a learner of $\mathfrak{K}$ and let $\Gamma : 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ be a continuous function which induces a reduction from $\mathrm{LD}(\mathfrak{K})$ to E_0. To this end, it suffices to fix a countably infinite transversal $(p_i)_{i \in \mathbb{N}}$ of E_0 (i.e., a set intersecting countably many equivalence classes of E_0 in exactly one point) and define Γ as $\Gamma(q)(s) := p_{\mathbf{M}(q[s])}(s)$. Here we use the following convention:

$$\text{if } \mathbf{M}(q[s]) = \ulcorner \mathcal{A}_i \urcorner, \text{ then } p_{\mathbf{M}(q[s])} = p_i \text{ and if } \mathbf{M}(q[s]) = ?, \text{ then } p_{\mathbf{M}(q[s])} = 0^{\mathbb{N}}.$$

To verify that Γ induces a reduction from $\mathrm{LD}(\mathfrak{K})$ to E_0, it is enough to observe the following: if $q \in 2^{\mathbb{N}}$ encodes a copy of some $\mathcal{A}_i$ from $\mathrm{LD}(\mathfrak{K})$, then $(\exists s_0)(\forall s \geqslant s_0)(\mathbf{M}(q[s]) = \ulcorner \mathcal{A}_i \urcorner)$, and thus $\Gamma(q)\ E_0\ p_i$. So, since the p_i's form a transversal for E_0, we deduce that, if q_0 and q_1 encode copies of $\mathcal{A}_i$ and $\mathcal{A}_j$ respectively, then $\Gamma(q_0)\ E_0\ \Gamma(q_1) \iff i = j$ and this concludes the proof. □

IV.2 A novel approach: E-learnability

Replacing E_0 with any equivalence relation E, we define the notion of *E-learnability.*

Definition IV.2.1. A family of structures $\mathfrak{K}$ is *E-learnable* if there is a function $\Gamma : 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ which continuously reduce $\mathrm{LD}(\mathfrak{K})$ to E.

As already mentioned in the introduction, the definition of E-learnability, contrarily to the definition of learnability, seems to be independent of a "classical" approach to learnability. On the other hand, Theorem *IV*.1.1 shows that the latter coincides with what now we can call E_0-learnability. Informally, this means that we can consider the Γ in the definition of E_0-learnability as a "learner" which converges if it reduces $\mathrm{LD}(\mathfrak{K})$ to E_0: with this interpretation in mind, it is not surprising that, the notion of learnability in Definition 2, which is defined *in the limit*, corresponds to E_0-learnability where an E_0-equivalence class is given by all the infinite sequences that are equivalent *in the limit*. Once we have this correspondence, we have a tool for comparing the (non) learnability of a family of structures $\mathfrak{K}$. Indeed, replacing E_0 with other equivalence relations of increasing complexity, we are relaxing the condition of convergence of the "learner" Γ obtaining stronger notions of learnability, and we can naturally compare them, as described in the following definition.

Definition IV.2.2. Let E and F be Borel equivalence relations. We say that E is *countable-learning reducible* to F (in symbols, $E \leqslant^{\mathbb{N}}_{\mathsf{Learn}} F$), if every countable E-learnable family is also F-learnable. We say that E is *finite-learning reducible* to F (in symbols, $E \leqslant^{<\mathbb{N}}_{\mathsf{Learn}} F$), if every finite E-learnable family is also F-learnable.

As justified in this chapter's introduction, we are going to study the learning power of some benchmark Borel equivalence relations: before doing so, we give the following straightforward proposition.

Proposition IV.2.3. *If E is continuously reducible to F then $E \leqslant^{\mathbb{N}}_{\mathsf{Learn}} F$. Furthermore, if $E \leqslant^{\mathbb{N}}_{\mathsf{Learn}} F$ then $E \leqslant^{<\mathbb{N}}_{\mathsf{Learn}} F$: the converse may fail.*

IV.2.1 When oracle equivalence relations don't help: E_1 and E_2

In this section, we analyze the learning power of E_1 and E_2. These equivalence relations are incomparable and strictly above E_0 with respect to continuous reductions; in fact, the same is true if one requires computable reductions. But, as it is proven in Theorem IV.2.4 and Theorem IV.2.6, E_1 and E_2 coincide and collapse to E_0 with respect to their learning power.

Recall that given $p, q \in 2^{\mathbb{N}\times\mathbb{N}}$,

$$p \; E_1 \; q \iff (\forall^\infty n \in \mathbb{N})(\forall i)(p(m,i) = q(m,i)).$$

Theorem IV.2.4. *A family $\mathfrak{K}$ is E_1-learnable if and only if $\mathfrak{K}$ is E_0-learnable. That is, $E_1 \equiv^{\mathbb{N}}_{\mathsf{Learn}} E_0$.*

Proof. For the right-to-left direction, since E_0 is continuously reducible to E_1 (see Figure I.2) it suffices to apply Proposition IV.2.3.

For the opposite direction, let $\mathfrak{K} := \{\mathcal{A}_i : i \in \mathbb{N}\}$ be an E_1-learnable family and let $\Gamma : 2^{\mathbb{N}} \to 2^{\mathbb{N}\times\mathbb{N}}$ induce a continuous reduction from $\mathrm{LD}(\mathfrak{K})$ to E_1. For each $i \in \mathbb{N}$, we choose $q_i \in 2^{\mathbb{N}\times\mathbb{N}}$ such that,

$$(\forall \mathcal{S} \in \mathrm{LD}(\mathfrak{K}))(\mathcal{S} \cong \mathcal{A}_i \implies \Gamma(\mathcal{A}_i) = q_i),$$

i.e., Γ maps all copies of $\mathcal{A}_i$ into the class $[q_i]_{E_1}$. Fix a computable bijection ξ from the set $\{(i,j) \in \mathbb{N}^2 : i \neq j\} \times \mathbb{N}$ onto $\mathbb{N}$. We build a set $X := \{\langle c_s, r_s\rangle : s \in \mathbb{N}\}$ as follows. Put $\langle c_s, r_s\rangle := \langle 0, 0\rangle$. At stage $s = \xi(i,j,t)$, let

$$c_{s+1} := \min\{c > m_s : (\forall k)(q_i(c,k)q_j(c,k))\} \wedge r_{s+1} := \min\{r : q_i(c_{s+1}, r) \neq q_j(c_{s+1}, r)\}.$$

Notice that $\langle c_{s+1}, r_{s+1}\rangle$ exists as $(q_i \not\!E_1 \; q_j)$, and it is straightforward to check that for every $c \in \mathbb{N}$, there is at most one r such that $\langle c, r\rangle \in X$.

We define the X-computable (and hence continuous) operator $\Psi : 2^{\mathbb{N}\times\mathbb{N}} \to 2^{\mathbb{N}}$ as follows: for every $p \in 2^{\mathbb{N}\times\mathbb{N}}$ and $s \in \mathbb{N}$, let $\Psi(p)(s) := p(c_s, r_s)$.

We show that the operator $\Phi := \Psi \circ \Gamma$ provides a continuous reduction from $\mathrm{LD}(\mathfrak{K})$ to E_0. Let $p \in 2^{\mathbb{N}}$ encode a copy of some $\mathcal{A}_{i_0}$: since $\Gamma(p)\ E_1\ q_{i_0}$, we obtain that $(\forall^\infty s)(p(c_s, r_s) = q_{i_0}(c_s, r_s))$ and thus $\Phi(p)\ E_0\ \Psi(q_{i_0})$. Instead, if $i \neq i_0$. Then,

$$(\forall^\infty t)\big(p(c_{\xi(i,i_0,t)}, r_{\xi(i,i_0,t)}) = q_{i_0}(c_{\xi(i,i_0,t)}, r_{\xi(i,i_0,t)}) \neq q_i(c_{\xi(i,i_0,t)}, r_{\xi(i,i_0,t)})\big).$$

This implies that $(\Phi(p) \not{E_0}\ \Psi(q_i))$. Therefore, we deduce that our family $\mathfrak{K}$ is E_0-learnable. The theorem is proved. □

Remark IV.2.5. The previous theorem can also be restated in purely descriptive set theoretic terms as follows. Let $(q_i)_{i\in\mathbb{N}} \in (2^{\mathbb{N}\times\mathbb{N}})^{\mathbb{N}}$ and $q_i E_1 q_j$ if and only if $i = j$. Then, there is a continuous function Γ such that for every $p \in \bigcup_{i\in\mathbb{N}}[q_i]_{E_1}$ we have that pE_1q_j if and only if $\Gamma(p)\ E_0\ \Gamma(q_j)$, i.e. E_1, with domain restricted to $(q_i)_{i\in\mathbb{N}}$, continuously reduces to E_0 via Γ. Theorem *IV*.2.6, Proposition *IV*.2.7, and Theorem *IV*.3.2 admit similar characterizations, but we will not make them explicit.

Given $p, q \in 2^{\mathbb{N}}$,

$$p\ E_2\ q \iff \sum_{k=0}^{\infty} \frac{(p\triangle q)(k)}{k+1} < \infty.$$

Theorem IV.2.6. *A countable family $\mathfrak{K}$ is E_2-learnable if and only if $\mathfrak{K}$ is E_0-learnable, i.e.* $E_2 \equiv^{\mathbb{N}}_{\mathsf{Learn}} E_0$.

Proof. For the right-to-left direction, since E_0 is continuously reducible to E_2 (see Figure I.2) it suffices to apply Proposition *IV*.2.3.

For the opposite direction, let $\mathfrak{K} := \{\mathcal{A}_i : i \in \mathbb{N}\}$ be an E_2-learnable family and let Γ be an operator, which induces a continuous reduction from $\mathrm{LD}(\mathfrak{K})$ to E_2. For $i \in \mathbb{N}$, we fix $q_i \in 2^{\mathbb{N}}$ such that Γ maps all copies of $\mathcal{A}_i$ into $[q_i]_{E_2}$.

By Theorem *I*.2.2, there exist an oracle X and a Turing operator Φ such that $\Gamma(p) = \Phi^{X\oplus p}$ for all $p \in 2^{\mathbb{N}}$.

Construction.

We define a $(X \oplus \bigoplus_{i\in\mathbb{N}} q_i)$-computable operator $\Psi : 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ that, given in input $p \in 2^{\mathbb{N}}$ acts as follows. For $s \in \mathbb{N}$, let $\ell_s := \max\{\ell : (\forall x \leqslant \ell)(\Phi^{(X\oplus p)\upharpoonright s}(x)[s] \text{ is defined})\}$. Without loss of generality, one may assume that ℓ_s is defined for every s. For $i, s \in \mathbb{N}$, we consider the partial sum

$$p(i,s) := \sum_{k=0}^{\ell_s} \frac{(q_i \triangle \Phi^{X\oplus p})(k)}{k+1}.$$

At a stage s, we define auxiliary values $i_s, b_s \in \mathbb{N}$ and $c_s \in \{0,1\}$. Similarly to the proof of Theorem *IV*.1.1, these parameters control the flow of the construction. Moreover, at each stage s, we set $\Psi(p)(s) := q_{i_s}(s)$. Our construction ensures that $i_s \leqslant b_s$ for every s. At stage 0, set $i_0 := 0$, $b_0 := 1$, and $c_0 := 0$. At stage $s+1$, we assume that the parameters b_s, c_s, and i_s are already defined.

- *Case* 1: if $p(i_s, s+1) \leqslant b_s$, then do not change anything.
- *Case* 2: if $p(i_s, s+1) > b_s$ and $c_s = 0$, then put $i_{s+1} := 0$ and $c_s := 1$.
- *Case* 3: suppose that $p(i_s, s+1) > b_s$, $c_s = 1$, and $i_s < b_s$. Define $i_{s+1} := i_s + 1$.
- *Case* 4: suppose that $p(i_s, s+1) > b_s$, $c_s = 1$, and $i_s = b_s$. Let

$$i_0 := \min\{i \leqslant b_s + 1 : p(i, s+1) = \min\{p(j, s+1) : j \leqslant b_s + 1\}\}$$

and set $i_{s+1} := i_0$, $c_{s+1} := 0$ and $b_{s+1} := \max\{x : x \in \{b_s + 1, \text{the integer part of } p(i_0, s+1) + 1\}\}$.

This concludes the description of the construction. It is clear that the operator Ψ is Y-computable.

Verification.

Suppose that $p \in 2^{\mathbb{N}}$ encodes a copy of the structure $\mathcal{A}_{i_0}$, and define $N_0 := \sum_{k=0}^{\infty} \frac{(q_{i_0} \triangle \Gamma(p))(k)}{k+1}$.

We claim that there exists a finite limit $b^* = \lim_s b_s$ and, in addition, $b^* \geqslant i_0$. To do so, we distinguish two cases. First, assume that $b_s < i_0$ for all s. Then we have that for every s, $i_s < i_0$. Furthermore, since the sequence b_s is nondecreasing, there exists $b^* = \lim_s b_s$ with $b^* < i_0$. Since $(\Gamma(p) \not{E_2}\, q_j)$ for all $j \neq i_0$, there exists a stage s_0 such that $p(j, s_0) > i_0$ for all $j < i_0$, and $b_s = b^*$ for all $s \geqslant s_0$. Then, our construction ensures that after the stage s_0, there is a stage s_1 satisfying *Case 4*. This implies that $b_{s_1} \geqslant b^* + 1$, which gives a contradiction. Thus, we deduce that there must exist a stage s'_0 such that $b_{s'_0} \geqslant i_0$. Second, assume that $\lim_s b_s = \infty$. This implies that there are infinitely many stages $s > s'_0$ satisfying *Case 4*. Choose a stage $s_1 > s'_0$ such that s_1 satisfies *Case 4* and $b_{s_1} \geqslant N_0 + 1$. Consider the value $i^* := i_{s_1}$.

- If $i^* = i_0$, then for every s, we have $p(i^*, s) < b_{s_1}$. This implies that every stage $s > s_1$ satisfies *Case 1*, which gives a contradiction.
- If $i^* \neq i_0$, then the stage $s_2 := \{\min s > s_1 : p(i^*, s_2) > b_{s_1}\}$ satisfies *Case 2*, and we have $c_{s_2} = 1$. Therefore, *Case 3* of the construction ensures that there is a sequence of stages

$$s_2 = s''_0 < s''_1 < \cdots < s''_{i_0}$$

such that $i_{s''_k} = k$ for every $k \leqslant i_0$. Again, every stage $s > s''_{i_0}$ satisfies *Case 1*, which provides a contradiction.

Therefore, we proved that there is a finite limit $b^* = \lim_s b_s$, and $b^* \geqslant i_0$ and this ends the proof of the claim.

