Which Classes of Structures Are Both Pseudo-elementary and Definable by an Infinitary Sentence?

Barbara F. Csima*, Nancy A. Day, Matthew Harrison-Trainor†

August 5, 2018

Abstract

When classes of structures are not first-order definable, we might still try to find a nice description. There are two common ways for doing this. One is to expand the language, leading to notions of pseudo-elementary classes, and the other is to allow infinite conjuncts and disjuncts. In this paper we examine the intersection. Namely, we address the question: Which classes of structures are both pseudo-elementary and $\mathcal{L}_{\omega_1\omega}$-elementary? We find that these are exactly the classes that can be defined by an infinitary formula that has no infinitary disjunctions.

1 Introduction

It is well-known that many properties are not definable in elementary first-order logic, even by a theory rather than a single sentence. Common examples are the property (of graphs) of being connected, the property (of abelian groups) of being torsion, and the property (of linear orders) of being well-founded. To capture such properties, one can pass to extensions of elementary first-order logic. This paper is about a characterization of the common expressive power of two such extensions.

The first extension of elementary first-order logic that we consider is to allow countably infinite conjunctions and disjunctions; this is, morally, similar to allowing quantifiers over the (standard) natural numbers. One can then define properties such as being torsion by saying "for each group element x, there is n such that $nx = 0$", or formally,

$$(\forall x) \bigvee_{n \in \mathbb{N}} nx = 0.$$

This work grew out of initial discussions with Vakili about the generality of expressing properties not definable in first-order logic in a pseudo-elementary way, and whether such phenomena might be of use for model checking (as the pseudo-elementary definability of graph reachability was used for model checking by Vakili in his thesis [Vak16] and with the second author in [VD14]).

*Partially supported by Canadian NSERC Discovery Grant 312501.

†Supported by an NSERC Banting Fellowship.

This infinitary logic is known as $\mathcal{L}_{\omega_1\omega}$. One loses compactness, but gains other powerful tools. For example, every countable structure is described, up to isomorphism among countable structures, by a sentence of $\mathcal{L}_{\omega_1\omega}$ [Sco65].

The second extension of elementary first-order logic is to allow existential second-order quantifiers. For example, the property of a linear order being non-well-founded can be defined by the sentence "there is a set with no least element". We say that such a property is pseudo-elementary. More formally, a property P of τ-structures is pseudo-elementary if there is an expanded language $\tau^* \supseteq \tau$ and an τ^*-sentence φ (or τ^*-theory T) such that the τ-structures admitting an τ^*-expansion to a model of φ (respectively T) are exactly the structures satisfying P. We will describe both of these extensions of first-order logic in more detail later.

These two extensions of elementary first-order logic have different descriptive powers. For example, the property of being non-well-founded is pseudo-elementary but not $\mathcal{L}_{\omega_1\omega}$-definable. Also, the negation of a pseudo-elementary property is not necessarily pseudo-elementary, but the negation of an $\mathcal{L}_{\omega_1\omega}$-definable property is again $\mathcal{L}_{\omega_1\omega}$-definable. Nevertheless, there are properties that are not elementary first-order definable, but that are both pseudo-elementary and $\mathcal{L}_{\omega_1\omega}$-definable. The property of a graph being disconnected is such an example; we provide a more detailed discussion of various examples in Section 3. The main result of this paper is a complete classification of such properties.

Theorem 1.1. *Let $\mathbb{K}$ be a class of structures closed under isomorphism. The following are equivalent:*

- $\mathbb{K}$ *is both a pseudo-elementary* (PC_Δ) *class and* $\mathcal{L}_{\omega_1\omega}$*-elementary.*
- $\mathbb{K}$ *is defined by a* $\bigwedge\!\!\!\bigwedge$*-sentence.*

There is some notation in this theorem that we must explain. First, there are some subtleties in the definition of what it means for a property to be pseudo-elementary, and in fact there are four different natural definitions (giving rise to three distinct notions). Two of them are as follows.

Definition 1.2. We say that a class $\mathbb{K}$ of τ-structures is a PC-class if there is a language $\tau^* \supseteq \tau$ and an elementary first-order τ^*-sentence ϕ such that

$$\mathbb{K} = \{\mathcal{M} \mid \text{there is an } \tau^*\text{-structure } \mathcal{M}^* \text{ expanding } \mathcal{M} \text{ with } \mathcal{M}^* \models \phi\}.$$

We say that $\mathbb{K}$ is a PC_Δ-class if ϕ is replaced by an elementary first-order theory.

So the theorem above is concerned with pseudo-elementary classes where one is allowed to use a theory in the definition.

The $\bigwedge\!\!\!\bigwedge$-sentences in the theorem are the $\mathcal{L}_{\omega_1\omega}$ sentences which (in normal form) involve infinitary conjunctions, but no infinitary disjunctions. For example, the property of being infinite is definable by the $\bigwedge\!\!\!\bigwedge$-sentence

$$\bigwedge\!\!\!\bigwedge_{n\in\mathbb{N}} \exists x_1,\dots,x_n \Big(\bigwedge_{i\neq j} x_i \neq x_j\Big).$$

The negation, the property of being finite, is $\mathcal{L}_{\omega_1\omega}$-definable by the sentence

$$\bigvee\!\!\!\!\bigvee_{n\in\mathbb{N}} \forall x_1,\ldots,x_n(\bigvee_{i\neq j} x_i = x_j)$$

but this sentence is not a $\bigwedge\!\!\!\!\bigwedge$-sentence because it involves an infinitary disjunct.

Definition 1.3. The $\bigwedge\!\!\!\!\bigwedge$-formulas are defined inductively as follows:

- every finitary quantifier-free formula is a $\bigwedge\!\!\!\!\bigwedge$-formula
- if φ is a $\bigwedge\!\!\!\!\bigwedge$-formula, then so are $(\exists x)\varphi$ and $(\forall x)\varphi$
- if $(\varphi_i)_{i\in\omega}$ are $\bigwedge\!\!\!\!\bigwedge$-formulas with finitely many free variables, then so is $\bigwedge\!\!\!\!\bigwedge_{i\in\omega}\varphi_i$.

In one direction, the proof of Theorem 1.1 passes through a modification of the proof of Craig Interpolation for $\mathcal{L}_{\omega_1\omega}$. This was originally proved by Lopez-Escobar [LE65] who also gave the following corollary: a property which is both pseudo-elementary and co-pseudo-elementary with respect to $\mathcal{L}_{\omega_1\omega}$ (i.e., both $\mathbf{\Sigma}^1_1$ and $\mathbf{\Pi}^1_1$) is actually $\mathcal{L}_{\omega_1\omega}$-definable. So it is not too surprising that we can use similar ideas.

In the other direction, the proof passes through the coding of computable formulas in expansions of models of arithmetic. One of the results along the way is of independent interest.

Theorem 1.4. *Let $\mathbb{K}$ be a class of structures in a finite language that is axiomatized by a computable $\bigwedge\!\!\!\!\bigwedge$-sentence. Then $\mathbb{K}$ is pseudo-elementarily defined using a single sentence (PC').*

We have not yet defined the PC$'$ classes, but they are similar to the PC classes in that they are axiomatized with a single sentence rather than a theory. (We note that for classes which have only infinite models, PC and PC$'$ coincide.) In particular, noting the fact that the conjunction of the sentences in an elementary first-order theory is a $\bigwedge\!\!\!\!\bigwedge$-sentence, we get the following corollary.

Corollary 1.5. *Every computably axiomatizable class in a finite language is a* PC$'$ *class.*

What this says is that an elementary class that has an infinite but computable axiomatization can actually be axiomatized by a single sentence in an expanded language. Unfortunately, we do not know how to reverse these results. We have the following conjecture.

Conjecture 1.6. *A* PC *or* PC$'$ *class which is also $\mathcal{L}_{\omega_1\omega}$-axiomatizable is axiomatizable by a computable $\bigwedge\!\!\!\!\bigwedge$-sentence.*

The reader may wonder here why the connection with computability theory arises. It is, essentially, because a single elementary first-order sentence forms a computable theory with a computably enumerable set of consequences, and, on the other hand, that a computable set has a an elementary first-order Σ_1 definition in arithmetic.

The following result, in the special case of classes which are closed under substructures, has been known since Tarski [Tar54a, Tar54b] and Mal'cev [Mal41].

Theorem 1.7 (Mal'cev, Tarski, see Theorem 6.6.7 of [Hod08] or Theorem 9.14 of [HH02]). *If $\mathbb{K}$ is a PC_Δ-class which is closed under substructures, then it axiomatized by a set of universal sentences. Moreover, if the theory T by which $\mathbb{K}$ is PC_Δ is computably enumerable, then so is the set of universal sentences.*

It is obvious that a class axiomatized by universal sentences is PC_Δ and closed under substructures. We get the following new reversal of the "moreover" clause:

Theorem 1.8. *Let $\mathbb{K}$ be a class of structures closed under isomorphism. The following are equivalent:*

- *$\mathbb{K}$ is a PC'-class that is closed under substructures,*
- *$\mathbb{K}$ is axiomatized by a computably enumerable universal theory.*

The paper is laid out as follow. In Section 2 we introduce formal definitions of infinitary logic and pseudo-elementary classes and some basic results. In Section 3, we will give several examples. In Section 4, we prove an interpolation theorem which gives one direction of Theorem 1.1. Finally, in Section 5 we prove the other direction by coding infinitary sentences into models of arithmetic.

2 Notation and Definitions

2.1 Infinitary Logic

For the most part, we follow Marker's new book [Mar16]. We want to be precise with our definitions here, because we will need to encode infinitary formulas in first-order sentences. We first define $\mathcal{L}_{\omega_1\omega}$ formulas. Throughout the paper, let τ be a countable language.

