



universität  
wien

# MASTERARBEIT

Titel der Masterarbeit

Semantic methods in bounded arithmetic

verfasst von

Thomas Imre, BSc

angestrebter akademischer Grad

Master of Science (MSc)

Wien, 2015

Studienkennzahl lt. Studienblatt: A 066 821

Studienrichtung lt. Studienblatt: Mathematik

Betreut von: O.Univ.-Prof. Sy-David Friedman

Mitbetreut von: Dr. Moritz Müller

# Contents

<b>Abstract</b>	<b>3</b>
<b>Zusammenfassung</b>	<b>5</b>
<b>Acknowledgements</b>	<b>7</b>
<b>1 Preliminaries</b>	<b>8</b>
1.1 Logic and arithmetic . . . . .	8
1.2 Complexity theory . . . . .	13
<b>2 Witnessing and conservation</b>	<b>15</b>
2.1 Conservation by Herbrand-saturation . . . . .	16
2.2 More on Herbrand-saturated structures . . . . .	21
2.3 Buss's Witnessing Theorem . . . . .	23
2.3.1 Defining $UPV_i$ . . . . .	24
2.3.2 Defining $US_2^1(FP_i)$ . . . . .	27
2.3.3 Conservation of $US_2^1(FP_i)$ over $UPV_i$ . . . . .	30
<b>3 Simulation of propositional refutations</b>	<b>35</b>
3.1 Propositional formulas . . . . .	35
3.2 Frege refutation systems . . . . .	41
3.3 Propositional translation . . . . .	45
3.4 Coding literals . . . . .	52

<i>CONTENTS</i>	2
3.5 Coding propositional formulas . . . . .	59
3.6 Coding refutations . . . . .	73
3.7 Simulation of propositional refutations . . . . .	79
3.7.1 The main theorem . . . . .	82
3.7.2 The main theorem for other Frege refutation systems .	85
<b>Bibliography</b>	<b>87</b>
<b>Curriculum Vitae</b>	<b>88</b>

# Abstract

The focus of this thesis is on *Bounded Arithmetic*, which describes a family of theories in the language  $\mathcal{L}_{BA} = \{0, 1, +, \cdot, <, \lfloor \frac{x}{2} \rfloor, |x|, \#\}$ . Those theories are often augmented with induction-like schemes for formulas in which all quantifiers are bounded.

The first part establishes a model-theoretic method to prove conservation of theories. A theory  $T$  is  $\forall\exists$ -conservative over another theory  $T'$  in the same language, if all  $\forall\exists$ -sentences that follow from  $T$  also follow from  $T'$ . To achieve the goal of finding the right conditions on theories to obtain conservation for all  $\forall\exists$ -sentences, we introduce the notion of *Herbrand-saturated* structures. With such structures we are able to prove the main result of this chapter: If every Herbrand-saturated model  $\mathcal{M}$  of a universal theory  $T'$  is also a model of a theory  $T$  in the same language, then  $T$  is  $\forall\exists$ -conservative over  $T'$ .

As an application of this theorem, we show the following: Let  $UPV_i$  be the theory of all true universal sentences in the language consisting of polynomial time functions with  $\Sigma_i^p$ -oracle in the standard model and let  $US_2^1(FP_i)$  be the theory  $UPV_i$  plus length minimization for strict  $\Sigma_1^b$ -formulas in that language. Then  $US_2^1(FP_i)$  is  $\forall\exists$ -conservative over  $UPV_i$ .

The second main topic discussed in this thesis is *simulation of propositional refutations*. In this chapter, we develop a method for translating

first-order sentences into propositional formulas. In order to work in a non-standard model of arithmetic, we show how we code such formulas as well as refutations. Further, we introduce a truth formula for codes of propositional formulas. We show that a bounded sentence is true if and only if the truth formula states that the code of its propositional translation is true.

Combining all the results, we prove the main theorem of this part: Let  $\varphi$  be an unnested sentence in a bigger language than the language of all definable functions and relations in the standard model of arithmetic and let  $\mathcal{M}$  be a countable nonstandard model of arithmetic. If the sentence  $\varphi$  with all quantifiers bounded by some nonstandard integer is true in an expansion of  $\mathcal{M}$ , then there are no polynomial size refutations of the propositional translations of  $\varphi$ . We prove this result by utilizing the previously defined coding functions and formulas.

# Zusammenfassung

Der Schwerpunkt dieser Arbeit ist *Bounded Arithmetic*. Diese beschreibt eine Familie von Theorien in der Sprache  $\mathcal{L}_{BA} = \{0, 1, +, \cdot, <, \lfloor \frac{x}{2} \rfloor, |x|, \#\}$ . Solche Theorien werden oft um ein Induktions-ähnliches Axiomenschema für Sätze, bei denen alle Quantoren beschränkt sind, erweitert.

Im ersten Teil beschreiben wir eine modelltheoretische Methode mit der wir Konservativität von Theorien zeigen können. Eine Theorie  $T$  ist  $\forall\exists$ -konservativ über einer Theorie  $T'$  der gleichen Sprache, wenn alle  $\forall\exists$ -Sätze, die aus  $T$  folgen auch aus  $T'$  folgen. Um die richtigen Anforderungen an Theorien zu finden, sodass wir die  $\forall\exists$ -Konservativitätseigenschaft erhalten, führen wir *Herbrand-saturierte* Strukturen ein. Mit solchen zeigen wir anschließend das Haupttheorem dieses Kapitels: Wenn jedes Herbrand-saturierte Modell einer universellen Theorie  $T'$  auch Modell einer Theorie  $T$  der gleichen Sprache ist, dann ist  $T$  konservativ über  $T'$  für  $\forall\exists$ -Sätze.

Als Anwendung dieses Theorems beweisen wir folgendes: Sei  $UPV_i$  die Theorie aller wahren universellen Sätze in der Sprache, die aus den Funktionen besteht, die in polynomieller Zeit von einer Turing-Maschine mit  $\Sigma_i^p$ -Orakel berechenbar sind. Darüber hinaus sei  $US_2^1(FP_i)$  die Theorie  $UPV_i$  mit zusätzlicher length minimization für strikte  $\Sigma_1^b$ -Formeln in dieser Sprache. Dann ist  $UPV_i$  konservativ über  $US_2^1(FP_i)$  für  $\forall\exists$ -Sätze.

Das zweite große Thema, das in dieser Arbeit behandelt wird, ist die

*Simulation von Widerspruchsbeweisen aussagenlogischer Formeln.* In diesem Kapitel entwickeln wir eine Methode, prädikatenlogische Formeln erster Stufe in aussagenlogische Formeln zu übersetzen. Um in Nichtstandardmodellen der Arithmetik arbeiten zu können, zeigen wir, wie wir solche Formeln und Widerspruchsbeweise codieren. Weiters führen wir eine Wahrheitsformel für Codes aussagenlogischer Formeln ein. Mit dieser zeigen wir, dass ein beschränkt quantifizierter Satz genau dann wahr ist, wenn die Wahrheitsformel sagt, dass der Code der aussagenlogischen Übersetzung wahr ist.

Wir wenden diese Resultate an, um das Haupttheorem dieses Kapitels zu beweisen: Sei  $\varphi$  ein Satz in einer größeren Sprache als der Sprache aller definierbaren Funktionen und Relationen im Standardmodell der Arithmetik und  $\mathcal{M}$  ein abzählbares Nichtstandardmodell der Arithmetik. Wenn der Satz  $\varphi$ , bei dem alle Quantoren durch eine nichtstandard Zahl beschränkt sind, in einer Expansion von  $\mathcal{M}$  wahr ist, dann existieren keine polynomiell großen Widerspruchsbeweise der aussagenlogischen Übersetzungen von  $\varphi$ . Wir beweisen dies mit Hilfe der zuvor definierten Codierungs-Funktionen und -Formeln.

# Acknowledgements

First of all, I would like to thank Sy-David Friedman for the opportunity to write my master thesis at the Kurt Gödel Research Center. Further, I want to express my gratitude to Moritz Müller, whose idea it was to work on this topic. He spent countless hours answering my questions, discussing my thesis, and helping me shape this thesis.

A very important part in my studies was my family. Without their support I would not and could not have come so far. So thank you to my parents for supporting me over the years. Thank you to my brother Martin and to Helena, our weekly science sessions will not be forgotten.

Furthermore, I want to express my gratitude to my colleagues and friends Martin Köberl and Markus Müller. You inspired me to see math as something more and not only contributed a lot mathematically but also made it very enjoyable. Beyond that, I want to thank Markus for his friendship and extended support in the last few years.

Lastly, I want to say my thanks to the most important person in my studies and in my life, Angela Henke. Thank you for always being by my side and keeping me motivated. There are not enough words to do justice to how grateful I am.



# Chapter 1

## Preliminaries

In this chapter we revisit some important notions. We assume some basic knowledge of logic and complexity theory but give the following definitions as a reminder.

### 1.1 Logic and arithmetic

We start by giving a reminder on basic concepts in logic and arithmetic. For more details we refer the reader to [7], [8] and [9].

**Definition.** A language  $\mathcal{L}$  is a set of constants, function symbols and relation symbols. Function symbols and relation symbols have a positive arity, i.e., the number of arguments accepted by the function symbol or relation symbol.

**Note.** In this thesis we use the convention that every language contains at least one constant.

**Definition.** Let  $\mathcal{L}$  be a language and  $\mathbb{X}$  be a set of variables. We define  $A_{\mathcal{L}} = \{\neg, \vee, \wedge, \forall, \exists, =, \cdot, \cdot\} \cup \mathbb{X} \cup \mathcal{L}$  as the alphabet of the language  $\mathcal{L}$ .

**Definition.** Let  $\mathcal{L}$  be a language. An  $\mathcal{L}$ -term is a string over the alphabet  $A_{\mathcal{L}}$  built by the following rules.

- Every variable is an  $\mathcal{L}$ -term.
- Every constant in  $\mathcal{L}$  is an  $\mathcal{L}$ -term.
- If  $f$  is a function symbol in  $\mathcal{L}$  with arity  $r$  and  $t_1, \dots, t_r$  are  $\mathcal{L}$ -terms, then so is  $f(t_1, \dots, t_r)$ .

**Definition.** Let  $\mathcal{L}$  be a language. An  $\mathcal{L}$ -formula is a string over the alphabet  $A_{\mathcal{L}}$  built by the following rules.

- (a) If  $s$  and  $t$  are  $\mathcal{L}$ -terms, then  $s = t$  is an  $\mathcal{L}$ -formula.
- (b) If  $R \in \mathcal{L}$  is a relation symbol with arity  $r$  and  $t_1, \dots, t_r$  are  $\mathcal{L}$ -terms, then  $R(t_1, \dots, t_r)$  is an  $\mathcal{L}$ -formula.
- (c) If  $\varphi$  is an  $\mathcal{L}$ -formula, then  $\neg\varphi$  is an  $\mathcal{L}$ -formula.
- (d) If  $\varphi_1$  and  $\varphi_2$  are  $\mathcal{L}$ -formulas, then  $(\varphi_1 \vee \varphi_2)$  and  $(\varphi_1 \wedge \varphi_2)$  are  $\mathcal{L}$ -formulas.
- (e) If  $\varphi$  is an  $\mathcal{L}$ -formula and  $x$  is a variable, then  $\exists x\varphi$  and  $\forall x\varphi$  are  $\mathcal{L}$ -formulas. We call  $\exists$  and  $\forall$  quantifiers.

**Definition.**

- We call  $\mathcal{L}$ -formulas atomic, if they are derived by only applying rule (a) or (b).
- If  $\varphi$  is an atomic  $\mathcal{L}$ -formula or the negation of an atomic  $\mathcal{L}$ -formula, we say  $\varphi$  is an  $\mathcal{L}$ -literal.
- We say an  $\mathcal{L}$ -formula is quantifier-free, if it has no quantifiers, i.e., it is derived by only applying the rules (a)-(d).

**Notation.** We use the common abbreviation  $\varphi \rightarrow \psi$  for  $\neg\varphi \vee \psi$  where  $\varphi$  and  $\psi$  are  $\mathcal{L}$ -formulas in a language  $\mathcal{L}$ .

