CODING AND DEFINABILITY IN COMPUTABLE STRUCTURES

ANTONIO MONTALBÁN

In computable structure theory, we study the computational aspects of mathematical structures. We are interested in questions like the following: How difficult is it represent a certain structure? Which structures can be represented computably? How difficult is it to recognize a given structure? How can information be coded in the isomorphism type of a structure? How difficult is to compute certain relations on a structure, or perform certain constructions on it? We are particularly interested in answers that connect computational properties with algebraic or combinatorial properties of the structure.

Let $\mathbb{K}$ be a class of countable structures, like, for example, the class of all countable linear orderings. Let n be a natural number. The reader may start assuming $n = 1$, as this case is already interesting enough. In this course we will analyse the following two questions: Can we characterize all the relations on the structures of $\mathbb{K}$ that can be defined within n Turing jumps? How much information can be encoded into the $(n-1)$th Turing jump of the structures in $\mathbb{K}$? We will see that these two questions are closely connected. Furthermore, we will see that these questions are connected with other a structural property of the class $\mathbb{K}$, namely the number of n-back-and-forth equivalence classes in $\mathbb{K}$.

The idea of the course is to introduce some basic concepts about computable structures and to develop all the background necessary to present the main result from [Mon10]. We will give lots of examples along the way. A large number of these examples will be about the class of linear orderings, as this is a class that has been well studied by computability theorist and that presents an interesting behavior.

We will start the paper introducing the notions of Turing degree and degree spectrum of a structure. Then, in the second section, we will look at the information that is encoded on a structure and possible ways to decode it. Section 3 is about the relations that can be defined in a structure within a certain number of jumps. In Section 4 we will present a standard technique to build copies of a structures that we will use to prove some fundamental theorems from the previous sections. Then, in Section 5, we introduce the notion of the jump of a structure. Finally, in the last section, we will show the main theorem from [Mon10], that for a class of structures $\mathbb{K}$ and for a number n, either we can nicely characterize all the relations in the structures of $\mathbb{K}$ that are defined within n jumps, or we can (weakly) code any set in the $(n-1)$st jump of some structure from $\mathbb{K}$, but not both–either one or the other. This proof requires introducing the useful notion of n-back-and-forth relations.

---

[0] Saved: on Oct 28

Compiled: November 22, 2010

These lecture notes were produced after a one-month-, 10-hour-long course Montalbán gave at the University of Notre Dame in September 2010. An original version of these notes, which are now the back-bone of this paper, were produced during the course by Notre Dame students: Jesse Johnson, Steve VanDenDriessche, Quinn Culver and Victor Ocasio Gonzalez.

1. Degrees of Structures

Throughout this course we will use L to denote a countable language, that is, a set of symbols for constants, functions and relations. We will study countable L-structures from a computable view point.

Definition 1.1. L is a *computable language* if there is a computable procedure that, given a symbol, tells what kind of symbol it is and also gives the arity of the symbol, if the symbol is a relation or a function. For this to make sense, every symbol in L has to have an associated Gödel number.

All the languages we will consider are computable.

We would like to have some notion of computational complexity for structures. Since computability theory is developed on the natural numbers we need to work with structures whose elements can be enumerated by natural numbers. Given a structure $\mathscr{A}$, a *presentation of* $\mathscr{A}$ is nothing more than isomorphic copy of $\mathscr{A}$ whose domain is either ω or an initial segment of ω (the latter case being only possible when $\mathscr{A}$ is finite). Since we will consider only countable structures, all structures have presentations, and whenever we are given a structure, we will assume we are given a presentation for it.

When L is finite, the Turing degree of a presentation can be defined to be the join the Turing degrees of its relations and functions (which are subsets of ω^k for relations of arity k and subsets of ω^{k+1} for functions of arity k). When L is infinite, the situation is slightly more delicate, and we need to take an infinite join taking in consideration the Gödel numbering of each symbol. Instead of doing this, we will use a different, but equivalent, definition of degree of a presentation.

For each natural number i, we consider a constant element b_i. Given a presentation $\mathscr{B}$ with domain $B \subseteq \omega$, for each $i \in B$, we interpret b_i as i. We enumerate all the atomic formulas $\{\phi_0, \phi_1, \dots\}$ of the language $L \cup \{b_0, b_1, \dots\}$ in some effective way.

Definition 1.2. The *degree of a presentation* $\mathscr{B}$ is *deg*$(D(\mathscr{B}))$, where $D(\mathscr{B})$ is the atomic diagram of $\mathscr{B}$, that is

$$D(\mathscr{B}) = \{i \in \omega : \mathscr{B} \models \phi_i\} \subseteq \omega,$$

and *deg*(X) is the Turing degree of X. We say that $Y \subseteq \omega$ *computes a copy of* $\mathscr{A}$, if $D(\mathscr{B}) \leq_T Y$ for some presentation $\mathscr{B}$ of $\mathscr{A}$.

Note that this definition is no different from our first notion of degree since atomic formulas determine, nothing more, and nothing less, than the relations among elements and the values of the functions.

This notions of degree of a presentation, is clearly dependent on the particular presentation chosen for a certain structure, and two isomorphic presentations of the same structures might have different degree. We would like to have a way of measuring the complexity of an isomorphism type of a structure that is independent of the particular presentation chosen.

Definition 1.3 (Jockusch [Ric81])**.** Given $X \subseteq \omega$, we say that an L-structure $\mathscr{A}$ has *Turing degree* X if

$$(\forall Y \subseteq \omega)\ Y \text{ computes a copy of } \mathscr{A} \Leftrightarrow Y \geq_T X.$$

It is clear that if such an X exists, it determines the complexity of the structure $\mathscr{A}$. But the is no reason to assume that, for a structure $\mathscr{A}$, such a set X exists. Let us see a few examples.

Example 1.4. *$\mathscr{A}$ has a computable copy iff $\mathscr{A}$ has Turing degree* $\mathbf{0}$.

Example 1.5. *Fix $X \subseteq \omega$. Let G be a graph that consists of disjoint cycles where if $n \in X$, then G has a cycle of length $2n+3$, and if $n \notin X$, then G has a cycle of length $2n+4$, and there are no other cycles in G.*

Claim 1. G has Turing degree X.

Proof. ($\Leftarrow$): Suppose $Y \geq_T X$. We need to show that Y computes a copy of G. We build G step by step. Recall that G will have domain ω, that is, each vertex will be represented by a natural number. At the first step, if $0 \in X$, we build a cycle in G using the first three natural numbers, and if $0 \notin X$, we use the first four natural numbers. At the $(n+1)$-st step, using Y as an oracle, we can determine whether or not $n \in X$. If $n \in X$, then we use the next $2n+3$ numbers to make a cycle. Otherwise, we use the next $2n+4$ numbers.

($\Rightarrow$): Suppose Y computes a copy of G. We need to show that $Y \geq_T X$. So given n, using oracle Y, we want to determine if $n \in X$. Again using Y as an oracle, we can look through our copy of G element by element. As we search, we can see which elements are part of a cycle, and we can easily determine the length of these cycles once we find them. So we search through our graph until we find a cycle of length $2n+3$ or $2n+4$, exactly one of which will appear by our construction of G. If we find a cycle of length $2n+3$, then $n \in X$. If we find a cycle of length $2n+4$, then $n \notin X$. Therefore, $Y \geq_T X$. $\square$

We have shown that for every set X there is a graph with Turing degree X.

Example 1.6. *The situation with linear orderings is quite different.*

Theorem 1.7 (Richter [Ric81]). *Every linear ordering has two presentations, $\mathscr{A}$ and $\mathscr{B}$, such that*

$$deg(\mathscr{A}) \wedge deg(\mathscr{B}) = \mathbf{0}.$$

Corollary 1.8. *Only if $X \equiv_T \mathbf{0}$, we can have a linear order $\mathcal{L}$ with Turing degree X.*

Proof. Suppose that $\mathcal{L}$ has a Turing degree X. Consider the presentations $\mathscr{A}$ and $\mathscr{B}$ of $\mathcal{L}$ that satisfy the previous theorem. Then $deg(\mathscr{A}) \geq_T X$ and $deg(\mathscr{B}) \geq_T X$. So $deg(\mathscr{A}) \wedge deg(\mathscr{B}) \geq_T X$. Therefore, by the choice of $\mathscr{A}$ and $\mathscr{B}$, $X \equiv_T 0$. $\square$

Since there are continuum many linear orderings, and only countably many of them have computable copies, this corollary shows that most linear orderings do not have Turing degree. This indicates that our definition for degrees of structures may not be as good as we would like. The following definition works for all structures.

Definition 1.9. Given a structure $\mathscr{A}$, we define the *degree spectrum of $\mathscr{A}$* to be

$$\mathrm{Spec}(\mathscr{A}) = \{deg(\mathscr{B}) : \mathscr{B} \text{ is a copy of } \mathscr{A}\} \subseteq \mathbf{D},$$

where $\mathbf{D}$ is the set of all Turing degrees.

Notice that a structure $\mathscr{A}$ has Turing degree X if and only if $\mathrm{Spec}(\mathscr{A}) = \{deg(Y) : Y \geq_T X\}$, the cone above $deg(X)$. But degree spectra do not always need to be shaped as a cone above a degree.

To introduce the next theorem, we must say what a trivial structure is. A structure is *trivial* if there are finitely many elements such that any permutation of the domain of the structure which leaves these elements fixed is is an automorphism. For example, a complete graph, where all elements are related, is trivial as any permutation of the vertices is an automorphism.

Theorem 1.10 (Knight [Kni98])**.** *For every non-trivial structure $\mathscr{A}$,*

$$\mathrm{Spec}(\mathscr{A}) = \{\mathbf{x} \in \mathbf{D} : \mathbf{x} \textit{ computes a copy of } \mathscr{A}\}.$$

Thus, $\mathrm{Spec}(\mathscr{A})$ *is upwards closed in the Turing degrees.*

2. Information coded on a structure

Knight's theorem above implies that, given a non-trivial structure $\mathscr{A}$, we have that, for every set $X \subseteq \omega$, there is a copy of $\mathscr{A}$ that computes X. In short, every non-trivial stucture has a copy that, in a sense, encodes any information we want. However, if we want to look at the information that is encoded in the isomorphism type of a structure, we would like this information to encoded in every copy of $\mathscr{A}$.

Definition 2.1. A set $D \subseteq \omega$ is *coded by a structure* $\mathscr{A}$ if D is computably enumerable in the degree of every presentation of $\mathscr{A}$.

A set $D \subseteq \omega$ is *strongly coded by a structure* $\mathscr{A}$ if D is computable in every presentation of $\mathscr{A}$.

Note that D is strongly coded in $\mathscr{A}$ if and only if D and $\overline{D}$ are coded in $\mathscr{A}$ (where $\overline{D}$ is the complement of D). Also note that any c.e. set is coded by any structure.

Example 2.2. *Linear orders cannot strongly code anything except* $\mathbf{0}$*. This follows from Richter's Theorem 1.7 above.*

Example 2.3. *Consider our graph G from Example 1.5 above. Notice that G strongly codes X. Let G_Y be a graph consisting of cycles where it has a cycle of length $n+3$ if and only if $n \in Y$. Then Y is coded by G_Y. Note that our original example was $G_{X\oplus\overline{X}}$.*

Sometimes, information is not coded in such a direct way.