Definition 2.1. The $\mathcal{L}_{\omega_1\omega}(\tau)$-formulas are defined inductively as follows:

- every atomic τ-formula is an $\mathcal{L}_{\omega_1\omega}(\tau)$-formula,
- if φ is an $\mathcal{L}_{\omega_1\omega}(\tau)$-formula, then so are $\neg\varphi$, $(\exists x)\varphi$ and $(\forall x)\varphi$,
- if $(\varphi_i)_{i\in\omega}$ are $\mathcal{L}_{\omega_1\omega}(\tau)$-formulas with finitely many free variables, then so are $\bigwedge\!\!\!\bigwedge_{i\in\omega}\varphi_i$ and $\bigvee\!\!\!\bigvee_{i\in\omega}\varphi_i$.

In general, we will drop the reference to τ when it is clear what we mean.

Definition 2.2. An $\mathcal{L}_{\omega_1\omega}$ formula is in $\mathcal{L}_{\omega_1\omega}$ *normal form* if the $\neg$ only occurs applied to atomic formulas.

Every $\mathcal{L}_{\omega_1\omega}$ can be placed into a normal form. The negation $\neg\varphi$ of a sentence φ in normal form is not immediately in normal form itself. This gives rise to the formal negation $\sim\varphi$, which is logically equivalent to $\neg\varphi$ but is in normal form.

Definition 2.3. For any $\mathcal{L}_{\omega_1\omega}$-formula φ, the formula $\sim\varphi$ is defined inductively as follows:

- if φ is atomic, $\sim\varphi$ is $\neg\varphi$,
- $\sim\neg\varphi$ is φ, $\sim(\exists x)\varphi$ is $(\forall x)\sim\varphi$ and $\sim(\forall x)\varphi$ is $(\exists x)\sim\varphi$,
- $\sim \bigwedge\!\!\!\bigwedge_{i\in\omega} \varphi_i$ is $\bigvee\!\!\!\bigvee_{i\in\omega} \sim\varphi_i$ and $\sim \bigvee\!\!\!\bigvee_{i\in\omega} \varphi_i$ is $\bigwedge\!\!\!\bigwedge_{i\in\omega} \sim\varphi_i$.

We repeat again the definition of a $\bigwedge\!\!\!\bigwedge$-formula.

Definition 2.4. An $\mathcal{L}_{\omega_1\omega}$-sentence φ is a $\bigwedge\!\!\!\bigwedge$-formula if it can be written in normal form without any infinite disjunctions. More concretely, the $\bigwedge\!\!\!\bigwedge$-formulas are defined inductively as follows:

- every finitary quantifier-free sentence is a $\bigwedge\!\!\!\bigwedge$-formula,
- if φ is a $\bigwedge\!\!\!\bigwedge$-formula, then so are $(\exists x)\varphi$ and $(\forall x)\varphi$,
- if $(\varphi_i)_{i\in\omega}$ are $\bigwedge\!\!\!\bigwedge$-formulas with finitely many free variables, then so is $\bigwedge\!\!\!\bigwedge_{i\in\omega} \varphi_i$.

An $\mathcal{L}_{\omega_1\omega}$ formula is computable if, essentially, there is a computable syntactic representation of the formula. To define the computable $\mathcal{L}_{\omega_1\omega}$ formulas, we will encode in a labeled tree the way that a formula is built from atomic formulas. The labeled trees build formulas in normal form. The condition (a) below is somewhat non-standard in that we allow disjunctions but not conjunctions; this is merely a cosmetic change which will make our lives easier later.

Definition 2.5. Assume that we have a fixed Gödel coding of atomic τ-formulas, connectives, and quantifiers. A *labeled tree* is a non-empty tree $T \subseteq \omega^{<\omega}$ with functions l and v such that for each $\sigma \in T$, one of the following is true:

(a) σ is a terminal node of T and $l(\sigma) = \ulcorner\psi\urcorner$ where ψ is a finite disjunct of atomic and negated atomic formulas, and $\{x_i \mid i \in v(\sigma)\}$ is the set of free variables in ψ;

(b) $l(\sigma) = \ulcorner\exists x_i\urcorner$, $\sigma\hat{\ }0$ is the unique successor of σ in T, and $v(\sigma) = v(\sigma\hat{\ }0) \setminus \{i\}$;

(c) $l(\sigma) = \ulcorner\forall x_i\urcorner$, $\sigma\hat{\ }0$ is the unique successor of σ in T, and $v(\sigma) = v(\sigma\hat{\ }0) \setminus \{i\}$;

(d) $l(\sigma) = \ulcorner\bigwedge\!\!\!\bigwedge\urcorner$ and $v(\sigma) = \bigcup_{\sigma\hat{\ }i\in T} v(\sigma\hat{\ }i)$ and is finite;

(e) $l(\sigma) = \ulcorner\bigvee\!\!\!\bigvee\urcorner$ and $v(\sigma) = \bigcup_{\sigma\hat{\ }i\in T} v(\sigma\hat{\ }i)$ and is finite.

A *formula code* is a well-founded labeled tree (T, l, v). A *sentence code* is a formula code where $v(\varnothing) = \varnothing$. The formula coded by a formula code (T, l, v) can be defined inductively in the obvious way. A formula is *computable* if it has a computable formula code, i.e., if T, l, and v are computable. Moreover, this relativizes: a formula is C-computable if it has a C-computable formula code.

Every formula is C-computable for sets C of sufficiently high degree. A computable $\bigwedge\!\!\!\bigwedge$-formula is just one that has a computable formula code that does not involve any infinitary disjunctions.

Definition 2.6. We say φ is a (C-)computable $\bigwedge\!\!\!\bigwedge$-formula if φ has a (C-)computable formula code (T, l, v) such that for no $\sigma \in T$ do we have $l(\sigma) = \ulcorner \bigvee\!\!\!\bigvee \urcorner$.

It seems possible that a formula might be both a computable $\mathcal{L}_{\omega_1\omega}$-sentence, and a $\bigwedge\!\!\!\bigwedge$-sentence, without being a computable $\bigwedge\!\!\!\bigwedge$-sentence. This is analogous to the fact that a sentence might be both computable $\mathcal{L}_{\omega_1\omega}$ and infinitary Π^0_1 without being computable Π^0_1.

The satisfaction relation for $\mathcal{L}_{\omega_1\omega}$ is easy to define, but we will make use of the following witnesses to the truth value of a formula. If η is a (possibly partial) variable assignment, we will use the notation $\eta^{x\mapsto a}$ to denote the modification of η to map x to a.

Definition 2.7. Let $\mathcal{M}$ be a model and (T, l, v) a labeled tree. We define "f is a truth definition for (T, l, v) in $\mathcal{M}$" as follows:

- The domain of f is pairs (σ, η) where $\sigma \in T$ and η is an valuation of the free variables, and $f(\sigma, \eta) \in \{0, 1\}$.
- If $l(\sigma) = \ulcorner \psi \urcorner$ an atomic $\mathcal{L}$-formula, then $f(\sigma, \eta) = 1$ if and only if ψ is true in $\mathcal{M}$ when we use η to assign the free variables.
- If $l(\sigma) = \ulcorner \exists x_i \urcorner$ then $f(\sigma, \eta) = 1$ if and only if for some a in $\mathcal{M}$ we have $f(\sigma\hat{\ }0, \eta^{x_i\mapsto a}) = 1$.
- If $l(\sigma) = \ulcorner \forall x_i \urcorner$ then $f(\sigma, \eta) = 1$ if and only if for all $a \in \mathcal{M}$, $f(\sigma\hat{\ }0, \eta^{x_i\mapsto a}) = 1$.
- If $l(\sigma) = \ulcorner \bigvee\!\!\!\bigvee \urcorner$ then $f(\sigma, \eta) = 1$ if and only if there is an i such that $\sigma\hat{\ }i \in T$ and $f(\sigma\hat{\ }i, \eta) = 1$.
- If $l(\sigma) = \ulcorner \bigwedge\!\!\!\bigwedge \urcorner$ then $f(\sigma, \eta) = 1$ if and only if for every i such that $\sigma\hat{\ }i \in T$, $f(\sigma\hat{\ }i, \eta) = 1$.

If φ is a formula with code (T, l, v), $\mathcal{M} \models \varphi(\bar{a})$ if and only if there is a truth definition f for (T, l, v) in $\mathcal{M}$ with $f(\varnothing, \eta) = 1$, where η is the assignment of the free variables to $\bar{a}$.

We can even keep track of the existential witnesses that make a formula true.

Definition 2.8. A Skolem function for $\mathcal{M}$ and (T, l, v) is a function g which assigns to each $\sigma \in T$ and valuation on $v(\sigma)$ an element of $\mathcal{M}$. We define "f is a truth definition for (T, l, v) with respect to g in $\mathcal{M}$" by replacing the third condition above with:

- If $l(\sigma) = \ulcorner \exists x_i \urcorner$ then $f(\sigma, \eta) = 1$ if and only if for $a = g(\sigma, \eta)$ we have $f(\sigma\hat{\ }0, \eta^{x_i\mapsto a}) = 1$.

If f is a truth definition for (T, l, v) in $\mathcal{M}$, then there is a Skolem function g such that f is a truth definition with respect to g.

2.2 Pseudo-elementary Classes

In this section, we follow the book by Hodges [Hod08]. There are four different types of pseudo-elementary classes: PC, PC$'$, PC$_\Delta$, and PC$'_\Delta$. The Δ means that we are allowed a full theory rather than a single sentence, and the $'$ means that we are allowed to add additional sorts (elements) to the structure. PC and PC$_\Delta$ have already been defined, but we repeat the definition.