Further, we often write  $\bigvee_{1 \leq i \leq r} \varphi_i$  or  $\bigwedge_{1 \leq i \leq r} \varphi_i$  if  $\varphi_i$  ( $1 \leq i \leq r$ ) are  $\mathcal{L}$ -formulas for  $(\dots(\varphi_1 \vee \varphi_2) \vee \dots) \vee \varphi_r$  or  $(\dots(\varphi_1 \wedge \varphi_2) \wedge \dots) \wedge \varphi_r$  respectively.

We also omit parentheses for better readability. If  $r = 0$ , we define  $\bigvee_{1 \leq i \leq r} \varphi_i$  as  $\perp$  and  $\bigwedge_{1 \leq i \leq r} \varphi_i$  as  $\top$ .

**Notation.** We denote tuples of variables with a bar, e.g.,  $\bar{x} = (x_1, \dots, x_r)$  for  $r \in \mathbb{N}$ .

**Definition.** Let  $\mathcal{L}$  be a language and  $\varphi$  be an  $\mathcal{L}$ -formula. A variable  $x$  is called free if it is not in the scope of  $\forall$  or  $\exists$ . An  $\mathcal{L}$ -formula without free variables is called  $\mathcal{L}$ -sentence or first-order sentence.

**Notation.** Let  $t$  be an  $\mathcal{L}$ -term and  $\varphi$  be an  $\mathcal{L}$ -formula in a language  $\mathcal{L}$ . If we write  $t(x_1, \dots, x_r)$  or  $\varphi(x_1, \dots, x_r)$ , we mean that all free variables of  $t$  or  $\varphi$  are in  $\{x_1, \dots, x_r\}$ .

**Note.** We abbreviate  $\forall x (x \leq t(\bar{y}) \rightarrow \varphi(x, \bar{y}, \bar{z}))$  by  $\forall x \leq t(\bar{y}) \varphi(x, \bar{y}, \bar{z})$  and  $\exists x \leq t(\bar{y}) \varphi(x, \bar{y}, \bar{z})$  abbreviates  $\exists x (x \leq t(\bar{y}) \wedge \varphi(x, \bar{y}, \bar{z}))$ .

**Definition.** We say a first-order sentence is universal, if it is of the form

$$\forall x_1, \dots, x_r \psi(x_1, \dots, x_r)$$

for an  $r \in \mathbb{N}$  where  $\psi(x_1, \dots, x_r)$  is a quantifier-free first-order formula.

We say a first-order sentence is existential, if it is of the form

$$\exists x_1, \dots, x_r \psi(x_1, \dots, x_r)$$

for an  $r \in \mathbb{N}$  where  $\psi(x_1, \dots, x_r)$  is a quantifier-free first-order formula.

**Notation.** We abbreviate  $\forall x_1, \dots, x_r \psi(x_1, \dots, x_r, \bar{y})$  by writing  $\forall \bar{x} \psi(\bar{x}, \bar{y})$  and  $\exists x_1, \dots, x_r \psi(x_1, \dots, x_r, \bar{y})$  by  $\exists \bar{x} \psi(\bar{x}, \bar{y})$ .

**Definition.** Let  $\mathcal{L}$  be a language. An  $\mathcal{L}$ -structure  $\mathcal{M}$  is defined as a pair  $\mathcal{M} = (M, (S^{\mathcal{M}})_{S \in \mathcal{L}})$  such that  $M$  is a nonempty set and

- $S^{\mathcal{M}} \in M$ , if  $S$  is a constant,

- $S^{\mathcal{M}} : M^r \rightarrow M$ , if  $S$  is a function symbol with arity  $r$ , and
- $S^{\mathcal{M}} \subseteq M^r$ , if  $S$  is a relation symbol with arity  $r$ .

We call  $S^{\mathcal{M}}$  the interpretation of  $S \in \mathcal{L}$  in  $\mathcal{M}$  and say  $M$  is the universe of  $\mathcal{M}$ .

**Definition.** Let  $\mathcal{M} = (M, (S^{\mathcal{M}})_{S \in \mathcal{L}})$  be an  $\mathcal{L}$ -structure for a language  $\mathcal{L}$ . Further let  $t(x_1, \dots, x_r)$  be an  $\mathcal{L}$ -term and  $a_1, \dots, a_r \in M$ . The interpretation of an  $\mathcal{L}$ -term in  $\mathcal{M}$  is defined inductively on the complexity of  $t(x_1, \dots, x_r)$ :

- If  $t(x_1, \dots, x_r)$  is the variable  $x_i$ , then  $t^{\mathcal{M}}(a_1, \dots, a_r) = a_i$ .
- If  $t(x_1, \dots, x_r)$  is a constant  $c \in \mathcal{L}$ , then  $t^{\mathcal{M}}(a_1, \dots, a_r) = c^{\mathcal{M}}$ .
- If  $t(x_1, \dots, x_r)$  is of the form  $f(t_1(x_1, \dots, x_r), \dots, t_s(x_1, \dots, x_r))$  for  $\mathcal{L}$ -terms  $t_i$  ( $1 \leq i \leq s$ ), then

$$t^{\mathcal{M}}(a_1, \dots, a_r) = f^{\mathcal{M}}(t_1^{\mathcal{M}}(a_1, \dots, a_r), \dots, t_s^{\mathcal{M}}(a_1, \dots, a_r)).$$

**Note.** By the definition above, every term  $t(x_1, \dots, x_r)$  defines, interpreted in  $\mathcal{M}$ , a function  $t^{\mathcal{M}} : M^r \rightarrow M$ .

**Definition.** Let  $\mathcal{M} = (M, (S^{\mathcal{M}})_{S \in \mathcal{L}})$  be an  $\mathcal{L}$ -structure for a language  $\mathcal{L}$ . Assume  $N \subseteq M$  and  $N \neq \emptyset$ , all interpretations  $c^{\mathcal{M}}$  of  $\mathcal{L}$ -constants  $c$  in  $\mathcal{M}$  are in  $N$  and  $N$  is closed under all functions  $f^{\mathcal{M}}$ . If we restrict the interpretations of the symbols in  $\mathcal{L}$  on  $N$ , we obtain a structure  $\mathcal{N} = (N, (S^{\mathcal{N}})_{S \in \mathcal{L}})$ , which we call substructure of  $\mathcal{M}$ .

**Definition.** Let  $\mathcal{M} = (M, (S^{\mathcal{M}})_{S \in \mathcal{L}})$  be an  $\mathcal{L}$ -structure for a language  $\mathcal{L}$  and let  $\mathcal{L}'$  be a language such that  $\mathcal{L} \subseteq \mathcal{L}'$ . If  $\mathcal{N} = (N, (S^{\mathcal{N}})_{S \in \mathcal{L}'})$  is an  $\mathcal{L}'$ -structure such that  $N = M$  and  $S^{\mathcal{N}} = S^{\mathcal{M}}$  for all  $S \in \mathcal{L}$ , then we call  $\mathcal{N}$  an  $\mathcal{L}'$ -expansion of  $\mathcal{M}$ .

**Definition.** Let  $\mathcal{L}$  be a language. A set  $T$  of  $\mathcal{L}$ -sentences is called  $\mathcal{L}$ -theory.

**Definition.** Let  $T$  be a theory in a language  $\mathcal{L}$ . A model  $\mathcal{M}$  of  $T$  is an  $\mathcal{L}$ -structure such that for all sentences  $\varphi$  in  $T$ ,  $\mathcal{M} \models \varphi$ .

**Notation.** We often do not distinguish between  $\mathcal{M}$  and its universe  $M$  and write  $\mathcal{M}$  for both. It should be clear from the context which of them is meant.

**Definition.** Let  $\mathcal{L}$  be a language.

- An  $\mathcal{L}$ -formula  $\varphi$  is valid, if it is true in all  $\mathcal{L}$ -structures. We write  $\models \varphi$ .
- Let  $T$  be an  $\mathcal{L}$ -theory and  $\varphi$  an  $\mathcal{L}$ -sentence. We write  $T \models \varphi$ , if for all  $\mathcal{L}$ -models  $\mathcal{M}$  of  $T$ ,  $\mathcal{M} \models \varphi$  also holds.

**Definition.** Let  $\mathcal{L}$  be a language and  $T$  an  $\mathcal{L}$ -theory.  $T$  is consistent if there exists a model  $\mathcal{M}$  of  $T$ .

**Definition.** The structure  $\mathfrak{N} = (\mathbb{N}, 0^{\mathfrak{N}}, S^{\mathfrak{N}}, +^{\mathfrak{N}}, \cdot^{\mathfrak{N}}, <^{\mathfrak{N}})$  is called the standard model of arithmetic where  $\mathbb{N}$  denotes the set of natural numbers,  $0^{\mathfrak{N}}$  the integer 0,  $S^{\mathfrak{N}}$  the successor function,  $+^{\mathfrak{N}}$  the addition,  $\cdot^{\mathfrak{N}}$  the multiplication and  $<^{\mathfrak{N}}$  the less-than relation.

**Definition.** Let  $\mathcal{M} = (M, (S^{\mathcal{M}})_{S \in \mathcal{L}})$  and  $\mathcal{N} = (N, (S^{\mathcal{N}})_{S \in \mathcal{L}})$  be  $\mathcal{L}$ -structures for a language  $\mathcal{L}$ .

- An  $\mathcal{L}$ -structure homomorphism is a map  $f : M \rightarrow N$  such that
  - For every constant  $c \in \mathcal{L}$ ,  $f(c^{\mathcal{M}}) = c^{\mathcal{N}}$ .
  - For every relation symbol  $R \in \mathcal{L}$  with arity  $r > 0$  and every tuple  $(a_1, \dots, a_r) \in M^r$ ,  
if  $(a_1, \dots, a_r) \in R^{\mathcal{M}}$ , then  $(f(a_1), \dots, f(a_r)) \in R^{\mathcal{N}}$ .
  - For every function symbol  $F \in \mathcal{L}$  with arity  $r > 0$  and every tuple  $(a_1, \dots, a_r) \in M^r$ ,  
 $f(F^{\mathcal{M}}(a_1, \dots, a_r)) = F^{\mathcal{N}}(f(a_1), \dots, f(a_r))$ .

- An elementary embedding is an  $\mathcal{L}$ -structure homomorphism  $f : M \rightarrow N$  such that for every  $\mathcal{L}$ -formula  $\varphi(x_1, \dots, x_r)$  and all  $a_1, \dots, a_r \in M$ , the following holds

$$\mathcal{M} \models \varphi(a_1, \dots, a_r) \text{ if and only if } \mathcal{N} \models \varphi(f(a_1), \dots, f(a_r)).$$

- The structure  $\mathcal{N}$  is called elementary extension if  $M \subseteq N$  and the inclusion map  $f : M \rightarrow N$  is an elementary embedding. It is called a proper elementary extension, if  $M \subsetneq N$ .

**Definition.** Let  $\mathcal{M}$  be a proper elementary extension of the standard model of arithmetic. We call  $\mathcal{M}$  a nonstandard model of arithmetic. The elements of the universe of  $\mathcal{M}$  which are not in  $\mathbb{N}$  are called nonstandard integers.

**Notation.** From now on, we omit "of arithmetic" and refer to the models defined above as the standard model or nonstandard models.

## 1.2 Complexity theory

The definitions and results of this sections as well as more details can be found in [1].

**Definition.** A problem  $Q$  is a subset of  $\mathbb{N}$ .

**Definition.** Let  $x \in \mathbb{N}$ . We denote the length of  $x$  by  $|x|$  which is equal to  $\lceil \log_2(x+1) \rceil$ .

**Definition.** Let  $f : \mathbb{N}^r \rightarrow \mathbb{N}$  be a function. We say  $f$  is a polynomial time function if there is a polynomial  $p$  and a Turing machine  $\mathbb{T}$  such that for every  $x_1, \dots, x_r \in \mathbb{N}$  after at most  $p(|x_1|, \dots, |x_r|)$ -many steps on input  $(x_1, \dots, x_r)$  in binary,  $\mathbb{T}$  reaches the halting state and  $f(x_1, \dots, x_r)$  is written on the work tape in binary.