Example 2.4. *Let $X \subseteq \omega$. For each n, construct a linear order*

$$\mathcal{L}_n \simeq \begin{cases} \mathbb{Z} & \text{if } n \notin X \\ \mathbb{Z} + (n+1) + \mathbb{Z} & \text{if } n \in X, \end{cases}$$

where $\mathbb{Z} + (n+1) + \mathbb{Z}$ means we have an order consisting of a $\mathbb{Z}$-chain, followed by $n+1$ elements, followed by another $\mathbb{Z}$-chain. Let

$$\mathcal{L}_X = \mathcal{L}_0 + \mathcal{L}_1 + \mathcal{L}_2 + \cdots .$$

It is clear that, in some way, the set X is encoded in $\mathcal{L}_X$. How difficult is it to decode it this information from $\mathcal{L}_X$? Unfortunately, it is not that easy.

Claim 2. If Y computes a presentation of $\mathcal{L}_X$, then X is c.e. in Y''.

Proof. We know that $n \in X$ if and only if we can find $n+1$ elements in the linear ordering with a few properties: these elements must form a chain with no other elements in between them, and this chain must be in between two $\mathbb{Z}$-chains. We can express these conditions in the following formula about $\mathcal{L}_X$:

$$n \in X \Leftrightarrow \exists x_0, \dots, x_n \in \mathcal{L}_X \left(\begin{array}{l} x_0 < x_1 < \cdots < x_n \ \& \\ \forall y (x_0 \leq y \leq x_n \to y = x_0 \vee \cdots \vee y = x_n) \ \& \\ \forall y < x_0 \exists z (y < z < x_0) \ \& \\ \forall y > x_n \exists z (x_n < z < y) \end{array} \right).$$

Since $\mathcal{L}_X$ is Y-computable, notice that the information inside the large parentheses is a Π_2^Y statement. So Y'' computes it. The outside existential quantifier makes membership in X a Σ_3^Y statement. This is equivalent to saying that X is c.e. in Y''. □

This example motivates the following definition.

Definition 2.5. D is *coded by the nth jump of a structure* if D is c.e. in the nth Turing jump of the degree of any presentation of $\mathscr{A}$.

Example 2.6. *So, in the example above we get that X is coded in the 2nd jump of L_X. We will now show that the statement of the claim above is sharp.*

Claim 3. Y can compute a presentation of $\mathcal{L}_X \Leftrightarrow X$ is Σ_3^Y.

Proof. ($\Rightarrow$) This direction was done in the previous claim.

($\Leftarrow$) Suppose X is Σ_3^Y. Then there is a $\Sigma_3^0(Y)$ formula $\exists x \phi(n,x)$, where $\phi(n,x)$ is $\Pi_2^0(Y)$ and $n \in X \leftrightarrow \exists x \phi(n,x)$. Let $\phi(n,x) = \forall y \theta(n,x,y)$ where θ is $\Sigma_1^0(Y)$. We want to make two standard assumptions on our formulas ϕ and θ.

- If $\exists x \phi(n,x)$, then $\exists! x \phi(n,x)$.
- If $\theta(n,x,y)$, then $\forall y' < y \theta(n,x,y)$.

For the first assumption, we need to change $\phi(n,x)$ for a formula that says that $\langle x', y' \rangle$ is a pair such that x' is the first witness for $\phi(n,x)$ and y' is the least element below which we can find witnesses showing that $\phi(n,x_1)$ does not hold for any $x_1 < x'$. (See the Figure fig W of $W = \theta$.) All we need to do is replace $\phi(n,x)$ by the formula

$$(x = \langle x', y \rangle)) \ \wedge \ \phi(n,x') \ \wedge \ (\forall x_1 < x')(\exists y_1 < y') \neg \theta(n,x_1,y_1) \wedge \\ \neg((\forall x_1 < x')(\exists y_1 < y' - 1) \neg \theta(n,x_1,y_1)).$$

Note that this formula is $\Pi_2^0(Y)$. Once we are assuming ϕ satisfies the first assumption, for the second assumption all we need to do is replace $\theta(n,x,y)$ by $\forall y' \leq y \theta(n,x,y')$.

We may proceed with the proof. Fix some n. We want to build $\mathcal{L}_n$ uniformly in n. Let $W = \{(x,y) \mid \theta(n,x,y)\}$. Let $\mathscr{A}_n = (\omega, \leq_{\mathscr{A}_n})$ be a computable presentation of $\omega + (n+1) + \omega^*$ where ω^* is the ordering of the negative integers.

Using $\mathscr{A}_n$, we will define an ordering $\leq_W$ on W essentially by restricting the product ordering $(\omega, \leq) \times (\omega, \leq_{\mathscr{A}_n})$ on ω^2 to W. We define $\leq_W$ as follows:

$$(x_1, y_1) \leq_W (x_2, y_2) \Leftrightarrow ((x_1 < x_2) \vee (x_1 = x_2 \wedge y_1 \leq_{\mathscr{A}_n} y_2)).$$

This means that $(x_1, y_1) \leq_W (x_2, y_2)$ if and only if either the x_1 column is to the left of the x_2 column or if the points are in the same column then the y_1 entry appears below the y_2 entry in the $\mathscr{A}_n$ ordering.

If $n \notin X$, then every column of W if finite. So our final ordering will be an infinite sequence of finite linear orders, and hence will look like ω. If $n \in X$, then we will have exactly one column of 1's, as in Figure 1. In this case, inside this column, the ordering is isomorphic to $\mathscr{A}_n$. Therefore, our final ordering would look like (finite order) $+ \mathscr{A}_n + \omega$.

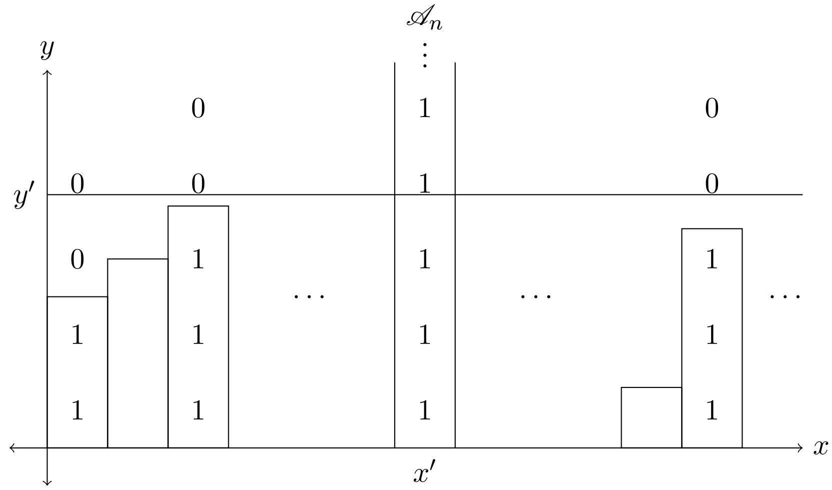

FIGURE 1. Figure of $W \subseteq \omega^2$ when $n \in X$.

The domain of this ordering is W which is c.e. in Y, but not necessarily computable in Y. If we consider a Y-computable one-to-one enumeration of W, say $\{w_0, w_1, ...\}$, we can pull back the ordering $\leq_W$ to ω: Let $\leq_V$ be an ordering on ω such that $i \leq_V j$ if $w_i \leq_W w_j$. So we have that $\mathscr{V} = (\omega, \leq_V)$ is a Y-computable linear ordering that is isomorphic to either ((finite order) $+\mathscr{A}_n + \omega$) or ω depending on whether $n \in X$ or not.

Finally, let $\mathcal{L}_n = \omega^* + \mathscr{V}$. Then, if $n \in X$, we have that $\mathcal{L}_n = \omega^* + ($ finite order $+\omega + (n + 1) + \omega^*) + \omega \simeq \mathbb{Z} + (n+1) + \mathbb{Z}$, and if $n \notin X$ we have that $\mathcal{L}_n = \omega^* + \omega \simeq \mathbb{Z}$, as desired. Since this Y-computable construction of $\mathcal{L}_n$ is uniform in n, Y can compute a presentation of $\mathcal{L}_X = \mathcal{L}_0 + \mathcal{L}_1 + \cdots$. □

Definition 2.7. Given $X \geq_T 0^{(n)}$, we say that a structure $\mathscr{A}$ has *nth-jump Turing degree* X if and only if $\forall Y$ (Y can compute a copy of $\mathscr{A} \leftrightarrow Y^{(n)} \geq_T X$).

Example 2.8. *Observe that for every $X \subseteq \omega$, in the example above we have that $\mathcal{L}_{X \oplus \bar{X}}$ has 2nd-jump Turing degree X since $X \leq_T Y'' \leftrightarrow X \in \Sigma_3^Y \wedge \bar{X} \in \Sigma_3^Y$.*

Theorem 2.9 (Knight [Kni86])**.** *Every linear order has two copies $\mathscr{A}$, $\mathscr{B}$ such that*

$$(deg(\mathscr{A}))' \wedge (deg(\mathscr{B}))' = \mathbf{0}'.$$

Corollary 2.10. *Only for $X \equiv_T 0'$, there exists linear orderings which have 1st-jump Turing degree X.*

Proof. Suppose that $\mathcal{L}$ is a linear ordering with 1st-jump Turing degree X. Therefore $(\forall Y)$, Y computes a copy of $\mathcal{L} \leftrightarrow Y' \geq_T X$. Let Y_1 and Y_2 be the degrees of the two copies of $\mathcal{L}$ that satisfy the previous theorem. Thus, $Y_1' \wedge Y_2' = 0'$. Since $Y_1' \geq X$ and $Y_2' \geq X$, we have $0' \geq X$. So $X = 0'$. □

Definition 2.11 (Jockusch and Soare [JS94])**.** Given a class of structures $\mathbb{K}$, and an ordinal α, we say that $\mathbb{K}$ has *Turing ordinal* α if for every $X \geq_T 0^\alpha$, there is a structure in $\mathbb{K}$ with α-th jump Turing degree X, and for every $\beta < \alpha$, only for $X \equiv_T 0^\beta$ can a structure in $\mathbb{K}$ have βth-jump Turing degree X.

Example 2.12.
(1) *Graphs have Turing ordinal 0, as it follows from Example 1.5.*
(2) *Linear Orderings have Turing ordinal 2, as it follows from Theorem thm: kni86 and Example 2.6.*

(3) *Boolean Algebras have Turing ordinal* ω *(*[JS94]*).*
(4) *Equivalence structures have Turing ordinal* 1 *(see* [Mon10] *for a proof).*

2.1. **Coding and enumeration reducibility.** In this section we will give a characterization of the sets coded in a structure. We will delay the proofs to Section 4 below. Let us start by recalling the notion of enumeration reducibility.