Now choose a stage s^* such that $b_{s^*} = b^*$. There exists a stage $s_1 \geqslant s^*$ such that every $i \leqslant b^*$ satisfies the following: if $i \neq i_0$, then $p(i, s_1) > b^*$. Since after the stage s^*, there are no stages satisfying *Case 4*, it is not hard to deduce that for every $s \geqslant s_1 + b^* + 2$, we must have $i_s = i_0$. This implies that $\Psi(p)\ E_0\ q_{i_0}$. For all $i \neq j$, we have $(q_i \not{E_2}\, q_j)$ and clearly this implies $(q_i \not{E_0}\, q_j)$. Hence, we conclude that our operator Ψ provides a continuous reduction from $\mathrm{LD}(\mathfrak{K})$ to E_0. In other words, the family $\mathfrak{K}$ is E_0-learnable, as desired. □

IV.2.2 Characterizing the learning power of E_0^ω

All equivalence relations considered so far (i.e., E_0, E_1, and E_2) are inseparable with respect to their learning power. In fact, by Theorem *IV*.1.1, they don't expand the boundaries of our original framework. The case of E_0^ω, to be discussed in this section, is different. Namely, E_0^ω has strictly more learning power than E_0—but this fact is only witnessed by infinite families. Recall that given $(p_n)_{n\in\mathbb{N}}, (q_n)_{n\in\mathbb{N}} \in (2^{\mathbb{N}})^{\mathbb{N}}$,

$$((p_n)_{n\in\mathbb{N}}\ E_0^\omega\ (q_n)_{n\in\mathbb{N}}) \iff (\forall m \in \mathbb{N})(p_m\ E_0\ q_m).$$

Proposition IV.2.7. *A finite family $\mathfrak{K}$ is E_0^ω-learnable if and only if $\mathfrak{K}$ is E_0-learnable. That is, $E_0^\omega \equiv_{\mathsf{Learn}}^{<\mathbb{N}} E_0$.*

Proof. For the right-to-left direction, since E_0 is continuously reducible to E_0^ω (see Figure I.2) it suffices to apply Proposition *IV*.2.3.

For the other direction, let $\mathfrak{K} := \{\mathcal{A}_i : i \leqslant n\}$ be a finite E_0^ω-learnable family and let Γ induce a continuous reduction from $\mathrm{LD}(\mathfrak{K})$ to E_0^ω. For $i \leqslant n$, choose $(q_n^i)_{n\in\mathbb{N}}$ such that Γ maps all copies of $\mathcal{A}_i$ into $[(q^i_n)_{n\in\mathbb{N}}]_{E_0^\omega}$. For every $\langle i,j\rangle$ such that $i \neq j$, let $d(i,j)$ be such that $q^i_{[d(i,j)]} \not{E}_0\, q^j_{[d(i,j)]}$.

Then, we define a Turing operator $\Psi : (2^{\mathbb{N}})^{\mathbb{N}} \to 2^{\mathbb{N}}$ as follows.

$$\Psi((p_n)_{n\in\mathbb{N}}) := \bigoplus_{i\neq j\leqslant n} p_{[d(i,j)]}.$$

The operator $\Phi := \Psi \circ \Gamma$ provides a continuous reduction from $\mathrm{LD}(\mathfrak{K})$ to E_0. Indeed, let $p \in 2^{\mathbb{N}}$ encode a copy of $\mathcal{A}_{i_0}$. Then $\Gamma(p)\ E_0^\omega\ (q^{i_0})_{n\in\mathbb{N}}$ and $\Phi(p)\ E_0\ \Psi((q^{i_0})_{n\in\mathbb{N}})$. If $i \neq i_0$, then we have

$$(p\ E_0\ q^{i_0}_{[d(i,i_0)]} \not{E}_0\ q^i_{[d(i,i_0)]}) \text{ and } (\Phi(p) \not{E}_0\ \Psi((q^i_n)_{n\in\mathbb{N}})).$$

Therefore, the family $\mathfrak{K}$ is E_0-learnable. □

We continue separating E_0^ω-learnability by E_0-learnability.

Theorem IV.2.8. $E_0^\omega <^{\mathbb{N}}_{\mathsf{Learn}} E_0$.

To prove the theorem above, we provide examples of two families that are E_0^ω learnable but not E_0-learnable. Of course a single example would suffice, but we prefer to provide both families, as they have interesting properties. Indeed, to prove the statement for the first family we need to exploit the model-theoretic characterization of E_0-learnability given at the end of Part 2's introduction (i.e., [BFSM20, Theorem 3]), while the second one provides an example of a more "natural" family separating E_0 from E_0^ω. Hence, the proof of the theorem above is obtained combining Lemmas *IV*.2.9 and *IV*.2.10 or equivalently, by Lemma *IV*.2.11.

We define the first family that is a family of directed graphs $\mathfrak{K}_{gr}$. For the sake of exposition, first, we define a family $\mathfrak{K}_b := \{\mathcal{A}_i : i \in \mathbb{N}\}$ where the signature of the class $\mathfrak{K}_b$ is allowed to be infinite. After that, we provide comments on how to pass from $\mathfrak{K}_b$ to the desired $\mathfrak{K}^{gr}$. Thus, consider the signature $L = \{R_j : j \in \mathbb{N}\} \cup \{\leqslant\}$, where R_j are unary predicates. Given $p \in 2^{\mathbb{N}}$, we define an L-structure $\mathcal{D}(p)$ as follows:

- Inside $\mathcal{D}(p)$, the relations R_j, $j \in \mathbb{N}$, are pairwise disjoint. We say that the set $R_j^{\mathcal{D}(p)}$ is the R_j*-box* of $\mathcal{D}(p)$.
- The R_j-box of $\mathcal{D}(p)$ contains a linear order L_j such that

$$L_j \cong \begin{cases} \omega, & \text{if } p(j) = 0, \\ \omega^*, & \text{if } p(j) = 1. \end{cases}$$

where ω and ω^* are respectively the order types of the positive and negative integers. For a finite string $\sigma \in 2^{<\mathbb{N}}$, let $\mathcal{A}_\sigma$ be the structure $\mathcal{D}(\sigma^\frown 10^{\mathbb{N}})$. Our family $\mathfrak{K}_b$ consists of all $\mathcal{A}_\sigma$, $\sigma \in 2^{<\mathbb{N}}$. Notice that the relation $\leqslant$ in L provides an order between elements in L_j and does not provide any order between elements in different R_j-boxes.

In order to obtain the family of directed graphs $\mathfrak{K}_{gr}$, which has the same properties as the family $\mathfrak{K}_b$, one can proceed as follows. Instead of distinguishing an R_j-box via the predicate R_j, one attaches to every element a of L_j of the corresponding R_j-box the graph C_{j+3}. Indeed, for each $a \in L_j$ use fresh elements $c_{a,1}, c_{a,2} \ldots, c_{a,j+3}$ and put the edges $(a, c_{a,1})$, $(c_{a,j+3}, c_{a,1})$ and, for $i \leqslant j+2$, $(c_{a,i}, c_{a,i+1})$.

Lemma IV.2.9. *The family $\mathfrak{K}_{gr}$ is E_0^ω-learnable.*

Proof. By the comments above, it suffices to prove the statement for $\mathfrak{K}_b$. Recall that the family $\{\omega, \omega^*\}$ is learnable, as they are distinguishable by Σ_2^{inf} formulas [BFSM20, Theorem 3]. By employing this fact, we can easily define a Turing operator Φ that, given in input $p \in 2^{\mathbb{N}}$, it treats p as a code for the atomic diagram of a countable partial order $\mathcal{L}$. Then:

- If $\mathcal{L}$ is a copy of ω, then the output $\Phi(p) E_0 0^{\mathbb{N}}$.
- If $\mathcal{L} \cong \omega^*$, then we have $(\Phi(p)\ E_0\ 1^{\mathbb{N}})$.

For every $j \in \mathbb{N}$, we define a Turing operator Ψ_j that, given in input $p \in 2^{\mathbb{N}}$, it treats p as a code of a countable L-structure $\mathcal{A}$. The output $\Psi_j(p)$ encodes the partial order, which is contained inside the R_j-box of $\mathcal{A}$.

Finally, we define an operator Θ. For $p \in 2^{\mathbb{N}}$ and for $j, k \in \mathbb{N}$, we set

$$\Theta(p)(\langle j, k\rangle) := (\Phi \circ \Psi_j(p))(k).$$

Observe the following. Given $p, q \in 2^{\mathbb{N}}$, if p encodes a copy of the structure $\mathcal{D}(q)$, then for every $j \in \mathbb{N}$, we have:

- if $q(j) = 0$, then the j-th column $(\Theta(p))^{[j]} E_0 0^{\mathbb{N}}$;
- if $q(j) = 1$, then $(\Theta(p))^{[j]}\ E_0\ 1^{\mathbb{N}}$.

This observation implies that the operator Θ witnesses the E_0^ω-learnability of our family $\mathfrak{K}$. □

Lemma IV.2.10. *The family $\mathfrak{K}_{gr}$ is not E_0-learnable.*

Proof. As we have done for Lemma *IV*.2.9, by the comments above, it suffices to prove the statement for $\mathfrak{K}_b$. Towards a contradiction, assume that the family $\mathfrak{K}_b$ is E_0-learnable. By Theorem *IV*.1.1, $\mathfrak{K}_b$ is learnable, and by [BFSM20, Theorem 3.1], one can choose an infinitary Σ_2^c sentence θ such that $\mathcal{A}_0 \models \theta$ and for every $\sigma \neq 0$, we have $\mathcal{A}_\sigma \not\models \theta$. Without loss of generality, one may assume that

$$\theta = \exists \bar{x} \bigwedge\!\!\!\bigwedge_{i \in I} \forall \bar{y}_i \psi_i(\bar{x}, \bar{y}_i),$$

where every ψ_i is a quantifier-free formula. Fix a tuple $\bar{c}$ from the structure $\mathcal{A}_0$ such that

$$\mathcal{A}_0 \models \bigwedge\!\!\!\bigwedge_{i \in I} \forall \bar{y}_i \psi_i(\bar{c}, \bar{y}_i).$$

Choose a natural number N such that for every $j \geqslant N$, the R_j-box of $\mathcal{A}_0$ does not contain elements from $\bar{c}$.

Consider a string $\tau := 010^N$ and the corresponding structure $\mathcal{A}_\tau = \mathcal{D}(\tau^\frown 10^{\mathbb{N}})$. It is clear that for every $j < N$, the (contents of the) R_j-boxes inside $\mathcal{A}_0$ and $\mathcal{A}_\tau$ are isomorphic. Therefore, one can choose a tuple $\bar{d}$ inside $\mathcal{A}_\tau$ as isomorphic copies of $\bar{c}$ (with respect to the isomorphism of the R_j-boxes, $j < N$).

We claim that the structures $(\mathcal{A}_0, \bar{c})$ and $(\mathcal{A}_\tau, \bar{d})$ satisfy the same $\exists$-sentences. To do so, it suffices to verify that every quantifier-free formula $\psi(\bar{x}, \bar{y})$ satisfies

$$\mathcal{A}_0 \models \exists \bar{y} \psi(\bar{c}, \bar{y}) \quad \Rightarrow \quad \mathcal{A}_\tau \models \exists \bar{y} \psi(\bar{d}, \bar{y}).$$

The other direction ($\Leftarrow$) can be obtained via a similar argument. Choose a tuple $\bar{b}$ from $\mathcal{A}_0$ such that $\mathcal{A}_0 \models \psi(\bar{c}, \bar{b})$. Suppose that $\bar{b} = b_0, b_1, \dots, b_m$. We define a new tuple $\bar{b}' = b'_0, b'_1, \dots, b'_m$ from $\mathcal{A}_\tau$ as follows:

- If b_k lies in an R_j-box, which contains elements from $\bar{c}$, then b'_k is defined as the copy of b_k with respect to the natural isomorphism of R_j-boxes, $j < N$.

- Suppose that b_k belongs to an R_j-box, which does not contain elements from $\bar{c}$. Then b'_k can be chosen as any element from the R_j-box of $\mathcal{A}_\tau$, while preserving the ordering $\leqslant$. More formally, one needs to ensure the following: if $b_k \neq b_\ell$ both belong to this R_j-box, then we have:
$$\mathcal{A}_0 \models b_k \leqslant b_\ell \iff \mathcal{A}_\tau \models b'_k \leqslant b'_\ell.$$

It is clear that the tuples $\bar{c}, \bar{b}$, and $\bar{d}, \bar{b}'$ satisfy the same atomic formulas. Therefore, we deduce that the structure $\mathcal{A}_\tau$ satisfies $\psi(\bar{d}, \bar{b}')$, and $\mathcal{A}_\tau \models \exists \bar{y} \psi(\bar{d}, \bar{y})$. This ends the proof of the claim.

The claim we have just proven implies that
$$\mathcal{A}_\tau \models \bigwedge_{i \in I} \forall \bar{y}_i \psi_i(\bar{d}, \bar{y}_i),$$
and hence, $\mathcal{A}_\tau \models \theta$, which contradicts the choice of θ. We deduce that the family $\mathfrak{K}_b$ is not E_0-learnable. □

Recall from §I.1 that $\overset{n}{\otimes} C_3$ denotes the disconnected union of n-many copies of C_3, while $\overset{\infty}{\otimes} C_3$ is the disconnected union of infinitely many copies of C_3.

Lemma IV.2.11. *The family $\mathfrak{K} := \{\overset{n}{\otimes} C_3 : n > 0\} \cup \{\overset{\infty}{\otimes} C_3\}$ is E_0^ω-learnable, but not E_0-learnable.*

Proof. By Theorem *IV*.1.1 E_0-learnability and learnability coincide. We can use a similar argument to the one given in [FKSM19, Example 5] to show that $\mathfrak{K}$ is not learnable (in [FKSM19], the authors write "**InfEx**-learnable" instead of "learnable"). Informally, given $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ such that $\mathcal{S} \cong \overset{\infty}{\otimes} C_3$ no learner $\mathbf{M}$ is able to learn $\overset{n}{\otimes} C_3$ for some $n > 0$ as, for any s, $\mathcal{S}\restriction_s$ may be extended either to a copy of $\overset{\infty}{\otimes} C_3$ or to a copy of $\overset{m}{\otimes} C_3$ for $m \geqslant n$.