**Definition.** Let  $O \subseteq \mathbb{N}$  be a problem. An oracle Turing machine  $\mathbb{T}$  with oracle  $O$  is a Turing machine augmented with an extra read/write tape which is called oracle tape and three states  $q_?$ ,  $q_y$ ,  $q_n$ . Every time  $\mathbb{T}$  reaches the state  $q_?$ , the Turing machine  $\mathbb{T}$  goes into state  $q_y$  if what is written on the oracle tape (in binary) is in  $O$  and goes into state  $q_n$  if it is not.

**Definition.** Let  $i \geq 1$ . A problem  $Q$  is in  $\Sigma_i^p$  if there exists a polynomial time function  $f : \mathbb{N}^{i+1} \rightarrow \mathbb{N}$ , and a polynomial  $p$  such that  $x \in Q$  if and only if

$$\exists y_1 < 2^{p(|x|)} \forall y_2 < 2^{p(|x|)} \dots Q_i y_i < 2^{p(|x|)} f(x, y_1, \dots, y_i) = 1$$

where  $Q_i$  denotes  $\forall$  if  $i$  is even and  $Q_i$  denotes  $\exists$  if  $i$  is odd.

**Notation.** We code sequences of natural numbers by converting them into binary representation with  $*$  between two entries of the sequence and then applying the map  $0 \mapsto 00$ ,  $1 \mapsto 11$  and  $*$   $\mapsto$   $01$ . Then we convert the resulting binary number back to its integer representation.

The tupling functions  $\langle \cdot \rangle_r : \mathbb{N}^r \rightarrow \mathbb{N}$  with  $r \in \mathbb{N}$  map an  $r$ -ary tuple of natural numbers to its code. We omit the index  $r$  and just write  $\langle \cdot \rangle$  for any  $r$ -ary tupling function. Further, we use the function  $\|x\|$  which maps  $x$  to the length of the sequence which  $x$  codes.

Furthermore, when we write  $(x)_i$  for an  $i \in \mathbb{N}$ , we mean the output of the function which maps  $(x, i)$  to the  $i$ -th entry of the sequence which is coded by  $x$ . We use the convention that a sequence starts with the 0th entry.

**Note.** By our choice of coding, the functions  $\langle \cdot \rangle$ ,  $\|x\|$  and  $(x)_i$  above are polynomial time functions.

Now we have given all necessary definitions needed and are able to start with the main part.

# Chapter 2

## Witnessing and conservation

In this chapter, we introduce the notion of Herbrand-saturated structures and show how to use them to prove conservation of one theory over another. By conservation we mean the following.

**Definition.** *Let  $T_1$  and  $T_2$  be theories in the same language and let  $\Phi$  be a set of first-order formulas in this language. We say  $T_1$  is conservative for  $\Phi$ -formulas (or  $\Phi$ -conservative) over  $T_2$ , if for every  $\varphi \in \Phi$ ,*

$$T_1 \models \varphi \text{ implies } T_2 \models \varphi.$$

We show that for two theories  $T_1$  and  $T_2$  in the same language where  $T_2$  is universal, the following holds: If every Herbrand-saturated model of  $T_2$  is also a model of  $T_1$ , then  $T_1$  is  $\forall\exists$ -conservative over  $T_2$ .

This particular method is based on a paper by Avigad [2]. It serves as an alternative to the more common proof-theoretic approach.

With this technique we show a similar result to the so-called Buss's Witnessing Theorem [3]. This was first proven in [3] in a proof-theoretic manner like many other conservation results. Our approach will be semantically.

In the first section of this chapter we define Herbrand-saturated structures and show how we obtain  $\forall\exists$ -conservation of a theory over another theory in the same language by using such structures. We show some more results



about Herbrand-saturated structures afterwards. In the last section of this chapter, we prove a similar result to Buss's Witnessing Theorem with the techniques and methods developed in the first section.

## 2.1 Conservation by Herbrand-saturation

**Definition.** *A theory  $T$  is universal, if it only consists of universal sentences.*

Next is Herbrand's theorem which will be useful later on. This theorem is the most important part when it comes to witnessing results.

**Theorem 2.1** (Herbrand's Theorem). *Let  $T$  be a universal theory and assume that  $T \models \forall \bar{x} \exists \bar{y} \psi(\bar{x}, \bar{y})$  where  $\psi$  is a quantifier-free formula and  $\bar{y} = (y_1, \dots, y_r)$ .*

*Then there are terms  $t_i^j(\bar{x})$  and an  $s \in \mathbb{N}$  with  $1 \leq i \leq r$  and  $1 \leq j \leq s$ , such that*

$$T \models \forall \bar{x} \bigvee_{j=1}^s \psi(\bar{x}, t_1^j(\bar{x}), t_2^j(\bar{x}), \dots, t_r^j(\bar{x})).$$

*Proof.* Assume  $T \not\models \bigvee_{j=1}^s \psi(\bar{x}, t_1^j(\bar{x}), t_2^j(\bar{x}), \dots, t_r^j(\bar{x}))$ . Then  $T$  is consistent with the set

$$S = \{\neg \psi(\bar{c}, t_1(\bar{c}), \dots, t_r(\bar{c})) \mid t_1(\bar{x}), \dots, t_r(\bar{x}) \text{ terms in the language of } T\}$$

where  $\bar{c}$  is a new tuple of constants. Thus, there is a model  $\mathcal{M}$  of  $S \cup T$ . Now let  $\mathcal{N}$  be the substructure of  $\mathcal{M}$  with the universe  $N = \{t(\bar{c}) \mid t(\bar{x}) \text{ is a term in the language of } T\}$ . Note that  $N$  contains all constants of the language of  $T$  interpreted in  $\mathcal{M}$  and is closed under the interpretations of the function symbols in  $\mathcal{M}$ . Then  $\mathcal{N}$  is a model of  $T$  where  $\exists \bar{y} \psi(\bar{c}, \bar{y})$  fails.  $\square$

The next definition gives us a useful property of theories.

**Definition.** A theory  $T$  supports definition by cases, if for every finite number of terms  $t_1(\bar{x}), \dots, t_s(\bar{x})$  and quantifier-free formulas  $\theta_1(\bar{x}), \dots, \theta_{s-1}(\bar{x})$  in the language of  $T$  there is a function symbol  $F$  such that:

$$T \models \forall \bar{x} F(\bar{x}) = \begin{cases} t_1(\bar{x}) & \text{if } \theta_1(\bar{x}) \\ t_2(\bar{x}) & \text{if } \neg\theta_1(\bar{x}) \wedge \theta_2(\bar{x}) \\ \vdots \\ t_{s-1}(\bar{x}) & \text{if } \neg\theta_1(\bar{x}) \wedge \neg\theta_2(\bar{x}) \wedge \dots \wedge \neg\theta_{s-2}(\bar{x}) \wedge \theta_{s-1}(\bar{x}) \\ t_s(\bar{x}) & \text{otherwise.} \end{cases}$$

If a theory supports definition by cases, then we can use a function symbol for distinct terms depending on the truth of quantifier-free formulas. For example, in the proof of Herbrand's Theorem we get a function symbol eliminating all the terms.

**Corollary 2.2.** Let  $T$  be a universal theory that supports definition by cases and assume  $T \models \forall \bar{x} \exists \bar{y} \psi(\bar{x}, \bar{y})$  where  $\psi$  is a quantifier-free formula and  $\bar{y} = (y_1, \dots, y_r)$  with  $r \in \mathbb{N}$ .

Then there are function symbols  $F_1, \dots, F_r$  such that

$$T \models \forall \bar{x} \psi(\bar{x}, F_1(\bar{x}), F_2(\bar{x}), \dots, F_r(\bar{x})).$$

*Proof.* Theorem 2.1 implies that there are terms  $t_i^j(\bar{x})$  and an  $s \in \mathbb{N}$  with  $1 \leq i \leq r$  and  $1 \leq j \leq s$ , such that

$$T \models \forall \bar{x} \bigvee_{j=1}^s \psi(\bar{x}, t_1^j(\bar{x}), t_2^j(\bar{x}), \dots, t_r^j(\bar{x})).$$

Since  $\psi(\bar{x}, y_1, \dots, y_r)$  is quantifier-free and  $T$  supports definition by cases, there exist function symbols  $F_i$  for  $1 \leq i \leq r$  such that in models of  $T$  the

following holds

$$\forall \bar{x} F_i(\bar{x}) = \begin{cases} t_i^1(\bar{x}) & \text{if } \psi(\bar{x}, t_1^1(\bar{x}), \dots, t_r^1(\bar{x})) \\ t_i^2(\bar{x}) & \text{if } \neg\psi(\bar{x}, t_1^1(\bar{x}), \dots, t_r^1(\bar{x})) \wedge \psi(\bar{x}, t_1^2(\bar{x}), \dots, t_r^2(\bar{x})) \\ \vdots & \\ t_i^{s-1}(\bar{x}) & \text{if } \bigwedge_{j=1}^{s-2} \neg\psi(\bar{x}, t_1^j(\bar{x}), \dots, t_r^j(\bar{x})) \wedge \\ & \psi(\bar{x}, t_1^{s-1}(\bar{x}), \dots, t_r^{s-1}(\bar{x})) \\ t_i^s(\bar{x}) & \text{otherwise.} \end{cases}$$

Hence, we obtain

$$T \models \forall \bar{x} \psi(\bar{x}, F_1(\bar{x}), F_2(\bar{x}), \dots, F_r(\bar{x})).$$

□

Next we define Herbrand-saturated structures. Structures with this property will play an important role in proving conservation results.

**Definition.** Let  $\mathcal{L}$  be a language and  $\mathcal{M}$  be an  $\mathcal{L}$ -structure.

- The language  $\mathcal{L}(\mathcal{M})$  is the language  $\mathcal{L}$  together with additional constants for the elements of the universe of  $\mathcal{M}$ .
- The universal diagram of  $\mathcal{M}$  is the set of all universal sentences in the language  $\mathcal{L}(\mathcal{M})$  which are true in  $\mathcal{M}$ .
- We say  $\varphi$  is an  $\exists\forall$ -sentence, if  $\varphi = \exists\bar{x}\forall\bar{y} \psi(\bar{x}, \bar{y})$  where  $\psi(\bar{x}, \bar{y})$  is quantifier-free.  
Similarly, we say that  $\varphi$  is a  $\forall\exists$ -sentence, if  $\varphi = \forall\bar{x}\exists\bar{y} \psi(\bar{x}, \bar{y})$  for a quantifier-free formula  $\psi(\bar{x}, \bar{y})$ .
- The  $\mathcal{L}$ -structure  $\mathcal{M}$  is Herbrand-saturated if for any  $\exists\forall$ -sentence  $\varphi$  in the language  $\mathcal{L}(\mathcal{M})$  which is consistent with the universal diagram of

$\mathcal{M}$ , we have  $\mathcal{M} \models \varphi$ .

- Let  $T_1$  and  $T_2$  be theories in the language  $\mathcal{L}$ .  
We say  $T_1$  is  $\forall\exists$ -conservative over  $T_2$ , if  $T_1$  is  $\Phi$ -conservative over  $T_2$  where  $\Phi$  is the set of all  $\forall\exists$ -sentences in  $\mathcal{L}$ .

**Proposition 2.3.** *Every consistent universal theory  $T$  has an Herbrand-saturated model.*

*Proof.* Let  $\mathcal{L}$  be the language of  $T$ . For simplicity we assume that  $\mathcal{L}$  is countable. The same argument works for uncountable languages using transfinite induction. Let  $\mathcal{L}'$  denote an extension of  $\mathcal{L}$  with countably many new constants  $c_0, c_1, \dots$ . Furthermore, we enumerate all quantifier-free formulas of  $\mathcal{L}'$  with  $\theta_1(\bar{x}_1, \bar{y}_1), \theta_2(\bar{x}_2, \bar{y}_2), \dots$  where  $\bar{x}_i = (x_1, \dots, x_{r_i})$  and  $\bar{y}_i = (y_1, \dots, y_{s_i})$  for  $r_i, s_i \in \mathbb{N}$  and  $i \geq 1$ . The next step is constructing an increasing sequence of sets of universal sentences to obtain the universal diagram of the desired model.