Theorem 2.13 (Selman [Sel71])**.** *Let $A, B \subseteq \omega$. The following are equivalent:*
(1) *There exists a Turing functional Φ such that for every onto function $f\colon \omega \to B$, Φ^f is a and onto function from ω to A.*
(2) *For every onto function $f\colon \omega \to B$, there exists $g \leq_T f$ which is an onto function from ω to A.*
(3) *For every $X \subseteq \omega$, if B is c.e. in X, then A is c.e. in X.*
(4) *There exists a c.e. set $\Gamma \subseteq \mathcal{P}_f(\omega) \times \omega$, (where $\mathcal{P}_f(\omega)$ is the set of finite subsets of ω) such that*

$$A = \{n \in \omega : (\exists D \in P_f(\omega))\ \langle D, n\rangle \in \Gamma \wedge D \subseteq B\}.$$

Definition 2.14. If A and B satisfy any of the conditions of the theorem above, we say that A is *enumeration* reducible to B, and we write $A \leq_e B$.

There is one other bit of notation that we need before our characterization of the sets coded in a structure. Given $\bar{a} \in A^{<\omega}$, we let $\Sigma_1\text{-tp}_{\mathscr{A}}(\bar{a}) \subseteq \omega$, the Σ_1-*type of* $\bar{a}$, be the set of indices of finitary Σ_1 formulas $\phi(\bar{x})$ such that $\mathscr{A} \models \phi(\bar{a})$. Notice that the set $\Sigma_1\text{-tp}_{\mathscr{A}}(\bar{a})$ is defined independently of the given presentation of $\mathscr{A}$. It is not hard to see that for every $\bar{a}$, $\Sigma_1\text{-tp}_{\mathscr{A}}(\bar{a})$ is coded in $\mathscr{A}$. The next theorem says that, essentially, these are the only sets that are coded in a structure $\mathscr{A}$.

Theorem 2.15 (Knight [AK00])**.** *A set X is coded in a structure A if and only if for some $\bar{a} \in A^{<\omega}$,*

$$X \leq_e \Sigma_1\text{-tp}_{\mathscr{A}}(\bar{a}).$$

The proof of this theorem is somewhat similar to the one of Theorem 4.2.

2.2. **Weakly coding.** There is another way of coding information into a structure without taking jumps. We first need to recall the notion of left c.e. set. For $\sigma, \tau \in 2^{<\omega}$, we let $\sigma \leq_{\mathbb{Q}} \tau$ if for $\gamma = \sigma \cap \tau$, we have that σ is compatible with $\gamma^\frown 0$ and τ is compatible with $\gamma^\frown 1$. It not hard to see that $(2^{<\omega}, \leq_{\mathbb{Q}})$ is isomorphic to the ordering on the rationals. We can then extend this ordering to $2^{\leq\omega}$ in the obvious way, getting the lexicographic ordering when restricted to 2^ω. We say that a $D \in 2^\omega$ is *left c.e.* if $\{\sigma \in 2^{<\omega} : \sigma <_{\mathbb{Q}} D\}$ is c.e.. These reals are also sometimes called left-approximable or c.e. reals.

Definition 2.16. D is *weakly coded in the nth jump* of $\mathscr{A}$ if for every $\mathscr{B} \cong \mathscr{A}$, D is left c.e. in $D(\mathscr{B})^{(n)}$.

We will that in some cases weakly coding is all we can do.

Example 2.17. *We will now define a class of structure $\mathbb{K}$, such that every structure $\mathscr{A}$ of $\mathbb{K}$ is determiened by a $\leq_{\mathbb{Q}}$-downwards closed subset $R_{\mathscr{A}}$ of $2^{<\omega}$, and such that $R_{\mathscr{A}}$ is coded in $\mathscr{A}$ (i.e. it is c.e. in every copy of $\mathscr{A}$).*

The language for these structures consists of two unary relations A and B, a function symbol f, and a constant symbol c_q for each $q \in 2^{<\omega}$. The set $R_{\mathscr{A}}$ that we mention above will be decoded from the set of c_q's which are in the range of f. Let $\mathbb{K}$ be the class of structures on this language which satisfy the following properties:

- *A and B partition the universe in two sets.*
- *Every element of B is named by some constant c_q, and no element of A is.*
- *Different constants are assigned to different elements.*
- *The range of f is included in B.*
- *f is the identity on the elements of B.*
- *f is one-to-one on the elements of A.*
- *If $q <_{\mathbb{Q}} r \in 2^{<\omega}$ and $(\exists x \in A) f(x) = c_r$, then $(\exists y \in A) f(y) = c_q$.*

It is not hard to see that each structure $\mathscr{A}$ of $\mathbb{K}$ is completely determined by the set $R_{\mathscr{A}} = \{q \in 2^{<\omega} : \mathscr{A} \models (\exists x \in A) f(x) = c_q\}$ which is an initial segment of $(2^{<\omega}, \leq_{\mathbb{Q}})$, an could be any given initial segment of $(2^{<\omega}, \leq_{\mathbb{Q}})$. Furthermore, $R_{\mathscr{A}}$ is coded by $\mathscr{A}$. Therefore, for every $D \in 2^\omega$, there is a structure $\mathscr{A} \in \mathbb{K}$ with $R_{\mathscr{A}} = \{\sigma \in 2^{<\omega} : \sigma <_{\mathbb{Q}} D\}$, and hence $\mathscr{A}$ weakly codes D.

3. Relations on a structure

Definition 3.1. A relation R on a structure $\mathscr{A} (R \subseteq A^k)$ is *relatively intrinsically computably enumerable (r.i.c.e.)* if for every copy $(\mathscr{B}, Q)$ of $(\mathscr{A}, R)$, Q is c.e. in $D(\mathscr{B})$.

Example 3.2. *Let $\mathcal{L}$ be a linear order and let $\texttt{Succ}(x,y) \equiv x < y \wedge \forall z \neg(x < z < y)$. $\neg\,\texttt{Succ}(x,y)$ is r.i.c.e.. To see this, given two elements x, y, for $\neg\,\texttt{Succ}(x,y)$ to hold, either $y < x$, which we can tell computably, or there is a z such that $x < z < y$, which we can search computably.*

Example 3.3. *On a graph, the relation $\texttt{Conn}(x,y) \equiv$ (x and y are joined by a path) is r.i.c.e.. To see this, just enumerate all the paths in the graph looking for a path between x and y. This is a c.e. process.*

Note that there is no 1st order formula in the language of graphs that defines connectedness.

The definition of relatively intrinsically computably enumerable relation can be extended in an obvious way to the whole arithmetic hierarchy.

Definition 3.4. A relation R on a structure $\mathscr{A}$ is *relatively intrinsically* Σ^0_n if for every copy $(\mathscr{B}, Q)$ of $(\mathscr{A}, R)$, Q is many-one reducible to $D(\mathscr{B})^{(n)}$.

Thus, these relations are exactly the ones that can be define within n Turing jumps of the structure, independently of the presentation of the structure. Our goal now is to characterize the relatively intrinsically Σ^0_n relations on a structure.

Definition 3.5. Given a set L of relation, function and constant symbols, we introduce the *infinitary language* over it. $L_{\omega_1,\omega}$ is the least set of formulas such that

- All first order L-formulas are in $L_{\omega_1,\omega}$.
- If $\{\phi_0, \phi_1, \dots\} \subseteq L_{\omega_1,\omega}$ and altogether they use only finitely many free variables then $\bigwedge_{i\in\omega} \phi_i$ and $\bigvee_{i\in\omega} \phi_i \in L_{\omega_1,\omega}$.
- If $\phi \in L_{\omega_1,\omega}$, then $\forall x \phi_i \in L_{\omega_1,\omega}$ and $\exists x \phi_i \in L_{\omega_1,\omega}$.

The interpretation of an infinitary formula on an L-structure is defined in the obvious way.

The hierarchy of $L_{\omega_1,\omega}$ formulas is defined as follows. The $\Sigma_0^{\mathtt{in}}$ and $\Pi_0^{\mathtt{in}}$ formulas are formulas without quantifiers and without infinite disjunctions or conjunctions. The $\Sigma_n^{\mathtt{in}}$ formulas are the ones of the form $\bigvee_{i\in\omega} \exists \bar{x}\phi_i(\bar{x})$, where ϕ_i is $\Pi_m^{\mathtt{in}}$ for some $m < n$, and the $\Pi_n^{\mathtt{in}}$ formulas are the ones of the form $\bigwedge_{i\in\omega} \forall \bar{x}\phi_i(\bar{x})$, where ϕ_i is $\Sigma_m^{\mathtt{in}}$ for some $m < n$. This definition can be extended throughout the ordinals, but in this paper we only consider the finite levels. (See [AK00, Chapter 6].)

A formula $\phi \in L_{\omega_1,\omega}$ is *computably infinitary* if all its conjunctions and disjunctions are of c.e. sets of formulas. We then denote $\Sigma_n^{\mathtt{c}}$ for the computably infinitary $\Sigma_n^{\mathtt{in}}$ formulas, and $\Pi_n^{\mathtt{c}}$ for the computably infinitary $\Pi_n^{\mathtt{in}}$ formulas. (See [AK00, Chapter 7].)

For this definition to make sense, that is, to be able to talk about c.e. set of formulas, we need to assign a Gödel number to each computably infinitary formula. This is done from the bottom up. That is, we define the codes for the $\Sigma_n^{\mathtt{c}}$ and $\Pi_n^{\mathtt{c}}$ by recursion on n: Once all the $\Sigma_{n-1}^{\mathtt{c}}$ and $\Pi_{n-1}^{\mathtt{c}}$ formulas have Gödel numbers, we can give codes to the $\Sigma_n^{\mathtt{c}}$ and $\Pi_n^{\mathtt{c}}$ formulas using the index for the c.e. sets of formulas being considered.

Example 3.6. *On a graph* (V, E), $\mathtt{Conn}(x, y) \equiv \bigvee_{n\in\omega} \exists x_1, \dots, x_n(xEx_1 \wedge x_1Ex_2 \wedge \cdots \wedge x_nEy)$. *Note this is a* $\Sigma_1^{\mathtt{c}}$ *formula.*

Example 3.7. *"A group is torsion" (all elements have finite order) can be defined by* $\forall x \bigvee_{n\in\omega} x^n = 1$. *This one is a* $\Pi_2^{\mathtt{c}}$ *sentence.*

Observation 3.8. If $\phi(\bar{x})$ is a $\Sigma_n^{\mathtt{c}}$ formula, then $\{\bar{a} \in A^{|\bar{x}|} : \mathscr{A} \models \phi(\bar{a})\}$, as a subset of ω, is Σ_n^0 in $D(\mathscr{A})$. Furthermore, this is uniform in ϕ. That is, if $\varphi_i(\bar{x}_i)$ denotes the ith $\Sigma_n^{\mathtt{c}}$ formula in a standard enumeration, then $\{\langle i, \bar{a}\rangle : i \in \omega, i \in A^{|\bar{x}_i|}, \mathscr{A} \models \phi_i(\bar{a})\}$ is also Σ_n^0 in $D(\mathscr{A})$.

The following theorem gives the first characterization of set relatively intrinsically Σ_n^0 relations. Notice how this theorem provides an equivalence between a computational notion that is defined in terms of the presentations of a structure and a syntactical notion that is completely independent of the presentations involved.