Definition 2.9. We say that a class $\mathbb{K}$ of $\mathcal{L}$-structures is a PC-class if there is a language $\tau^* \supseteq \tau$ and an elementary first-order τ^* sentence ϕ such that

$$\mathbb{K} = \{\mathcal{M} \mid \text{there is a } \tau^*\text{-structure } \mathcal{M}^* \text{ expanding } \mathcal{M} \text{ with } \mathcal{M}^* \vDash \phi\}.$$

We say that $\mathbb{K}$ is a PC_Δ-class if ϕ is replaced by an elementary first-order theory.

The classes PC' and PC'_Δ are a little more complicated to define. We need the following definition, which one should think of as throwing away a sort from a structure.

Definition 2.10. Let $\tau \subseteq \tau^*$ be a pair of languages, with a unary predicate $P \in \tau^* \setminus \tau$. Given a τ^*-structure $\mathcal{A}$, we denote by $\mathcal{A}_P$ the substructure of $\mathcal{A} \mid \tau$ whose domain is $P^{\mathcal{A}}$ (if this is a τ-structure; otherwise $\mathcal{A}_P$ is not defined).

The classes PC' and PC'_Δ differ from PC and PC_Δ respectively in that in addition to expanding the language, one is allowed to add additional elements.

Definition 2.11. We say that a class $\mathbb{K}$ of τ-structures is a PC'-class if there is a language $\tau^* \supseteq \tau$, with a unary relation $P \in \tau^* \setminus \tau$, and a τ^*-formula ϕ, such that

$$\mathbb{K} = \{\mathcal{A}_P \mid \mathcal{A} \vDash \phi \text{ and } \mathcal{A}_P \text{ is defined}\}.$$

We say that $\mathbb{K}$ is a PC'_Δ-class if ϕ is a first-order theory.

Note that, if the language is finite (or we are dealing with a PC'_Δ-class) it suffices to ask that

$$\mathbb{K} = \{\mathcal{A}_P \mid \mathcal{A} \vDash \phi\}$$

as ϕ can say that $\mathcal{A}_P$ is defined.

Though we have four different definitions, they give rise to only three different notions (and only two if we consider classes which consist only of infinite structures).

Theorem 2.12 (Theorem 5.2.1 of [Hod08])**.** *Let $\mathbb{K}$ be a class of structures.*

- $\mathbb{K}$ *is a* PC_Δ*-class if and only if it is a* PC'_Δ*-class.*
- *If all the structures in* $\mathbb{K}$ *are infinite, then* $\mathbb{K}$ *is a* PC*-class if and only if it is a* PC'*-class.*

In Example 3.5 we give a class which is PC' but not PC.

The proof of the first point in [Hod08] is not obvious and quite interesting. For the second, essentially the only reason that PC and PC' are different is that the model might be finite; if a model is infinite, one could just have the elements of the model "wear two hats", on the one hand being the domain of the expansion of the original model, and on the other hand playing the role of the elements of the new sort P. In our proofs below, we want to add on a model of arithmetic to a given structure; if the structure was finite, then as every model of arithmetic is infinite, we must add new elements. This is why it is PC' and not PC that appears in, for example, Theorem 1.4.

3 Examples

We give here a few examples of properties that are definable in various combinations of expansions of elementary first-order logic, including some applications of the theorems.

Example 3.1. Let $\tau = \{R\}$ the language of graphs. The class $\mathbb{K}$ of non-connected graphs is a PC-class. Indeed, an undirected graph $\mathcal{G} = (G, R)$ is disconnected if and only if there is a binary relation C of connectedness such that

- $(\forall x)(\forall y)[R(x,y) \to C(x,y)]$,
- $(\forall x)(\forall y)(\forall z)[C(x,y) \wedge C(y,z) \to C(x,z)]$, and
- $\neg(\forall x)(\forall y)C(x,y)$.

An undirected graph $\mathcal{G}$ is also disconnected if and only if

$$(\exists x \neq y) \bigwedge\!\!\!\bigwedge_{n\in\omega} (\forall u_0, \dots, u_n)[x \neq u_0 \vee \neg R(u_0, u_1) \vee \neg R(u_1, u_2) \vee \cdots \vee \neg R(u_{n-1}, u_n) \vee u_n \neq y].$$

So $\mathbb{K}$ is also defined by a $\bigwedge\!\!\!\bigwedge$-sentence.

Example 3.2. Let $\tau = \{<\}$ the language of linear orders. The class $\mathbb{K}$ of non-well-founded linear orders is a PC-class as a linear order $(S, <)$ is non-well-founded if and only if there is a unary relation U such that

$$(\forall x)\big[x \in U \to (\exists y)[y \in U \ \wedge \ y < x]\big].$$

$\mathbb{K}$ is not definable by any $\mathcal{L}_{\omega_1\omega}$ formula.

Example 3.3. Let τ be any language and ϕ a τ-sentence. The class $\mathbb{K}$ of infinite models of ϕ is a PC-class as $\mathcal{A} \models \phi$ is infinite if and only if there is a linear order $<$ on $\mathcal{A}$ such that $(\forall x)(\exists y)[x < y]$. $\mathbb{K}$ is also defined by the $\bigwedge\!\!\!\bigwedge$-sentence

$$\bigwedge\!\!\!\bigwedge_{n\in\omega} (\exists x_0, \dots, x_n) \left[\bigwedge_{i\neq j} x_i \neq x_j \right].$$

Example 3.4. Orderable groups are a PC-class. They are also universally axiomatizable by saying that every finite subset can be ordered in a way that is compatible with the group operation.

Example 3.5. There is a c.e. universal theory T whose models do not form a PC-class; by Theorem 1.8 they are, however, a PC$'$-class. The language of T will be the language of graphs. Fix an enumeration of the sentences ϕ_n in finite languages $\mathcal{L}_n$ expanding the language of graphs. Note that for every finite graph G, we can decide effectively whether there is an expansion of G to a model of ϕ_n. For each n, let C_n be cycle of length n. Then, let T be the theory that says that there is no cycle of length n for exactly those n where C_n does not have an expansion to a model of ϕ_n. Note that T is c.e. and universal, and that it is different from each PC-class.

4 An Application of Craig Interpolation

To prove the direction (1) implies (2) of Theorem 1.1, we will adapt a proof of the Craig Interpolation Theorem for $\mathcal{L}_{\omega_1\omega}$. The proof we adapt is not the original proof by Lopez-Escobar, but one that appears in the book by Marker [Mar16]. We begin with a few preliminaries.

Lemma 4.1. *If φ_1 and φ_2 are $\bigwedge\!\!\!\bigwedge$-formulas, $\varphi_1 \vee \varphi_2$ is equivalent to a $\bigwedge\!\!\!\bigwedge$-formula.*

Proof. We argue by induction on the complexity of φ_1 and φ_2 together. If φ_1 and φ_2 are both finitary quantifier-free, then so is $\varphi_1 \vee \varphi_2$. For the inductive steps, we will give the argument which reduces the complexity of φ_1; the arguments for φ_2 are similar.

If φ_1 is of the form $(Qx)\varphi_1'(x)$, where Q is either $\exists$ or $\forall$, then $\varphi_1 \vee \varphi_2$ is equivalent to $(Qv)[\varphi_1'(v) \vee \varphi_2]$ where v is not free in φ_2; by the inductive hypothesis, $\varphi_1'(v) \vee \varphi_2$ is equivalent to a $\bigwedge\!\!\!\bigwedge$-formula and so $(Qv)[\varphi_1'(v) \vee \varphi_2]$ is as well.

Finally, if φ_1 is of the form $\bigwedge\!\!\!\bigwedge_{\phi \in X} \phi$, where each ϕ is a $\bigwedge\!\!\!\bigwedge$-formula, then $\varphi_1 \vee \varphi_2$ is equivalent to $\bigwedge\!\!\!\bigwedge_{\phi \in X}[\phi \vee \varphi_2]$, and by the induction hypothesis, each $\phi \vee \varphi_2$ is a $\bigwedge\!\!\!\bigwedge$-formula. $\square$

The proof of Craig Interpolation makes use of consistency properties. Consistency properties are the infinitary equivalent of Henkin-style constructions in finitary logic. The following definition, due to Makkai, is what we need to do to perform such a construction.

Definition 4.2 (Definition 4.1 of [Mar16])**.** Let C be a countable collection of new constants. A consistency property Σ is a collection of countable sets σ of $\mathcal{L}_{\omega_1\omega}$-sentences with the following properties. For $\sigma \in \Sigma$:

1. if $\mu \subseteq \sigma$, then $\mu \in \Sigma$;
2. if $\phi \in \sigma$, then $\neg\phi \notin \sigma$;
3. if $\neg\phi \in \sigma$, then $\sigma \cup \{\sim \phi\} \in \Sigma$;
4. if $\bigwedge\!\!\!\bigwedge_{\phi \in X} \phi \in \sigma$, then for all $\phi \in X$, $\sigma \cup \{\phi\} \in \Sigma$;
5. if $\bigvee\!\!\!\bigvee_{\phi \in X} \phi \in \sigma$, then there is $\phi \in X$ such that $\sigma \cup \{\phi\} \in \Sigma$;
6. if $(\forall v)\phi(v) \in \sigma$, then for all $c \in C$, $\sigma \cup \{\phi(c)\} \in \Sigma$;
7. if $(\exists v)\phi(v) \in \sigma$, then there is $c \in C$ such that $\sigma \cup \{\phi(c)\} \in \Sigma$;
8. let t be a term with no variables and let $c, d \in C$,
   (a) if $c = d \in \sigma$, then $\sigma \cup \{d = c\} \in \Sigma$;
   (b) if $c = t \in \sigma$ and $\phi(t) \in \sigma$, then $\sigma \cup \{\phi(c)\} \in \Sigma$;
   (c) there is $e \in C$ such that $\sigma \cup \{e = t\} \in \Sigma$.

A consistency property is in some sense a recipe for building a model.