- Let  $S_0 = T$ . Since  $T$  is universal, so is  $S_0$ .
- At stage  $i + 1$  we try to satisfy  $\forall \bar{y}_{i+1} \theta_{i+1}(\bar{x}_{i+1}, \bar{y}_{i+1})$ .

To achieve the latter, we pick a tuple of the newly introduced constants  $\bar{c}$  which does not occur in  $S_i$  or  $\theta_{i+1}$  and let

$$S_{i+1} := \begin{cases} S_i \cup \{\forall \bar{y}_{i+1} \theta_{i+1}(\bar{c}, \bar{y}_{i+1})\} & \text{if this is consistent,} \\ S_i & \text{otherwise.} \end{cases}$$

By induction on  $i$  we show that every  $S_i$  is consistent. The theory  $S_0 = T$  is consistent by assumption. Now assume that  $S_i$  is consistent. If  $S_{i+1} = S_i \cup \{\forall \bar{y}_{i+1} \theta_{i+1}(\bar{c}, \bar{y}_{i+1})\}$ , it has to be consistent by definition. Otherwise,  $S_{i+1} = S_i$  and thus, it is consistent by the induction hypothesis. Therefore,  $S = \bigcup_{i \in \mathbb{N}} S_i$  is also consistent.

For uncountable languages we use the same construction as above at successor stages and the union of all previously defined  $S_i$  at limit stages.

Let  $\mathcal{N}$  be a model of  $S$  and  $\mathcal{M}$  be a substructure of  $\mathcal{N}$  such that the universe of  $\mathcal{M}$  is the set  $\{t^{\mathcal{N}} \mid t \text{ closed term in } \mathcal{L}'\}$ . Since  $S$  only consists of universal sentences, every substructure of  $\mathcal{N}$  is also a model of  $S$ . Hence,  $\mathcal{M}$  is a model of  $S$  and thus,  $\mathcal{M} \models T$ .

Note that each element of  $\mathcal{M}$  is denoted by one of the  $c_i$ 's we introduced earlier. This follows from the fact that every element of  $\mathcal{M}$  can be written as a term  $t$  in  $\mathcal{L}'$ . Now pick  $j$  such that  $\theta_j(\bar{x}_j, \bar{y}_j)$  is the statement  $x = t$ . Hence for the constant  $c$  introduced in this step of the construction, the formula  $c = t$  is an element of  $S_{j+1}$ .

Finally, we show that  $\mathcal{M}$  is Herbrand-saturated. Assume there is a quantifier-free formula  $\varphi$  and a tuple of parameters  $\bar{a}$  in  $\mathcal{M}$  such that we have  $\mathcal{M} \not\models \exists \bar{x} \forall \bar{y} \varphi(\bar{x}, \bar{y}, \bar{a})$ . We claim that this sentence is inconsistent with the universal diagram of  $\mathcal{M}$ .

By the remark above, every entry of  $\bar{a}$  is denoted by a constant in  $\mathcal{L}'$ . Let  $\bar{b}$  be the tuple of constants in  $\mathcal{L}'$  such that  $b_i$  denotes  $a_i$  for all  $a_i$  in  $\bar{a}$ . Choose an index  $i_0$  such that  $\theta_{i_0+1}(\bar{x}, \bar{y}) = \varphi(\bar{x}, \bar{y}, \bar{b})$ . Let  $\bar{c}$  be the constants used at stage  $i_0 + 1$  in the construction. Then  $\mathcal{M} \not\models \forall \bar{y} \varphi(\bar{c}, \bar{y}, \bar{b})$  and hence, the sentence is inconsistent with  $S_i$ . Since  $\bar{c}$  does not occur in  $S_i$ , the formula  $\exists \bar{x} \forall \bar{y} \varphi(\bar{x}, \bar{y}, \bar{b})$  is also inconsistent with  $S_i$ .

Now we rename  $\bar{b}$  to  $\bar{a}$  and the constants in  $S_i$  to the corresponding elements of the universe of  $\mathcal{M}$ . If we choose to name the constants of  $\mathcal{L}(\mathcal{M})$  the same as the element in the universe they are representing, we obtain that the renamed  $S_i$  is a subset of the universal diagram of  $\mathcal{M}$ . This proves the desired inconsistency.  $\square$

We will now turn to the main result of this section, which provides us with

all the requirements on models of theories we need for proving conservation.

**Theorem 2.4.** *Let  $T_1$  and  $T_2$  be theories in the same language such that  $T_2$  is universal.*

*If every Herbrand-saturated model of  $T_2$  is also a model of  $T_1$ , then for every  $\forall\exists$ -sentence  $\varphi$ ,  $T_1 \models \varphi$  implies  $T_2 \models \varphi$ .*

*Proof.* Assume that every Herbrand-saturated model of  $T_2$  is also a model of  $T_1$ . Let  $\varphi(\bar{x}, \bar{y})$  be a quantifier-free formula in the language of  $T_2$ . Suppose that  $T_2 \not\models \forall\bar{x}\exists\bar{y} \varphi(\bar{x}, \bar{y})$ . Then we have to show that  $T_1 \not\models \forall\bar{x}\exists\bar{y} \varphi(\bar{x}, \bar{y})$ . Assume this sentence is not true in some models of  $T_2$ . Then we obtain that  $T'_2 := T_2 \cup \{\forall\bar{y} \neg\varphi(\bar{c}, \bar{y})\}$  is consistent and universal where  $\bar{c}$  is a new tuple of constants.

By Proposition 2.3 there is an Herbrand-saturated model  $\mathcal{M}$  of  $T'_2$ . Let  $\mathcal{N}$  be the restriction of  $\mathcal{M}$  to the language of both theories. It is easy to see that  $\mathcal{N}$  is Herbrand-saturated: Let  $\psi$  be an  $\exists\forall$ -sentence in  $\mathcal{L}(\mathcal{N})$  that is consistent with the universal diagram of  $\mathcal{N}$ . Since  $\mathcal{N}$  is a restriction of  $\mathcal{M}$ , it follows that  $\psi$  is an  $\exists\forall$ -sentence in  $\mathcal{L}(\mathcal{M})$  that is consistent with the universal diagram of  $\mathcal{M}$ . Thus,  $\mathcal{M} \models \psi$  because  $\mathcal{M}$  is Herbrand-saturated. Therefore, we also obtain  $\mathcal{N} \models \psi$  since  $\psi$  is in  $\mathcal{L}(\mathcal{N})$  and  $\mathcal{N}$  is restriction of  $\mathcal{M}$ .

By construction,  $\mathcal{N}$  is an Herbrand-saturated model of  $T_2$  such that  $\mathcal{N} \models \exists\bar{x}\forall\bar{y} \neg\varphi(\bar{x}, \bar{y})$ . Therefore,  $\mathcal{N}$  is also a model of  $T_1$  in which  $\forall\bar{x}\exists\bar{y} \varphi(\bar{x}, \bar{y})$  is false.  $\square$

## 2.2 More on Herbrand-saturated structures

In this section we show some consequences for Herbrand-saturated structures from previously proven results.

**Proposition 2.5.** *Let  $\mathcal{M}$  be an Herbrand-saturated structure in a language  $\mathcal{L}$ . Suppose  $\mathcal{M} \models \forall\bar{x}\exists\bar{y} \varphi(\bar{x}, \bar{y}, \bar{a})$  where  $\varphi$  is quantifier-free,  $\bar{a}$  is a tuple of*

parameters in  $\mathcal{M}$  and  $\bar{y} = (y_1, \dots, y_r)$ . Then there is a universal formula  $\psi(\bar{z}, \bar{w})$ , an integer  $s$  and terms  $t_i^j(\bar{x}, \bar{z}, \bar{w})$  with  $1 \leq i \leq r$  and  $1 \leq j \leq s$  such that  $\mathcal{M} \models \exists \bar{w} \psi(\bar{a}, \bar{w})$  and

$$\models \psi(\bar{z}, \bar{w}) \rightarrow \bigvee_{j=1}^s \varphi(\bar{x}, t_1^j(\bar{x}, \bar{z}, \bar{w}), t_2^j(\bar{x}, \bar{z}, \bar{w}), \dots, t_r^j(\bar{x}, \bar{z}, \bar{w}), \bar{z}).$$

*Proof.* Assume  $\mathcal{M} \models \forall \bar{x} \exists \bar{y} \varphi(\bar{x}, \bar{y}, \bar{a})$  holds. Hence,  $\exists \bar{x} \forall \bar{y} \neg \varphi(\bar{x}, \bar{y}, \bar{a})$  is not true in  $\mathcal{M}$ . Since it is an  $\exists \forall$ -sentence and  $\mathcal{M}$  is Herbrand-saturated, it is inconsistent with the universal diagram of  $\mathcal{M}$ . Therefore, there is a universal formula  $\psi(\bar{z}, \bar{w})$  and a tuple of parameters  $\bar{b}$  satisfying

$$\begin{aligned} \mathcal{M} &\models \psi(\bar{a}, \bar{b}) \text{ and} \\ &\models \psi(\bar{a}, \bar{b}) \rightarrow \exists \bar{y} \varphi(\bar{x}, \bar{y}, \bar{a}). \end{aligned}$$

Now replace the tuples  $\bar{a}$  and  $\bar{b}$  with variables  $\bar{z}$  and  $\bar{w}$ . First we obtain  $\mathcal{M} \models \exists \bar{w} \psi(\bar{a}, \bar{w})$ . Now consider the formula  $\psi(\bar{z}, \bar{w}) \rightarrow \exists \bar{y} \varphi(\bar{x}, \bar{y}, \bar{z})$ . We can rewrite the formula to the equivalent statement  $\exists \bar{y} (\psi(\bar{z}, \bar{w}) \rightarrow \varphi(\bar{x}, \bar{y}, \bar{z}))$ . Because this existential formula is valid, we can apply Herbrand's Theorem. So there is an integer  $s$  and terms  $t_i^j(\bar{x}, \bar{z}, \bar{w})$  with  $1 \leq i \leq r$  and  $1 \leq j \leq s$  such that

$$\models \psi(\bar{z}, \bar{w}) \rightarrow \bigvee_{j=1}^s \varphi(\bar{x}, t_1^j(\bar{x}, \bar{z}, \bar{w}), t_2^j(\bar{x}, \bar{z}, \bar{w}), \dots, t_r^j(\bar{x}, \bar{z}, \bar{w}), \bar{z}).$$

□

**Corollary 2.6.** *Let  $\mathcal{M}$  be an Herbrand-saturated structure for a language  $\mathcal{L}$ . Suppose  $\mathcal{M} \models \forall \bar{x} \exists \bar{y} \varphi(\bar{x}, \bar{y}, \bar{a})$  where  $\varphi$  is quantifier-free and  $\bar{a}$  is a tuple of parameters in  $\mathcal{M}$  and  $\bar{y} = (y_1, \dots, y_r)$ . Then there is an integer  $s$ , terms  $t_i^j(\bar{x}, \bar{z}, \bar{w})$  with  $1 \leq i \leq r$  and  $1 \leq j \leq s$  and a tuple of parameters  $\bar{b} \in \mathcal{M}$*

such that

$$\mathcal{M} \models \forall \bar{x} \bigvee_{j=1}^s \varphi(\bar{x}, t_1^j(\bar{x}, \bar{a}, \bar{b}), t_2^j(\bar{x}, \bar{a}, \bar{b}), \dots, t_r^j(\bar{x}, \bar{a}, \bar{b}), \bar{a})$$

for an  $s \in \mathbb{N}$ .