Theorem 3.9. *[Ash, Knight, Manasse, Slaman; Chishholm] Given a relation* R *on* $\mathscr{A}$, *the following are equivalent:*

(1) R *is relatively intrinsically* Σ_n^0.
(2) *There are a* $\Sigma_n^{\mathtt{c}}$ *formula* $\phi(\bar{x}, \bar{y})$ *and parameters* $\bar{b} \in A$ *such that*

$$(\forall \bar{a} \in A^k) \quad \bar{a} \in R \Leftrightarrow \mathscr{A} \models \phi(\bar{a}, \bar{b}).$$

We will prove this theorem at the end of Section 5. Now, we will see how, in some cases, one can find a much better characterization of the relatively intrinsically Σ_n^0 relations.

Example 3.10. *The class of linear orderings gives us again a nice example.*

Lemma 3.11. *In the class of linear orderings, every* $\Sigma_1^{\mathtt{c}}$ *formula is equivalent to a finitary* Π_1 *formula in the language* $(\leq, \mathtt{Succ})$.

Before proving this lemma, we need to prove the following auxiliary result.

Lemma 3.12. *For $(a_1, a_2, \dots, a_n)$ and $(b_1, b_2, \dots, b_n) \in \omega^n$, we declare $(a_1, \dots, a_n) \leq (b_1, \dots, b_n)$ if $(\forall i \leq n)\ a_i \leq b_i$. Then, every $A \subseteq \omega^n$ has a finite subset $B \subseteq A$ such that*

$$\forall x \in A \exists y \in B (y \leq x).$$

This lemma says is that $\leq$ is a well-quasi-ordering.

Proof. Since $\leq$ is clearly well founded, A has a subset B of minimal elements, satisfying $\forall x \in A \exists y \in B (y \leq x)$. We need to prove that B is finite. Note that all the elements of B are incomparable, so, it will be enough to show that $(\omega^n, \leq)$ has no infinite antichains. We claim that for every sequence $\{\bar{x}_i : i \in \omega\} \subseteq \omega^n$, $\exists i, j (i \leq j) \wedge (\bar{x}_i \leq \bar{x}_j)$. The proof of this claim is done by induction on n: Split each $\bar{x}_i$ as $\bar{y}_i {}^\frown m_i$ where $\bar{y}_i \in \omega^{n-1}$ and $m_i \in \omega$. There is a subsequence where the m_i's are non-decreasing. Along this subsequence, we know by induction, that there are $i < j$ such that $\bar{y}_i \leq \bar{y}_j$. Since $m_i \leq m_j$ for all $i < j$ in that subsequence, we get $\bar{x}_i \leq \bar{x}_j$. This completes our induction step. □

We now prove the Lemma 3.11.

Proof. Let $\bar{x} = (x_1, \dots, x_n)$ and $\phi(\bar{x}) = \bigvee_{i \in \omega} \exists \bar{y}_i \psi_i(\bar{x}, \bar{y}_i)$, where the $\bar{y}_i$ can be of different lengths for different i's. We need to show that this is equivalent to a finitary Π_1 formula. For each finite map f from the set of variables $\{\bar{x}, \bar{y}_i\}$ to an initial segment of ω, let $\psi_f(\bar{x}, \bar{y}_i)$ be the formula that says that these variables appear in the same order as their image through f. That is $\psi_f(\bar{x}, \bar{y}_i)$ is conjunction of the formulas $w < z$ for $w, z \in \{\bar{x}, \bar{y}_i\}$ with $f(w) < f(z)$ and the formulas $w = z$ for $w, z \in \{\bar{x}, \bar{y}_i\}$ with $f(w) = f(z)$. It is not hard to see that each each ψ_i is equivalent to a finite disjunction of formulas of the form ψ_f. So, by pulling the disjunction out, we can assume all the ψ_i are of this form.

Since there are only finitely many ways to order $\bar{x}$, it is enough to show that $\phi(\bar{x}) \wedge (x_1 < x_2 < \dots < x_n)$ is equivalent to a Π_1 $(\leq, \mathtt{Succ})$-formula. So, we can assume all the ψ_i are consistent with $(x_1 < x_2 < \dots < x_n)$. Then ψ_i looks like

$$y_1 < y_2 < \dots < y_{l_0^i} < x_1 < y_{l_0^i+1} < \dots < y_{l_1^i} < x_2 < y_{l_1^i+1} < \dots < x_2 < \dots < x_3 < \dots < x_n.$$

Thus, if we let $\mathtt{D}_l(z, w)$ denote $\exists (y_1, \dots, y_l)(z < y_1 < \dots < y_l < w)$, then

$$\exists \bar{y} \psi_i \equiv \mathtt{D}_{l_0^i}(-\infty, x_1) \wedge \mathtt{D}_{l_1^i}(x_1, x_2) \wedge \dots \wedge \mathtt{D}_{l_n^i}(x_n, \infty).$$

Note that this formula is equivalent to a Π_1 formula over $\{\leq, \mathtt{Succ}\}$:

$$\mathtt{D}_l(z, w) \equiv \bigwedge_{k < l} (\forall y_1, \dots, y_k) \neg (\mathtt{Succ}(z, y_1) \wedge \mathtt{Succ}(y_1, y_2) \wedge \dots \wedge \mathtt{Succ}(y_k, w)).$$

Let $A = \{(l_0^i, l_1^i, \dots, l_n^i) : i \in \omega\} \subseteq \omega^{n+1}$. Then by lemma 4.11, there exists a finite $B \subseteq A$ such that $\forall \bar{l} \in A \exists \bar{m} \in B (\bar{m} \leq \bar{l})$. It follows that

$$\phi(\bar{x}) \wedge (x_1 < x_2 < \dots < x_n) \equiv \bigvee_{\bar{m} \in B} \mathtt{D}_{m_0}(-\infty, x_1) \wedge \mathtt{D}_{m_1}(x_1, x_2) \wedge \dots \wedge \mathtt{D}_{m_n}(x_n, \infty). \qquad \square$$

Observation 3.13. We can obtain the equivalent Π_1 formula computably in $0'$.

Corollary 3.14. *Every computably infinitary $\Sigma_2^{\mathtt{c}}$ formula about linear orderings is equivalent to a $0'$-computable disjunction of finitary Σ_1 formulas over the language $(\leq, \mathtt{Succ})$.*

Proof. From the lemma above, we get that every $\Pi_1^{\mathtt{c}}$ formula is equivalent to a finitary Σ_1 formulas over the language $(\leq, \mathtt{Succ})$. Then, use that $\Sigma_2^{\mathtt{c}}$ formulas are $\Sigma_1^{\mathtt{c}}$ over $\Pi_1^{\mathtt{c}}$ formulas. □

This corollary gives a nice characterization of the class or relatively intrinsically Σ^0_2 relations on a linear ordering. We are interested in finding for which other classes of structures and for which other n do we have such nice characterization of the class or relatively intrinsically Σ^0_n relations.

Definition 3.15. Given a class of structures $\mathbb{K}$, a computable set of $\Pi^{\mathtt{c}}_n$ formulas, $\{\phi_1, \phi_2, \dots\}$, is a *complete set of* Π^c_n *formulas for* $\mathbb{K}$ if every Σ^c_{n+1} formula is uniformly equivalent to a $0^{(n)}$-computable disjunction of finitary Σ_1 formulas over $L \cup \{\phi_1, \phi_2, \dots\}$.

Note that for the definition above, it is enough to ask that every Π^c_n formula is uniformly equivalent to a $0^{(n)}$-computable disjunction of finitary Σ_1 formulas over $L \cup \{\phi_1, \phi_2, \dots\}$. So, a complete set of Π^c_n formulas for $\mathbb{K}$ is a set of formulas that capture the whole $\Pi^{\mathtt{c}}_n$ structural content of the structures in $\mathbb{K}$.

Example 3.16. *In the class of linear orderings,* $\{\mathtt{Succ}\}$ *is a complete set of* $\Pi^{\mathtt{c}}_1$ *formulas. This is what we just proved.*

Example 3.17. *Let* $\mathtt{S}_n(x, y) \equiv \nexists(z_1, z_2, \dots, z_n)[x < z_1 < \cdots < z_n < y \wedge \mathtt{Succ}(z_1, z_2) \wedge \cdots \wedge \mathtt{Succ}(z_{n-1}, z_n)]$. *This says that in between* x *and* y *there does not exist an* n*-string of successor elements. Then, for instance,* $\mathtt{S}_2,(x, y)$ *says that the open interval between* x *and* y *is dense, and* $\mathtt{S}_1(x, y)$ *is equivalent to* $\mathtt{Succ}(x, y)$*. Let* $\mathtt{limleft}(x) \equiv \forall z < x \exists y(z < y < x)$*, the formula that says that* x *is a limit from the left, and let* $\mathtt{limright}(x) \equiv \forall z > x \exists y(x < y < z)$*. It is proved in* [Mon10] *that the set*

$$\{\mathtt{limleft}(\cdot), \mathtt{limright}(\cdot), \mathtt{S}_1(\cdot,\cdot), \mathtt{S}_2(\cdot,\cdot), \mathtt{S}_3(\cdot,\cdot), \dots, \mathtt{S}_1(-\infty,\cdot), \mathtt{S}_2(-\infty,\cdot), \dots, \mathtt{S}_1(\cdot,\infty), \dots\}$$

is complete for $\Pi^{\mathtt{c}}_2$ *formulas.*

Example 3.18. *The set of all* $\Pi^{\mathtt{c}}_n$ *formulas is a complete set of* $\Pi^{\mathtt{c}}_n$ *formulas.*

The following lemma provides one of the motivations for being interested in complete sets of Π^c_n formulas.

Lemma 3.19. *Let* $\{\phi_1, \dots, \phi_n, \dots\}$ *be a complete set of* Π^c_n *formulas for a class of structures,* $\mathbb{K}$*. Let* $\mathscr{A} \in \mathbb{K}$ *and* R *be a relatively intrinsically* Π_n *relation on* $\mathscr{A}$*. Then for all* $X \geq_T 0^{(n)}$*, if* X *computes a copy* $\mathscr{B}$ *of* $(\mathscr{A}, \phi_0^{\mathscr{A}}, \phi_1^{\mathscr{A}}, \dots)$*, then* $X \geq_T R^{\mathscr{B}}$*.*

The following theorem provides further motivation.

Theorem 3.20 (Jump Inversion Theorem)**.** *Let* $X \geq_T 0'$ *compute a copy of* $(\mathscr{A}, \psi_0^{\mathscr{A}}, \psi_1^{\mathscr{A}}, \dots)$ *where* $\{\psi_0, \psi_1, \dots\}$ *is a complete set of* Π^c_1 *formulas. Then there exists* Y *such that*

(1) $Y' \equiv_T X$
(2) Y *computes a copy of* $\mathscr{A}$*.*

We will prove this theorem in the next section. This theorem is due independently to [**?**] and to Soskov and A. Soskova [**?**]. In [**?**], they never state this theorem, and what they call the "Jump Inversion Theorem" is a different result. But this theorem follows from the proof of [**?**, Theorem 12].