We now show that $\mathfrak{K}$ is E_0^ω learnable. To do so, we define a Turing operator $\Psi : 2^{\mathbb{N}} \to (2^{\mathbb{N}})^{\mathbb{N}}$ acting as follows. For any $p \in 2^{\mathbb{N}}$ let $\Psi(p) := (p_n)_{n \in \mathbb{N}}$ where p_n is such that for any $s \in \mathbb{N}$, let
$$p_n(s) := \begin{cases} 1 & \text{if } \mathcal{S}\restriction_s \hookrightarrow \overset{n+1}{\otimes} C_3 \\ 0 & \text{otherwise.} \end{cases}$$

It is easy to verify that:

- if $\mathcal{S} \cong \overset{n}{\otimes} C_3$ for $n > 0$, then, $(\forall i < n)(p_i \; Id \; 1^{\mathbb{N}})$ while for any $(\forall j \geqslant n)(p_j \; E_0 \; 1^{\mathbb{N}})$;

- if $\mathcal{S} \cong \overset{\infty}{\otimes} C_3$ then for all $(\forall i)(p_i \; E_0 \; 0^{\mathbb{N}})$.

This concludes the proof. □

IV.2.3 A syntactic characterization of E_0^ω-learnability

As aforementioned, in [BFSM20] Bazhenov, Fokina, and San Mauro obtained a full syntactic characterization of which families of structures are learnable, by means of computable $\Sigma_2^{\inf}$ formulas (see [BFSM20, Theorem 3.1], at the end of §2). The next theorem offers an analogous characterization for E_0^ω-learning.

Theorem IV.2.12. *Let $\mathfrak{K} := \{\mathcal{A}_i : i \in \mathbb{N}\}$ be a countable family. The family $\mathfrak{K}$ is E_0^ω-learnable if and only if there exists a countable family of computable $\Sigma_2^{\inf}$ sentences Θ with the following* *properties:*

(a) *if θ is a formula from Θ, then there is a formula $\psi \in \Theta$ such that for every $\mathcal{A} \in \mathfrak{K}$,*

$$\mathcal{A} \models \theta \iff \mathcal{A} \models \neg\psi;$$

(b) *if $\mathcal{A} \not\cong \mathcal{B}$ are structures from $\mathfrak{K}$, then there is a sentence $\theta \in \Theta$ such that*

$$\mathcal{A} \models \theta \text{ and } \mathcal{B} \models \neg\theta.$$

The proof of the theorem is inspired by ideas from [BFSM20]. In particular, we adopt the technology of tc-embeddings and the Relativized Pullback Theorem reminded in the preliminaries. Before proving the theorem above, we recall the definition of the class $\mathfrak{K}_{st}$ from [BFSM20]: in the same paper, the authors show that $\mathfrak{K}_{st}$ is an archetypical E_0-learnable family, in the sense that a countable family $\mathfrak{C}$ is learnable if and only if there is a continuous embedding from the class $\mathfrak{C}$ into $\mathfrak{K}_{st}$. Thus, consider a signature $L_{st} := \{\leqslant\} \cup \{P_i : i \in \mathbb{N}\}$, where P_i are unary predicates. Given $i \in \mathbb{N}$, an L-structure $\mathcal{S}_i$ satisfies the following properties:

- Inside $\mathcal{S}_i$, the relations P_j are pairwise disjoint. In addition, if $x \in P_j$ and $y \in P_k$ for some $j \neq k$, then x and y are $\leqslant$-incomparable. Let η be the order type of the rational numbers.
- The predicate P_i contains an isomorphic copy of $1 + \eta$.
- Every P_j, for $j \neq i$, contains a copy of η.

For dealing with E_0^ω-learnability, we have to introduce a new, and more complicated, class $\mathfrak{C}_{st}$. But the informal idea behind $\mathfrak{C}_{st}$ is pretty simple: roughly speaking, this class contains all countable disjoint sums of the structures from $\mathfrak{K}_{st}$.

Consider a new signature $L_1 = L_{st} \cup \{Q_k : k \in \mathbb{N}\}$, where Q_k are unary predicates. The class $\mathfrak{C}_{st}$ contains all L-structures $\mathcal{M}$, which satisfy the following properties:

- Their relations Q_k, $k \in \mathbb{N}$, are pairwise disjoint. We say that the L_{st}-substructure with domain $\mathcal{M} \upharpoonright Q_k$ is the *Q_k-box* of $\mathcal{M}$.
- Every Q_k-box of $\mathcal{M}$ is isomorphic to a structure from the class $\mathfrak{K}_{st}$.

Note that our class $\mathfrak{C}_{st}$ is uncountable.

Lemma IV.2.13. *The class $\mathfrak{C}_{st}$ has a computable family of Σ_2^{inf} sentences Θ, which satisfies properties (a) and (b) from the formulation of Theorem IV.2.12.*

Proof. The desired family Θ contains the following Σ_2^{inf} sentences:

1. For each i and j, we add a finitary Σ_2^0 sentence $\theta_{i,j}$, which states the following: "the P_j-predicate inside the Q_i-box has a $\leqslant$-least element".
2. For each i and j, we add a Σ_2^{inf} sentence $\psi_{i,j}$, which is equivalent to the following formula:

$$\bigvee\!\!\!\bigvee_{k \neq j} \theta_{i,k}.$$

   In other words, there is some $k \neq j$ such that the P_k-predicate inside the Q_i-box has a least element.

Let $\mathcal{M}$ be an arbitrary structure from $\mathfrak{C}_{st}$. Since the Q_i-box of $\mathcal{M}$ is a structure from $\mathfrak{K}_{st}$, it is not hard to show that

$$\mathcal{M} \models \theta_{i,j} \iff \mathcal{M} \models \neg\psi_{i,j}.$$

Hence, we deduce that the class $\mathfrak{C}_{st}$ satisfies property (a) of Theorem IV.2.12.

Suppose that $\mathcal{M} \not\cong \mathcal{N}$ are structures from $\mathfrak{C}_{st}$. Then there exist indices i and j such that for the structures $\mathcal{M}$ and $\mathcal{N}$, their P_j-predicates inside Q_i-boxes are not isomorphic. Without loss of generality, one may assume that in this P_j-place, $\mathcal{M}$ has order-type $1 + \eta$, and $\mathcal{N}$ has order-type η. Then, it is clear that

$$(\mathcal{M} \models \theta_{i,j} \wedge \mathcal{M} \models \neg\psi_{i,j}) \text{ and } (\mathcal{N} \models \neg\theta_{i,j} \wedge \mathcal{N} \models \psi_{i,j}).$$

Therefore, $\mathfrak{C}_{st}$ satisfies property (b) of the theorem. □

Proof of Theorem IV.2.12. We first show the left-to-right direction: to do so, we build a continuous embedding from the given class $\mathfrak{K}$ to $\mathfrak{C}_{st}$. This embedding allows us to apply the Relativized Pullback Theorem (Theorem *I*.4.14) for finishing our argument.

Consider $\vec{\gamma} := (\gamma_i)_{i\in\mathbb{N}} \in (2^{\mathbb{N}})^{\mathbb{N}}$. We define an auxiliary continuous operator $\Psi_{\vec{\gamma}} : 2^{\mathbb{N}} \to 2^{\mathbb{N}}$, mapping any p to δ_p, where δ_p encodes the atomic diagram of an L-structure $\mathcal{S}(p)$.

We always assume that inside $\mathcal{S}(p)$, all predicates P_i are disjoint, every predicate P_i contains at least one element and $\mathcal{S}(p)$ has domain $\mathbb{N}$.

Construction.

The construction of $\mathcal{S}(p)$ proceeds in stages. At a stage s, for each $i \in \mathbb{N}$, we define the following auxiliary value:

$$v(i,s) = \begin{cases} \min\{t \leqslant s : (\forall x)[t \leqslant x \leqslant s \implies p(x) = q_i(x)]\}, & \text{if } p(s) = q_i(s), \\ \infty, & \text{otherwise.} \end{cases}$$

We also define two parameters $p(s)$ and $b(s)$. Roughly speaking, at a stage s, our current "guess" is that the input $p \in 2^{\mathbb{N}}$ is such that $p\ E_0\ q_{p(s)}$, where $p(s) \leqslant b(s) \leqslant s$. At stage 0 let $p(0) = 0$ and $b(s) = 0$. At stage $s+1$, consider the following cases.

- *Case* 1: suppose that there is $i \leqslant s+1$ such that $p(s+1) = q_i(s+1)$.

  If $v(p(s), s+1) \neq \infty$, then set $i_0 := p(s)$. Otherwise, i_0 is defined as follows.

  – If $p(s) < b(s)$, then $p(s+1) := p(s) + 1$ and $i_0 := p(s) + 1$;

  – If $p(s) = b(s)$, then let

  $$i_0 := \min\{i \leqslant s+1 : v(i, s+1) = \min\{v(j, s+1) : j \leqslant s+1\}\}.$$

  We set $b(s+1) := s+1$ and $p(s+1) := i_0$.

  Suppose that the relation P_{i_0} (at this particular moment) contains the following linear order: $a_0 < a_1 < \cdots < a_k$. We choose fresh elements $b_0, b_1, \ldots, b_k$, add them into P_{i_0}, and set:

  $$a_0 < b_0 < a_1 < b_1 < \cdots < a_k < b_k.$$

  Consider a $j \neq i_0$, and suppose that the relation P_j contains the ordering $c_0 < c_1 < \cdots < c_\ell$. Choose fresh elements $d_{-1}, d_0, d_1, \ldots, d_\ell$, put them into P_j, and define:

  $$d_{-1} < c_0 < d_0 < c_1 < d_1 < \cdots < c_\ell < d_\ell.$$

- *Case* 2: if $p(s) \neq q_i(s)$ for all $i \leqslant s+1$, then for every $j \in \mathbb{N}$, the relation P_j is arranged in the same way as described in *Case 1*.

  This concludes the description of the operator $\Psi_{\vec{\gamma}}$.

Verification.

Similarly to the previous proofs, it is not hard to verify the following properties of $\Psi_{\vec{\gamma}}$:

1. The operator $\Psi_{\vec{\gamma}}$ is $\left(\bigoplus_{i\in\mathbb{N}}\gamma_i\right)$-computable.

2. If $(p\ E_0\ \gamma_i)$ for some $i \in \mathbb{N}$, then the structure $\mathcal{S}(p)$ is isomorphic to $\mathcal{S}_i$.

Now, let Γ be a continuous operator which induces a reduction from $\mathrm{LD}(\mathfrak{K})$ to E_0^ω. For a structure $\mathcal{A}_i$ from $\mathfrak{K}$, fix $(q_n^i)_{n\in\mathbb{N}} \in (2^{\mathbb{N}})^{\mathbb{N}}$ such that Γ maps all copies of $\mathcal{A}_i$ into the class $[(q_n^i)_{n\in\mathbb{N}}]_{E_0^\omega}$.

We define a continuous operator Ξ as follows. Let $p \in 2^{\mathbb{N}}$.

1. First, we compute $\Gamma(p) := (p_n)_{n\in\mathbb{N}} \in (2^{\mathbb{N}})^{\mathbb{N}}$.

2. Second, for each $j \in \mathbb{N}$, we consider the sequence $\vec{q_j} := (q_j^i)_{i\in\mathbb{N}}$, and we compute $\delta_{p,j} := \Psi_{\vec{q_j}}(p_j) \in 2^{\mathbb{N}}$.

3. Finally, by using $(\delta_{p,j})_{j\in\mathbb{N}}$, we compute $\delta \in 2^{\mathbb{N}}$ encoding the atomic diagram of an L_1-structure $\mathcal{M}$ defined as follows. For each j, the Q_j-box of $\mathcal{M}$ is an isomorphic copy of the L_{st}-structure encoded by $\delta_{p,j}$, and this copy has domain $\{\langle j,k\rangle : k \in \mathbb{N}\}$. We set $\Xi(p) := \delta$.

It is straightforward to establish the following: the operator Ξ is a continuous embedding from the class $\mathfrak{K}$ into a countable subclass of $\mathfrak{C}_{st}$. So, by applying the relativized pullback theorem (Theorem *I*.4.14) to the continuous embedding Ξ, we recover a countable family of formulas with the desired properties. Indeed, the following holds:

- by Lemma *IV*.2.13, $\mathfrak{C}_{st}$ has a family of Σ_2^{inf} sentences Θ, which satisfies properties (a) and (b) of Theorem *IV*.2.12;

- by Theorem *I*.2.2, Φ is equivalent to a Turing X-operator, for a suitable oracle X.

Hence, we can apply Theorem *I*.4.14, and deduce that $\mathfrak{K}$ has a family $\Theta^\star$ of Σ_2^{inf} sentences Θ, which satisfies (a) and (b) of Theorem *IV*.2.12, as desired. This concludes the proof of the left-to-right direction of the Theorem.

The right-to-left direction essentially follows from previous results. Assume that $\mathfrak{K} := \{\mathcal{A}_i : i \in \mathbb{N}\}$ has a family Θ of Σ_2^{inf} formulas which satisfy the properties (a) and (b) of the theorem. Then it's easy to check that the formulas of Θ can be arranged to satisfy the existence of a collection of pairs of formulas $(\rho_{i_0}, \rho_{i_1})_{i\in\mathbb{N}}$ so that, for all structures $\mathcal{A}$ and $\mathcal{B}$ from $\mathfrak{K}$,

(*i*) $\bigcup_{i\in\mathbb{N}}\{\rho_{i_0}, \rho_{i_1}\} = \Theta$;

(*ii*) for all $i \in \mathbb{N}$, $\mathcal{A}$ satisfies exactly one formula between ρ_{i_0} and ρ_{i_1};

(*iii*) if $\mathcal{A} \not\cong \mathcal{B}$, then $(\exists j \in \mathbb{N})(\mathcal{A} \models \rho_{j_0} \iff \mathcal{B} \models \rho_{j_1})$.

We claim that, for all i, there is a continuous operator $\Gamma_i : 2^{\mathbb{N}} \to 2^{\mathbb{N}}$ such that,

$$(\forall \mathcal{S} \in \mathrm{LD}(\mathfrak{K}))((\mathcal{S} \models \rho_{i_0} \implies \Gamma_i(\mathcal{S})\ E_0\ 0^{\mathbb{N}}) \wedge (\mathcal{S} \models \rho_{i_1} \implies \Gamma_i(\mathcal{S})\ E_0\ 1^{\mathbb{N}})).$$

To prove the claim we combine (a limited case of) [BFSM20, Theorem 3.1] (see at the end of §2) with Theorem *IV*.1.1. Notice that the proof is similar to that of the direction (2) $\implies$ (1) of [BFSM20, Theorem 3]. Let $i \in \mathbb{N}$. For $k \in \{0,1\}$, without loss of generality assume that

$$\rho_{i_k} := (\exists \bar{x}) \bigwedge\!\!\!\bigwedge_{j\in J_{i_k}} \forall \bar{y} \phi_{i_k,j}(\bar{x}, \bar{y}).$$

For a finite structure $\mathcal{F}$, say that ϕ_{i_k} is *$\mathcal{F}$-compatible* via tuple $\bar{a}$ if within the domain of $\mathcal{F}$ there is no pair $(j, \bar{b})$ with $j \in J_{i_k}$ such that $\mathcal{F} \models \neg\phi_{i_k,j}(\bar{a}, \bar{b})$.
Construction.