*Proof.* This follows from the proof of Proposition 2.5 by using the same tuple  $\bar{b}$ .  $\square$

**Corollary 2.7.** *Let  $T$  be a universal theory which supports definition by cases. Assume  $\mathcal{M}$  is an Herbrand-saturated model of  $T$  and suppose  $\mathcal{M} \models \forall \bar{x} \exists \bar{y} \varphi(\bar{x}, \bar{y}, \bar{a})$  where  $\varphi$  is quantifier-free and  $\bar{a}$  is a tuple of parameters in  $\mathcal{M}$  and  $\bar{y} = (y_1, \dots, y_r)$ .*

*Then there are function symbols  $F_1(\bar{x}, \bar{z}, \bar{w}), \dots, F_r(\bar{x}, \bar{z}, \bar{w})$  and a tuple of parameters  $\bar{b}$  in  $\mathcal{M}$  such that  $\mathcal{M} \models \forall \bar{x} \varphi(\bar{x}, F_1(\bar{x}, \bar{a}, \bar{b}), \dots, F_r(\bar{x}, \bar{a}, \bar{b}), \bar{a})$ .*

*Proof.* Corollary 2.6 implies that there are terms  $t_i^j(\bar{x}, \bar{z}, \bar{w})$  with  $1 \leq i \leq r$  and  $1 \leq j \leq s$  for an integer  $s$  and that there is a tuple of parameters  $\bar{b} \in \mathcal{M}$  such that

$$\mathcal{M} \models \forall \bar{x} \bigvee_{j=1}^s \varphi(\bar{x}, t_1^j(\bar{x}, \bar{a}, \bar{b}), t_2^j(\bar{x}, \bar{a}, \bar{b}), \dots, t_r^j(\bar{x}, \bar{a}, \bar{b}), \bar{a})$$

for an  $s \in \mathbb{N}$ . Now define  $F_i$  as in the proof of Corollary 2.2.  $\square$

## 2.3 Buss's Witnessing Theorem

In this section we apply previous results to show a model-theoretic version of Buss's Witnessing Theorem [3]. The difference between our approach and the original proof is that in [3] the result is shown by using proof-theoretic methods. For a proof-theoretic discussion we refer to [4].



We will not show the exact same result proven by Buss, but the theorem we prove is similar to his statement. The definitions of this section are based on [4] and [8] which provide more details on bounded arithmetic.

### 2.3.1 Defining $UPV_i$

First of all, we define the proper theories for proving this result. We start with the theory of polynomial time computable functions with  $\Sigma_{i-1}^p$ -oracles.

**Definition.**

- Let  $FP_1$  be the set of polynomial time functions.
- For  $i \geq 2$ , let  $FP_i$  be the set of polynomial time functions with  $\Sigma_{i-1}^p$ -oracle.
- Let  $i \geq 1$ . The theory  $UPV_i$  is the universal theory in the language

$$\mathcal{L}_{PV_i} = \{<\} \cup FP_i$$

*consisting of all true universal  $\mathcal{L}_{PV_i}$ -sentences in the expansion of the standard model to  $\mathcal{L}_{PV_i}$  where all  $FP_i$ -functions are interpreted the common way.*

Now we show that  $UPV_i$  supports definition by cases. Intuitively this is clear: We have a list of  $FP_i$ -functions to choose from and depending on the output of the characteristic functions of quantifier-free formulas, we compute the output of the function we want to choose. Thus, we need to show the next lemma first.

**Lemma 2.8.** *Let  $i \geq 1$ . Every quantifier-free formula  $\varphi(\bar{x})$  in the language  $\mathcal{L}_{PV_i}$  has a characteristic function  $\chi_\varphi(\bar{x})$  in  $FP_i$ , i.e., there is a function  $\chi_\varphi(\bar{x}) \in FP_i$  such that  $UPV \models \forall \bar{x} (\varphi(\bar{x}) \leftrightarrow \chi_\varphi(\bar{x}) = 1)$ .*

*Proof.* The interpretation of every  $\mathcal{L}_{\text{PV}_i}$ -term in the standard model defines a function that is in  $\text{FP}_i$ , because  $\text{FP}_i$  is closed under composition. The equality can be stated as a universal sentence, and therefore it holds in all models of  $\text{UPV}_i$ . Since the relations  $=$  and  $<$  can be checked in polynomial time, the lemma holds for atomic formulas. It is clear, that the formulas for which the lemma holds are closed under conjunctions and negations.  $\square$

**Proposition 2.9.** *Let  $i \geq 1$ . The theory  $\text{UPV}_i$  supports definition by cases.*

*Proof.* By definition, we have to show that for every finite number of terms  $t_1(\bar{x}), \dots, t_k(\bar{x})$  and quantifier-free formulas  $\theta_1(\bar{x}), \dots, \theta_{k-1}(\bar{x})$  there is a function symbol  $F$  such that:

$$\text{UPV}_i \models \forall \bar{x} F(\bar{x}) = \begin{cases} t_1(\bar{x}) & \text{if } \theta_1(\bar{x}) \\ t_2(\bar{x}) & \text{if } \neg\theta_1(\bar{x}) \wedge \theta_2(\bar{x}) \\ \vdots & \\ t_{k-1}(\bar{x}) & \text{if } \bigwedge_{i=1}^{k-2} \neg\theta_i(\bar{x}) \wedge \theta_{k-1}(\bar{x}) \\ t_k(\bar{x}) & \text{otherwise.} \end{cases}$$

Since all  $\theta_j (1 \leq j < k)$  are quantifier-free, by Lemma 2.8 they have a characteristic function  $\chi_{\theta_j}$  in  $\text{FP}_i$ . Furthermore, from the proof of Lemma 2.8 we obtained that every term  $t_j$  defines an  $\text{FP}_i$ -function  $f_j$ .

Let  $F(\bar{x})$  be computed by the algorithm below (with algorithms  $\mathbb{A}_j$  computing  $\chi_{\theta_j}$  and  $\mathbb{B}_j$  the algorithm computing the  $\text{FP}_i$ -function  $f_j$ ).

Every algorithm that is run in the algorithm computing  $F(\bar{x})$  computes its output in polynomial time. Therefore, we run at most  $k$ -many polynomial time algorithms for the characteristic functions and one for the term. Adding the time of all those computations is adding polynomials. Thus, we get another polynomial – so the algorithm runs in polynomial time. Since all algorithms are allowed to make  $\Sigma_{i-1}^p$ -oracle queries, we obtain that overall this

algorithm runs in polynomial time and makes  $\Sigma_{i-1}^p$ -oracle queries. Therefore,  $F(\bar{x})$  is an  $\text{FP}_i$ -function.

```

Input:  $\bar{x}$ 
if  $\mathbb{A}_1(\bar{x}) = 1$  then
   $output \leftarrow \mathbb{B}_1(\bar{x})$ 
  return  $output$ 
else if  $\mathbb{A}_2(\bar{x}) = 1$  then
   $output \leftarrow \mathbb{B}_2(\bar{x})$ 
  return  $output$ 
   $\vdots$ 
else if  $\mathbb{A}_{k-1}(\bar{x}) = 1$  then
   $output \leftarrow \mathbb{B}_{k-1}(\bar{x})$ 
  return  $output$ 
else
   $output \leftarrow \mathbb{B}_k(\bar{x})$ 
  return  $output$ 
end if

```

□

Now we apply the theorems we proved in the previous section. First of all we can apply *Herbrand's Theorem*. Since  $\text{UPV}_i$  supports definition by cases, we get the following corollary.

**Corollary 2.10.** *Let  $i \geq 1$ . If  $\text{UPV}_i \models \forall \bar{x} \exists \bar{y} \psi(\bar{x}, \bar{y})$  and  $\psi(\bar{x}, \bar{y})$  is a quantifier-free formula where  $\bar{y} = (y_1, \dots, y_r)$ , then there are function symbols  $F_1(\bar{x}), \dots, F_r(\bar{x})$  in  $\text{FP}_i$  such that  $\text{UPV}_i \models \forall \bar{x} \psi(\bar{x}, F_1(\bar{x}), \dots, F_r(\bar{x}))$ .*

*Proof.* By Proposition 2.9,  $\text{UPV}_i$  supports definition by cases. Since it is a universal theory by definition, we can apply Corollary 2.2. □

Clearly the application of *Herbrand's Theorem*, namely Proposition 2.5 works for  $\text{UPV}_i$ .

**Corollary 2.11.** *Let  $i \geq 1$ . Assume  $\mathcal{M}$  is an Herbrand-saturated model of  $\text{UPV}_i$  and  $\mathcal{M} \models \forall \bar{x} \exists \bar{y} \psi(\bar{x}, \bar{y}, \bar{a})$  where  $\psi(\bar{x}, \bar{y}, \bar{z})$  is quantifier-free,  $\bar{y} = (y_1, \dots, y_r)$  and  $\bar{a}$  is a tuple of parameters in  $\mathcal{M}$ . Then there exist function symbols  $F_1(\bar{x}, \bar{z}, \bar{w}), \dots, F_r(\bar{x}, \bar{z}, \bar{w})$  in  $\text{FP}_i$  and parameters  $\bar{b} \in \mathcal{M}$  such that  $\mathcal{M} \models \forall \bar{x} \psi(\bar{x}, F_1(\bar{x}, \bar{a}, \bar{b}), \dots, F_r(\bar{x}, \bar{a}, \bar{b}), \bar{a})$ .*

*Proof.* Since  $\text{UPV}_i$  supports definition by cases and is a universal theory by definition, we can apply Corollary 2.7.  $\square$

### 2.3.2 Defining $\text{US}_2^1(\text{FP}_i)$

Next we define a weak fragment of arithmetic. We start with the language of arithmetic and add a unary symbol  $\lfloor \frac{x}{2} \rfloor$  for cutting off the last digit in the binary representation, a unary symbol  $|x|$  for the length of a number which is equal to  $\lceil \log_2(x+1) \rceil$  and the binary smash symbol  $x \# y$  defined as  $2^{|x| \cdot |y|}$ .

**Definition.** *The language of bounded arithmetic is*

$$\mathcal{L}_{BA} = \left\{ 0, 1, +, \cdot, <, \left\lfloor \frac{x}{2} \right\rfloor, |x|, \# \right\}.$$

**Note.** *It is clear that  $0, 1, +, \cdot, <, \lfloor \frac{x}{2} \rfloor, |x|$  and  $\#$  are in  $\text{FP}_1$ . Thus, we obtain that  $\mathcal{L}_{BA} \subseteq \mathcal{L}_{\text{PV}_i}$  for all  $i \geq 1$ .*

**Note.** *We use the common abbreviation " $x \leq y$ " for " $x < y \vee x = y$ ".*

**Definition.** *Let  $i \geq 1$ . The set of  $\Delta_0^b(\text{FP}_i)$ -formulas is the set of formulas satisfying:*

- *If  $\varphi(\bar{x})$  is a quantifier-free  $\mathcal{L}_{\text{PV}_i}$ -formula, then  $\varphi(\bar{x}) \in \Delta_0^b(\text{FP}_i)$ .*
- *If  $\varphi(\bar{x})$  and  $\psi(\bar{x})$  are  $\Delta_0^b(\text{FP}_i)$ -formulas, then  $\neg\varphi(\bar{x})$ ,  $\varphi(\bar{x}) \wedge \psi(\bar{x})$  and  $\varphi(\bar{x}) \vee \psi(\bar{x})$  are  $\Delta_0^b(\text{FP}_i)$ -formulas.*
- *If  $\varphi(x, \bar{y})$  is a  $\Delta_0^b(\text{FP}_i)$ -formula and  $t(\bar{y})$  is an  $\mathcal{L}_{\text{PV}_i}$ -term, then  $\exists x \leq |t(\bar{y})| \varphi(x, \bar{y})$  and  $\forall x \leq |t(\bar{y})| \varphi(x, \bar{y})$  are  $\Delta_0^b(\text{FP}_i)$ -formulas.*

The set of  $\Sigma_1^{b,s}(\text{FP}_i)$ -formulas is the set of formulas  $\varphi(\bar{y})$  that are of the form  $\exists x \leq t(\bar{y}) \psi(x, \bar{y})$  where  $\psi(x, \bar{y})$  is a  $\Delta_0^b(\text{FP}_i)$ -formula and  $t(\bar{y})$  is an  $\mathcal{L}_{\text{PV}_i}$ -term.