Example 3.21. *The following corollary was proved independently by Frolov as a tool to obtain other results.*

Corollary 3.22 (Frolov [Fro06])**.** *If* $0'$ *computes a linear ordering* $(L, \leq, \mathtt{Succ})$*, then* $(L, \leq)$ *has a low copy.*

Proof. Here, use the Jump Inversion Theorem, letting $X = 0'$ and using the fact that $\{\mathtt{Succ}\}$ is $\Pi^{\mathtt{c}}_1$-complete. □

4. Building copies of a structure

Given some structure $\mathscr{A}$, we would like to build a 'generic copy' of $\mathscr{A}$. Let $\mathbb{P}$ be the set of finite tuples of distinct elements from $\mathscr{A}$. We want to build sequences $p_1 \subseteq p_2 \subseteq \cdots \in \mathbb{P}$ such that every element of $\mathscr{A}$ appears in some tuple in the sequence. Here $p_i \subseteq p_{i+1}$ means that p_i is an initial segment of p_{i+1}. Let

$$G = \cup_{i \in \omega} p_i \mathscr{A}^\omega.$$

So, $G\colon \omega \to \mathscr{A}$ is one-to-one and onto. Then, we obtain a structure with domain ω by pulling back $\mathscr{A}$. Call this structure $\mathscr{B}$. So, if R is a relation on $\mathscr{A}$, then $R^{\mathscr{B}} = G^{-1}(R^{\mathscr{A}})$.

Recall that $|\mathscr{B}| = B = \{b_0, b_1, \dots\}$ is a set of constants naming the natural numbers. Using this, we are able to obtain an enumeration via Gödel numbering of atomic $(L \cup B)$-sentences, $\{\phi_0, \phi_1 \dots\}$.

Given $p \in \mathbb{P}$, we say $p \models \phi_i(b_0, ..., b_k)$ (where the constants that appear in ϕ are among the shown ones) if $k < |p|$, and $\mathscr{A} \models \phi(p(0), \dots, p(k))$ (where $p(j)$ is the jth element of p).

Definition 4.1. Given $n \in \omega$, if L is a finite language, let k_n be the number of $L \cup b_0, ..., b_n$ atomic formulas, using all symbols in L as relation symbols. If L is an infinite language, then let k_n be the number of such formulas which only use the first n many relations. We will always assume that in our enumeration of atomic formulas, the k_n formulas just mentioned appear first, and that this is true for every n. Given $p \in \mathbb{P}$, we let $D(p) \in 2^{k_{|p|}}$ be such that for $i < k_{|p|}$,

$$D(p)(i) = \begin{cases} 1 & \text{if } p \models \phi_i \\ 0 & \text{otherwise.} \end{cases}$$

Notice that

$$D(\mathscr{B}) = \cup_{i \in \omega} D(p_i) \in 2^\omega.$$

Now we have the machinery to prove the Jump Inversion Theorem 3.20 and Theorems 2.15 and 3.9.

Proof of Theorem 3.20. We want to build $G \leq_T X$ such that $(D(\mathscr{B}))' \leq_T X$.

Step 0: Let $p_0 = \emptyset$.

Step $s+1 = e$: Suppose we have already defined p_s; we now define p_{s+1}: We ask if $\exists q \in \mathbb{P}$ such that $q \supseteq p_s$ and

$$\{e\}^{D(q)}(e) \downarrow .$$

(Here, we are using $\{e\}$ for the eth partial computable function, and we use the convention that if a oracle is a finite string of length s, then the computation does not run for more than s steps.) If so, let q_{s+1} be the q found in the search. Otherwise, let $q_{s+1} = p_s$. In either case, let $p_{s+1} = q_{s+1}{}^\frown a$ where a is the first element in A not in the range of q_{s+1}. This later part of the construction is to make G onto A.

We claim that the construction is computable in X and that $(D(\mathscr{B}))' \leq_T X$. Note that the statement

$$\exists q \in \mathbb{P}((q \supseteq p_s) \wedge (\{e\}^{D(q)}(e) \downarrow))$$

holds if and only if

$$\mathscr{A} \models \bigvee_{\substack{\sigma \in 2^{<\omega} \\ \{e\}^{\sigma}(e)\downarrow}} \exists a_1, \dots, a_{n=|\sigma|-|p_s|} \in A\Big(D(p_s {}^\frown \langle a_1, \dots, a_n \rangle) = \sigma\Big),$$

where $D(p) = \sigma$ can be written as $\bigwedge_{i:\sigma(i)=1} \phi_i(p) \wedge \bigwedge_{i:\sigma(i)=0} \neg\phi_i(p)$, which is a quantifier-free finitary formula. So, the formula above is a $\Sigma_1^{\mathtt{c}}$ formula with parameters p_s. Since X computes $0'$ and $(\mathscr{A}, \phi_0^{\mathscr{A}}, \phi_1^{\mathscr{A}}, \dots)$, X can compute the $\Sigma_1^{\mathtt{c}}$ formula above, and hence X can run the construction above. Furthermore, $(D(\mathscr{B}))' \leq_T X$, because $e \in (D(\mathscr{B}))'$ if and only if at stage $s+1 = e$ there existed such q. □

We now prove the case $n = 1$ of Theorem 3.9.

Theorem 4.2. *Given a relation R on $\mathscr{A}$, the following are equivalent:*

(1) *R is r.i.c.e.*
(2) *There are a $\Sigma_1^{\mathtt{c}}$ formula $\phi(\bar{x}, \bar{y})$ and $\bar{b} \in A$ such that*

$$(\forall \bar{a} \in A^k) \quad \bar{a} \in R \Leftrightarrow \mathscr{A} \models \phi(\bar{a}, \bar{b}).$$

Proof. (2) ⇒ (1): This is the easy direction. It follows from Observation 3.8.

(1) ⇒ (2): We will build a copy $\mathscr{B}$ of $\mathscr{A}$ by building a sequence of $p_s \in \mathbb{P}$ as above, and at step $s+1 = e$ we will try to diagonalize $R^{\mathscr{B}}$ against $W_e^{D(\mathscr{B})}$. One of these attempts will have to fail, and we will use its failure to define ϕ as wanted.

Step 0: Let $p_0 = \emptyset$.

Step $s+1 = e$: We try to make $R^{\mathscr{B}} \neq W_e^{D(\mathscr{B})}$. Ask if

$$(\exists q \supseteq p_s)(\exists n < |q|) \;\; n \in W_e^{D(q)} \wedge q(n) \notin R.$$

If so, let $q_{s+1} = q$. Otherwise, let $q_{s+1} = p_s$. In any case, let $p_{s+1} = q_{s+1} {}^\frown a$, where a is the first element in A not in the range of q_{s+1}.

We now have a sequence $p_1 \subseteq p_2 \subseteq \dots$ and define G and $\mathscr{B}$ as above. Since $\mathscr{B}$ is isomorphic to $\mathscr{A}$, and R is relatively intrinsically c.e., for some e, $R^{\mathscr{B}} = W_e^{D(\mathscr{B})}$, where $R^{\mathscr{B}} = G^{-1}(R)$. Let $s = e - 1$. We now observe that for $a \in A$,

$$a \in R \iff (\exists q \supseteq p_s)(\exists n < |q|)\; n \in W_e^{D(q)} \wedge q(n) = a.$$

The direction from left to right follows from the fact that $G^{-1}(R) = W_e^{D(\mathscr{B})}$, so all we need is $n = G^{-1}(a)$ and q a sufficiently large initial segment of G. For the right to left direction, we need to observe that if $(\exists q \supseteq p_s)(\exists n < |q|)\; n \in W_e^{D(q)} \wedge q(n) = a \wedge a \notin R$, then at stage $s+1$ we would have acted and prevented $R^{\mathscr{B}} = W_e^{D(\mathscr{B})}$.

Now the right-hand side of the equation above can be written as the following $\Sigma_1^{\mathtt{c}}$ formula:

$$\bigvee_{\substack{\sigma \in 2^{<\omega} \\ n \in W_e^{\sigma}}} \exists \bar{c}\big(D(p_s {}^\frown \bar{c}) = \sigma \wedge (p_s {}^\frown c)(n) = a\big),$$

obtaining a $\Sigma_1^{\mathtt{c}}$ definition of R with parameters p_s. □

5. The jump of a structure

We start by defining the notion of the jump of a structure. Note that this definition is independent of the presentation of the given structure.

Definition 5.1. If $\{\phi_0, \phi_1, \ldots\}$ is a complete set of $\Pi^{\mathtt{c}}_n$ relations on $\mathscr{A}$, we say that $(\mathscr{A}, \phi_0^{\mathscr{A}}, \phi_1^{\mathscr{A}}, \ldots)$ is *an nth jump of* $\mathscr{A}$, written $\mathscr{A}^{(n)}$. When $\{\phi_0, \phi_1, \ldots\}$ is the sequence of all $\Pi^{\mathtt{c}}_n$ formulas, we say that $(\mathscr{A}, \phi_0^{\mathscr{A}}, \phi_1^{\mathscr{A}}, \ldots)$ is *the canonical nth jump of* $\mathscr{A}$.

Other definitions of the jump of a structure in slightly different settings were given independently by Baleva [**?**] and further studied by Soskov and A. Soskova, and also independently by Morozov and Puzarenko [**?**, **?**], and then further studied by Stukachev.

Observation 5.2. It is worth observing that an nth jump of a kth jump of a structure is an $(n+k)$th jump because a complete set of $\Pi^{\mathtt{c}}_k$ formulas over a complete set of $\Pi^{\mathtt{c}}_n$ formulas yields a complete set of $\Pi^{\mathtt{c}}_{n+k}$ formulas.

Observation 5.3. If $X \in \mathrm{Spec}(\mathscr{A})$, then $X' \in \mathrm{Spec}(\mathscr{A}')$. If $Y \in \mathrm{Spec}(\mathscr{A}')$, and $Y \geq_T 0'$, then there is $X \in \mathrm{Spec}(\mathscr{A}')$ such that $X' \equiv_T Y$ by the jump inversion theorem. Thus

$$\mathrm{Spec}(\mathscr{A}') \cap \mathbf{D}_{(\geq 0')} = \{\mathbf{x}' \colon \mathbf{x} \in \mathrm{Spec}(\mathscr{A})\},$$

where $\mathbf{D}_{(\geq 0')}$ is the set of Turing degrees that compute $0'$.

If $\mathscr{A}'$ is the canonical jump of $\mathscr{A}$, then $\mathscr{A}'$ strongly codes $0'$ (because there is a computable sequence of $\Pi^{\mathtt{c}}_1$ sentences ψ_i such that $\mathscr{A} \models \psi_i$ if and only if $i \notin 0'$, and hence $\mathscr{A}'$ codes the complement of $0'$.) Therefore $\mathrm{Spec}(\mathscr{A}') \supseteq \mathbf{D}_{(\geq 0')}$, and hence $\mathrm{Spec}(\mathscr{A}') = \{X' \colon X \in \mathrm{Spec}(\mathscr{A})\}$.