Theorem 4.3 (Model Existence Theorem; see Theorem 4.1.6 of [Mar16])**.** *If Σ is a consistency property and $\sigma \in \Sigma$, there is $\mathcal{M} \models \sigma$.*

We are now ready to prove our variant of the Craig Interpolation Theorem. We strengthen the hypotheses to assume that one of the sentences is a $\bigwedge\!\!\!\bigwedge$-sentence, and in return, we get that the interpolant is also a $\bigwedge\!\!\!\bigwedge$-sentence. The proof follows the same structure as that of the Craig Interpolation Theorem in [Mar16] (Theorem 4.3.1).

Theorem 4.4 (Variation on Craig Interpolation Theorem)**.** *Suppose ϕ_1 is a $\bigwedge\!\!\!\bigwedge$-sentence and ϕ_2 is an $\mathcal{L}_{\omega_1\omega}$-sentence with $\phi_1 \vDash \phi_2$. There is a $\bigwedge\!\!\!\bigwedge$-sentence θ such that $\phi_1 \vDash \theta$, $\theta \vDash \phi_2$, and every relation, function and constant symbol occurring in θ occurs in both ϕ_1 and ϕ_2.*

Proof. Let C be a countable collection of new constants. Let τ_i be the smallest language containing ϕ_i and C, and let $\tau = \tau_1 \cap \tau_2$.

Let Σ be the collection of finite σ containing only finitely many new constants that can be written as $\sigma = \sigma_1 \cup \sigma_2$, where σ_1 is a finite set of $\bigwedge\!\!\!\bigwedge$-τ_1-sentences and σ_2 is a finite set of τ_2-sentences, and such that for all τ-sentences ψ_1 and ψ_2, with ψ_1 a $\bigwedge\!\!\!\bigwedge$-sentence, if $\sigma_1 \vDash \psi_1$ and $\sigma_2 \vDash \psi_2$ then $\psi_1 \wedge \psi_2$ is satisfiable.

In the rest of the proof, we make the convention that if $\sigma \in \Sigma$ and we write $\sigma = \sigma_1 \cup \sigma_2$, then σ_1 and σ_2 are the witnesses that $\sigma \in \Sigma$, i.e., σ_1 consists of $\bigwedge\!\!\!\bigwedge$-τ_2-sentences, σ_2 consists of τ_2-sentences, and they satisfy the satisfiability condition above.

We claim that Σ is a consistency property. The following claim will verify many of the conditions.

Claim. *Fix $\sigma \in \Sigma$ and write $\sigma = \sigma_1 \cup \sigma_2$. If ϕ is a τ_i-sentence (and a $\bigwedge\!\!\!\bigwedge$-sentence if $i = 1$) with $\sigma_i \vDash \phi$, then $\sigma \cup \{\phi\} \in \Sigma$.*

Proof. We will show the case $i = 1$. We can write $\sigma \cup \{\phi\} = (\sigma_1 \cup \{\phi\}) \cup \sigma_2$. If $\sigma_1 \cup \{\phi\} \vDash \psi_1$ and $\sigma_2 \vDash \psi_2$, with ψ_1 a $\bigwedge\!\!\!\bigwedge$-sentence, then since $\sigma_1 \vDash \phi$, $\sigma_1 \vDash \psi_1$. Hence $\psi_1 \wedge \psi_2$ is satisfiable. □

We now check the conditions of a consistency property.

1. If $\mu \subseteq \sigma$ with $\sigma \in \Sigma$, write $\mu = \mu_1 \cup \mu_2$ and $\sigma = \sigma_1 \cup \sigma_2$ where $\mu_1 \subseteq \sigma_1$ and $\mu_2 \subseteq \sigma_2$. Given $\mu_1 \vDash \psi_1$ and $\mu_2 \vDash \psi_2$, we have $\sigma_1 \vDash \psi_1$ and $\sigma_2 \vDash \psi_2$; hence $\psi_1 \wedge \psi_2$ is satisfiable. So $\mu \in \Sigma$.

2. If $\phi, \neg\phi \in \sigma = \sigma_1 \cup \sigma_2$, say $\phi, \neg\phi \in \sigma_i$, $\sigma_i \vDash \phi \wedge \neg\phi$ which is not satisfiable. The other possible case is that $\phi \in \sigma_i$, $\neg\phi \in \sigma_j$, $i \neq j$, in which case $\sigma_i \vDash \phi$ and $\sigma_j \vDash \neg\phi$, and $\phi \wedge \neg\phi$ is not satisfiable.

3. This follows from the claim.

4. This follows from the claim.

5. Write $\sigma = \sigma_1 \cup \sigma_2$. We have two cases which are different, depending on whether $\bigvee\!\!\!\bigvee_{\phi \in X} \phi \in \sigma_1$ or $\bigvee\!\!\!\bigvee_{\phi \in X} \phi \in \sigma_2$.

   First suppose that $\bigvee\!\!\!\bigvee_{\phi \in X} \phi \in \sigma_2$. Let $\sigma_{2,\phi} = \sigma_2 \cup \{\phi\}$. We claim that for some $\phi \in X$, $\sigma_{2,\phi} \cup \sigma_1 \in \Sigma$. If not, then for each $\phi \in X$ there are τ-sentences $\psi_{2,\phi}$ and $\psi_{1,\phi}$, with $\psi_{1,\phi}$ a $\bigwedge\!\!\!\bigwedge$-sentence, such that $\sigma_{2,\phi} \vDash \psi_{2,\phi}$ and $\sigma_1 \vDash \psi_{1,\phi}$, and such that $\psi_{2,\phi} \wedge \psi_{1,\phi}$ is unsatisfiable. So $\psi_{2,\phi} \vDash \neg\psi_{1,\phi}$. Since

$$\sigma_2 \vDash \bigvee\!\!\!\bigvee_{\phi \in X} \phi$$

we have that

$$\sigma_2 \vDash \bigvee_{\phi \in X} \psi_{2,\phi}.$$

On the other hand,

$$\sigma_1 \vDash \bigwedge_{\phi \in X} \psi_{1,\phi}.$$

This formula is a $\bigwedge$-sentence as each $\psi_{1,\phi}$ is. Finally,

$$\bigvee_{\phi \in X} \psi_{2,\phi} \vDash \neg \bigwedge_{\phi \in X} \psi_{1,\phi}$$

which contradicts that $\sigma \in \Sigma$.

Now suppose that $\bigvee_{\phi \in X} \phi \in \sigma_1$; then X is finite. We begin in a similar way as before. Let $\sigma_{1,\phi} = \sigma_1 \cup \{\phi\}$. We claim that for some $\phi \in X$, $\sigma_{1,\phi} \cup \sigma_2 \in \Sigma$. If not, there for each $\phi \in X$ there are τ-sentences $\psi_{1,\phi}$ and $\psi_{2,\phi}$, with $\psi_{1,\phi}$ a $\bigwedge$-sentence, such that $\sigma_{1,\phi} \vDash \psi_{1,\phi}$ and $\sigma_2 \vDash \psi_{2,\phi}$, and such that $\psi_{1,\phi} \wedge \psi_{2,\phi}$ is unsatisfiable. So $\psi_{1,\phi} \vDash \neg \psi_{2,\phi}$. Since

$$\sigma_1 \vDash \bigvee_{\phi \in X} \phi$$

we have that

$$\sigma_1 \vDash \bigvee_{\phi \in X} \psi_{1,\phi}.$$

As X is finite, by Lemma 4.1 this is equivalent to a $\bigwedge$-sentence. On the other hand,

$$\sigma_2 \vDash \bigwedge_{\phi \in X} \psi_{2,\phi}$$

and

$$\bigvee_{\phi \in X} \psi_{1,\phi} \vDash \neg \bigwedge_{\phi \in X} \psi_{2,\phi}$$

which contradicts that $\sigma \in \Sigma$.

6. This follows from the claim as $(\forall x)\phi(x) \vDash \phi(c)$ for all $c \in C$.

7. If $(\exists x)\phi(x) \in \sigma$, then choose $c \in C$ which does not appear in σ. Suppose that $(\exists x)\phi(x) \in \sigma_1$; the case where $(\exists x)\phi(x) \in \sigma_2$ is similar. We claim that $\sigma \cup \{\phi(c)\} \in \Sigma$. Suppose that $\sigma_1 \cup \{\phi(c)\} \vDash \psi_1$ and $\sigma_2 \vDash \psi_2$, where ψ_1 is a $\bigwedge$-sentence. Write $\psi_1 = \theta_1(c)$ and $\psi_2 = \theta_2(c)$. We have $\sigma_1 \vDash \phi(c) \to \theta_1(c)$, and so since c does not appear in σ_1, $\sigma_1 \vDash (\forall x)[\phi(x) \to \theta_1(x)]$. Similarly, $\sigma_2 \vDash (\forall x)\theta_2(x)$. Also, $\sigma_1 \vDash (\exists x)\phi(x)$ and so $\sigma_1 \vDash (\exists x)\theta_1(x)$. Since $(\exists x)\phi(x) \in \sigma_1$, $\phi(x)$ is a $\bigwedge$-formula. So $(\exists x)\theta_1(x) \wedge (\forall x)\theta_2(x)$ is satisfiable, say in a model $\mathcal{M}$. Note that the constant c does not appear in the formula $(\exists x)\theta_1(x) \wedge (\forall x)\theta_2(x)$, so we may choose the interpretation of c in $\mathcal{M}$ such that $\mathcal{M} \vDash \theta_1(c)$. Then $\mathcal{M} \vDash \theta_1(c) \wedge \theta_2(c)$.

8. let t be a term with no variables and let $c, d \in C$,

   (a) This follows from the claim.