**Definition.** Let  $\Phi$  be a set of formulas. Length minimization  $\text{LMIN}(\Phi)$  is the set of sentences

$$\forall x, \bar{w} (\varphi(x, \bar{w}) \rightarrow \exists y \leq x \forall z \leq x (\varphi(y, \bar{w}) \wedge (|z| < |y| \rightarrow \neg \varphi(z, \bar{w}))))$$

for all formulas  $\varphi(x, \bar{w}) \in \Phi$ .

**Definition.** Let  $i \geq 1$ . The theory  $\text{US}_2^1(\text{FP}_i)$  is the theory  $\text{UPV}_i$  together with  $\text{LMIN}(\Sigma_1^{b,s}(\text{FP}_i))$ .

**Proposition 2.12.** Let  $i \geq 1$ . In  $\text{UPV}_i$  every  $\Delta_0^b(\text{FP}_i)$ -formula is equivalent to an atomic formula.

*Proof.* By Lemma 2.8, every quantifier-free  $\mathcal{L}_{\text{PV}_i}$ -formula has a characteristic function in  $\text{FP}_i$ . Hence, if  $\varphi(\bar{y})$  is quantifier-free, then  $\text{UPV}_i \models \forall \bar{y} (\varphi(\bar{y}) \leftrightarrow \chi_\varphi(\bar{y}) = 1)$  which proves the claim for quantifier-free formulas.

Assume now that  $\varphi(\bar{y}) = \exists x \leq |t(\bar{y})| \psi(x, \bar{y})$  where  $\psi(x, \bar{y})$  is a quantifier-free formula in the language  $\mathcal{L}_{\text{PV}_i}$  and  $t(\bar{y})$  is a term in this language. First of all, we show that there is an  $\text{FP}_i$ -function  $F(\bar{y})$  such that

$$\text{UPV}_i \models \forall \bar{y} (\exists x \leq |t(\bar{y})| \psi(x, \bar{y}) \leftrightarrow \psi(F(\bar{y}), \bar{y})).$$

We show that the left-hand side implies the right-hand side: Since  $\psi(x, \bar{y})$  is quantifier-free, there is an  $\text{FP}_i$ -function  $\chi_\psi$ , such that its algorithm  $\mathbb{A}$  computes on input  $(x, \bar{y})$  the value of  $\chi_\psi(x, \bar{y})$ . Let  $F(\bar{y})$  be the function with the following algorithm:

```

Input:  $\bar{y}$ 
 $z \leftarrow t(\bar{y})$ 
 $u \leftarrow |z|$ 
 $x \leftarrow 0$ 
while  $x \leq u$  do
  if  $\mathbb{A}(x, \bar{y}) = 1$  then
     $output \leftarrow x$ 
    return output
  else
     $x \leftarrow x + 1$ 
  end if
end while
return output

```

Let  $\bar{y} = (y_1, \dots, y_r)$ . Computing  $t(\bar{y})$  can be done in less than  $p(|y_1|, \dots, |y_r|)$  steps with  $p$  being a polynomial, because all  $\mathcal{L}_{\text{UPV}_i}$ -terms define  $\text{FP}_i$ -functions. Hence,  $|t(\bar{y})| = |z| \leq p(|y_1|, \dots, |y_r|)$  and therefore,  $|z|$  is computable in polynomial time. Now the loop computes the characteristic function of  $\psi$  at most  $|z|$ -many times. Since this function is in  $\text{FP}_i$ , there is a polynomial  $p'$  such that this function is computable in time  $p'(|x|, |y_1|, \dots, |y_r|)$ . Since  $x \leq |z|$ , we obtain that  $|x| \leq |z|$  also holds. Combining these facts, we obtain that this algorithm needs polynomial time to compute. Note that, if  $i > 1$ , in all the algorithms of  $t(\bar{y})$  and in  $\mathbb{A}$ ,  $\Sigma_{i-1}^p$ -oracle queries are allowed. Hence, in the algorithm above, queries to  $\Sigma_{i-1}^p$ -oracles are allowed as well. In conclusion, whether  $i = 1$  or  $i > 1$ ,  $F(\bar{y})$  is an  $\text{FP}_i$ -function. The other implication is trivial.

Since  $\psi(F(\bar{y}), \bar{y})$  is quantifier-free, we obtain that it is equivalent to an atomic formula in models of  $\text{UPV}_i$ .  $\square$

### 2.3.3 Conservation of $\text{US}_2^1(\text{FP}_i)$ over $\text{UPV}_i$

**Notation.** Let  $i \geq 1$ . We write  $\mathfrak{N}_{\text{PV}_i}$  for the expansion of the standard model  $\mathfrak{N}$  to  $\mathcal{L}_{\text{PV}_i}$  where all  $\mathcal{L}_{\text{PV}_i}$ -symbols are interpreted the obvious way.

**Theorem 2.13.** Let  $i \geq 1$ . The theory  $\text{US}_2^1(\text{FP}_i)$  is  $\forall\exists$ -conservative over  $\text{UPV}_i$ .

*Proof.* By Theorem 2.4 the only thing we need to show is that, if  $\mathcal{M}$  is an Herbrand-saturated model of  $\text{UPV}_i$ , then  $\mathcal{M}$  is a model of  $\text{US}_2^1(\text{FP}_i)$ .

Since  $\text{UPV}_i$  is the theory of true universal sentences in  $\mathfrak{N}_{\text{PV}_i}$ , the only step remaining to prove that  $\mathcal{M} \models \text{US}_2^1(\text{FP}_i)$  is showing that  $\text{LMIN}(\Sigma_1^{b,s}(\text{FP}_i))$  holds in  $\mathcal{M}$ , i.e.,

$$\mathcal{M} \models \varphi(a, \bar{b}) \rightarrow \exists x \leq a \forall y \leq a \left( \varphi(x, \bar{b}) \wedge (|y| < |x| \rightarrow \neg\varphi(y, \bar{b})) \right)$$

for a  $\Sigma_1^{b,s}(\text{FP}_i)$ -formula  $\varphi(x, \bar{u})$  and  $a, \bar{b} \in \mathcal{M}$ . Assume  $\mathcal{M} \models \varphi(a, \bar{b})$  holds and

$$\mathcal{M} \not\models \exists x \leq a \forall y \leq a \left[ \varphi(x, \bar{b}) \wedge (|y| < |x| \rightarrow \neg\varphi(y, \bar{b})) \right].$$

As  $\varphi(x, \bar{u})$  does not involve  $y$ , we can rewrite the formula to get the quantifier inside the parentheses right in front of the first occurrence of  $y$ , i.e.,

$$\mathcal{M} \not\models \exists x \leq a \left[ \varphi(x, \bar{b}) \wedge \forall y \leq a \left( |y| < |x| \rightarrow \neg\varphi(y, \bar{b}) \right) \right].$$

Since  $\varphi(x, \bar{u})$  is a  $\Sigma_1^{b,s}(\text{FP}_i)$ -formula, there exists a term  $t(x, \bar{u})$  not involving  $z$  and a  $\Delta_0^b(\text{FP}_i)$ -formula  $\psi(x, z, \bar{u})$ , such that  $\varphi(x, \bar{u}) = \exists z \leq t(x, \bar{u}) \psi(x, z, \bar{u})$ . By Proposition 2.12, in models of  $\text{UPV}_i$  the formula  $\psi(x, z, \bar{u})$  is equivalent to an atomic formula of the form  $\chi_\psi(x, z, \bar{u}) = 1$  where  $\chi_\psi$  is in  $\text{FP}_i$ . Without loss of generality, we assume  $\psi(x, z, \bar{u})$  is such a formula. Since  $\mathcal{M} \models \text{UPV}_i$ ,

we obtain

$$\begin{aligned} \mathcal{M} \not\models \exists x \leq a \left[ \exists z \leq t(x, \bar{b}) \psi(x, z, \bar{b}) \wedge \right. \\ \left. \forall y \leq a \left( |y| < |x| \rightarrow \neg \exists z' \leq t(y, \bar{b}) \psi(y, z', \bar{b}) \right) \right]. \end{aligned}$$

We can rewrite the statement above by using standard arguments. Hence, the above is equivalent to

$$\begin{aligned} \mathcal{M} \models \forall x, z \exists y, z' \left( x \leq a \rightarrow \left( z \leq t(x, \bar{b}) \rightarrow \left( y \leq a \wedge z' \leq t(y, \bar{b}) \wedge \right. \right. \right. \\ \left. \left. \left. \left[ \psi(x, z, \bar{b}) \rightarrow (|y| < |x| \wedge \psi(y, z', \bar{b})) \right] \right) \right) \right). \end{aligned}$$

This is a statement of the form  $\mathcal{M} \models \forall x, z \exists y, z' \theta(x, z, y, z', \bar{b})$  where the formula  $\theta(x, z, y, z', \bar{b})$  is quantifier-free. By Corollary 2.11, there exist functions  $G_1(x, z, \bar{u}, \bar{v})$  and  $G_2(x, z, \bar{u}, \bar{v})$  in  $\text{FP}_i$  and a tuple of parameters  $\bar{c}$  in  $\mathcal{M}$  such that

$$\begin{aligned} \mathcal{M} \models \forall x, z \left( x \leq a \rightarrow \left( z \leq t(x, \bar{b}) \rightarrow \left( G_1(x, z, \bar{b}, \bar{c}) \leq a \wedge \right. \right. \right. \\ \left. \left. \left. G_2(x, z, \bar{b}, \bar{c}) \leq t(G_1(x, z, \bar{b}, \bar{c}), \bar{b}) \wedge [\psi(x, z, \bar{b}) \rightarrow \right. \right. \right. \\ \left. \left. \left. (|G_1(x, z, \bar{b}, \bar{c})| < |x| \wedge \psi(G_1(x, z, \bar{b}, \bar{c}), G_2(x, z, \bar{b}, \bar{c}), \bar{b})) \right] \right) \right) \right). \end{aligned}$$

Putting the abbreviations  $\forall x \leq a$  and  $\forall z \leq t(x, \bar{b})$  in place, we obtain the statement

$$\begin{aligned} \mathcal{M} \models \forall x \leq a \forall z \leq t(x, \bar{b}) \left( G_1(x, z, \bar{b}, \bar{c}) \leq a \wedge \right. \\ \left. G_2(x, z, \bar{b}, \bar{c}) \leq t(G_1(x, z, \bar{b}, \bar{c}), \bar{b}) \wedge [\psi(x, z, \bar{b}) \rightarrow \right. \\ \left. (|G_1(x, z, \bar{b}, \bar{c})| < |x| \wedge \psi(G_1(x, z, \bar{b}, \bar{c}), G_2(x, z, \bar{b}, \bar{c}), \bar{b})) \right] \right). \quad (\star) \end{aligned}$$

Assume the algorithms  $\mathbb{A}_1$  and  $\mathbb{A}_2$  compute on input  $(x, z, \bar{u}, \bar{v})$  the output of  $G_1(x, z, \bar{u}, \bar{v})$  and  $G_2(x, z, \bar{u}, \bar{v})$  respectively. Let  $\mathbb{B}_\psi$  be the algorithm com-



puting  $\chi_\psi(x, z, \bar{u})$ . Because  $t(x, \bar{u})$  is a term in  $\mathcal{L}_{\text{PV}_i}$ , by the proof of Lemma 2.8 it defines an  $\text{FP}_i$ -function. Assume  $\mathbb{B}_t(x, \bar{u})$  is the algorithm computing on input  $(x, \bar{u})$  the output of this function. Let  $F(x, z, \bar{u}, \bar{v})$  be the function that is computed by the algorithm below.

```

Input:  $x, z, \bar{u}, \bar{v}$ 
  if  $\mathbb{B}_\psi(x, z, \bar{u}) = 0$  then
     $w \leftarrow 1$ 
  else
     $w \leftarrow 0$ 
  end if
  while  $w = 0$  do
     $x' \leftarrow \mathbb{A}_1(x, z, \bar{u}, \bar{v})$ 
     $z' \leftarrow \mathbb{A}_2(x, z, \bar{u}, \bar{v})$ 
    if  $\mathbb{B}_\psi(x', z', \bar{u}) = 0 \vee |x| \leq |x'| \vee z' > \mathbb{B}_t(x', \bar{u})$  then
       $w \leftarrow 1$ 
    else
       $x \leftarrow x'$ 
       $z \leftarrow z'$ 
    end if
  end while
   $output \leftarrow \langle x, z \rangle$ 
  return output

```

The algorithms  $\mathbb{A}_1, \mathbb{A}_2, \mathbb{B}_\psi$  and  $\mathbb{B}_t$  compute their output in polynomial time because they are algorithms of  $\text{FP}_i$ -functions. Hence, there is a polynomial bounding the computation time of one run of the *while*-loop. The *while*-loop is computed at most  $|x|$ -many times because every time we run this loop, we check if the reassigned  $x'$  has the property that  $|x'|$  is smaller than in the previous run of the *while*-loop, starting with  $x'$  being the input  $x$ . Hence, the computation time of the algorithm is bounded by a polynomial as well. Thus,  $F(x, z, \bar{u}, \bar{v})$  is an  $\text{FP}_i$ -function.