Example 5.4. *If $\mathcal{L}$ is a linear order, then Lemma 3.11 and Example 3.17 show that*

$$\mathcal{L}' = (\mathcal{L}, \mathtt{Succ})$$

and

$$\mathcal{L}'' = (\mathcal{L}, \mathtt{limleft}(\cdot), \mathtt{limright}(\cdot), \mathtt{S}_1(\cdot,\cdot), \mathtt{S}_2(\cdot,\cdot), \ldots, \mathtt{S}_1(-\infty,\cdot), \ldots, \mathtt{S}_1(\cdot,\infty), \ldots).$$

Example 5.5. *Boolean Algebras provide a very interesting example. The relations needed to get the first four jumps of a Boolean algebra were considered by Knight and Stob* [KS00]*, and a proof that they are actually complete sets of relations at the right level can be indirectly obtained from* [HM]*. For example, if $\mathscr{B}$ is a Boolean algebra, we have that that $\mathscr{B}' = (\mathscr{B}, \mathtt{atom})$ and $\mathscr{B}'' = (\mathscr{B}, \mathtt{atom}, \mathtt{inf}, \mathtt{atomless})$. This was then extended to all $n \in \mathbb{N}$ by Harris and Montalbán.*

Theorem 5.6 (Harris, Montalbán [HM])**.** *For every n there is a finite complete set of $\Pi^{\mathtt{c}}_n$ relations for the class of Boolean algebras.*

The relations used for the first four jumps of a Boolean algebra were used to prove the following lemma.

Lemma 5.7. *Let $\mathscr{B}$ be a Boolean algebra. For every $X \subseteq \omega$:*

1. *X' computes a copy of $\mathscr{B}'$ iff X computes a copy of $\mathscr{B}$ (Downey, Jockusch* [DJ94]*);*
2. *X' computes a copy of $\mathscr{B}''$ iff X computes a copy of $\mathscr{B}'$ (Thurber* [Thu95]*);*
3. *X' computes a copy of $\mathscr{B}^{(3)}$ iff X can compute a copy of $\mathscr{B}''$ (Knight, Stob* [KS00]*);*
4. *X' computes a copy of $\mathscr{B}^{(4)}$ iff X computes a copy of $\mathscr{B}^{(3)}$ (Knight, Stob* [KS00]*).*

Notice that these statements are stronger than the jump inversion theorem. The jump inversion theorem would only give us that if X' computes a copy of $\mathscr{B}'$, then there is a copy of $\mathscr{B}$ that is low over X.

Corollary 5.8. *Every* low$_4$ *Boolean algebra has a computable copy.*

Proof. If $\mathscr{B}$ is a low$_4$ Boolean algebra, then we know that $0^{(4)}$ computes a copy of $\mathscr{B}^{(4)}$. Working backwards through the statements in the lemma, we conclude that $\emptyset^{(3)}$ computes a copy of $\mathscr{B}^{(3)}$, $\emptyset^{(2)}$ computes a copy of $\mathscr{B}^{(2)}$, $\emptyset'$ computes a copy of $\mathscr{B}'$, and finally $\emptyset$ computes a copy of $\mathscr{B}$. □

The following open question was already posed in [DJ94].

Question 1. Does every low$_n$ Boolean algebra have a computable copy?

Let us now re-state the jump inversion theorem using the jump notation.

Theorem 5.9. *If $X \geq \emptyset^{(n)}$ computes $\mathscr{A}^{(n)}$, then there is a Y such that Y computes a copy $\mathscr{B}$ of $\mathscr{A}$ and $Y^{(n)} \equiv_T X$. Furthermore, an isomorphism between $\mathscr{A}$ and $\mathscr{B}$ can be found computably in X.*

This version of the theorem follows immediately form the proof of Theorem 3.20 and Observation 5.2. We will now use it as a tool to prove the full version of Theorem 3.9. Recall that in Section 4 we only proved the case $n = 1$.

Proof of Theorem 3.9. We already knew that (2) ⇒ (1). We will now prove (1) ⇒ (2). So, we have that R is relatively intrinsically Σ^0_{n+1}. We now claim that R is r.i.c.e. over $\mathscr{A}^{(n)}$, where $\mathscr{A}^{(n)}$ is the canonical nth jump of $\mathscr{A}$. To prove this claim, suppose $\mathscr{B}_n$ is a copy of $\mathscr{A}^{(n)}$, that $\mathscr{B}_n = \mathscr{B}^{(n)}$ and that X computes $D(\mathscr{B}^{(n)})$. By the jump inversion there is a Y such that $Y^{(n)} \equiv_T X$ and Y computes a copy $\mathscr{C}$ of $\mathscr{B}$, and X computes an isomorphism between $\mathscr{C}$ and $\mathscr{B}$. Since R is relatively intrinsically Σ^0_{n+1}, the relation $R^{\mathscr{C}}$ is Σ^0_{n+1} in Y and hence also c.e. in X, so that $R^{\mathscr{B}}$ is also c.e. in X (because the isomorphism is computable in X). Therefore, R is r.i.c.e. in $\mathscr{A}^{(n-1)}$ as claimed.

The just-proven claim implies that R is definable in $\mathscr{A}^{(n)}$ by a $\Sigma^{\mathtt{c}}_1$ formula. Since $\mathscr{A}^{(n)}$ comes equipped with a complete set of $\Pi^{\mathtt{c}}_n$ relations on $\mathscr{A}$, R is definable in $\mathscr{A}$ by a $\Sigma^{\mathtt{c}}_{n+1}$ formula. □

6. Connecting the notions

Given a class of structures $\mathbb{K}$ and $n \in \omega$, we ask the following questions: Does there exist a "natural" complete set of $\Pi^{\mathtt{c}}_n$ relations for $\mathbb{K}$? Is there, for every $D \subseteq \omega$, a structure $\mathscr{A} \in \mathbb{K}$ that encodes D in its n^{th} jump?

Of course, to answer the first question we would need to give a precise meaning to the idea of "natural" complete set of $\Pi^{\mathtt{c}}_n$ formulas. For this we will use the fact that all natural concepts in computability are relativizable. That is, if a natural set of formulas is complete $\Pi^{\mathtt{c}}_n$, it should also be complete $\Pi^{\mathtt{c}}_n$ relative to any oracle. Notice that this is the case with our natural examples, like $\{\mathtt{Succ}\}$, but it is not the case with the sequence of all $\Pi^{\mathtt{c}}_n$ formulas.

For this we will look at the boldface version of this notion.

Definition 6.1. A set of $\Pi^{\mathtt{in}}_n$ formulas $\{\phi_0, \phi_1, \ldots\}$ is a *complete set of* $\Pi^{\mathtt{in}}_n$ *formulas* if every $\Pi^{\mathtt{in}}_n$ formula is equivalent to a $\Sigma^{\mathtt{in}}_1$ formula over $L \cup \{\phi_0, \phi_1, \ldots\}$.

We aim to prove the following dichotomy theorem, whose proof is postponed pending further machinery that will be developed in the next sub-section.

Theorem 6.2. *Fix a class of structures $\mathbb{K}$ and $n \in \omega$. Either*

(1) *there is a countable complete set of $\Pi^{\mathtt{in}}_n$ formulas for $\mathbb{K}$ and*
(2) *no set D is coded in the $(n-1)^{st}$ jump of any structure $\mathscr{A} \in \mathbb{K}$ unless $D \leq_1 0^{(n)}$,*

or

(1) *there is no countable complete set of $\Pi^{\mathtt{in}}_n$ formulas and*
(2) *every set D is weakly coded in the $(n-1)^{st}$ jump of some structure $\mathscr{A} \in \mathbb{K}$,*

all relative to some oracle.

6.1. **Back-and-Forth Relations.** The main tool to prove theorem 6.2 will be the back-and-forth relations.

Definition 6.3. Fix a class $\mathbb{K}$ of structures. We define a relation $\leq_n$ for each n on pairs $(\mathscr{A}, \bar{a})$, where $\mathscr{A} \in \mathbb{K}$ and $\bar{a} \in A^{<\omega}$. Given $\mathscr{A}, \mathscr{B} \in \mathbb{K}$, $\bar{a} \in A^{<\omega}$, $\bar{b} \in B^{<\omega}$, with $|\bar{a}| = |\bar{b}|$. The relation $\leq_0$ is defined by $(\mathscr{A}, \bar{a}) \leq_0 (\mathscr{B}, \bar{b})$ if for any atomic formula ϕ (with index $\leq k_{|a|}$, where k_n is defined in 4.1) we have

$$A \models \phi(\bar{a}) \Leftrightarrow B \models \phi(\bar{b}),$$

or equivalently, if $D(\bar{a}) = D(\bar{b})$.

Supposing $\leq_n$ to be defined, we define

$$(\mathscr{A}, \bar{a}) \leq_{n+1} (\mathscr{B}, \bar{b}) \iff \forall \bar{d} \in B^{<\omega} \exists \bar{c} \in A^{<\omega} (A, \bar{a}, \bar{c}) \geq_n (\mathscr{B}, \bar{b}, \bar{d}).$$

To help understand this definition we present a few examples.

Example 6.4.

- *If $\mathscr{A}$ and $\mathscr{B}$ are linear orders, $\bar{a} = \langle a_1, ..., a_k \rangle \in A^k$, and $\bar{b} = \langle b_1, ..., b_k \rangle \in B^k$, then $(\mathscr{A}, \bar{a}) \leq_0 (\mathscr{B}, \bar{b})$ if and only if $a_i < a_j \Leftrightarrow b_i < b_j$ for $i, j \leq k$. Further, $(\mathscr{A}, \bar{a}) \leq_1 (\mathscr{B}, \bar{b})$ if and only if $|[a_i, a_{i+1}]| \geq |[b_i, b_{i+1}]|$ for each $i \leq k$, thinking of a_0 as $-\infty$ and a_{k+1} as ∞.*
- *If $\mathscr{A} = (\mathbb{Z}, <)$ and $\mathscr{B} = (\mathbb{Q}, <)$, then taking a_0 and b_0 to be one-element sequences in $\mathscr{A}$ and $\mathscr{B}$ respectively, we have $(\mathscr{A}, a_0) \equiv_1 (\mathscr{B}, b_0)$ but $(\mathscr{A}, a_0) \lneq_2 (\mathscr{B}, b_0)$. To see why the latter inequality is strict, note that by selecting $a_1 = a_0 + 1$ as the $\bar{d}$ in the definition, there is no $b_1 \in \mathbb{Q}$ so that $(\mathbb{Z}, a_0, a_1) \leq_1 (\mathbb{Q}, b_0, b_1)$ because we can find an element in $\mathbb{Q}$ between b_0 and b_1, but not an element in $\mathbb{Z}$ between a_0 and a_1.*

For the next theorem, we use the notation $\Pi^{\mathtt{in}}_n\text{-tp}_{\mathscr{A}}(\bar{a})$ to mean the set of all $\Pi^{\mathtt{in}}_n$ formulas satisfied by $\bar{a}$ (i.e. the $\Pi^{\mathtt{in}}_n$-*type of* $\bar{a}$).