(b) Suppose $c = t \in \sigma$ and $\phi(t) \in \sigma$. Write $\sigma = \sigma_1 \cup \sigma_2$. Consider $\mu = \sigma \cup \{\phi(c)\} = \sigma_1 \cup \sigma_2 \cup \{\phi(c)\}$. Suppose $c = t \in \sigma_i$ and $\phi(t) \in \sigma_j$. The case $i = j$ follows from the claim, so we consider the case $i \neq j$. Suppose that $\sigma_i \models \psi_i$ and $\sigma_j \cup \{\phi(c)\} \models \psi_j$. Then $\sigma_i \models c = t \wedge \psi_i$ and $\sigma_j \models c = t \to \psi_j$, so $c = t \wedge \psi_i \wedge (c = t \to \psi_i)$ is satisfiable. So $\psi_i \wedge \psi_j$ is satisfiable.

(c) Pick $e \in C$ which does not appear in $\sigma = \sigma_1 \cup \sigma_2$. Then if $\sigma_1 \cup \{e = t\} \models \psi_1$ and $\sigma_2 \cup \{e = t\} \models \psi_2$, write $\psi_1 = \theta_1(e)$ and $\psi_2 = \theta_2(e)$. Then since e does not appear in σ_1 or σ_2, $\sigma_1 \models \theta_1(t)$ and $\sigma_2 \models \theta_2(t)$. Thus $\theta_1(t) \wedge \theta_2(t)$ is satisfiable. Given a model of $\theta_1(t) \wedge \theta_2(t)$, setting the interpretation of c to t, we get a model of $\psi_1 \wedge \psi_2$. So $\psi_1 \wedge \psi_2$ is satisfiable.

Since $\phi_1 \models \phi_2$, $\{\phi_1, \neg\phi_2\} \notin \Sigma$ as otherwise by the Model Existence Theorem there would be a model of $\phi_1 \wedge \neg\phi_2$. By definition of Σ, there are τ-sentences ψ_1 and ψ_2, with ψ_1 a $\bigwedge\!\!\!\bigwedge$-sentence, such that $\phi_1 \models \psi_1$, $\neg\phi_2 \models \psi_2$, and $\psi_1 \wedge \psi_2$ is not satisfiable. So we have that $\phi_1 \models \psi_1$, $\psi_1 \models \neg\psi_2$, and $\neg\psi_2 \models \phi_2$. Hence $\phi_1 \models \psi_1$ and $\psi_1 \models \phi_2$.

Thus ψ_1 is the desired interpolant, except that it may contain constants from C. Write $\psi_1 = \theta(\bar{c})$, where θ is an τ-formula with no constants from $\bar{c}$. Neither ϕ_1 nor ϕ_2 contains constants from C, and so $\phi_1 \models (\forall \bar{x})\theta(\bar{x})$ and $(\exists \bar{x})\theta(\bar{x}) \models \phi_2$. Since $(\forall \bar{x})\theta(\bar{x}) \models (\exists \bar{x})\theta(\bar{x})$, we can take $(\forall \bar{x})\theta(\bar{x})$ as the interpolant. □

We get the following corollary, which is (1) implies (2) of Theorem 1.1. Interestingly, when we apply the interpolation theorem in the proof, one of the languages contains the other (i.e., we have $\tau_1 \supseteq \tau_2$ so that $\tau = \tau_1 \cap \tau_2 = \tau_2$). If it were not for our added assumptions on the form of the formulas involved, finding an interpolant would be trivial as we could just take the sentence in the smaller language.

Corollary 4.5. *Let $\mathbb{K}$ be a class of τ-structures closed under isomorphism. If $\mathbb{K}$ is both a PC_Δ-class and $\mathcal{L}_{\omega_1\omega}$-elementary, then it is defined by a $\bigwedge\!\!\!\bigwedge$-sentence.*

Proof. Let $\tau^* \supseteq \tau$ be an expanded language and let X be a set of first-order sentences such that $\mathbb{K}$ is the class of reducts to τ of models of $\psi_1 = \bigwedge\!\!\!\bigwedge_{\phi \in X} \phi$. Note that ψ_1 is a $\bigwedge\!\!\!\bigwedge$-sentence.

Let ψ_2 be an $\mathcal{L}_{\omega_1\omega}(\tau)$-sentence defining $\mathbb{K}$. We have that $\psi_1 \models \psi_2$, so by the Interpolation Theorem, there is a $\bigwedge\!\!\!\bigwedge$-τ-sentence θ such that $\psi_1 \models \theta$ and $\theta \models \psi_2$.

Every $\mathcal{M} \in \mathbb{K}$ has an expansion which is a model of ψ_1 and hence is itself a model of θ; and every model of θ is a model of ψ_2, and hence in the class $\mathbb{K}$. So θ defines $\mathbb{K}$. □

5 Coding Sentences in Models of Arithmetic

This section is devoted to proving the direction (2) implies (1) of Theorem 1.1. First, we will recall some facts about weak theories of arithmetic.

5.1 Weak Theories of Arithmetic

We will work with PA^-, Peano arithmetic without the axiom of induction, which is the axiomatization of the non-negative parts of discretely ordered rings. A good reference for many of the facts that we use is the book by Kaye [Kay91]. We use PA^- because it is finitely axiomatizable, and yet strong enough for our coding.

Recall that every model of PA^- is an end extension of $\mathbb{N}$. Moreover, if $\mathcal{N}$ is an end extension of $\mathcal{M}$, and φ is a Π_1 formula in the language of arithmetic, then if $\mathcal{N} \models \varphi$ then $\mathcal{M} \models \varphi$. So if a Π_1 sentence holds in any model of PA^-, then it holds in $\mathbb{N}$.

We will also use the fact that computable subsets of $\mathbb{N}$ are definable by Σ_1 and Π_1 formulas.

Theorem 5.1 (See Corollary 3.5 of [Kay91])**.** *Fix $C \subseteq \mathbb{N}$. $A \subseteq \mathbb{N}^k$ is C-computable if and only if there is a Σ_1 formula $\varphi(x_1, \dots, x_k)$ and a Π_1 formula $\psi(x_1, \dots, x_k)$ in the language of arithmetic with an additional unary predicate such that, for all $\bar{a} \in \mathbb{N}$, $\bar{a} \in A$ if and only if $(\mathbb{N}, C) \models \varphi(\bar{a})$.*

We will use this fact implicitly throughout to write computable predicates inside of a formula of arithmetic, meaning that one should replace the computable predicate by the corresponding Σ_1 or Π_1 formula.

5.2 Adding Tuples to a Theory

Fix a language τ. We will describe a theory T in an expanded language which is essentially the theory of a τ-structure augmented with finite tuples from that structure. Because we will be working within elementary first-order logic, we cannot guarantee that the tuples are actually finite, but rather that their lengths take values within an associated model of arithmetic.

Definition 5.2. Given a language τ, let τ_{tup} be the three-sorted language with sorts M, a τ-structure; N, a structure in the language of arithmetic; and M_{tup}, a structure representing tuples from M. The sort M_{tup} is equipped with a length function $|\cdot|: M_{\text{tup}} \to N$ and an indexing relation $\text{ind} \subseteq M_{\text{tup}} \times N \times M$; we interpret $\text{ind}(\pi, n, m)$ as saying that the nth entry of π is m.

Definition 5.3. Let T_{tup} be the τ_{tup}-theory which consists of the following formulas:

1. $N \models PA^-$,
2. M_{tup} consists of N-indexed sequences from M:
   (a) for all $\pi \in M_{\text{tup}}$, $i \in N$, and $m \in M$, if $\text{ind}(\pi, i, m)$ then $i < |\pi|$.
   (b) for all $\pi \in M_{\text{tup}}$, $i \in N$, and $m, m' \in M$, if $\text{ind}(\pi, i, m)$ and $\text{ind}(\pi, i, m')$ then $m = m'$.
   (c) for all $\pi \in M_{\text{tup}}$ and $i \in N$, if $i < |\pi|$ then there is $m \in M$ such that $\text{ind}(\pi, i, m)$.
   (d) for all $\pi, \rho \in M_{\text{tup}}$, if $|\pi| = |\rho|$ and for all $i < |\pi| = |\rho|$ there is m such that $\text{ind}(\pi, i, m)$ and $\text{ind}(\rho, i, m)$ then $\pi = \rho$.
   (e) there is $\pi \in M_{\text{tup}}$ with $|\pi| = 0$.
   (f) for all $\pi \in M_{\text{tup}}$ and $m \in M$, there is $\rho \in M_{\text{tup}}$ with $|\rho| = |\pi| + 1$, $\text{ind}(\rho, |\pi|, m)$, and for all $i < |\pi|$ and m, $\text{ind}(\pi, i, m)$ if and only if $\text{ind}(\rho, i, m)$.

(g) for all $\pi \in M_{\text{tup}}$ with $|\pi| \geq 1$, there is $\rho \in M_{\text{tup}}$ with $|\rho| = |\pi| - 1$, and for all $i < |\rho|$ and m, $\text{ind}(\pi, i, m)$ if and only if $\text{ind}(\rho, i, m)$.

In a model $\mathcal{A}$ of T_{tup}, we write A, $\mathcal{A}_N$, and $\mathcal{A}_{M_{\text{tup}}}$ for the underlying sets of the three sorts. The theory T_{tup} implies that given $\pi \in M_{\text{tup}}$ and $i < |\pi|$, there is a unique m such that $\text{ind}(\pi, i, m)$; we write $\pi(i) = m$. This is the ith entry of the tuple π. The axioms also say that natural operations such as the concatenation of π by a single element m, $\pi\hat{\ }m$, exists, or that π without its last entry, π^-, exists. We can also view elements of the sort N as tuples using the standard coding of tuples of natural numbers.

Definition 5.4. Given a τ-structure $\mathcal{M} = (M, \ldots)$, we consider $\mathcal{M}^{<\mathbb{N}} = (\mathcal{M}, \mathbb{N}, M^{<\mathbb{N}})$ a τ_{tup}-structure with the natural interpretation of the language.