Let $p \in 2^{\mathbb{N}}$ and denote by $\mathcal{F}_{p[s]}$ the finite structure (in the signature of $\mathfrak{K}$) encoded by the initial segment $p[s]$ of p. The continuous operator Γ_i is defined by stages. At stage 0, let $\Gamma_i(p)(0) := 0$ and $\Gamma_i(p)(1) := 1$. At stage $s+1$ we define $\Gamma_i(p)(2s)$ and $\Gamma_i(p)(2s+1)$. To this end, we distinguish three cases:

- *Case* 1: there is a tuple $\bar{c}$ so that ϕ_{i_0} is $\mathcal{F}_{p[s]}$-compatible via $\bar{c}$, and ϕ_{i_1} is not $\mathcal{F}_{p[s]}$-compatible for all tuples $< \bar{c}$. If so, let $\Gamma_i(p)(2s) = \Gamma_i(p)(2s+1) := 0$;
- *Case* 2: there is a tuple $\bar{c}$ so that ϕ_{i_1} is $\mathcal{F}_{p[s]}$-compatible via $\bar{c}$, and ϕ_{i_0} is not $\mathcal{F}_{p[s]}$-compatible for all tuples $\leqslant \bar{c}$. If so, let $\Gamma_i(p)(2s) = \Gamma_i(p)(2s+1) := 1$;
- *Case* 3: if neither of the above cases holds, then let $\Gamma_i(p)(2s) := 0$ and $\Gamma_i(p)(2s+1) := 1$.

Verification
The continuity of Γ_i immediately follows from the construction. Next, suppose that $q \in 2^{\mathbb{N}}$ encodes a copy of a structure $\mathcal{S} \in \mathfrak{K}$. By (*ii*), $\mathcal{S}$ satisfies exactly one formula between ϕ_{i_0} and ϕ_{i_1}; without loss of generality, assume that $\mathcal{S} \models \phi_{i_1}$. This means that there is a tuple $\bar{c}$ and a stage t_0 so that ϕ_{i_1} is $\mathcal{F}_{q[t]}$-compatible via $\bar{c}$, for all $t \geqslant t_0$. On the other hand, since $\mathcal{S} \not\models \phi_{i_0}$, it must be the case that for all tuples $\bar{d}$ (and, in particular, all tuples $\leqslant \bar{c}$), there must be a stage t_1 so that, for all $t \geqslant t_1$, ϕ_{i_0} is not $\mathcal{F}_{q[t]}$-compatible. So, for all sufficiently large x, $\Gamma_i(q)(x)$ is defined by performing action (2) above. Thus, $\Gamma_i(q)\ E_0\ 1^{\mathbb{N}}$, as desired. This ends the proof of the claim.

We can now compute a continuous reduction Γ from $\mathrm{LD}(\mathfrak{K})$ to E_0^ω as follows. Given $p \in 2^{\mathbb{N}}$ coding some $\mathcal{A} \in \mathfrak{K}$, let $\Gamma(p) := (p_n)_{n\in\mathbb{N}}$ be such that $p_n := \Gamma_n(p)$.

It is an easy consequence of the claim above that, if q^0 and q^1 code the same structure $\mathcal{S} \in \mathfrak{K}$, then $\Gamma(q^0)\ E_0^\omega\ \Gamma(q^1)$. To deduce that Γ is the desired reduction, suppose that q^0 codes $\mathcal{A} \in \mathfrak{K}$ and q^1 codes $\mathcal{B} \in \mathfrak{K}$ for $\mathcal{A} \not\cong \mathcal{B}$. By (*iii*), there are $j \in \mathbb{N}$ and $k \in \{0,1\}$ so that $\mathcal{A} \models \phi_{i_k}$ and $\mathcal{B} \models \phi_{i_{1-k}}$. But then, by the claim above, it follows that $\Gamma(q^0) := (q^0_n)_{n\in\mathbb{N}}$ and $\Gamma(q^1) := (q^1_n)_{n\in\mathbb{N}}$ differ on the j-th column, that is,

$$q^0_j\ E_0\ k^{\mathbb{N}} \text{ but } q^1_j\ E_0\ (1-k)^{\mathbb{N}}.$$

Thus, $\Gamma(q^0)\ \not\!\!E_0^\omega\ \Gamma(q^1)$ and this concludes the proof of Theorem *IV*.2.12. □

IV.3 Learning with the help of Z_0 and $=^+$

We conclude our examination of the learning power of combinatorial Borel equivalence relations by briefly focusing on two further examples: Z_0 and $=^+$. Here, the main goal is to finally individuate a Borel equivalence relation which is able to learn a finite family beyond the reach of our original framework.

IV.3.1 Z_0-learning

Before proceeding to a new result, we give a simple useful fact. Let $p, q \in 2^{\mathbb{N}}$ and $s \in \mathbb{N}$. We use the following notation:

$$dn(p,q;s) = \frac{|\{i \leqslant s : p\triangle q(i) = 1\}|}{s+1}.$$

Recall that the equivalence relation Z_0 is given by

$$(p\ Z_0\ q) \iff \lim_{k\to\infty} dn(p,q;k) = 0$$

Lemma IV.3.1. *Suppose that $(p\ Z_0\ q)$. Then, for every $x \in 2^{\mathbb{N}}$,*

$$\limsup_s dn(p, x; s) = \limsup_s dn(q, x; s).$$

Proof. Let $a := \limsup_s dn(q, x; s)$. It is sufficient to show that for any ε such that $0 < \varepsilon < a$, we have

$$\limsup_s dn(p, x; s) \geqslant a - \varepsilon.$$

Let $N \in \mathbb{N} \backslash \{0\}$ and fix a number s_0 such that $dn(p, q; s) < \frac{a-\epsilon}{N}$ for all $s \geqslant s_0$. There exists a sequence $(s_j)_{j \in \mathbb{N}}$, where $s_0 < s_1 < s_2 < \dots$ such that, for every $j \in \mathbb{N}$ $dn(q, x; s_j) > r - \epsilon$ for all j. Note that every s satisfies the following:

$$|\{i \leqslant s : p \triangle x(i) = 1\}| \geqslant |\{i \leqslant s : q \triangle x(i) = 1\}| - |\{i \leqslant s : q \triangle p(i) = 1\}|.$$

Hence, we have:

$$dn(p, x; s_j) \geqslant dn(q, x; s_j) - dn(p, q; s_j) > r - \epsilon - \frac{r - \epsilon}{N} = (r - \epsilon) \cdot \frac{N-1}{N}.$$

Since N was chosen as an arbitrary natural number, we deduce that for any $\delta > 0$, we have $\limsup_s dn(p, x; s) > r - \epsilon - \delta$. This implies $\limsup_s dn(p, x; s) \geqslant r - \epsilon$ and concludes the proof of the lemma. ☐

We show that learnability by finite families cannot distinguish between E_0 and Z_0:

Theorem IV.3.2. *A finite family $\mathfrak{K}$ is Z_0-learnable if and only if $\mathfrak{K}$ is E_0-learnable. That is, Z_0 and E_0 are* $\mathrm{Learn}^{<\mathbb{N}}$*-equivalent.*

Proof. Since E_0 is continuously reducible to Z_0 (see Figure I.2), every E_0-learnable family is also Z_0-learnable.

Suppose that $\mathfrak{K} := \{\mathcal{A}_i : i \in \mathbb{N}\}$ is a Z_0-learnable family. Let Γ be an operator which induces a continuous reduction from $\mathrm{LD}(\mathfrak{K})$ to Z_0. For $i \leqslant n$, we fix q_i such that Γ maps all copies of $\mathcal{A}_i$ into $[q_i]_{Z_0}$. Notice that the q_i's are pairwise not E_0-equivalent.

We fix a positive rational a_0 such that

$$a_0 < \min\{\limsup_s dn(q_i, q_j; s) : i < j \leqslant n\}.$$

There exists an oracle X and a Turing operator Φ such that $\Gamma(p) = \Phi^{X \oplus p}$ for all $p \in 2^{\mathbb{N}}$. We define an $(X \oplus \bigoplus_{i \leqslant n} q_i)$-computable operator Ψ. Let $p \in 2^{\mathbb{N}}$ and, for $s \in \mathbb{N}$, let

$$\ell[s] := \max\{n : (\forall x \leqslant n :)(\Phi^{(X \oplus p) \upharpoonright s}(x)[s] \text{ is defined})\}.$$

At a stage s, for each $i \leqslant n$, we compute the value

$$m_i[s] := \left|\{t \leqslant \ell[s] : dn(\Phi^{X \oplus p}, q_i; t) > a_0\}\right|,$$

we defined

$$j' := \min\{j \leqslant n : m_j[s] = \min\{m_i[s] : i \leqslant n\}\},$$

and we set $\Psi(p)(s) := q_{j'}(s)$. This concludes the description of the operator Ψ.

Suppose that $p \in 2^{\mathbb{N}}$ encodes a copy of a structure $\mathcal{A}_{i_0}$ for some $i_0 \leqslant n$. Then by Lemma IV.3.1, we have:

$$(\forall i \neq i_0)(\lim_s dn(\Gamma(p), q_{i_0}; s) = 0 \wedge \limsup_s dn(\Gamma(p), q_i; s) > a_0).$$

Choose a number t_0 such that $(\forall t \geqslant t_0)(dn(\Phi^{X\oplus p}, q_{i_0}; t) \leqslant a_0)$ and fix a stage s_0 with $t_0 \leqslant \ell[s_0]$. Then for all $s \geqslant s_0$, we have $m_{i_0}[s] = m_{i_0}[s_0]$. On the other hand, it is not hard to show that for every $(\forall i \neq i_0)(\lim_s m_i[s] = \infty)$. This implies that $\Psi(p)\ E_0\ q_{i_0}$ and hence Ψ provides a continuous reduction from $\mathrm{LD}(\mathfrak{K})$ to E_0. Theorem IV.3.2 is proved. □

It is known that E_0^ω is continuously reducible to Z_0 (see Figure I.2). So, E_0^ω is $\mathrm{Learn}^{\mathbb{N}}$ reducible to Z_0. The next question, which is left open, asks if the converse hold.

We leave open whether there exists a countable Z_0-learnable family, which is not $=^+$-learnable.

IV.3.2 $=^+$-learning

A distinctive feature of our learning framework is that there are finite families of structures that are not learnable. This is the case, most notably, of the pair of linear orders $\{\omega, \zeta\}$, where ζ is the order type of the integers. Such a feature is in sharp contrast with classical paradigms, since, e.g., any finite collection of recursive functions is **InfEx**-learnable. Yet, we have observed that all Borel equivalence relations so far considered are $\mathrm{Learn}^{<\mathbb{N}}$-equivalent to E_0. So, a question comes naturally: how high in the Borel hierarchy one needs to climb to reach an equivalence relation E which is able to learn a nonlearnable finite family? The next proposition shows that $=^+$ suffices. Recall that, given $(p_n)_{n\in\mathbb{N}}, (q_n)_{n\in\mathbb{N}} \in (2^{\mathbb{N}})^{\mathbb{N}}$

$$(p_n)_{n\in\mathbb{N}} =^+ (q_n)_{n\in\mathbb{N}} :\iff \{p_n : n \in \mathbb{N}\} = \{q_n : n \in \mathbb{N}\}.$$

Given a family of structures $\mathfrak{K}$, we can consider the Γ witnessing the reduction from $\mathrm{LD}(\mathfrak{K})$ to $=^+$ as a learner that has the freedom to “shuffle” and repeat finitely many times the infinitely many sequences in $2^{\mathbb{N}}$ that it outputs. This seems to be a rather relaxed notion of “convergence” for the learner, as confirmed by the next theorem.

Theorem IV.3.3. *The family $\{\omega, \zeta\}$ is $=^+$-learnable, implying that $E_0 <^{<\mathbb{N}}_{\mathsf{Learn}} =^+$ (and therefore, $E_0 \leqslant^{\mathbb{N}}_{\mathsf{Learn}} =^+$).*

Proof. Given $p \in 2^{\mathbb{N}}$ encoding a linear order with infinite domain $A \subseteq \mathbb{N}$, we have an effective and uniform (with respect to p) procedure to recover a list $(a_i)_{i\in\mathbb{N}}$, enumerating A without repetitions.

We define a Turing operator $\Psi : 2^{\mathbb{N}} \to (2^{\mathbb{N}})^{\mathbb{N}}$. For $p \in 2^{\mathbb{N}}$ and $i, s \in \mathbb{N}$, let B_s be the finite linear order, which is encoded by the finite string $p \upharpoonright s$ (note that B_s can be empty) and consider the element a_i (from the list discussed above). We define $\Psi(p) := (p_n)_{n\in\mathbb{N}}$ such that

$$p_{2i}(s) := \begin{cases} 0, & \text{if } s < i, \\ 1, & \text{otherwise}, \end{cases} \qquad p_{2i+1}(s) := \begin{cases} 0, & \text{if } a_i \notin B_s \vee a_i = \min\{a : (\forall b \in B_s)(a \leqslant_{B_s} b)\}, \\ 1, & \text{otherwise}. \end{cases}$$

Suppose that $p \in 2^{\mathbb{N}}$ encodes a copy of $\mathcal{A} \in \{\omega, \zeta\}$:

- If $\mathcal{A} \cong \zeta$, then it is clear that $\{p_n : n \in \mathbb{N}\} = \{0^i 1^{\mathbb{N}} : i \in \mathbb{N}\}$.
- If $\mathcal{A} \cong \omega$, then there is an element a_{i_0}, which is $\leqslant_{\mathcal{A}}$-least. This implies that $\{p_n : n \in \mathbb{N}\} = \{0^i 1^{\mathbb{N}} : i \in \mathbb{N}\} \cup \{0^{\mathbb{N}}\}$.

Therefore, we deduce that the family $\{\omega, \zeta\}$ is $=^+$-learnable. □

Since some of the results in this chapter are strictly related to the ones in Chapter V, we postpone the conclusions and the open questions about this chapter to 138 V.4.

V

Calculating the Mind Change Complexity of Learning Algebraic Structures

All the results in this chapter are a joint work with Nikolay Bazhenov and Luca San Mauro and can be found in [BCSM22].