Theorem 6.5. *The following are equivalent:*

(1) $(\mathscr{A}, \bar{a}) \leq_n (\mathscr{B}, \bar{b})$
(2) $\Pi^{\mathtt{in}}_n\text{-tp}_{\mathscr{A}}(\bar{a}) \subseteq \Pi^{\mathtt{in}}_n\text{-tp}_{\mathscr{B}}(\bar{b})$
(3) *Given that a structure $(\mathscr{C}, \bar{c})$ that is isomorphic to either $(\mathscr{A}, \bar{a})$ or $(\mathscr{B}, \bar{b})$, deciding whether $(\mathscr{C}, \bar{c}) \cong (\mathscr{A}, \bar{a})$ is $\mathbf{\Sigma}^0_n$-hard; i.e. given a $\mathbf{\Sigma}^0_n$ set $S \subseteq 2^\omega$, there is a continuous function $f\colon 2^\omega \to \mathbb{K} \times \omega^{|a|}$ such that*

$$f(X) \cong \begin{cases} (\mathscr{A}, \bar{a}) & \text{if } X \in S \\ (\mathscr{B}, \bar{b}) & \text{if } X \notin S. \end{cases}$$

Observation 6.6. By the (2) on the previous Theorem we can easily prove that the relation $\leq_n$ is both reflexive and transitive. Therefore $\leq_n$ imposes an equivalence relation $\equiv_n$ on $\mathbb{K}$.

Notation 6.7. We will use lowercase greek letters, α, β, etc., for the equivalence classes of $\equiv_n$. Further, we say that a tuple $(\mathscr{A}, \bar{a})$ has n-type α, and we write $n\text{-tp}(\mathscr{A}, \bar{a}) = \alpha$, if $(\mathscr{A}, \bar{a})$ belongs to the equivalence class α. Of course, α can be seen as a complete $\Pi_n^{\mathtt{in}}$-type, as all the tuples in α have the same $\Pi_n^{\mathtt{in}}$-type. We use $\Pi_n^{\mathtt{in}}\text{-tp}(\alpha)$ to denote this type.

Definition 6.8. $\mathbf{bf}_n(\mathbb{K}) = \{(\mathscr{A}, \bar{a}) : \mathscr{A} \in \mathbb{K}, \bar{a} \in A^{<\omega}\} \,/ \equiv_n$ denotes the set of the n-back-and-forth equivalence classes.

Note that $(\mathbf{bf}_n(\mathbb{K}), \leq_n)$ is a partial ordering. We will see that the size of $\mathbf{bf}_n(\mathbb{K})$ will give us useful information about the structures in $\mathbb{K}$. Since by definition $\leq_n$ is Borel, the following theorem, due to Silver, reduces the posibilies to just two.

Theorem 6.9 (Silver [Sil80])**.** *Every Borel equivalence relation on 2^ω has either countable or $2^{\aleph_0}$ many equivalence classes.*

Corollary 6.10. *$|\mathbf{bf}_n(\mathbb{K})|$ is either countable or $2^{\aleph_0}$.*

Example 6.11. *All these examples require proofs which we won't include here.*

(1) *If $\mathbb{K}$ is the class of Boolean Algebras, then $\forall n \in \omega$, $|\mathbf{bf}_n(\mathbb{K})| \leq \aleph_0$.*
(2) *If $\mathbb{K}$ is the class of Linear Orderings, then* $|\mathbf{bf}_n(\mathbb{K})| = \begin{cases} \aleph_0 & \textit{for } n = 1, 2; \\ 2^{\aleph_0} & \textit{for } n \geq 3. \end{cases}$
(3) *If $\mathbb{K}$ is the class of Equivalence Structures, then* $|\mathbf{bf}_n(\mathbb{K})| = \begin{cases} \aleph_0 & \textit{for } n = 1; \\ 2^{\aleph_0} & \textit{for } n \geq 2. \end{cases}$

Notation 6.12. Since we have defined $\leq_n$ between pairs of the form $(\mathscr{A}, \bar{a})$, if α is the n-type of $(\mathscr{A}, \bar{a})$, we denote $|\alpha|$ to be lenght of the tuple $\bar{a}$.

For $\alpha \in \mathbf{bf}_n(\mathbb{K})$, given a $\Pi_n^{\mathtt{in}}$ formula $\varphi(\bar{x})$ with $|\bar{x}| = |\alpha|$, we write $\alpha \models \varphi$ if $\varphi \in \Pi_n^{\mathtt{in}}\text{-tp}(\alpha)$. For each $\alpha \in \mathbf{bf}_n(\mathbb{K})$, we let

$$ext_n(\alpha) \subseteq \mathbf{bf}_{n-1}(\mathbb{K})$$

be the set of all $\delta \in \mathbf{bf}_{n-1}(\mathbb{K})$ such that for all $(\mathscr{A}, \bar{a})$ with $n\text{-tp}(\mathscr{A}, \bar{a}) = \alpha$, there exists $\bar{c}$ such that $(n-1)\text{-tp}(\mathscr{A}, \bar{a}, \bar{c}) \geq_{n-1} \delta$.

Observation 6.13. Straight from the definition of $ext_n(\alpha)$ we have:

- $ext_n(\alpha)$ is closed downwards under $\leq_{n-1}$;
- $\alpha \leq_n (\mathscr{B}, \bar{b}) \Leftrightarrow (\forall \bar{d} \in B^{<\omega})\ (n-1)\text{-tp}(\mathscr{B}, \bar{b}, \bar{d}) \in ext_n(\alpha)$; and
- $\alpha \leq_n \beta \Leftrightarrow ext_n(\alpha) \supseteq ext_n(\beta)$.

We now begin building the machinery needed for the proof of Theorem 6.16.

Lemma 6.14. *If $\mathbf{bf}_{n-1}(\mathbb{K})$ is countable, then for each $\alpha \in \mathbf{bf}_n(\mathbb{K})$ there exists a $\Pi_n^{\mathtt{in}}$ formula, $\varphi_\alpha(\bar{x})$, such that for every $\mathscr{B} \in \mathbb{K}$, and $\bar{b} \in B^{|\alpha|}$,*

$$\alpha \leq_n (\mathscr{B}, \bar{b}) \iff \mathscr{B} \models \varphi_\alpha(\bar{b}) \iff \varphi_\alpha \in \Pi_n^{\mathtt{in}}\text{-tp}_{\mathscr{B}}(\bar{b}).$$

Proof. Suppose that for each $\delta \in \mathbf{bf}_{n-1}(\mathbb{K})$ we already have a $\Pi^{\mathtt{in}}_{n-1}$ formula φ_δ as wanted. Then have that

$$\begin{aligned}
\alpha \leq_n (\mathscr{B}, \bar{b}) &\Leftrightarrow (\forall \bar{d} \in B^{<\omega})(n-1)\text{-tp}(\mathscr{B}, \bar{b}, \bar{d}) \in ext_n(\alpha) \\
&\Leftrightarrow \neg(\exists \bar{d} \in B^{<\omega})(n-1)\text{-tp}(\mathscr{B}, \bar{b}, \bar{d}) \notin ext_n(\alpha) \\
&\Leftrightarrow \neg(\exists \bar{d} \in B^{<\omega}) \bigvee_{\substack{\delta \in \mathbf{bf}_{n-1}(\mathbb{K}); \\ \delta \notin ext_n(\alpha)}} \delta \leq_{n-1} (\mathscr{B}, \bar{b}, \bar{d}) \\
&\Leftrightarrow \mathscr{B} \models \neg \bigvee_{\substack{\delta \in \mathbf{bf}_{n-1}(\mathbb{K}) \\ \delta \notin ext_n(\alpha)}} (\exists \bar{y}) \varphi_\delta(\bar{b}, \bar{y}).
\end{aligned}$$

Where the third equivalence uses that $ext_n(\alpha)$ is closed downwards. Notice that the formula in the last line is $\Pi^{\mathtt{in}}_{n-1}$, and that the infinitary disjunction is countable because $\mathbf{bf}_{n-1}(\mathbb{K})$ is countable. Therefore, $\varphi_\alpha(\bar{x}) = \bigwedge_{\substack{\delta \in \mathbf{bf}_{n-1}(\mathbb{K}) \\ \delta \notin ext_n(\alpha)}} (\forall \bar{y}) \neg \varphi_\delta(\bar{x}, \bar{y})$ is as wanted. $\square$

Lemma 6.15. *If $|\mathbf{bf}_n(\mathbb{K})| \leq \aleph_0$, then there exists a countable complete set of $\Pi^{\mathtt{in}}_n$ formulas.*

Proof. We will show that $\{\varphi_\alpha : \alpha \in \mathbf{bf}_n(\mathbb{K})\}$ is $\Pi^{\mathtt{in}}_n$-complete. Let ψ be any $\Pi^{\mathtt{in}}_n$ formula. We claim that

$$\psi(\bar{x}) \Leftrightarrow \bigvee_{\substack{\alpha \in \mathbf{bf}_n(\mathbb{K}) \\ |\alpha| = |\bar{x}| \\ \alpha \models \psi}} \varphi_\alpha.$$

($\Rightarrow$) Assume $\mathscr{A} \models \psi(\bar{a})$ and let α be the n-type of $(\mathscr{A}, \bar{a})$. Then $\alpha \models \psi$ and $\mathscr{A} \models \varphi_\alpha(\bar{a})$. Therefore $(\mathscr{A}, \bar{a})$ satisfies the right-hand-side.

($\Leftarrow$) Suppose $(\mathscr{A}, \bar{a})$ satisfies the right-hand-side. Then, for some α from the infinitary disjunction, $\mathscr{A} \models \varphi_\alpha(\bar{a})$. Therefore, $\alpha \leq_n (\mathscr{A}, \bar{a})$ and $\alpha \models \psi$. Since ψ is $\Pi^{\mathtt{in}}_n$, $\mathscr{A} \models \psi(\bar{a})$ too.

This proves the claim and the lemma. $\square$

Notation 6.16. We let $\Pi^{\mathtt{in}}_n$-impl(φ_α) denote the set of all $\Pi^{\mathtt{in}}_n$-formulas implied by φ_α in the class $\mathbb{K}$.

Observation 6.17. Let $\alpha \in \mathbf{bf}_n(\mathbb{K})$, then from Lemma 6.14 above, we get that $\Pi^{\mathtt{in}}_n$-tp$(\alpha) = \Pi^{\mathtt{in}}_n$-impl$(\varphi_\alpha)$, because both are equal to $\bigcap_{\beta \geq_n \alpha} \Pi^{\mathtt{in}}_n$-tp$(\beta)$.

The following theorem provides the first big step towards proving Theorem 6.16 while at the same time unifying the concepts discussed in this section and those of complete set of formulas.