Let  $F_x(x, z, \bar{u}, \bar{v})$  be the function that computes  $F(x, z, \bar{u}, \bar{v})$  first and then decodes the first entry of the pair that  $F$  outputs on input  $(x, z, \bar{u}, \bar{v})$ . Analogously, let  $F_z(x, z, \bar{u}, \bar{v})$  be the decoded second entry of  $F(x, z, \bar{u}, \bar{v})$ . Since  $F$  is an  $\text{FP}_i$ -function and the decoding can be done in polynomial time, the functions  $F_x$  and  $F_z$  are in  $\text{FP}_i$  as well.

**Claim 1.**  $\mathfrak{N}_{\text{PV}_i} \models \forall x, \bar{u}, \bar{v} \forall z \leq t(x, \bar{u}) F_x(x, z, \bar{u}, \bar{v}) \leq x$

*Proof.* Let  $a_x, a_z \in \mathbb{N}$  and  $\bar{e}_u$  and  $\bar{e}_v$  be  $\mathbb{N}$ -tuples such that  $a_z \leq t(a_x, \bar{e}_u)$ . Either  $F_x(a_x, a_z, \bar{e}_u, \bar{e}_v) = a_x$ , or, if the algorithm runs the *while*-loop at least twice,  $|F_x(a_x, a_z, \bar{e}_u, \bar{e}_v)| < |a_x|$  holds. ■

**Claim 2.**  $\mathfrak{N}_{\text{PV}_i} \models \forall x, \bar{u}, \bar{v} \forall z \leq t(x, \bar{u}) F_z(x, z, \bar{u}, \bar{v}) \leq t(F_x(x, z, \bar{u}, \bar{v}), \bar{u})$

*Proof.* Let  $a_x, a_z \in \mathbb{N}$  and  $\bar{e}_u$  and  $\bar{e}_v$  be  $\mathbb{N}$ -tuples such that  $a_z \leq t(a_x, \bar{e}_u)$ . Analogously to Claim 1, we obtain that either  $F_x(a_x, a_z, \bar{e}_u, \bar{e}_v) = a_x$  and  $F_z(a_x, a_z, \bar{e}_u, \bar{e}_v) = a_z$ , which by assumption is  $\leq t(a_x, \bar{e}_u)$ , or we have that  $F_z(a_x, a_z, \bar{e}_u, \bar{e}_v) \leq t(F_x(a_x, a_z, \bar{e}_u, \bar{e}_v), \bar{e}_u)$  which follows from the algorithm. ■

The next claim is obvious.

**Claim 3.**

$\mathfrak{N}_{\text{PV}_i} \models \forall x, \bar{u}, \bar{v} \forall z \leq t(x, \bar{u}) (\psi(x, z, \bar{u}) \rightarrow \psi(F_x(x, z, \bar{u}, \bar{v}), F_z(x, z, \bar{u}, \bar{v}), \bar{u}))$

In the following claim, we abbreviate  $F_x$  for  $F_x(x, z, \bar{u}, \bar{v})$  and  $F_z$  for  $F_z(x, z, \bar{u}, \bar{v})$  for better readability. Note, that in the claim  $x, z, \bar{u}, \bar{v}$  are in the scope of the universal quantifiers.

**Claim 4.**

$$\begin{aligned} \mathfrak{N}_{\text{PV}_i} \models \forall x, \bar{u}, \bar{v} \forall z \leq t(x, \bar{u}) & (|G_1(F_x, F_z, \bar{u}, \bar{v})| \geq |F_x| \vee \\ & t(G_1(F_x, F_z, \bar{u}, \bar{v}), \bar{u}) < G_2(F_x, F_z, \bar{u}, \bar{v}) \vee \\ & \neg \psi(G_1(F_x, F_z, \bar{u}, \bar{v}), G_2(F_x, F_z, \bar{u}, \bar{v}), \bar{u})) \end{aligned}$$

*Proof.* Assume otherwise, then the *while*-loop would not have stopped, since these three disjuncts are equivalent to the three conditions to run the *while*-loop again.  $\blacksquare$

All the statements of the four claims above are universal and true in  $\mathfrak{N}_{\text{PV}_i}$  and therefore, they hold in  $\mathcal{M}$  as well.

By assumption,  $\mathcal{M} \models \varphi(a, \bar{b})$ . Thus, there is a  $d \in \mathcal{M}$  with  $\mathcal{M} \models d \leq t(a, \bar{b})$  such that  $\mathcal{M} \models \psi(a, d, \bar{b})$ . Let  $a_0 = F_x^{\mathcal{M}}(a, d, \bar{b}, \bar{c})$  and  $d_0 = F_z^{\mathcal{M}}(a, d, \bar{b}, \bar{c})$ . Then the following holds:

- From Claim 1, we obtain that  $\mathcal{M} \models a_0 \leq a$ .
- From Claim 2, we obtain that  $\mathcal{M} \models d_0 \leq t(a_0, \bar{b})$ .
- From Claim 3, we obtain that  $\mathcal{M} \models \psi(a_0, d_0, \bar{b})$ .
- From Claim 4, we obtain that

$$\mathcal{M} \models |G_1(a_0, d_0, \bar{b}, \bar{c})| \geq |a_0| \vee t(G_1(a_0, d_0, \bar{b}, \bar{c}), \bar{b}) < G_2(a_0, d_0, \bar{b}, \bar{c}) \vee \neg \psi(G_1(a_0, d_0, \bar{b}, \bar{c}), G_2(a_0, d_0, \bar{b}, \bar{c}), \bar{b}).$$

Thus, the statement  $(\star)$  does not hold in  $\mathcal{M}$  for  $x = a_0$  and  $z = d_0$ , which is a contradiction.  $\square$

# Chapter 3

## Simulation of propositional refutations

In the first two sections of this chapter we introduce propositional formulas and refutations. After that we define how we translate first-order formulas into propositional formulas. In the subsequent sections, we show how we code propositional translations and refutations. The last section of this chapter focuses on proving the main theorem of this part which can be understood as follows: Let  $\mathcal{L}$  be the language of all definable functions in the standard model with new symbols added, let  $\varphi$  be an  $\mathcal{L}$ -sentence and let  $\mathcal{M}$  be a countable nonstandard model of arithmetic. If there is an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  where  $\varphi$  is true on an initial segment, then there are no polynomial size refutations of the propositional translations of  $\varphi$ . We postpone further details.

### 3.1 Propositional formulas

**Definition.** Propositional formulas *are built from* propositional variables and connectives  $\wedge$ ,  $\vee$  and  $\neg$  *using the following rules:*

- *A propositional variable  $p$  and its negation  $\neg p$  are propositional formulas called literals.*

- If  $\Phi$  is a finite set of propositional formulas, then  $\bigwedge \Phi$  is a propositional formula called conjunction.
- If  $\Phi$  is a finite set of propositional formulas, then  $\bigvee \Phi$  is a propositional formula called disjunction.

**Notation.** We write  $\top$  for  $\bigwedge \emptyset$  and  $\perp$  for  $\bigvee \emptyset$ .

**Definition.** If  $\varphi$  is a propositional formula, then we define the propositional formula  $\neg\varphi$  recursively:

- If  $\varphi = p$  for a propositional variable  $p$ , then  $\neg\varphi = \neg p$ .
- If  $\varphi = \neg p$  for a propositional variable  $p$ , then  $\neg\varphi = p$ .
- If  $\varphi = \bigwedge \Phi$  where  $\Phi$  is a finite set of propositional formulas, then  $\neg\varphi = \bigvee \{\neg\psi \mid \psi \in \Phi\}$ .
- If  $\varphi = \bigvee \Phi$  where  $\Phi$  is a finite set of propositional formulas, then  $\neg\varphi = \bigwedge \{\neg\psi \mid \psi \in \Phi\}$ .

**Note.** By definition  $\neg\perp = \top$  and  $\neg\top = \perp$ .

**Notation.** We use the common abbreviation  $\varphi \wedge \psi$  and  $\varphi \vee \psi$  for propositional formulas  $\varphi$  and  $\psi$ . Those abbreviations should be seen as functions with the following meaning.

Let  $\varphi$  and  $\psi$  be propositional formulas and let  $\Phi$  and  $\Psi$  be finite sets of propositional formulas. Then

$$\varphi \wedge \psi = \begin{cases} \varphi & \text{if } \psi = \top \\ \psi & \text{if } \varphi = \top \\ \bigwedge (\Phi \cup \Psi) & \text{if } \varphi = \bigwedge \Phi \text{ and } \psi = \bigwedge \Psi \text{ where } \Phi \neq \emptyset \text{ and } \Psi \neq \emptyset \\ \bigwedge (\Phi \cup \{\psi\}) & \text{if } \varphi = \bigwedge \Phi \text{ with } \Phi \neq \emptyset \text{ and } \psi \text{ is not a conjunction} \\ \bigwedge (\{\varphi\} \cup \Psi) & \text{if } \varphi \text{ is not a conjunction and } \psi = \bigwedge \Psi \text{ with } \Psi \neq \emptyset \\ \bigwedge \{\varphi, \psi\} & \text{if } \varphi \text{ and } \psi \text{ are not conjunctions} \end{cases}$$

and

$$\varphi \vee \psi = \begin{cases} \varphi & \text{if } \psi = \perp \\ \psi & \text{if } \varphi = \perp \\ \bigvee (\Phi \cup \Psi) & \text{if } \varphi = \bigvee \Phi \text{ and } \psi = \bigvee \Psi \text{ where } \Phi \neq \emptyset \text{ and } \Psi \neq \emptyset \\ \bigvee (\Phi \cup \{\psi\}) & \text{if } \varphi = \bigvee \Phi \text{ with } \Phi \neq \emptyset \text{ and } \psi \text{ is not a disjunction} \\ \bigvee (\{\varphi\} \cup \Psi) & \text{if } \varphi \text{ is not a disjunction and } \psi = \bigvee \Psi \text{ with } \Psi \neq \emptyset \\ \bigvee \{\varphi, \psi\} & \text{if } \varphi \text{ and } \psi \text{ are not disjunctions} \end{cases}$$

**Note.** It is easy to see, that  $\varphi \vee \psi = \neg(\neg\varphi \wedge \neg\psi)$  holds for all propositional formulas  $\varphi$  and  $\psi$ .

**Notation.** Because of the associativity of  $\wedge$  and  $\vee$ , we can iterate the notations above for 3 or more propositional formulas. For  $r$ -many propositional formulas  $\varphi_1, \dots, \varphi_r$  we write  $\bigwedge_{1 \leq i \leq r} \varphi_i$  instead of  $(\dots(\varphi_1 \wedge \varphi_2) \wedge \dots) \wedge \varphi_r$  and  $\bigvee_{1 \leq i \leq r} \varphi_i$  instead of  $(\dots(\varphi_1 \vee \varphi_2) \vee \dots) \vee \varphi_r$ .