Here we study the number of mind changes made by a learner while learning a given family. This was already done for formal languages (e.g., in [[FS93, AJS97]]), where the authors study which families are learnable when the number of mind changes allowed is bounded by some ordinal α. In this chapter, we put the same constraints on the problem of learning algebraic structures. We give two different characterizations of the mind change complexity of a family of structures: a topological and a combinatorial one. We explore the first in §V.1, using the notion of Id-learnability (see Definition IV.2.1), we characterize families that are learnable with α many mind changes, where α is a countable ordinal in topological terms. In §V.2 we focus on the second one, restricting our attention to particular types of families, that we call *limit-free*, and considering how the height of the partial order (poset) given by some suitable embedding relation on the family $\mathfrak{K}$ and the mind change complexity of $\mathfrak{K}$ relate. Finally, in §V.3 we address how the complexity of a learner, defined in terms of Turing reducibility, affects the number of mind changes required to learn a given family. This leaves further directions open, starting from the definition, suggested here, of the *learning degree* of a family of structures. Notice that questions regarding how the (non) computability of a learner affects the learnability of a problem were considered by Bazhenov and San Mauro in [BSM21]: here the authors showed that finite families are always learnable by an oracle computing the halting problem, while there exists a family of two structures that is not learnable by any computable learner.

V.1 A topological characterization of the mind change complexity

In this section, we study the relations between Id-learnability and α-learnability, where α is a countable ordinal. The next proposition highlights the relation between Id-learnability and 0-learnability.

Proposition V.1.1. *If a family $\mathfrak{K}$ of structures is* 0*-learnable, then it is also Id-learnable. The converse is not true, i.e., there exists a family that is Id-learnable but not* 0*-learnable.*

Proof. We first prove the implication. Let $\mathbf{M}$ be a learner that 0-learns $\mathfrak{K}$. Let $p \in 2^{\mathbb{N}}$ encode the atomic diagram of a structure $\mathcal{S}$ isomorphic to some $\mathcal{A}$ in $\mathfrak{K}$. By definition of 0-learnability,

$$(\exists s_0, i)(\forall s \geqslant s_0)(\forall t < s_0)(\mathbf{M}(\mathcal{S}\upharpoonright_s) = \ulcorner \mathcal{A}_i \urcorner \wedge \mathbf{M}(\mathcal{S}\upharpoonright_t) = ?).$$

Then, we define our continuous operator Γ as $\Gamma(p) = i_c^{\mathbb{N}}$, where i_c is the binary translation of i. Trivially given two structures p_i, p_j identifying respectively $\mathcal{A}_i$ and $\mathcal{A}_j$ we have that $\Gamma(p_i)\ Id\ \Gamma(p_j) \iff i = j$.

We now define a family $\mathfrak{G}$ that is Id-learnable but not 0-learnable. Recall from §I.1 that R_ω is the one-way infinite line and, for $i > 2$, C_i is the cycle graph of length i. Let $\mathfrak{G} := \{R_\omega\} \cup \{C_i : i > 2\}$: we first prove that $\mathfrak{G}$ is Id-learnable. A continuous reduction from LD($\mathfrak{K}$) to Id is induced by a Turing operator Ψ. Let $p \in 2^{\mathbb{N}}$ encode the atomic diagram of a structure $\mathcal{S}$ such that $\mathcal{S} \cong \mathcal{A}_i$ for some i. Let $\Psi(p)(0) := 0$ and let

$$\Psi(p)(s+1) := \begin{cases} 1 & \text{if } \mathcal{S}\upharpoonright_{s+1} \cong C_i \text{ for some } i, \\ 0 & \text{if } \mathcal{S}\upharpoonright_{s+1} \ncong C_i \wedge |E(\mathcal{S}\upharpoonright_{s+1})| > |E(\mathcal{S}\upharpoonright_s)|, \\ \langle\rangle & \text{otherwise.} \end{cases}$$

This concludes the description of Ψ. It is clear that for every $i \geqslant 1$, we have:

$$\mathcal{S} \cong R_\omega \iff \Psi(p) = 0^{\mathbb{N}} \text{ and } \mathcal{S} \cong C_i \iff \Psi(p) = 0^i 1^{\mathbb{N}}.$$

Therefore, the family $\mathfrak{G}$ is Id-learnable.

We now show that $\mathfrak{G}$ is not 0-learnable. Let $\mathcal{S} \in$ LD($\mathfrak{K}$) be such that $\mathcal{S} \cong R_\omega$ and suppose that $\mathfrak{G}$ is 0-learnable by $\mathbf{M}$. Let $s := \min\{t : \mathbf{M}(\mathcal{S}\upharpoonright_t) \neq ?\}$. If $\mathbf{M}(\mathcal{S}\upharpoonright_s) \neq \ulcorner R_\omega \urcorner$, then $\mathbf{M}$ fails to learn $\mathfrak{G}$, hence $\mathbf{M}(\mathcal{S}\upharpoonright_s) = \ulcorner R_\omega \urcorner$. On the other hand, it is straightforward to build a copy $\mathcal{S}'$ such that $\mathcal{S}\upharpoonright_s \cong \mathcal{S}'\upharpoonright_s$ and $\mathcal{S} \ncong R_\omega$. Indeed, $\mathcal{S}\upharpoonright_s$ is isomorphic to finitely many rays of finite length plus possibly finitely many isolated vertices, hence it suffices to let $\mathcal{S}' \cong C_n$ for some n large enough to include $\mathcal{S}\upharpoonright_s$. To conclude the proof it suffices to notice that $\mathbf{M}$ is forced to make one mind change to learn $\mathcal{S}'$. □

Corollary V.1.2. *The family $\mathfrak{G}$ defined in the proof of Proposition V.1.1 is proper 1-learnable.*

Proof. We define a learner $\mathbf{M}$ that 1-learns $\mathfrak{G}$: since in Proposition V.1.1 we showed that $\mathfrak{G}$ is not 0-learnable we also obtain that $\mathfrak{G}$ is proper 1-learnable. Given $\mathcal{S} \in$ LD($\mathfrak{K}$) let,

$$\mathbf{M}(\mathcal{S}\upharpoonright_s) := \begin{cases} \ulcorner C_i \urcorner & \text{if } \mathcal{S}\upharpoonright_s \cong C_i \text{ for some } i \\ \ulcorner R_\omega \urcorner & \text{otherwise.} \end{cases}$$

Notice that if $\mathcal{S} \cong R_\omega$ then $\mathbf{M}$ makes 0 mind changes. Otherwise, if $\mathcal{S} \cong C_i$ for some i, $\mathbf{M}$ changes its mind from $\ulcorner R_\omega \urcorner$ to $\ulcorner C_i \urcorner$: on the other hand, since $C_i \nsubseteq_{\mathbf{is}} \mathcal{A}$ for any $\mathcal{A} \in \mathfrak{G}\backslash\{C_i\}$, $\mathbf{M}$ does not need to change its mind anymore and this concludes the proof. □

Using non limit-free families similar to the one used in the previous proof it is not hard to show that we can define families that are Id-learnable but not n-learnable for some $n \in \mathbb{N}$. On the other hand, even for $n = 1$, it is possible to define families that are 1-learnable but not Id-learnable. We could give proof of this fact even now, but we prefer to wait until the end of this section, where we characterize α-learnability for some countable ordinal α for those families that are Id-learnable. To do so, we give a "learning-theoretic" characterization of the Cantor-Benxison derivative (see Definition I.3.8). The following definitions and results are inspired by [LS06, §3.1 and §3.2]: here Lemma V.1.3 and Theorem V.1.4 are respectively the analogs of [LS06, Lemma 3.1(1) and Theorem 3.1(1)].

Let $\mathfrak{K} = \{\mathcal{A}_i : i \in \mathbb{N}\}$, and let Γ be an operator that induces a continuous reduction from LD($\mathfrak{K}$)

to Id. Given $\sigma \in 2^{<\mathbb{N}}$, we define the *cone above* σ (with respect to Γ) as

$$N^{\Gamma}_{\sigma} = \{\mathcal{A}_i : \Gamma(\sigma) \sqsubset \mathcal{S} \cong \mathcal{A}_i\}$$

Similarly to N_σ defined in §I.1, notice that the collection of N^{Γ}_{σ} is a base for LD($\mathfrak{K}$), i.e., $\bigcup_{\sigma \in 2^{<\mathbb{N}}} N^{\Gamma}_{\sigma} \supseteq \mathrm{LD}(\mathfrak{K})$. Then for $\mathcal{A}_i \in N^{\Gamma}_{\sigma}$, let

$$CB_{\Gamma}(\mathcal{A}_i, \sigma) = \max\{\alpha : \Gamma(\mathcal{A}_i) \in \mathsf{range}(\Gamma)^{\alpha} \wedge \Gamma(\sigma) \sqsubset \Gamma(\mathcal{A}_i)\}$$

and

$$CB_{\Gamma}(\sigma) = \sup\{CB_{\Gamma}(\mathcal{A}_i, \sigma) : \mathcal{A}_i \in N^{\Gamma}_{\sigma}\}.$$

We say that $\mathcal{A}_i \in N^{\Gamma}_{\sigma}$ *identifies* N^{Γ}_{σ} if $CB_{\Gamma}(\mathcal{A}_i, \sigma) = CB_{\Gamma}(\sigma)$. If for all $j \neq i$, $\mathcal{A}_j$ does not identify N^{Γ}_{σ} we say that $\mathcal{A}_i$ *uniquely identifies (u.i.)* N^{Γ}_{σ}. We say that $\mathsf{range}(\Gamma)$ is scattered, if there exists a countable ordinal α such that $\mathsf{range}(\Gamma)^{\alpha} = \varnothing$ (see Theorem I.3.9). Trivially, if $\sigma \sqsupset \tau$, then $CB_{\Gamma}(\sigma) \leqslant CB_{\Gamma}(\tau)$. It is easy to notice that if $\mathfrak{K}$ is an Id-learnable family via some continuous operator Γ, then $\mathsf{range}(\Gamma)$ is scattered.

Lemma V.1.3. *Let $\mathfrak{K}$ be a Id-learnable family via some continuous operator Γ. Then for any $\mathcal{A}_i \in \mathfrak{K}$ and for every $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ with $\mathcal{S} \cong \mathcal{A}_i$ there exists a stage s such that $\mathcal{A}_i$ u.i. $N^{\Gamma}_{\mathcal{S}\restriction_s}$.*

Proof. Suppose there exists $\mathcal{A}_i \in \mathfrak{K}$ and $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ with $\mathcal{S} \cong \mathcal{A}_i$ such that for every $s \in \mathbb{N}$, there is a structure $\mathcal{A}_j \in N^{\Gamma}_{\mathcal{S}\restriction_s}$, where $j \neq i$ and $CB_{\Gamma}(\mathcal{A}_j, \mathcal{S}\restriction_s) \geqslant CB_{\Gamma}(\mathcal{A}_i, \mathcal{S}\restriction_s) = \alpha$. Then $\mathcal{A}_i$ is not isolated in $\mathsf{range}(\Gamma)^{\alpha}$ that, by definition, contains all structures $\mathcal{A}_j$ such that $CB_{\Gamma}(\mathcal{A}_j, \mathcal{S}\restriction_s) \geqslant \alpha$. Hence, we have $CB_{\Gamma}(\mathcal{A}_i, \mathcal{S}\restriction_s) \geqslant \alpha + 1$, contradiction. ☐

Theorem V.1.4. *Let $\mathfrak{K}$ be a Id-learnable family via some continuous operator Γ. $\mathfrak{K}$ is α-learnable if and only if $\mathsf{range}(\Gamma)^{1+\alpha} = \varnothing$.*

Proof. Suppose $\mathfrak{K}$ is α-learnable by a learner $\mathbf{M}$, i.e., set the mind change counter $c(\langle\rangle) = \alpha$. As $\mathsf{range}(\Gamma)$ is scattered, this implies that it is also nonempty (recall that the empty set is perfect by definition). This means that $\mathsf{range}(\Gamma)^{\alpha} = \varnothing$ if and only if $\alpha > 0$. By transfinite induction we prove that if $CB_{\Gamma}(\langle\rangle) > 1 + \alpha$, then $\mathbf{M}$ does not α-learn $\mathfrak{K}$ (i.e., c is not a valid mind change counter).

Suppose that for all $\beta < 1 + \alpha$ the claim holds and consider the case for $1 + \alpha$. By contradiction, suppose that $\mathsf{range}(\Gamma)^{1+\alpha} \neq \varnothing$: this means that there exists $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ and an s such that $CB_{\Gamma}(\mathcal{S}\restriction_s) > 1 + \alpha$ or, equivalently, that there is some $\mathcal{A}_i \in N^{\Gamma}_{\mathcal{S}\restriction_s}$ such that $CB_{\Gamma}(\mathcal{A}_i, \mathcal{S}\restriction_s) \geqslant 1 + \alpha + 1$. We have two cases: either $\mathbf{M}(\mathcal{S}\restriction_s) = \ulcorner \mathcal{A}_i \urcorner$ or $\mathbf{M}(\mathcal{S}\restriction_s) \neq \ulcorner \mathcal{A}_i \urcorner$.

- Suppose $\mathbf{M}(\mathcal{S}\restriction_s) = \ulcorner \mathcal{A}_i \urcorner$. Then since $\Gamma(\mathcal{A}_i)$ is not isolated in $\mathsf{range}(\Gamma)^{1+\alpha}$, there is $\Gamma(\mathcal{A}_j) \in \mathsf{range}(\Gamma)^{1+\alpha}$ with $j \neq i$ such that $CB_{\Gamma}(\mathcal{A}_j, \mathcal{S}\restriction_s) \geqslant 1 + \alpha$. Suppose that $\mathcal{S} \cong \mathcal{A}_j$. Since $\mathbf{M}$ learns $\mathfrak{K}$ by hypothesis, there is a stage s' such that $\mathbf{M}(\mathcal{S}\restriction_{s'}) = \ulcorner \mathcal{A}_j \urcorner$ and $\mathbf{M}(\mathcal{S}\restriction_{s'}) \neq \mathbf{M}(\mathcal{S}\restriction_s)$. Since this is a mind change, we have that $c(\mathcal{S}\restriction_{s'}) < c(\mathcal{S}\restriction_s)$ and $c(\mathcal{S}\restriction_{s'}) = \beta < 1 + \alpha$. On the other hand, $CB_{\Gamma}(\mathcal{A}_j, \mathcal{S}\restriction_{s'}) \geqslant 1 + \alpha$ and so $CB_{\Gamma}(\mathcal{S}\restriction_{s'}) > \beta$. By induction hypothesis for β, this is not a valid mind change counter for $\mathbf{M}$.

- Suppose $\mathbf{M}(\mathcal{S}\restriction_s) \neq \ulcorner \mathcal{A}_i \urcorner$ and $\mathcal{S} \cong \mathcal{A}_i$. Since $\mathbf{M}$ learns $\mathfrak{K}$ by hypothesis, there is a stage s' such that $\mathbf{M}(\mathcal{S}\restriction_{s'}) = \ulcorner \mathcal{A}_i \urcorner$. Similarly to the first case, we get that $c(\mathcal{S}\restriction_{s'}) < c(\mathcal{S}\restriction_s) \leqslant 1+\alpha$, but $CB_{\Gamma}(\mathcal{S}\restriction_{s'}) > 1 + \alpha$, and so c is not a mind change counter for $\mathbf{M}$.