There is a natural generalization of end extensions of models of arithmetic to models of T_{tup} with a fixed first coordinate $\mathcal{M}$. The structure $\mathcal{M}^{<\mathbb{N}}$ plays the same role as the standard natural numbers here, i.e., every model is an end extension of $\mathcal{M}^{<\mathbb{N}}$. We will just prove the facts that we need rather than developing a full theory of end extensions. We will also want to expand the language a little more by adding a unary relation R on the arithmetic sort N, and a binary function $g\colon N \times M_{\text{tup}} \to M$. It is important that the range of the function g is M, the only sort which is fixed under end extensions. Let $\tau^*_{\text{tup}} = \tau_{\text{tup}} \cup \{R, g\}$.

The following lemma proves that $\mathcal{M}^{<\mathbb{N}}$ is the smallest model with M-sort $\mathcal{M}$. One could adapt the conclusion of the theorem to make a definition of what it means to be an end-extension in this context.

Lemma 5.5. *Let $\mathcal{A} = (A, R, g, \ldots)$ be a τ^*_{tup}-structure which is a model of T_{tup} with $\mathcal{A}_M = \mathcal{M}$. There is a unique embedding of $(\mathcal{M}^{<\mathbb{N}}, R\restriction_{\mathbb{N}}, g\restriction_{\mathbb{N}\times M^{<\mathbb{N}}})$ into $\mathcal{A}$ which is the identity on the sort M. Moreover, viewing $\mathcal{M}^{<\mathbb{N}}$ as a substructure of $\mathcal{A}$:*

- *$\mathcal{A}_N$ is an end extension of $\mathbb{N}$, and*
- *each $\pi \in \mathcal{A}_{M_{\text{tup}}}$ with $|\pi| \in \mathbb{N}$ has $\pi \in M^{<\mathbb{N}}$.*

Proof. Since $\mathcal{A}_N$ is a model of PA^-, it is an end extension of $\mathbb{N}$, and so there is a unique embedding of $\mathbb{N}$ into $\mathcal{A}_N$.

We define, by induction on $|\pi|$, an embedding $f\colon M^{<\mathbb{N}} \to \mathcal{A}_{M_{\text{tup}}}$. By 2(d) there is a unique $\varepsilon \in \mathcal{A}_{M_{\text{tup}}}$ with $|\varepsilon| = 0$. Define $f(\langle\rangle) = \varepsilon$. Given $\pi \in \mathcal{M}^{<\mathbb{N}}$ of length $n+1$, define $f(\pi) = f(\pi^-)\hat{\ }\pi(n)$.

Now viewing $\mathcal{M}^{<\mathbb{N}}$ as a substructure of $\mathcal{A}$, to see that if $\pi \in \mathcal{A}_{M_{\text{tup}}}$ has $|\pi| \in \mathbb{N}$, then $\pi \in M^{<\mathbb{N}}$, we argue by induction on $|\pi|$. For $|\pi| = 0$ this is clear. Given π with $|\pi| = n+1$, we have that $\pi^- \in M^{<\mathbb{N}}$. Thus $\pi = \pi^-\hat{\ }\pi(n) \in M^{<\mathbb{N}}$. □

As in models of arithmetic, the Π_1 formulas play an important role relative to end extensions. Since in this context the first sort is fixed, we allow any kind of quantifiers over the first sort, but only universal quantifiers (or bounded quantifiers) over the last two sorts. By a bounded quantifier over tuples, we mean that the lengths of the tuples are bounded.

Definition 5.6. We define the Π_1 τ^*_{tup}-formulas inductively:

- any finitary quantifier-free formula;

- $(\exists x \in M)\varphi$ or $(\forall x \in M)\varphi$ where φ is Π_1;
- $(\forall x \in N)\varphi$ or $(\forall \pi \in M_{\text{tup}})\varphi$ where φ is Π_1;
- $(\exists x \in N, x < t)\varphi$ or $(\exists \pi \in M_{\text{tup}}, |\pi| < t)\varphi$ where φ is Π_1.

These are all elementary first-order formulas.

As in arithmetic, if an end extension satisfies a Π_1 formula with parameters in the smaller model, then the smaller model satisfies the same formula.

Lemma 5.7. *Let $\mathcal{A} = (A, R, g, \dots)$ be a τ^*_{tup}-structure which is a model of T_{tup} with $\mathcal{A}_M = \mathcal{M}$. Let φ be a Π_1 formula of τ^*_{tup}. Given $\bar{a} \in \mathcal{M}^{<\mathbb{N}}$, if $\mathcal{A} \vDash \varphi(\bar{a})$, then $(\mathcal{M}^{<\mathbb{N}}, R\upharpoonright_{\mathbb{N}}, g\upharpoonright_{\mathbb{N}\times\mathcal{M}^{<\mathbb{N}}}) \vDash \varphi(\bar{a})$.*

Proof. By Lemma 5.5, we can view $(\mathcal{M}^{<\mathbb{N}}, R\upharpoonright_{\mathbb{N}}, g\upharpoonright_{\mathbb{N}\times M^{<\mathbb{N}}})$ as a submodel of $\mathcal{A}$ satisfying the end extension properties.

We argue by induction on φ. Suppose that $\mathcal{A} \vDash \varphi(\bar{a})$. If φ is finitary quantifier-free, then $\mathcal{A} \vDash \varphi(\bar{a})$ if and only if $(\mathcal{M}^{<\mathbb{N}}, R\upharpoonright_{\mathbb{N}}, g\upharpoonright_{\mathbb{N}\times M^{<\mathbb{N}}}) \vDash \varphi(\bar{a})$.

If φ is of the form $(\exists y \in M)\psi(\bar{x}, y)$, then there is $b \in \mathcal{M}$ such that $\mathcal{A} \vDash \psi(\bar{a}, b)$. Then, as ψ is Π_1, $(\mathcal{M}^{<\mathbb{N}}, R\upharpoonright_{\mathbb{N}}, g\upharpoonright_{\mathbb{N}\times M^{<\mathbb{N}}}) \vDash \psi(\bar{a}, b)$ and hence $(\mathcal{M}^{<\mathbb{N}}, R\upharpoonright_{\mathbb{N}}, g\upharpoonright_{\mathbb{N}\times M^{<\mathbb{N}}}) \vDash \varphi(\bar{a})$. A similar argument works when φ is of the form $(\forall y \in M)\psi(\bar{x}, y)$.

If φ is of the form $(\forall y \in N)\psi(\bar{x}, y)$, then given $b \in \mathbb{N}$, we have $\mathcal{A} \vDash \psi(\bar{a}, b)$. Thus $(\mathcal{M}^{<\mathbb{N}}, R\upharpoonright_{\mathbb{N}}, g\upharpoonright_{\mathbb{N}\times M^{<\mathbb{N}}}) \vDash \psi(\bar{a}, b)$. So $(\mathcal{M}^{<\mathbb{N}}, R\upharpoonright_{\mathbb{N}}, g\upharpoonright_{\mathbb{N}\times M^{<\mathbb{N}}}) \vDash \varphi(\bar{a})$. The same argument works when φ is of the form $(\forall \pi \in M_{\text{tup}})\psi(\bar{x}, \pi)$.

If φ is of the form $(\exists y \in N, y < t)\psi(\bar{x}, y)$, then $m = t(\bar{a}) \in \mathbb{N}$. Since $\mathcal{A} \vDash (\exists y \in N, y < m)\psi(\bar{a}, y)$, there is $b \in \mathbb{N}$ such that $\mathcal{A} \vDash \psi(\bar{a}, b)$. Since ψ is Π_1, $(\mathcal{M}^{<\mathbb{N}}, R\upharpoonright_{\mathbb{N}}, g\upharpoonright_{\mathbb{N}\times M^{<\mathbb{N}}}) \vDash \psi(\bar{a}, b)$. Hence $(\mathcal{M}^{<\mathbb{N}}, R\upharpoonright_{\mathbb{N}}, g\upharpoonright_{\mathbb{N}\times M^{<\mathbb{N}}}) \vDash \varphi(\bar{a})$. The same argument works, using the fact that each $\pi \in \mathcal{A}_{M_{\text{tup}}}$ with $|\pi| \in \mathbb{N}$ is in $M^{<\mathbb{N}}$, when φ is of the form $(\exists \pi \in M_{\text{tup}}, |\pi| < t)\psi(\bar{x}, \pi)$. ☐

5.3 Encoding Computable Infinitary Sentences

We are now ready for the main construction of this section. The idea is as follows. Given a computable $\bigwedge\!\!\!\bigwedge$-sentence φ in a language τ with no existential quantifiers, we encode that sentence into a finitary sentence in the language τ_{tup}. Essentially, we will write down (making use of the model of arithmetic and tuples in a τ_{tup}-structure) a finitary τ_{tup}-sentence ψ which expresses the truth conditions for φ. The sentence will be Π_1. Thus, given a τ-structure $\mathcal{M}$, we will have $\mathcal{M} \vDash \varphi$ if and only if $\mathcal{M}^{<\mathbb{N}} \vDash \psi$ (by checking that ψ does in fact express the truth conditions for φ in this model) if and only if there is some $\mathcal{A} \vDash \psi \wedge T_{\text{tup}}$ with $\mathcal{A}_M = \mathcal{M}$ (because ψ is Π_1). We can then show that φ defines a pseudo-elementary class. In fact, by relativizing we can apply this argument to any $\bigwedge\!\!\!\bigwedge$-sentence because every such sentence is C-computable for some C; this is what we use the relation R in τ^*_{tup} for. For existential quantifiers, we need to make use of Skolem functions for truth definitions of φ; this is what we use the function g in τ^*_{tup} for. There are also some extra complications when the language τ is infinite.

We begin with the following lemma which does most of the heavy lifting.