Theorem 6.18. *For a class of structures $\mathbb{K}$ and $n \in \omega$, we have that $|\mathbf{bf}_n(\mathbb{K})| = \aleph_0$ if and only if there exists a countable complete set of $\Pi^{\mathtt{in}}_n$-formulas.*

Proof. The left-to-right implication was proved in Lemma 6.15. To prove the other direction suppose that $\{R_1, R_2, \ldots\}$ is a countable complete set of $\Pi^{\mathtt{in}}_n$-formulas. We will prove, by induction on $k \leq n$, that $|\mathbf{bf}_k(\mathbb{K})| = \aleph_0$. So, suppose that $|\mathbf{bf}_{k-1}(\mathbb{K})| = \aleph_0$. We claim that for each $\alpha \in \mathbf{bf}_k(\mathbb{K})$ there exists a finitary Σ_1-formula ψ_α over $L \cup \{R_1, \ldots\}$ such that $\Pi^{\mathtt{in}}_n$-tp$(\alpha) = \Pi^{\mathtt{in}}_n$-impl$(\psi_\alpha)$. Then, since there are only $\aleph_0$ many such Σ_1

finitary formulas, the claim implies that $\mathbf{bf}_k(\mathbb{K})$ is countable, and the theorem follows. Let us now prove the claim. Since $|\mathbf{bf}_{k-1}(\mathbb{K})| = \aleph_0$, we know that for each $\alpha \in \mathbf{bf}_k(\mathbb{K})$, there exists a $\Pi^{\mathtt{in}}_k$-formula φ_α such that $\Pi^{\mathtt{in}}_k\text{-tp}(\alpha) = \Pi^{\mathtt{in}}_k\text{-impl}(\varphi_\alpha)$. Since $\{R_1, R_2, \ldots\}$ is a countable complete set of $\Pi^{\mathtt{in}}_n$-formulas, φ_a it is equivalent to a $\Sigma^{\mathtt{in}}_1$-formula over $L \cup \{R_1, \ldots\}$. So, $\varphi_\alpha \equiv \bigvee_{i\in\omega} \psi_i$, where each ψ_i is finitary Σ_1 over $L \cup \{R_1, \ldots\}$. Take $(\mathscr{A}, \bar{a})$ of type α, and, since $\mathscr{A} \models \varphi_\alpha(\bar{a})$, take i such that $\mathscr{A} \models \psi_i(\bar{a})$. Now,

$$\Pi^{\mathtt{in}}_k\text{-tp}(\alpha) = \Pi^{\mathtt{in}}_k\text{-impl}(\varphi_\alpha) \subseteq \Pi^{\mathtt{in}}_k\text{-impl}(\psi_i) \subseteq \Pi^{\mathtt{in}}_k\text{-tp}_{\mathscr{A}}(\bar{a}) = \Pi^{\mathtt{in}}_k\text{-tp}(\alpha).$$

Therefore $\Pi^{\mathtt{in}}_k\text{-tp}(\alpha) = \Pi^{\mathtt{in}}_k\text{-impl}(\psi_i)$ and the claim is proved. □

Lemma 6.19. *If $|\mathbf{bf}_n(\mathbb{K})| = \aleph_0$, then there exists an oracle X such that if D is encoded by the the $(n-1)$st jump of some structure in $\mathbb{K}$, then $D \leq_T X$.*

Proof. The reason is that there are countably many $\Sigma^{\mathtt{c}}_n$-types of tuples from structures in $\mathbb{K}$, and every set D coded by some structure in $\mathbb{K}$ has to be enumeration reducible to one of these. All we need to do is let X bound the jumps of these countably many $\Sigma^{\mathtt{c}}_n$-types. □

Observe that the previous results provide a proof for the first part of Theorem 6.2. The following discussion will focus on the case where $|\mathbf{bf}_n(\mathbb{K})|$ is uncountable.

Definition 6.20. The *bf-ordinal* of $\mathbb{K}$ is the least γ such that $|\mathbf{bf}_\gamma(\mathbb{K})| > \aleph_0$ if such a γ exists and ∞ otherwise.

If $\mathbb{K}$ is a class of countable structures, as all the ones we are considering, one can show that $\mathbb{K}$ has bf-ordinal ∞ if and only if $\mathbb{K}$ contains only countably many isomorphism types, and otherwise the bf-ordinal of $\mathbb{K}$ is at most ω_1. Also, it is not hard to prove that if $\mathbb{K}$ has bf-ordinal ω_1 then $\mathbb{K}$ has $\aleph_1$ many isomorphism types. This is the case, for instance, when $\mathbb{K}$ is the class of all countable well-orders. If $\mathbb{K}$ is first order axiomatizable, it is unknown whether $\mathbb{K}$ can have size $\aleph_1$, in the case when $\aleph_1 \neq 2^{\aleph_0}$. That this is not possible is the well-known Vaught conjecture. It is also not known in the case where $\mathbb{K}$ is a Borel class of countable structures.

Corollary 6.21. *If the Turing ordinal of $\mathbb{K}$ exists and is n, then the bf-ordinal of $\mathbb{K}$ is $\leq n$.*

Theorem 6.22. *If $|\mathbf{bf}_n(\mathbb{K})| = 2^{\aleph_0}$ then, relative to some oracle X, every $D \in 2^\omega$ can be weakly coded in $(n-1)$th jump of some $\mathscr{A} \in \mathbb{K}$.*

Proof. Suppose that there are countably many $(n-1)$-bftypes. Otherwise, replace the existing n by the least n such that there are continuum many n-bftypes, and note that if the theorem is true for the new value of n, it is true for all $m \geq n$. For some $k \in \omega$, we have that $\{\alpha \in \mathbf{bf}_n(\mathbb{K}) : |\alpha| = k\}$ has size continuum. We will assume $k = 0$ to simplify the notation needed in the proof; the general case is essentially the same.

Since $\mathbf{bf}_{n-1}(\mathbb{K})$ is countable, we know there is a complete set of $\Pi^{\mathtt{in}}_{n-1}$ formulas. Extend the language to $\hat{\mathcal{L}}$ by adding all these formulas. If $\hat{\mathcal{L}}$ is not computable, relativize the rest of the proof to the Turing degree of $\hat{\mathcal{L}}$ and of all the degrees of the formulas we just added. Thus, all the $\Sigma^{\mathtt{in}}_n$ $\mathcal{L}$-formulas are equivalent to $\Sigma^{\mathtt{in}}_1$ $\hat{\mathcal{L}}$-formulas, and the $\Sigma^{\mathtt{c}}_n$-$\mathcal{L}$-types of the tuples in $\mathbb{K}$ are determined by their finitary-Σ_1-$\hat{\mathcal{L}}$-types.

Now we define $t_{\mathscr{A}} \in 2^\omega$ to be the characteristic function of the finitary-Σ_1-$\hat{\mathcal{L}}$ theory of $\mathscr{A}$. More formally: Enumerate all the finitary-Σ_1-$\hat{\mathcal{L}}$ sentences in a list $(\psi_0, \psi_1, ...)$. For every structure $\mathscr{A}$ let $t_{\mathscr{A}} \in 2^\omega$ be such that $t_{\mathscr{A}}(i) = 1$ if $\mathscr{A} \models \psi_i$ and $t_{\mathscr{A}}(i) = 0$ otherwise. Observe that the set $\{i : t_{\mathscr{A}}(i) = 1\}$ can be coded by the $(n-1)$st jump of $\mathscr{A}$ (because the $(n-1)$st jump of any presentation of $\mathscr{A}$ can compute the relations in $\hat{\mathcal{L}}$ and then enumerate Σ_1-$\hat{\mathcal{L}}$-$\text{tp}_{\mathscr{A}}$). Let $R = \{t_{\mathscr{A}} : \mathscr{A} \in \mathbb{K}\} \subseteq 2^\omega$. Note that $\Sigma_n^{\mathtt{in}}$-$\text{tp}_{\mathscr{A}}$ is determined by $t_{\mathscr{A}}$, and hence $t_{\mathscr{A}} = t_{\mathscr{B}}$ if and only if $\mathscr{A} \equiv_n \mathscr{B}$. Thus, since $|\{\alpha \in \mathbf{bf}_n(\mathbb{K}) : |\alpha| = 0\}| = 2^{\aleph_0}$, R has size continuum. Notice that $R \subseteq 2^\omega$ is a $\mathbf{\Sigma}_1^1$ class, because R is the image of $\mathbb{K}$ under t, $\mathbb{K}$ is Borel, and t is arithmetic. Since R is uncountable and $\mathbf{\Sigma}_1^1$, Suslin's theorem (see [Mos80, Corollary 2C.3]) says that R has a perfect closed subset $[T]$, determined by some perfect tree $T \subseteq 2^{<\omega}$ (where $[T]$ is the set of paths through T). In what follows, we relativize our construction to T, so we assume T is computable. Thinking of T as an order-preserving map $2^\omega \to 2^\omega$, for $X \in 2^\omega$ we let $T(X)$ be the path through T obtained as the image of X under this map. For each X, $T(X)$ gives us a Σ_1-$\hat{\mathcal{L}}$-type that is consistent with $\mathbb{K}$ and of Turing degree X (modulo all the relativization we have already done). There is some $\mathscr{A} \in \mathbb{K}$ with Σ_1-$\hat{\mathcal{L}}$-type $t_{\mathscr{A}} = T(X)$, and hence $T(X)$ can be enumerated by the $(n-1)$st jump of any presentation of $\mathscr{A}$. One can show that $\{\sigma \in 2^{<\omega} : \sigma \leq_{\mathbb{Q}} X\}$ is enumeration reducible to $T(X)$. If follows that X is weakly coded by the $(n-1)$st jump of $\mathscr{A}$. We chose X arbitrarily, so any set can be weakly coded into the $(n-1)$st jump of some structure $\mathscr{A}$ of $\mathbb{K}$. $\square$

References

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

[DJ94] Rod Downey and Carl G. Jockusch. Every low Boolean algebra is isomorphic to a recursive one. *Proc. Amer. Math. Soc.*, 122(3):871–880, 1994.

[Fro06] A. N. Frolov. Δ_2^0-copies of linear orderings. *Algebra Logika*, 45(3):354–370, 376, 2006.

[HM] K. Harris and A. Montalbán. On the n-back-and-forth types of boolean algebras. To appear in the Transactions of the AMS.

[JS94] Carl G. Jockusch, Jr. and Robert I. Soare. Boolean algebras, Stone spaces, and the iterated Turing jump. *J. Symbolic Logic*, 59(4):1121–1138, 1994.

[Kni86] Julia F. Knight. Degrees coded in jumps of orderings. *J. Symbolic Logic*, 51(4):1034–1042, 1986.

[Kni98] J. F. Knight. Degrees of models. In *Handbook of recursive mathematics, Vol. 1*, volume 138 of *Stud. Logic Found. Math.*, pages 289–309. North-Holland, Amsterdam, 1998.

[KS00] Julia F. Knight and Michael Stob. Computable Boolean algebras. *J. Symbolic Logic*, 65(4):1605–1623, 2000.

[Mon10] A. Montalbán. Counting the back-and-forth types. *Journal of Logic and Computability*, 2010.

[Mos80] Yiannis N. Moschovakis. *Descriptive set theory*, volume 100 of *Studies in Logic and the Foundations of Mathematics*. North-Holland Publishing Co., Amsterdam, 1980.

[Ric81] Linda Jean Richter. Degrees of structures. *J. Symbolic Logic*, 46(4):723–731, 1981.

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

[Sil80] Jack H. Silver. Counting the number of equivalence classes of Borel and coanalytic equivalence relations. *Ann. Math. Logic*, 18(1):1–28, 1980.

[Thu95] John J. Thurber. Every low_2 Boolean algebra has a recursive copy. *Proc. Amer. Math. Soc.*, 123(12):3859–3866, 1995.