For the other direction, suppose $\mathsf{range}(\Gamma)^{1+\alpha} = \varnothing$. Let $\mathbf{M}$ be a learner with mind change counter c such that $c(\langle\rangle) = \alpha$. Recall that if $\mathbf{M}(\sigma) \neq \mathbf{M}(\sigma^{-})$ and $\mathbf{M}(\sigma^{-}) = ?$, this is not a mind change. Let $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ and $s \in \mathbb{N}$. $\mathbf{M}$ is defined as follows:

$$\mathbf{M}(\mathcal{S}\restriction_{s+1}) := \begin{cases} ? & \text{if } \mathcal{S}\restriction_{s+1} = \langle\rangle \text{ or } CB_\Gamma(\mathcal{S}\restriction_{s+1}) < CB_\Gamma(\mathcal{S}\restriction_s) \\ \ulcorner \mathcal{A}_i \urcorner & \text{if } CB_\Gamma(\mathcal{S}\restriction_{s+1}) = CB_\Gamma(\mathcal{S}\restriction_s) \text{ and } \mathcal{A}_i \text{ u.i. } [\mathcal{S}\restriction_s]_\Gamma \\ \mathbf{M}(\mathcal{S}\restriction_s) & \text{if } CB_\Gamma(\mathcal{S}\restriction_{s+1}) = CB_\Gamma(\mathcal{S}\restriction_s) \text{ and } (\forall i)(\mathcal{A}_i \text{ does not u.i. } N^\Gamma_{\mathcal{S}\restriction_s}) \end{cases}$$

Informally, the second disjunct in the first case of $\mathbf{M}$'s definition deals with the scenario in which $\mathbf{M}$ realizes that its conjecture is wrong and changes its mind to ?.

We immediately get that $\mathbf{M}$ learns $\mathfrak{K}$. Indeed, for any $\mathcal{A}_i \in \mathfrak{K}$ and for any $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ such that $\mathcal{S} \cong \mathcal{A}_i$, by Lemma V.1.3 there is a stage s such that for all $s' \geqslant s$, $\mathcal{A}_i$ u.i. $[\mathcal{S}\restriction_{s'}]_\Gamma$. So, for any $s' > s$ the second case of $\mathbf{M}$'s definition applies and $\mathbf{M}$ correctly learns $\mathcal{A}_i$. It remains to show that c is a mind change counter for $\mathbf{M}$ and $\mathfrak{K}$.

To do so we first show that if $\mathbf{M}$ rejects a hypothesis at a stage r, it will not output it in the future. More formally, we show the following.

(i) Let $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ and $r \in \mathbb{N}$. Suppose $\mathcal{A}_i \notin [\mathcal{S}\restriction_r]_\Gamma$. If $\mathbf{M}(\mathcal{S}\restriction_r) \neq \ulcorner \mathcal{A}_i \urcorner$, then for all $t > r$, $\mathbf{M}(\mathcal{S}\restriction_t) \neq \ulcorner \mathcal{A}_i \urcorner$.

To prove (i), let $t \geqslant r$ be such that $\mathbf{M}(\mathcal{S}\restriction_t) = \ulcorner \mathcal{A}_i \urcorner$. Then there is s such that $r < s \leqslant t$ such that $\mathbf{M}(\mathcal{S}\restriction_{s-1}) \neq \ulcorner \mathcal{A}_i \urcorner$ but $\mathbf{M}(\mathcal{S}\restriction_s) = \ulcorner \mathcal{A}_i \urcorner$. Then (as $\mathbf{M}(\mathcal{S}\restriction_r) \neq \ulcorner \mathcal{A}_i \urcorner$) by the second case of $\mathbf{M}$'s definition, we have that $\mathcal{A}_i$ u.i. $N^\Gamma_{\mathcal{S}\restriction_s}$. On the other hand, $\mathcal{A}_i \notin [\mathcal{S}\restriction_r]_\Gamma$ and consequently, by continuity of Γ, $\mathcal{A}_i \notin N^\Gamma_{\mathcal{S}\restriction_s}$, contradiction. This concludes the proof of (i).

We now show what happens if the second case of $\mathbf{M}$'s definition applies.

(ii) Let $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ and $s \in \mathbb{N}$. If the second case of $\mathbf{M}$'s definition applies at $\mathcal{S}\restriction_{s+1}$, then there is no mind change: that is, either $\mathbf{M}(\mathcal{S}\restriction_{s+1}) = \mathbf{M}(\mathcal{S}\restriction_s)$ or $\mathbf{M}(\mathcal{S}\restriction_s) = ?$

To prove (ii), suppose that $\mathbf{M}(\mathcal{S}\restriction_s) = \ulcorner \mathcal{A}_j \urcorner$ where $j \neq i$, $\mathcal{A}_i$ u.i. $[\mathcal{S}\restriction_{s+1}]_\Gamma$ and $CB_\Gamma(\mathcal{S}\restriction_{s+1}) = CB_\Gamma(\mathcal{S}\restriction_s)$. Let $r < s+1$ be the least stage such that $\mathbf{M}(\mathcal{S}\restriction_r) = \ulcorner \mathcal{A}_j \urcorner$. The second case of $\mathbf{M}$'s definition implies that $\mathcal{A}_j$ u.i. $[\mathcal{S}\restriction_r]_\Gamma$. But since $\mathcal{A}_i$ u.i. $[\mathcal{S}\restriction_{s+1}]_\Gamma$ and $i \neq j$, we immediately get that $CB_\Gamma(\mathcal{S}\restriction_{s+1}) < CB_\Gamma(\mathcal{S}\restriction_r)$, and so $r < s$ as $CB_\Gamma(\mathcal{S}\restriction_{s+1}) = CB_\Gamma(\mathcal{S}\restriction_s)$ by the second case of $\mathbf{M}$'s definition. So $CB_\Gamma(\mathcal{S}\restriction_s) < CB_\Gamma(\mathcal{S}\restriction_r)$. By the first case of $\mathbf{M}$'s definition, there is a stage m with $r < m < s$ such that $\mathbf{M}(\mathcal{S}\restriction_m) = ?$ and $\mathcal{A}_j \notin [\mathcal{S}\restriction_m]_\Gamma$. By ($i$), we obtain that $\mathbf{M}(\mathcal{S}\restriction_s) \neq \ulcorner \mathcal{A}_j \urcorner$, a contradiction. This concludes the proof of (ii).

We derive that $\mathbf{M}$ changes its mind only if the first clause of $\mathbf{M}$'s definition applies (i.e., the third clause clearly does not imply a mind change, and the second one was excluded by (ii)). As in the first part of the proof, recall that $\mathsf{range}(\Gamma)^\alpha = \varnothing$ if and only if $\alpha > 0$: this implies that for all $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ and for all $s \in \mathbb{N}$. $CB_\Gamma(\mathcal{S}\restriction_s) > 0$. So, whenever the first case of $\mathbf{M}$'s definition applies, we have that $0 < c(\mathcal{S}\restriction_{s+1}) = CB_\Gamma(\mathcal{S}\restriction_{s+1}) < CB_\Gamma(\mathcal{S}\restriction_s) = c(\mathcal{S}\restriction_s)$, and so c is a mind change counter for $\mathbf{M}$ and $\mathfrak{K}$, i.e., $\mathbf{M}$ α-learns $\mathfrak{K}$. This concludes the proof of Theorem V.1.4. □

Combining Proposition V.1.1 and Theorem V.1.4 we derive the following corollary that characterizes 0-learnability in terms of Id-learnability.

Corollary V.1.5. *$\mathfrak{K}$ is 0-learnable if and only if $\mathfrak{K}$ is Id-learnable via some continuous operator Γ such that* $\mathsf{range}(\Gamma)^1 = \varnothing$, *i.e., all points in* $\mathsf{range}(\Gamma)$ *are isolated.*

The last corollary shows that Id-learnability "contains" all 0-learnable families. It suffices to move to 1-lernability to show that this is not true anymore. As an example, let $\mathfrak{K} = \{\mathcal{A}, \mathcal{B}\}$, where $\mathcal{A} \cong \bigotimes^\infty C_3$, while $\mathcal{B} \cong \mathcal{A} \bigotimes C_4$: it is fairly easy to show that $\mathfrak{K}$ is proper 1-learnable (and actually we can prove it using Theorem V.2.3 stated in the next section). On the other hand, suppose that $\mathfrak{K}$ is Id-learnable via some continuous operator Γ. Then, as $\mathfrak{K}$ contains only two structures, the points in $\mathsf{range}(\Gamma)$ are two, and they are clearly isolated: Theorem V.1.4 implies that $\mathfrak{K}$ is

0-learnable, getting the desired contradiction.

V.2 Learnability and posets

In this section, we discuss some results about the relation between the number of mind changes made while learning a family $\mathfrak{K}$ and the structural properties of $\mathfrak{K}$. The first remark is that it is not always possible to define an upper bound to the number of mind changes. This may happen for two reasons. Either the family is not learnable at all, or the family is learnable but at any finite stage, it is always possible to extend the copy built so far to a structure different from the one the learner is conjecturing. To study when it is possible to define such a bound, we provide the following definition. Recall that $\hookrightarrow$ denotes the embedding relation between structures.

Definition V.2.1. Let $\mathcal{A}$ and $\mathcal{B}$ be two structures. $\mathcal{A}$ *finitely embeds* in $\mathcal{B}$ (notation $\mathcal{A} \hookrightarrow_{fin} \mathcal{B}$) if for all s, $\mathcal{A}\restriction_s \hookrightarrow \mathcal{B}$.

In general, $\hookrightarrow_{fin}$ is a preorder on $\mathfrak{K}$. In some nice cases, for example, if every structure in $\mathfrak{K}$ is finite, such relations are *partial orders* (posets). On the other hand, anti-symmetry is not guaranteed in the infinite case, as we may have two infinite structures $\mathcal{A}, \mathcal{B} \in \mathfrak{K}$ such that $\mathcal{A} \hookrightarrow_{fin} \mathcal{B}$ and vice versa but $\mathcal{A} \ncong \mathcal{B}$. In this section, we consider only families on which $\hookrightarrow_{fin}$ is a partial order, and we denote it by $(\mathfrak{K}, \hookrightarrow_{fin})$ (whenever we use this notation, we assume that $\hookrightarrow_{fin}$ is a partial order on $\mathfrak{K}$). We say that $\mathcal{A} \in \mathfrak{K}$ *has height* n, denoted by $\text{height}(\mathcal{A}) = n$, if in the corresponding poset there exists a chain (i.e., a totally ordered set) of length n having $\mathcal{A}$ as a maximal element but no chain of greater length has $\mathcal{A}$ as a maximal element. In case the structure of the greatest height in the poset has height n, we say that $(\mathfrak{K}, \hookrightarrow_{fin})$ has height n, and we denote this by $\text{height}((\mathfrak{K}, \hookrightarrow_{fin})) = n$.

Definition V.2.2. Let $\mathfrak{K}$ be a family of structures such that $\text{height}((\mathfrak{K}, \hookrightarrow_{fin})) = n$. Let $\mathfrak{K}_{=n} := \{\mathcal{A} \in \mathfrak{K} : \text{height}(\mathcal{A}) = n\}$. We say that $\mathfrak{K}$ is *limit-free* if

$$(\forall n)(\forall \mathcal{A} \in \mathfrak{K}_{=m})(\forall \mathcal{S} \cong \mathcal{A})(\exists s)(\{\mathcal{B} \in \mathfrak{K} : \mathcal{S}\restriction_s \hookrightarrow \mathcal{B}\} \cap \mathfrak{K}_{=n} = \{\mathcal{A}\}).$$

In this case we say that $\mathcal{A}$ is n-minimal on $\mathcal{S}\restriction_s$.

Intuitively, a limit-free $\mathfrak{K}$ allows a learner $\mathbf{M}$ not to change its mind between two structures having the same height. Indeed, given $\mathcal{S} \in \text{LD}(\mathfrak{K})$, if $\mathcal{S}\restriction_t \hookrightarrow \mathcal{A}$ where $\text{height}(\mathcal{A}) = n$, $\mathbf{M}$ can wait for a stage s such that $\mathcal{S}\restriction_s \hookrightarrow \mathcal{A}$, and $\mathcal{A}$ is the unique structure in $\mathfrak{K}_{=m}$ for some $m \leqslant n$ where $n = \text{height}((\mathfrak{K}, \hookrightarrow_{fin}))$. A trivial observation is that all finite families are clearly limit-free. For clarity, notice that the family $\mathfrak{G} := \{R_\omega\} \cup \{C_i : i > 2\}$ in Proposition V.1.1 is non limit-free and this is witnessed by R_ω. Indeed, given $\mathcal{S} \in \text{LD}(\mathfrak{K})$ such that $\mathcal{S} \cong R_\omega$, for all s, $\{\mathcal{B} \in \mathfrak{K} : \mathcal{S}\restriction_s \hookrightarrow \mathcal{B}\} \cap \mathfrak{K}_{=1}$ contains infinitely many structures, i.e., R_ω and for $i > 2$ all the C_i's such that $\mathcal{S}\restriction_s \hookrightarrow C_i$, which are cofinitely many. The next result gives a characterization of n-learnability for limit-free families.

Theorem V.2.3. Let $\mathfrak{K}$ be a limit-free family of structures. Then $\mathfrak{K}$ is n-learnable if and only if $\text{height}((\mathfrak{K}, \hookrightarrow_{fin})) \leqslant n + 1$.
Consequently, $\mathfrak{K}$ is proper n-learnable if and only if $\text{height}((\mathfrak{K}, \hookrightarrow_{fin})) = n + 1$.

Proof. Before starting, recall that the definition of limit-free already implies that $\hookrightarrow_{fin}$ is a partial order on $\mathfrak{K}$ and that $\text{height}((\mathfrak{K}, \hookrightarrow_{fin}))$ is bounded.