Lemma 5.8. *If φ is a C-computable $\bigwedge\!\!\!\bigwedge$-sentence in a language τ, then there is a sequence of Π_1 τ^*_{tup}-formulas χ, $\{\theta_s \mid s \in \mathbb{N}\}$ such that for all τ-structures $\mathcal{M}$,*

$$\mathcal{M} \models \varphi \iff (\exists g: \mathbb{N} \times M^{<\mathbb{N}} \to \mathcal{M})(\exists R \subseteq \mathbb{N}) \\ \Big[(\mathcal{M}^{<\mathbb{N}}, R, g) \models \{\chi\} \cup \{\theta_s \mid s \in \mathbb{N}\} \cup \{\underline{n} \in R \mid n \in C\} \cup \{\underline{n} \notin R \mid n \notin C\}\Big].$$

Before proving this lemma, we show how to use it together with results from the previous section to get Theorem 1.1.

Theorem 1.1. *Let $\mathbb{K}$ be a class of structures closed under isomorphism. The following are equivalent:*

- $\mathbb{K}$ *is both a pseudo-elementary* (PC_Δ) *class and $\mathcal{L}_{\omega_1\omega}$-elementary.*
- $\mathbb{K}$ *is defined by a $\bigwedge\!\!\!\bigwedge$-sentence.*

Proof. It is immediate that if $\mathbb{K}$ is defined by a $\bigwedge\!\!\!\bigwedge$-sentence φ, then it is $\mathcal{L}_{\omega_1\omega}$-elementary. We show that $\mathbb{K}$ is a PC'_Δ-class, and by Theorem 2.12, it is then a PC_Δ-class. Let C be such that φ is a C-computable $\bigwedge\!\!\!\bigwedge$-sentence. The expanded language is $\tau^*_{\text{tup}} = \tau_{\text{tup}} \cup \{R, g\}$, where R is a unary relation symbol on N and g is a binary function symbol $N \times M_{\text{tup}} \to M$. Then, given a τ-structure $\mathcal{M}$, by Lemma 5.8, if $\mathcal{M} \models \varphi$ then there are R and g such that

$$(\mathcal{M}^{<\mathbb{N}}, R, g) \models \{\chi\} \cup \{\theta_s \mid s \in \mathbb{N}\} \cup \{\underline{n} \in R \mid n \in C\} \cup \{\underline{n} \notin R \mid n \notin C\}.$$

In particular, there is an expansion of $\mathcal{M}$ to a model of these sentences together with T_{tup}. On the other hand, suppose that there is $(\mathcal{A}, R, g)$ a model of T_{tup} together with these sentences. Let $\mathcal{M} = \mathcal{A}_M$. Then, by Lemmas 5.5 and 5.7, and using the fact that χ and the θ_s are Π_1,

$$(\mathcal{M}^{<\mathbb{N}}, R{\restriction}_{\mathbb{N}}, g{\restriction}_{\mathbb{N}\times M^{<\mathbb{N}}}) \models \{\chi\} \cup \{\theta_s \mid s \in \mathbb{N}\} \cup \{\underline{n} \in R \mid n \in C\} \cup \{\underline{n} \notin R \mid n \notin C\}.$$

Thus $\mathcal{M} \models \varphi$. So we have shown that $\mathbb{K}$ is PC'_Δ, as defined by these sentences and T_{tup}.

On the other hand, if $\mathbb{K}$ is both a PC_Δ-class and $\mathcal{L}_{\omega_1\omega}$-elementary, then by Corollary 4.5, it is defined by a $\bigwedge\!\!\!\bigwedge$-sentence. □

We now prove the lemma.

Proof of Lemma 5.8. Fix a C-computable sentence code (T, l, v) for φ such that for no $\sigma \in T$ do we have $l(\sigma) = \ulcorner \bigvee\!\!\!\bigvee \urcorner$. We may assume that τ is relational, say consisting of $\{R_i \mid i \in \mathbb{N}\}$.

Let $\{\psi_n\}_{n\in\mathbb{N}}$ be an effective listing of all formulas that are finite disjuncts of atomic and negated atomic formulas over τ. That is, we can write,

$$\psi_n \iff \bigvee_{m=1}^{k(n)} P_{\ell(n,m)}(x_{i(n,m,1)}, \dots, x_{i(n,m,r(\ell(n,m)))})$$

where P_j range among the relations R_i and their negations, and the functions k, ℓ, i, and r are all computable.

The function g is supposed to represent a Skolem function for φ. A Skolem function, as defined in Definition 2.8, is a function which assigns to each $\sigma \in T$ and partial valuation v with domain $v(\sigma)$ an element of $\mathcal{M}$ (which is supposed to be the new value of x_i if $l(\sigma) = \ulcorner \exists x_i \urcorner$). It will be convenient to view g as a function of σ and any partial valuation, with the property that g maps any two valuations which agree on $v(\sigma)$ to the same element of $\mathcal{M}$; the two points of view are interchangeable. We can express that g is such a function with the following Π_1 sentence χ:

$$(\forall \sigma \in T)(\forall u, v \in M_{\text{tup}})\big[[|u|, |v| \geq \max(v(\sigma)) \wedge (\forall j \in v(\sigma)) u(j) = v(j)] \longrightarrow g(\sigma, u) = g(\sigma, v)\big].$$

To see that this sentence is Π_1, we use Theorem 5.1: T and $v(\sigma)$ are C-computable, and $v(\sigma)$ is finite with a computable upper bound.

Define θ_s to be the sentence

$$(\forall \sigma \in T)(\forall t \geq s \in N)(\forall \rho \in M_{\text{tup}})[\xi \to \zeta]$$

where ξ and ζ are defined below. This sentence will essentially say that for all sequences σ of natural numbers, and t, and sequences of tuples ρ, if (this is ξ) σ is a path through T ending in a leaf and ρ gives a partial valuation of the first t variables for each node on the path σ of the appropriate kind for a truth definition, then (this is ζ) the partial valuation at the leaf on σ makes the corresponding atomic formula true (with respect to the finite sublanguage consisting of the first s relations of τ). One should view ρ as a length $|\sigma|+1$ tuple of t-tuples, each t-tuple being a partial valuation. The formula ξ is

$$|\rho| = t|\sigma| + t \quad \wedge \quad (\forall i)\sigma\hat{\ }i \notin T \quad \wedge \quad (\forall i < |\sigma|) t \geq \max(v(\sigma\restriction_i)) \quad \wedge$$
$$(\forall i < |\sigma|)(\forall k < t)\Big[l(\sigma\restriction_i) = \ulcorner \textstyle\bigwedge\!\!\!\bigwedge \urcorner \longrightarrow \rho(it + t + k) = \rho(it + k)\Big] \quad \wedge$$
$$(\forall i < |\sigma|)(\forall j, k < t, j \neq k)\Big[[l(\sigma\restriction_i) = \ulcorner \forall v_j \urcorner \vee l(\sigma\restriction_i) = \ulcorner \exists v_j \urcorner] \longrightarrow \rho(it + t + k) = \rho(it + k)\Big] \quad \wedge$$
$$(\forall i < |\sigma|)(\forall j < t)\Big[l(\sigma\restriction_i) = \ulcorner \exists v_j \urcorner \longrightarrow \rho(it + t + j) = g(\sigma\restriction_i, \langle \rho(it), \rho(it+1), \dots, \rho(it + t - 1)\rangle)\Big]$$

The formula ζ is

$$\forall j\Big[l(\sigma) = \ulcorner \psi_j \urcorner \longrightarrow$$
$$(\exists m \leq k(j)) \bigwedge_{h=1}^{2s} h = \ell(j, m) \longrightarrow P_h(\rho(t|\sigma| + i(j, m, 1)), \dots, \rho(t|\sigma| + i(j, m, r(h))))\Big].$$

Essentially ζ is a satisfaction predicate. One has to see that θ_s can be expressed in a Π_1 way. To see this, note that many of the functions and relations appearing in the formula, such as the functions $j \mapsto \ulcorner \forall v_j \urcorner$ and $\sigma \mapsto \max(v(\sigma))$, the relation $\sigma \in T$, and so on, are computable or C-computable. The theory T_{tup} proves that $\sigma\restriction_i$ and $\langle \rho(it), \rho(it+1), \dots, \rho(it + t - 1)\rangle$ are unique, if they exist. From these facts, standard manipulations allow us to write θ_s in a Π_1 way.

We now show that χ and $\{\theta_s \mid s \in \mathbb{N}\}$ work as desired. First, suppose that $\mathcal{M} \models \varphi$. Let f be a truth definition for (T, l, v) in $\mathcal{M}$ with Skolem function g. We may expand g to be defined on pairs (σ, η), where η is an arbitrary partial valuation, by setting $g(\sigma, \eta) = g(\sigma, \eta^*)$

where η^* is η restricted to $v(\sigma)$ (and by choosing any value when η is not defined on all of $v(\sigma)$). Thus $(\mathcal{M}, C, g) \vDash \chi$. Fix s for which we claim that $(\mathcal{M}^{<\mathbb{N}}, C, g) \vDash \theta_s$. Suppose that we have σ, t, and ρ satisfying the antecedent. For each $i \leq |\sigma|$, let η_i be the partial valuation that maps x_k to $\rho(it+k)$. We claim, by induction, that $f(\sigma\upharpoonright_i, \eta_i) = 1$ for $0 \leq i \leq |\sigma|$. For $i = 0$, we have that $f(\varnothing, \eta_0) = 1$ since $\mathcal{M} \vDash \varphi$ and φ has no free variables. Suppose that $f(\sigma\upharpoonright_i, \eta_i) = 1$. We have a number of cases:

- We cannot have $l(\sigma\upharpoonright_i) = \ulcorner\psi\urcorner$ unless $i = |\sigma|$.
- If $l(\sigma\upharpoonright_i) = \ulcorner\exists x_k\urcorner$ then since $f(\sigma\upharpoonright_i, \eta_i) = 1$, for $a = g(\sigma\upharpoonright i, \eta_i)$, we have $f(\sigma\upharpoonright_i {}^\frown 0, \eta^{x_k \mapsto a}) = 1$. As $\sigma\upharpoonright_{i+1} \in T$, $\sigma\upharpoonright_{i+1} = \sigma\upharpoonright_i {}^\frown 0$. Also, $\eta_{i+1} = \eta_i^{x_k \mapsto a}$. So $f(\sigma\upharpoonright_{i+1}, \eta_{i+1}) = 1$
- If $l(\sigma) = \ulcorner\forall x_i\urcorner$ then since $f(\sigma\upharpoonright_i, \eta_i) = 1$, for each $a \in \mathcal{M}$, $f(\sigma\upharpoonright_i {}^\frown 0, \eta_i^{x_k \mapsto a}) = 1$. Once again, $\sigma\upharpoonright_{i+1} = \sigma\upharpoonright_i {}^\frown 0$, and $\eta_{i+1} = \eta_i^{x_k \mapsto a}$ for some a.
- If $l(\sigma) = \ulcorner\bigwedge\!\!\!\bigwedge\urcorner$ then since $f(\sigma\upharpoonright_i, \eta_i) = 1$, for each k with $\sigma\upharpoonright_i {}^\frown k \in T$, $f(\sigma\upharpoonright_i {}^\frown k, \eta_i) = 1$. Since $\sigma\upharpoonright_{i+1} = \sigma\upharpoonright_i {}^\frown k$ for some such k and $\eta_{i+1} = \eta_i$, $f(\sigma\upharpoonright_{i+1}, \eta_{i+1}) = 1$.

Thus we have that $f(\sigma, \eta_{|\sigma|}) = 1$. Fix ψ_j such that $l(\sigma) = \ulcorner\psi_j\urcorner$. Since $f(\sigma, \eta_{|\sigma|}) = 1$, we have that ψ_j is true with the valuation $\eta_{|\sigma|}$. So there is some $m \leq k(j)$ with $P_{l(j,m)}(\rho(t|\sigma| + i(j,m,1), \ldots, \rho(t|\sigma| + i(j,m,r(h)))))$.

Second, suppose that $\mathcal{M} \nvDash \varphi$. We claim that for each g with $(\mathcal{M}^{<\mathbb{N}}, C, g) \vDash \chi$ there is s such that $(\mathcal{M}^{<\mathbb{N}}, C, g) \nvDash \theta_s$. Fix g with $(\mathcal{M}^{<\mathbb{N}}, C, g) \vDash \chi$. We will define a sequence of finite strings $\sigma_0 \subseteq \sigma_1 \subseteq \cdots \subseteq \sigma_n$ in $\mathbb{N}^{<\mathbb{N}}$, and partial valuations η_i, such that for each i there is no truth definition f with Skolem function g on the subtree below σ_i with $f(\sigma_i, \eta_i) = 1$. As before, even though η_i might have in its domain more than the free variables $v(\sigma_i)$, by abuse of notation we can still write $f(\sigma_i, \eta_i)$ and $g(\sigma_i, \eta_i)$. Begin with $\sigma_0 = \varnothing$, $t_0 = 0$, and $\eta_0 : \varnothing \to \mathcal{M}$ the trivial valuation. As $\mathcal{M} \nvDash \varphi$, there is no f on T with $f(\sigma_0, \eta_0) = 1$. Given σ_i, t_i, and η_i, define σ_{i+1}, t_{i+1}, and η_{i+1} as follows.

- If $l(\sigma_i) = \ulcorner\psi\urcorner$, then end the construction.
- If $l(\sigma_i) = \ulcorner\exists x_j\urcorner$ then let $\sigma_{i+1} = \sigma_i {}^\frown 0$ and $\eta_{i+1} = \eta_i^{x_j \mapsto a}$, where $a = g(\sigma_i, \eta_i)$. Since there is no f with $f(\sigma_i, \eta_i) = 1$, there is no f with $f(\sigma_{i+1}, \eta_{i+1}) = 1$.
- If $l(\sigma_i) = \ulcorner\forall x_j\urcorner$ then there is $a \in \mathcal{M}$ such that there is no f with $f(\sigma_i {}^\frown 0, \eta_i^{x_j \mapsto a}) = 1$. Let $\sigma_{i+1} = \sigma_i {}^\frown 0$ and $\eta_{i+1} = \eta_i^{x_j \mapsto a}$.
- If $l(\sigma_i) = \ulcorner\bigwedge\!\!\!\bigwedge\urcorner$ then there is k with $\sigma {}^\frown k \in T$ such that there is no f with $f(\sigma_i {}^\frown k, \eta_i) = 1$. Let $\sigma_{i+1} = \sigma_i {}^\frown k$ and $\eta_{i+1} = \eta_i$.

Since (T, l, v) is well-founded, eventually this construction ends with $\sigma_0 \subset \cdots \subset \sigma_n$ and $\eta_0, \ldots, \eta_n$. Since σ_n is a leaf, $l(\sigma_n) = \ulcorner\psi_j\urcorner$ for some j. Then since there is no truth definition f with $f(\sigma_n, \eta_n) = 0$, if $l(\sigma_n) = \ulcorner\psi_j(\bar{x})\urcorner$, then $\mathcal{M} \nvDash_{\eta_n} \psi_j(\bar{x})$. Let $\sigma = \sigma_n$. Let t be the number of variables appearing in η_n. We can consider each η_i as a tuple of length t by assigning each unassigned variable to a fixed element of $\mathcal{M}$. Let $\rho = \eta_0 {}^\frown \cdots {}^\frown \eta_1$. Let s be sufficiently large that all of the relations P_h appearing in ψ_j have $h < s$. Then, by construction, this choice of σ, t, and ρ falsify θ_s. □

If we start with a computable formula in a finite language, we do not need any reference to the set C, and we only need one θ rather than the infinitely many θ_s from before.

Lemma 5.9. *If φ is a computable $\bigwedge\!\!\!\bigwedge$-sentence in a finite language τ, then there is a Π_1 τ_{tup}-formula θ such that for all τ-structures $\mathcal{M}$,*

$$\mathcal{M} \models \varphi \Longleftrightarrow (\exists g \colon \mathbb{N} \to \mathcal{M}) \Big[(\mathcal{M}^{<\mathbb{N}}, g) \models \theta \Big].$$

Proof. We can use the same proof as Lemma 5.8, but since the language is finite, we only need a single sentence θ_s for some sufficiently large s; so take θ to be the conjunction of χ with this formula. Moreover, we can replace R by $\varnothing$ as φ is $\varnothing$-computable. □

This lemma proves Theorem 1.4 in the same way that Lemma 5.8 proves Theorem 1.1; it is important here that PA^- and the axioms for T_{tup} consist of only finitely many sentences. Finally, we prove Theorem 1.8.

Theorem 1.8. *Let $\mathbb{K}$ be a class of structures closed under isomorphism. The following are equivalent:*

- $\mathbb{K}$ *is a* PC′*-class which is closed under substructures,*
- $\mathbb{K}$ *is axiomatized by a computably enumerable universal theory.*

Proof. If $\mathbb{K}$ is a PC′-class which is closed under substructures, then by Theorem 1.7, it is axiomatized by a set of universal sentences. Let ϕ be such that $\mathbb{K}$ is the class of reducts of models of ϕ; by listing out the consequences of ϕ, we can enumerate a set T of universal sentences which follow from ϕ.

On the other hand, suppose that $\mathbb{K}$ is axiomatized by a computably enumerable universal theory T. By Corollary 1.5, $\mathbb{K}$ is a PC′-class, and since it is axiomatized by a universal theory, it is closed under substructures. □

References

[HH02] R. Hirsch and I. Hodkinson. *Relation algebras by games*, volume 147 of *Studies in Logic and the Foundations of Mathematics*. North-Holland Publishing Co., Amsterdam, 2002. With a foreword by Wilfrid Hodges.

[Hod08] W. Hodges. *Model Theory.* Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2008.

[Kay91] R. Kaye. *Models of Peano arithmetic*, volume 15 of *Oxford Logic Guides.* The Clarendon Press, Oxford University Press, New York, 1991. Oxford Science Publications.

[LE65] E. G. K. Lopez-Escobar. An interpolation theorem for denumerably long formulas. *Fund. Math.*, 57:253–272, 1965.

[Mal41] A. Mal'cev. On a general method for obtaining local theorems in group theory. *Ivanov. Gos. Ped. Inst. Uč. Zap. Fiz.-Mat. Fak.*, 1(1):3–9, 1941.

[Mar16] D. Marker. *Lectures on infinitary model theory*, volume 46 of *Lecture Notes in Logic*. Association for Symbolic Logic, Chicago, IL; Cambridge University Press, Cambridge, 2016.

[Sco65] D. Scott. Logic with denumerably long formulas and finite strings of quantifiers. In *Theory of Models (Proc. 1963 Internat. Sympos. Berkeley)*, pages 329–341. North-Holland, Amsterdam, 1965.

[Tar54a] A. Tarski. Contributions to the theory of models. I. *Nederl. Akad. Wetensch. Proc. Ser. A.*, 57:572–581 = Indagationes Math. 16, 572–581 (1954), 1954.

[Tar54b] A. Tarski. Contributions to the theory of models. II. *Nederl. Akad. Wetensch. Proc. Ser. A.*, 57:582–588 = Indagationes Math. 16, 582–588 (1954), 1954.

[Vak16] A. Vakili. *Temporal Logic Model Checking as Automated Theorem Proving*. PhD thesis, University of Waterloo, 2016.

[VD14] A. Vakili and N. A. Day. Reducing ctl-live model checking to first-order logic validity checking. In *2014 Formal Methods in Computer-Aided Design (FMCAD)*, pages 215–218, Oct 2014.