For left-to-right direction, by contradiction, suppose that $\mathfrak{K}$ is n-learnable by a learner $\mathbf{M}$ and $\text{height}((\mathfrak{K}, \hookrightarrow_{fin})) = n + 2$. This means that there exists a chain of the form $\mathcal{A}_0 \hookrightarrow_{fin} \ldots \hookrightarrow_{fin} \mathcal{A}_{n+1}$. Towards a contradiction, we shall now construct $\mathcal{S} \in \text{LD}(\mathfrak{K})$ so that $\mathbf{M}$ does not n-learn $\mathcal{S}$. The construction of $\mathcal{S}$ is by stages:

- *Stage* 0. No number is *injured*, and we start building $\mathcal{S}$ as a copy of $\mathcal{A}_0$.
- *Stage* $s+1$. We are currently building $\mathcal{S}$ as a copy of $\mathcal{A}_i$, where i is the least number that is not injured. If $i > n$, we do nothing. Otherwise, we distinguish two cases. If $\mathbf{M}(\mathcal{S}\upharpoonright_s) = \ulcorner\mathcal{A}_i\urcorner$, we declare i injured, and we start building $\mathcal{S}$ as a copy of $\mathcal{A}_{i+1}$ (note that switching from copy $\mathcal{A}_i$ to copy $\mathcal{A}_{i+1}$ is always allowed, as $\mathcal{A}_i \hookrightarrow_{fin} \mathcal{A}_{i+1}$). Otherwise, we continue building $\mathcal{S}$ as a copy of $\mathcal{A}_i$.

It follows immediately from the construction that, if there is $i \leqslant n$ which is never injured, then $\mathcal{S}$ is eventually a copy of $\mathcal{A}_i$ which is not learned by $\mathbf{M}$, a contradiction. On the other hand, if all $i \leqslant n$ are injured, then $\mathcal{S}$ is eventually be a copy of $\mathcal{A}_{n+1}$ which forces $\mathbf{M}$ to have more than n mind changes, which is again a contradiction.

Let $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ and let s be such that $\mathcal{S}\upharpoonright_s \hookrightarrow \mathcal{A}_0$. If $\mathbf{M}(\mathcal{S}\upharpoonright_s) \neq \ulcorner\mathcal{A}_0\urcorner$, then $\mathbf{M}$ may fail to learn $\mathfrak{K}$ (i.e., $\mathcal{S}$ may be a copy of $\mathcal{A}_0$), so $\mathbf{M}(\mathcal{S}\upharpoonright_s) = \ulcorner\mathcal{A}_0\urcorner$ and $c(\mathcal{S}\upharpoonright_s) = n-1$. Suppose at stage $s_1 > s$, $\mathcal{S}\upharpoonright_{s_1} \not\hookrightarrow \mathcal{A}_0$ but $\mathcal{S}\upharpoonright_{s_1} \hookrightarrow \mathcal{A}_1$. For a similar reason, in order to learn $\mathfrak{K}$, $\mathbf{M}$ is forced to change its mind to $\mathcal{A}_1$. Proceeding in this fashion for all $\mathcal{A}_i$ with $i \leqslant n+1$, we may get at a stage in which $\mathbf{M}(\mathcal{S}\upharpoonright_{s_n}) = \mathcal{A}_n$ and $c(\mathcal{S}\upharpoonright_{s_n}) = 0$. On the other hand, $\mathcal{S}$ may be a copy of $\mathcal{A}_{n+1}$, contradicting that $\mathfrak{K}$ is n-learnable.

For the right-to-left direction, suppose the most general case when $\mathrm{height}((\mathfrak{K}, \hookrightarrow_{fin})) = n+1$, and let $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$. We define a learner $\mathbf{M}$ that n-learns the family. Set $c = n$: at stage s:

$$\mathbf{M}(\mathcal{S}\upharpoonright_s) = \begin{cases} \ulcorner\mathcal{A}\urcorner & \text{if } \mathcal{A} \text{ is } m\text{-minimal on } \mathcal{S}\upharpoonright_s \text{ for some } m \geqslant 1 \\ ? & \text{otherwise} \end{cases}$$

To see that $\mathbf{M}$ learns $\mathfrak{K}$, notice that $\mathcal{A} \in \mathfrak{K}$ is m-minimal for some m, and since $\mathfrak{K}$ is limit-free there is a stage s where $\mathcal{A}$ is the only structure in $\mathfrak{K}_{=m}$ such that $\mathcal{S}\upharpoonright_s \hookrightarrow \mathcal{A}$. As $m < n+2$, $\mathbf{M}$ eventually stabilizes to the correct conjecture in a finite amount of steps. To check that c is a mind change counter for $\mathbf{M}$ it is sufficient to notice that $\mathbf{M}$ changes its mind only in the first case of the definition. As the height of the poset is at most $n+1$, and $\mathbf{M}$ makes no mind change to identify the 1-minimal structure, the number of mind changes is at most n. ☐

V.3 Learner's complexity and number of mind changes

In this section, we show that for certain families, the complexity of the learner plays a role in the number of mind changes during the learning process.

Theorem V.3.1. *For any c.e. noncomputable set X, there exists a countable family of graphs $\mathfrak{K}$ such that:*

- *$\mathfrak{K}$ is 0-learnable by an A-computable learner if and only if $X \leqslant_T A$;*
- *$\mathfrak{K}$ is 1-learnable by a computable learner.*

Proof. Let X be a noncomputable c.e. set. We dynamically build $\mathfrak{K} = \{G_e : e \in \mathbb{N}\}$ as follows. In the construction below, when we write "add C_i to G_e" we mean that we append to some isolated vertex of G_e a copy of C_i

- *Stage* 0. for any $e \in \mathbb{N}$, let $G_{2e} \cong \mathsf{D} \bigotimes C_{4e}$. Similarly, $G_{2e+1} \cong \mathsf{D} \bigotimes C_{4e+1}$. Note that G_{2e} and G_{2e+1} are not isomorphic and are incomparable with respect to $\hookrightarrow_{fin}$.
- *Stage* $s+1$. if $e \in X_{s+1} \setminus X_s$, then
  - add C_{4e} to G_{2e+1} and C_{4e+1} to G_{2e};
  - add C_{4e+2} to G_{2e} and C_{4e+3} to G_{2e+1}.

Informally, if $e \in X_{s+1} \setminus X_s$ we first modify G_{2e} and G_{2e+1} so that they are isomorphic, and then we make them again nonisomorphic and incomparable with respect to $\hookrightarrow_{fin}$.

We first prove the first point of the theorem, starting from the right-to-left direction. Let A be such that $X \leqslant_T A$. We show that $\mathfrak{K}$ is 0-learnable by an A-computable learner. Let $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ and let $s > 1$. Then the following A-computable learner clearly 0-learns $\mathfrak{K}$.

$$\mathbf{M}^A(\mathcal{S}\restriction_s) = \begin{cases} ? & \text{if } (\forall n)(C_n \not\hookrightarrow \mathcal{S}\restriction_s) \vee (e \in X \wedge C_{4e+i} \hookrightarrow \mathcal{S}\restriction_s \text{ where} \\ & i \in \{0,1\} \wedge C_{4e+j} \not\hookrightarrow \mathcal{S}\restriction_s \text{ where } j \in \{2,3\}) \\ \ulcorner G_{2e} \urcorner & \text{if } (C_{4e+2} \hookrightarrow \mathcal{S}\restriction_s) \vee (e \notin X \wedge C_{4e} \hookrightarrow \mathcal{S}\restriction_s) \\ \ulcorner G_{2e+1} \urcorner & \text{if } (C_{4e+3} \hookrightarrow \mathcal{S}\restriction_s) \vee (e \notin X \wedge C_{4e+1} \hookrightarrow \mathcal{S}\restriction_s) \end{cases}$$

Informally, $\mathbf{M}^A(\mathcal{S}\restriction_s) = ?$ if either $\mathcal{S}\restriction_s$ contains no cycle, or the cycle(s) in $\mathcal{S}\restriction_s$ allow $\mathbf{M}^A$ only to distinguish that either $\mathcal{S} \cong G_{2e}$ or $\mathcal{S} \cong G_{2e+1}$. In the first case $\mathbf{M}^A$ waits for a stage $t > s$ such that $\mathcal{S}\restriction_t$ contains some cycle (the existence of such a stage is guaranteed by $\mathfrak{K}$'s construction). In the second case, as $e \in X$ and by $\mathfrak{K}$'s construction, $\mathbf{M}^A$ knows that only one between C_{4e+2} and C_{4e+3} is in $\mathcal{S}$ and this allows it to output the correct conjecture, i.e., depending on the length of the cycle, the first disjunct of the second or third case of $\mathbf{M}^A$'s definition applies. Trivially, if $\mathcal{S}\restriction_s$ contains C_{4e+2} or C_{4e+3} then $\mathbf{M}^A$ immediately outputs the correct conjecture, same if $\mathcal{S}\restriction_s$ contains C_{4e} or C_{4e+1} and $e \notin X$.

For the left-to-right direction, assume that there exists an A-computable learner $\mathbf{M}^A$ that 0-learns $\mathfrak{K}$ but $X \not\leqslant_T A$. We show that if it is the case, A can enumerate $\mathbb{N} \setminus X$, contradicting the fact that $X \not\leqslant_T A$. For any $e \in \mathbb{N}$, let $\mathcal{B}_{e,n}$ be a structure isomorphic to C_{4e} and n many disjoint vertices. It is clear that for any n, $\mathcal{B}_{e,n} \hookrightarrow G_{2e}$, independently of the presence/absence of e in X. In other words, there exists $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ such that $\mathcal{S} \cong G_{2e}$ and for any $n \in \mathbb{N}$, $\mathcal{B}_{e,n} \hookrightarrow \mathcal{S}$. This means that there exists an $n \in \mathbb{N}$ such that $\mathbf{M}^A(\mathcal{B}_{e,n}) = \ulcorner G_{2e} \urcorner$ and for all $n' > n$ $\mathbf{M}^A(\mathcal{B}_{e,n'}) = \ulcorner G_{2e} \urcorner$. Let f be the following function computing the characteristic function of X:

$$f(e) = \begin{cases} 1 & \text{if } e \in X \\ 0 & \text{if } (\exists n)(\mathbf{M}^A(\mathcal{B}_{e,n}) = \ulcorner G_{2e} \urcorner \wedge n = \min_m \mathbf{M}^A(\mathcal{B}_{e,m}) \neq ?) \end{cases}$$

To show that the second case of f's definition is correct, suppose that there is n such that $\mathbf{M}^A(\mathcal{B}_{e,n}) = \ulcorner G_{2e} \urcorner \wedge n = \min_m \mathbf{M}^A(\mathcal{B}_{e,m}) \neq ?$ but $e \in X$. Then, there is some $\mathcal{S}' \in \mathrm{LD}(\mathfrak{K})$ such that $\mathcal{B}_{e,n} \hookrightarrow \mathcal{S}'$ (i.e., by $\mathfrak{K}$'s construction, $C_{4e}, C_{4e+3} \hookrightarrow \mathcal{S}'$). This means that if $\mathbf{M}^A(\mathcal{B}_{e,n}) \downarrow = \ulcorner G_{2e} \urcorner$, $\mathbf{M}^A$ needs to change its mind to $\ulcorner G_{2e+1} \urcorner$ contradicting that $\mathbf{M}^A$ 0-learns the family. We derive that f is clearly A-computable and witness that $X \leqslant_T A$ getting the desired contradiction.

It remains to show that a computable learner $\mathbf{M}$ can 1-learn the family. Let $\mathcal{S} \in \mathrm{LD}(\mathfrak{K})$ and $s > 1$:

$$\mathbf{M}(\mathcal{S}\restriction_s) = \begin{cases} ? & \text{if } (\forall n)(C_n \not\hookrightarrow \mathcal{S}\restriction_s) \\ \ulcorner G_{2e} \urcorner & \text{if } (\exists n)(C_n \hookrightarrow \mathcal{S}\restriction_s \wedge (n = 4e \vee n = 4e+2)) \\ \ulcorner G_{2e+1} \urcorner & \text{if } (\exists n)(C_n \hookrightarrow \mathcal{S}\restriction_s \wedge (n = 4e+1 \vee n = 4e+3)) \end{cases}$$

$\mathbf{M}$ can change its mind only in the second and the third case and this may happen at most a single time, i.e., in case $C_{4e} \hookrightarrow \mathcal{S}\restriction_s$ and there exists $s' > s$ such that $C_{4e+3} \hookrightarrow \mathcal{S}\restriction_{s'}$ (similarly, for C_{4e+1} instead of C_{4e} and C_{4e+2} instead of C_{4e+3}). This shows that $\mathfrak{K}$ is 1-learnable by $\mathbf{M}$. □

V.4 Conclusions and open questions (Chapter IV and Chapter V)

The investigation conducted in Chapter IV has been fueled by the discovery of a connection between algorithmic learning theory and descriptive set theory. Namely, we proved that the task of learning a given family of algebraic structures (up to isomorphism) is equivalent to the task of defining a suitable continuous reduction to E_0. Then, we carefully analyzed the learning power of a number of well known benchmark Borel equivalence relations via the novel notion of countable-learning reducibility and finite-learning reducibility. This provided a solution to the lack of a method for calibrating the complexity of nonlearnable families and showed a correspondence between "classical" learning paradigms and E-learnability.

In Chapter V we studied different characterizations of mind change complexity.

There are numerous open questions in the novel area of algorithmic learning theory for algebraic structures, but we conclude this section (and actually, this thesis) mentioning some of them that originate from the above results.

In Chapter IV's introduction, we argued that it is natural to discuss the learning power of other Borel equivalence relations. There is a wide choice, even if one restricts to a small fragment of the Borel hierarchy.

Question V.4.1. For example, what happens if we restrict to $\mathbf{\Pi}^0_3$ equivalence relations (see [Gao09])?

The syntactic characterization of learnability notions, such as [BFSM20, Theorem 3.1] (see at the end of Part 2) and Theorem *IV*.2.12 is often very useful in proving the (non)learnability of families of structures.

Question V.4.2. Provide a syntactic characterization for $=^+$-learnability and Z_0-learnability.

Observe that our original framework was inherently limited to the countable case since the learner had to provide a conjecture (i.e., a finite object) for each isomorphism type of the observed family.

Question V.4.3. Study E-learnability for uncountable families.

The following direction was also suggested by one of the reviewers of [BCS23], and it is one of the main outcomes given by the interplay between algorithmic learning theory and descriptive set theory.

Question V.4.4. Locate analogs of classic learning criteria (such as partial learning or non-U-shape learning) in the learning hierarchy. (For now, in addition to learnability, we have addressed this issue in the present chapter for α-learnability).

The characterization given in §V.2 in terms of the height of the poset is restricted to limit-free families.

Question V.4.5. Extend (possibly a modification of) such a characterization to non limit-free families.

The proof of Theorem *V*.3.1 suggests the definition of *0-learning spectrum*: this is in analogy with several established notions from computable structure theory (see, e.g., [FKM10, BFRSM21]). Indeed, for any noncomputable c.e. set X we defined a family $\mathfrak{K}$ such that the oracles A for which there exists an A-computable learner that 0-learns $\mathfrak{K}$ coincide with the Turing cone above X,

$$\mathsf{Spec}_{0\text{-learn}}(\mathfrak{K}) = \{A : (\exists \mathbf{M})(\mathbf{M}^A \text{ 0-learns } \mathfrak{K})\} = \{A : X \leqslant_T A\}.$$

In the same spirit, it is natural to define $\mathsf{Spec}_{n\text{-learn}}(\mathfrak{K})$ as the collection of all oracles that allow learning $\mathfrak{K}$ with n many mind changes.

Question V.4.6. Analyze the structural properties of such spectra.

Bibliography

[ADK21] Paul-Elliot Anglès D'Auriac and Takayuki Kihara. A comparison of various analytic choice principles. *The Journal of Symbolic Logic*, 86(4):1452–1485, 2021.

[AJS97] Andris Ambainis, Sanjay Jain, and Arun Sharma. Ordinal mind change complexity of language identification. In Shai Ben-David, editor, *Computational Learning Theory, Third European Conference, EuroCOLT'97*, volume 1208 of *LNCS*, pages 301–315, Berlin, 1997. Springer.

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

[AMS92] Ron Aharoni, Menachem Magidor, and Richard A Shore. On the strength of König's duality theorem for infinite bipartite graphs. *Journal of Combinatorial Theory, Series B*, 54(2):257–290, 1992.

[Ash86] C.J. Ash. Stability of recursive structures in arithmetical degrees. *Annals of Pure and Applied Logic*, 32:113–135, 1986.

[BCS23] Nikolay Bazhenov, Vittorio Cipriani, and Luca San Mauro. Learning algebraic structures with the help of borel equivalence relations. *Theoretical Computer Science*, 951:113762, 2023.

[BCSM22] Nikolay Bazhenov, Vittorio Cipriani, and Luca San Mauro. Calculating the mind change complexity of learning algebraic structures. In *Revolutions and Revelations in Computability: 18th Conference on Computability in Europe, CiE 2022, Swansea, UK, July 11–15, 2022, Proceedings*, pages 1–12. Springer, 2022.

[BDP12] Vasco Brattka, Matthew De Brecht, and Arno Pauly. Closed choice and a uniform low basis theorem. *Annals of Pure and Applied Logic*, 163(8):986–1008, 2012. Continuity, Computability, Constructivity: From Logic to Algorithms.

[Bea76] Dwight R. Bean. Effective coloration. *The Journal of Symbolic Logic*, 41(2):469–480, 1976.

[BFRSM21] Nikolay Bazhenov, Ekaterina Fokina, Dino Rossegger, and Luca San Mauro. Degrees of bi-embeddable categoricity. *Computability*, 10(1):1–16, 2021.

[BFSM20] Nikolay Bazhenov, Ekaterina Fokina, and Luca San Mauro. Learning families of algebraic structures from informant. *Information and Computation*, 275:104590, 2020.

[BG89] Richard Beigel and William I. Gasarch. On the complexity of finding the chromatic number of a recursive graph i: the bounded case. *Annals of Pure and Applied Logic*, 45(1):1–38, 1989.

[BG08] Vasco Brattka and Guido Gherardi. Borel complexity of topological operations on computable metric spaces. *Journal of Logic and Computation*, 19(1):45–76, 08 2008.

[BG09] Vasco Brattka and Guido Gherardi. Weihrauch degrees, omniscience principles and weak computability. *Journal of Symbolic Logic*, 76, 05 2009.

[BG21] Vasco Brattka and Guido Gherardi. Completion of choice. *Annals of Pure and Applied Logic*, 172(3):102914, 2021.

[BGH15] Vasco Brattka, Guido Gherardi, and Rupert Hölzl. Probabilistic computability and choice. *Information and Computation*, 242:249–286, 2015.

[BGM12] Vasco Brattka, Guido Gherardi, and Alberto Marcone. The Bolzano-Weierstrass theorem is the jump of weak König's lemma. *Annals of Pure and Applied Logic*, 163:623–655, 2012.

[BGP21] Vasco Brattka, Guido Gherardi, and Arno Pauly. Weihrauch complexity in computable analysis. In *Handbook of computability and complexity in analysis*, pages 367–417. Springer, 2021.

[BHK17] Vasco Brattka, Rupert Hölzl, and Rutger Kuyper. Monte Carlo computability. In *34th Symposium on Theoretical Aspects of Computer Science (STACS 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.

[BHW21] Zach BeMent, Jeffry L. Hirst, and Asuka Wallace. Reverse mathematics and Weihrauch analysis motivated by finite complexity theory. *Computability*, 10(4):343–354, 2021. arXiv 2105.01719.

[BKMP16] Vasco Brattka, Akitoshi Kawamura, Alberto Marcone, and Arno Pauly. Measuring the complexity of computational content (Dagstuhl Seminar 15392). *Dagstuhl Reports*, 5(9):77–104, 2016.

[BP16] Vasco Brattka and Arno Pauly. On the algebraic structure of Weihrauch degrees. *Logical Methods in Computer Science*, 14, 04 2016.

[Bra04] Vasco Brattka. Effective Borel measurability and reducibility of functions. *Mathematical Logic Quarterly*, 51(1):19–44, 2004.

[BSM21] Nikolay Bazhenov and Luca San Mauro. On the Turing complexity of learning finite families of algebraic structures. *Journal of Logic and Computation*, 31(7):1891–1900, 2021.

[Cam05] Riccardo Camerlo. Continua and their σ-ideals. *Topology and its Applications*, 150(1-3):1–18, 2005.

[CCKM04] Wesley Calvert, Desmond F. Cummins, Julia F. Knight, and Sara Miller. Comparing classes of finite structures. *Algebra and Logic*, 43:374–392, 2004.

[CG01] Riccardo Camerlo and Su Gao. The completeness of the isomorphism relation for countable Boolean algebras. *Transactions of the American Mathematical Society*, 353(2):491–518, 2001.

[CH20] Antonin Callard and Mathieu Hoyrup. Descriptive complexity on non-Polish spaces. In *STACS 2020-37th Symposium on Theoretical Aspects of Computer Science*, volume 154, page 16, 2020.

[CJAJFK00] Chris J. Ash and Julia F. Knight. *Computable Structures and the Hyperarithmetical Hierarchy*, volume 144 of *Studies in Logic and the Foundations of Mathematics*. Elsevier Science B.V., Amsterdam, 2000.

[CMV22] Vittorio Cipriani, Alberto Marcone, and Manlio Valenti. The Weihrauch lattice at the level of $\boldsymbol{\Pi}^1_1\text{-}\mathsf{CA}_0$: the Cantor-Bendixson theorem. *Submitted, available at https://arxiv.org/abs/2210.15556*, 2022.

[CP22] Vittorio Cipriani and Arno Pauly. The uniform computational content of the (induced) subgraph problem. *in preparation*, 2022.

[CP23] Vittorio Cipriani and Arno Pauly. The complexity of finding supergraphs. *accepted for publication at the Computability in Europe proceedings*, 2023.

[CS83] John Case and Carl Smith. Comparison of identification criteria for machine inductive inference. *Theoretical Computer Science*, 25(2):193–220, 1983.

[DM22] Damir D Dzhafarov and Carl Mummert. *Reverse Mathematics: Problems, Reductions, and Proofs*. Springer Nature, 2022.

[DSY23] Damir D. Dzhafarov, Reed Solomon, and Keita Yokoyama. On the first-order parts of problems in the Weihrauch degrees. *Available at `https://arxiv.org/abs/2301.12733`*, 2023.

[FKM10] Ekaterina Fokina, Iskander Kalimullin, and Russell Miller. Degrees of categoricity of computable structures. *Archive for Mathematical Logic*, 49(1):51–67, 2010.

[FKSM19] Ekaterina Fokina, Timo Kötzing, and Luca San Mauro. Limit learning equivalence structures. In Aurélien Garivier and Satyen Kale, editors, *Proceedings of the 30th International Conference on Algorithmic Learning Theory*, volume 98 of *Proceedings of Machine Learning Research*, pages 383–403, Chicago, Illinois, 22–24 Mar 2019. PMLR.

[Fri75] Harvey Friedman. Some systems of second order arithmetic and their use. In *Proceedings of the international congress of mathematicians (Vancouver, BC, 1974)*, volume 1, pages 235–242. Citeseer, 1975.

[FS89] Harvey Friedman and Lee Stanley. A Borel reducibility theory for classes of countable structures. *J. Symbolic Logic*, 54(3):894–914, 1989.

[FS93] R. Freivalds and C.H. Smith. On the role of procrastination in machine learning. *Information and Computation*, 107(2):237–271, 1993.

[Gao09] Su Gao. *Invariant Descriptive Set Theory*. CRC Press, Boca Raton, FL, 2009.

[GHM15] Kirill Gura, Jeffry L Hirst, and Carl Mummert. On the existence of a connected component of a graph. *Computability*, 4(2):103–117, 2015.

[GKP17] Vassilios Gregoriades, Tamas Kispeter, and Arno Pauly. A comparison of concepts from computable analysis and effective descriptive set theory. *Mathematical Structures in Computer Science*, 27(8):1414–1436, 2017.

[Gly85] Clark Glymour. Inductive inference in the limit. *Erkenntnis*, 22:23–31, 1985.

[GM09] Guido Gherardi and Alberto Marcone. How incomputable is the separable Hahn-Banach theorem? *Notre Dame Journal of Formal Logic*, 50(4):393 – 425, 2009.

[Goh19] Jun Le Goh. Some computability-theoretic reductions between principles around ATR_0, 2019.

[Goh20] Jun Le Goh. Embeddings between well-orderings: Computability-theoretic reductions. *Annals of Pure and Applied Logic*, 171(6):102789, 2020.

[GPV21] Jun Le Goh, Arno Pauly, and Manlio Valenti. Finding descending sequences through ill-founded linear orders. *The Journal of Symbolic Logic*, 86(2):817–854, 2021.

[Har91] David Harel. Hamiltonian paths in infinite graphs. In *Proceedings of the twenty-third annual ACM symposium on Theory of Computing*, pages 220–229, 1991.

[Hir19] Jeffry Hirst. Leaf management. *Computability*, 9:1–6, 05 2019.

[HKL90] L. A. Harrington, A. S. Kechris, and A. Louveau. A Glimm-Effros dichotomy for Borel equivalence relations. *Journal of the American Mathematical Society*, 3(4):903–928, 1990.

[HL96] Jeffry L Hirst and Steffen Lempp. Infinite versions of some problems from finite complexity theory. *Notre Dame Journal of Formal Logic*, 37(4):545–553, 1996.

[HS07] Valentina S. Harizanov and Frank Stephan. On the learnability of vector spaces. *Journal of Computer and System Sciences*, 73(1):109–122, 2007.

[JORS99] Sanjay Jain, Daniel Osherson, James S. Royer, and Arun Sharma. *Systems that learn: An introduction to learning theory.* MIT Press, 1999.

[Kec12] A. Kechris. *Classical Descriptive Set Theory.* Graduate Texts in Mathematics. Springer New York, 2012.

[KMB07] Julia F. Knight, Sara Miller, and M. Vanden Boom. Turing computable embeddings. *The Journal of Symbolic Logic*, 72(3):901–918, 2007.

[KMP20] Takayuki Kihara, Alberto Marcone, and Arno Pauly. Searching for an analogue of ATR_0 in the Weihrauch lattice. *The Journal of Symbolic Logic*, 85(3):1006–1043, 2020.

[LG19] Jun Le Goh. Measuring the relative complexity of mathematical constructions and theorems. *Bulletin of Symbolic Logic*, 25(4):447–448, 2019.

[Lou17] A. Louveau. *Effective Descriptive Set Theory.* Unpublished notes, 2017.

[LS06] Wei Luo and Oliver Schulte. Mind change efficient learning. *Information and Computation*, 204(6):989–1011, 2006.

[Mek81] Alan H Mekler. Stability of nilpotent groups of class 2 and prime exponent. *The Journal of Symbolic Logic*, 46(4):781–788, 1981.

[Mil21] Russell Miller. Computable reducibility for Cantor space. In *Structure and Randomness in Computability and Set Theory*, pages 155–196. World Scientific, Singapore, 2021.

[MO98] Eric Martin and Daniel Osherson. *Elements of scientific inquiry.* MIT Press, 1998.

[Mon] Antonio Montalbán. Computable structure theory: Beyond the arithmetic.

[Mon21] Antonio Montalbán. *Computable structure theory: Within the arithmetic*. Cambridge University Press, 2021.

[Mos82] Yiannis Nicholas Moschovakis. Descriptive set theory. *Studia Logica*, 41(4):429–430, 1982.

[MR72] Alfred B Manaster and Joseph G Rosenstein. Effective matchmaking (recursion theoretic aspects of a theorem of Philip Hall). *Proceedings of the London Mathematical Society*, 3(4):615–654, 1972.

[MS04] Wolfgang Merkle and Frank Stephan. Trees and learning. *Journal of Computer and System Sciences*, 68(1):134–156, 2004.

[MV21] Alberto Marcone and Manlio Valenti. The open and clopen Ramsey theorems in the Weihrauch lattice. *Journal of Symbolic Logic*, 86(1):316–351, 2021.

[NP19] Hugo Nobrega and Arno Pauly. Game characterizations and lower cones in the Weihrauch degrees. *Logical Methods in Computer Science*, 15(3):Paper No. 11, 29, 2019.

[Pau16] Arno Pauly. On the topological aspects of the theory of represented spaces. *Computability*, 5(2):159–180, 2016.

[Rog87] Hartley Rogers. *Theory of Recursive Functions and Effective Computability.* MIT Press, Cambridge, MA, USA, 1987.

[Sac17] Gerald E. Sacks. *Higher Recursion Theory.* Perspectives in Logic. Cambridge University Press, 2017.

[Sch21] Matthias Schröder. *Admissibly Represented Spaces and Qcb-Spaces*, pages 305–346. Springer International Publishing, Cham, 2021.

[Sim09] Stephen G. Simpson. *Subsystems of Second Order Arithmetic*. Perspectives in Logic. Cambridge University Press, 2 edition, 2009.

[SV01] Frank Stephan and Yuri Ventsov. Learning algebraic structures from text. *Theoretical Computer Science*, 268(2):221–273, 2001.

[SV22] Giovanni Soldà and Manlio Valenti. Algebraic properties of the first-order part of a problem. *Available at https://arxiv.org/abs/2203.16298*, 2022.

[VM21] Manlio Valenti and Alberto Marcone. *A journey through computability, topology and analysis*. PhD thesis, Universitá degli Studi di Udine, 2021.

[Wei13] Klaus Weihrauch. *Computable Analysis: An Introduction.* Springer Publishing Company, Incorporated, 1st edition, 2013.

[Zie07] Martin Ziegler. Revising Type-2 computation and degrees of discontinuity. *Electronic Notes in Theoretical Computer Science*, 167:255–274, 2007.