**Definition.** If  $\varphi$  is a propositional formula, we define the set of subformulas  $sub_\varphi$  of  $\varphi$  recursively:

- If  $\varphi$  is a propositional variable  $p$ , then  $sub_\varphi = \{p\}$ .
- If  $\varphi$  is a negated propositional variable  $\neg p$ , then  $sub_\varphi = \{p, \neg p\}$ .
- If  $\varphi$  is of the form  $\bigwedge \Phi$  or  $\bigvee \Phi$  for a finite set of propositional formulas  $\Phi$ , then we define

$$sub_\varphi = \{\varphi\} \cup \bigcup_{\psi \in \Phi} sub_\psi.$$

**Definition.** The size of a propositional formula  $\varphi$  denoted by  $size_{\text{Fml}}(\varphi)$  is defined as follows.

- If  $\varphi = p$  for a propositional variable  $p$ , then  $size_{\text{Fml}}(\varphi) = 1$ .

- If  $\varphi = \neg p$  for a propositional variable  $p$ , then  $\text{size}_{\text{Fml}}(\varphi) = 2$ .
- If  $\varphi = \bigwedge \Phi$  or  $\varphi = \bigvee \Phi$  where  $\Phi$  is a finite set of propositional formulas, then  $\text{size}_{\text{Fml}}(\varphi) = 1 + \sum_{\psi \in \Phi} \text{size}_{\text{Fml}}(\psi)$ .

**Note.** From the definition we obtain  $\text{size}_{\text{Fml}}(\top) = \text{size}_{\text{Fml}}(\perp) = 1$ .

**Definition.** Let  $\varphi$  be a propositional formula. We define the depth of  $\varphi$  denoted by  $\text{dp}(\varphi)$  inductively:

- If  $\varphi \in \{p, \neg p\}$  where  $p$  is a propositional variable, then  $\text{dp}(\varphi) = 0$ .
- If  $\varphi = \bigwedge \Phi$ , then let  $\Phi_0 \subseteq \Phi$  be the set containing all conjunctions in  $\Phi$  and let  $\Phi_1$  be  $\Phi \setminus \Phi_0$ . The depth of a conjunction is defined as

$$\text{dp}(\varphi) = \max \{1 + \max\{\text{dp}(\psi) \mid \psi \in \Phi_1\}, \max\{\text{dp}(\psi) \mid \psi \in \Phi_0\}\}.$$

- If  $\varphi = \bigvee \Phi$ , then let  $\Phi_0 \subseteq \Phi$  be the set containing all disjunctions in  $\Phi$  and let  $\Phi_1$  be  $\Phi \setminus \Phi_0$ . The depth of a disjunction is defined as

$$\text{dp}(\varphi) = \max \{1 + \max\{\text{dp}(\psi) \mid \psi \in \Phi_1\}, \max\{\text{dp}(\psi) \mid \psi \in \Phi_0\}\}.$$

We set  $\max \emptyset = 0$ .

**Note.** By the definition above, we obtain  $\text{dp}(\top) = \text{dp}(\perp) = 1$ .

Further, if  $\varphi$  is a conjunction of depth  $d$ , then  $\neg\varphi$  is a disjunction of depth  $d$  and vice versa.

**Definition.** We define the sets of propositional formulas  $\Sigma_i^{\text{prop}}$  and  $\Pi_i^{\text{prop}}$  for  $i \in \mathbb{N}$  recursively. Let  $\mathbb{V}$  be a set of propositional variables. Then let  $\text{Lit} = \mathbb{V} \cup \{\neg p \mid p \in \mathbb{V}\}$ .

- $\Sigma_0^{\text{prop}} = \Pi_0^{\text{prop}} = \text{Lit}$
- If  $\varphi$  is a  $\Pi_i^{\text{prop}}$ -formula, then  $\varphi$  is a  $\Sigma_{i+1}^{\text{prop}}$ -formula.

- If  $\Phi$  is a set of  $\Pi_i^{prop}$ -formulas containing no disjunction, then  $\bigvee \Phi$  is a  $\Sigma_{i+1}^{prop}$ -formula.
- If  $\varphi$  is a  $\Sigma_i^{prop}$ -formula, then  $\varphi$  is a  $\Pi_{i+1}^{prop}$ -formula.
- If  $\Phi$  is a set of  $\Sigma_i^{prop}$ -formulas containing no conjunction, then  $\bigwedge \Phi$  is a  $\Pi_{i+1}^{prop}$ -formula.

**Note.** The set  $\text{Lit}$  is the set of all literals built from propositional variables in  $\mathbb{V}$ .

By definition, the propositional formula  $\perp = \bigvee \emptyset$  is in  $\Sigma_1^{prop}$  and  $\top = \bigwedge \emptyset$  is in  $\Pi_1^{prop}$  independent of the choice of  $\mathbb{V}$ .

If we write  $\varphi \vee \psi$  or  $\varphi \wedge \psi$  for  $\Sigma_d^{prop}$ - or  $\Pi_d^{prop}$ -formulas  $\varphi$  and  $\psi$  where  $d \in \mathbb{N}$ , then by the notational comment on page 36, the resulting formula is in  $\Sigma_{d'}^{prop}$ - or  $\Pi_{d'}^{prop}$  for another  $d' \in \mathbb{N}$ .

**Definition.** Let  $k \geq 1$  be an integer.

- A propositional formula is in disjunctive normal form (DNF), if it is in  $\Sigma_2^{prop}$ .
- If  $\bigvee \Phi$  is a DNF and for every  $\bigwedge \Psi \in \Phi$ , the set  $\Psi$  contains at most  $k$  literals, i.e., the cardinality of  $\Psi$  is  $\leq k$ , we call  $\bigvee \Phi$  a  $k$ -DNF.
- A propositional formula is in conjunctive normal form (CNF), if it is in  $\Pi_2^{prop}$ .
- If  $\bigwedge \Phi$  is a CNF and for every  $\bigvee \Psi \in \Phi$ , the set  $\Psi$  contains at most  $k$  literals, i.e., the cardinality of  $\Psi$  is  $\leq k$ , we call  $\bigwedge \Phi$  a  $k$ -CNF.

**Proposition 3.1.** For every  $d \in \mathbb{N}$ , every  $\Sigma_d^{prop}$ -formula and every  $\Pi_d^{prop}$ -formula has depth  $\leq d$ .

*Proof.* Literals have depth 0, hence  $\Sigma_0^{prop}$ -formulas and  $\Pi_0^{prop}$ -formulas have depth  $\leq 0$ .



Assume the claim holds for  $d - 1$ .

Then every  $\Sigma_{d-1}^{prop}$ -formula and every  $\Pi_{d-1}^{prop}$ -formula has depth  $\leq d - 1$ . Every  $\Sigma_d^{prop}$ -formula is either a  $\Pi_{d-1}^{prop}$ -formula or of the form  $\bigvee \Phi$  where  $\Phi$  is a finite set of  $\Pi_{d-1}^{prop}$ -formulas containing no disjunction. Assume the former, then the claim holds by the induction hypothesis. Assume the latter. Then the subset of disjunctions in  $\Phi$  is empty. In the notation of the definition of the depth of a propositional formula, we obtain  $\Phi_0 = \emptyset$  and  $\Phi_1 = \Phi$ . Hence,

$$\text{dp}\left(\bigvee \Phi\right) = \max\{1 + \max\{\text{dp}(\psi) \mid \psi \in \Phi\}, 0\} = 1 + \max\{\text{dp}(\psi) \mid \psi \in \Phi\}.$$

Since every  $\psi \in \Phi$  has depth  $\leq d - 1$ , we obtain that  $\text{dp}(\bigvee \Phi) \leq d$ .

Analogously we can show that  $\text{dp}(\bigwedge \Phi) \leq d$ .  $\square$

**Definition.** Let  $\mathbb{V}$  be a set of propositional variables.

- We define  $\mathbf{T}$  and  $\mathbf{F}$  as truth values. The truth value  $\mathbf{T}$  represents "true" and  $\mathbf{F}$  represents "false".
- An assignment is a function  $\alpha : \mathbb{V} \rightarrow \{\mathbf{T}, \mathbf{F}\}$ .
- Let  $\alpha$  be an assignment of all propositional variables in  $\mathbb{V}$ . The extension  $\bar{\alpha}$  of  $\alpha$  to propositional formulas  $\varphi$  with variables from  $\mathbb{V}$  is defined recursively:
  - If  $\varphi = p$  for a  $p \in \mathbb{V}$ , then  $\bar{\alpha}(\varphi) = \alpha(p)$ .
  - If  $\varphi = \bigwedge \Phi$  where  $\Phi$  is a finite set of propositional formulas, then  $\bar{\alpha}(\varphi) = \mathbf{T}$  if and only if  $\bar{\alpha}(\psi) = \mathbf{T}$  for all  $\psi \in \Phi$ .
  - If  $\varphi = \bigvee \Phi$  where  $\Phi$  is a finite set of propositional formulas, then  $\bar{\alpha}(\varphi) = \mathbf{T}$  if and only if there exists a  $\psi \in \Phi$  such that  $\bar{\alpha}(\psi) = \mathbf{T}$ .
  - If  $\varphi = \neg\varphi'$  for a propositional formula  $\varphi'$ , then  $\bar{\alpha}(\varphi) = \mathbf{T}$  if and only if  $\bar{\alpha}(\varphi') = \mathbf{F}$ .
- We say  $\bar{\alpha}(\varphi)$  is the truth value of  $\varphi$ .

- If  $\bar{\alpha}(\varphi) = \mathbf{T}$  where  $\varphi$  is a propositional formula and  $\bar{\alpha}$  is the extension of an assignment of all propositional variables occurring in  $\varphi$ , then we say " $\bar{\alpha}$  satisfies  $\varphi$ ".
- We say that a propositional formula  $\varphi$  is a tautology, if for all assignments  $\alpha$  their extension  $\bar{\alpha}$  to propositional formulas satisfies  $\varphi$ .
- We say that a propositional formula  $\varphi$  is unsatisfiable if there is no assignment  $\alpha$  such that its extension  $\bar{\alpha}$  to propositional formulas satisfies  $\varphi$ .
- We say that a set of propositional formulas  $\Phi$  is unsatisfiable if and only if there is no assignment  $\alpha$  such that its extension  $\bar{\alpha}$  to propositional formulas satisfies all  $\varphi \in \Phi$  simultaneously.
- We say the propositional formulas  $\varphi$  and  $\psi$  are logically equivalent if and only if for the extension  $\bar{\alpha}$  of any assignment  $\alpha$  of all propositional variables occurring in  $\varphi$  and  $\psi$ ,  $\bar{\alpha}(\varphi) = \bar{\alpha}(\psi)$ , i.e.,  $\varphi$  and  $\psi$  always have the same truth value. We write  $\varphi \equiv \psi$ .

**Note.** We obtain from the definition above that  $\bar{\alpha}(\perp) = \mathbf{F}$  and  $\bar{\alpha}(\top) = \mathbf{T}$  for the extension  $\bar{\alpha}$  of any assignment  $\alpha$ .

## 3.2 Frege refutation systems

**Definition.** Let  $\psi_1, \dots, \psi_r$  be propositional formulas and  $p_1, \dots, p_r$  distinct propositional variables where  $r$  is an integer. A substitution is a function  $\sigma : \{p_1, \dots, p_r\} \rightarrow \{\psi_1, \dots, \psi_r\}$  mapping  $p_i$  to  $\psi_i$  for  $1 \leq i \leq r$ . If  $\varphi$  is a propositional formula, then  $\sigma(\varphi)$  is the propositional formula which results from simultaneously replacing  $p_i$  by  $\psi_i$  for  $1 \leq i \leq r$  in the propositional formula  $\varphi$ .

**Definition.**

- A Frege rule  $R$  is an  $(r + 1)$ -tuple of propositional formulas  $\varphi_0, \dots, \varphi_r$  in propositional variables written as

$$R : \frac{\varphi_1 \ \dots \ \varphi_r}{\varphi_0}$$

such that the propositional formula  $\bigwedge_{1 \leq i \leq r} \varphi_i \rightarrow \varphi_0$  is a tautology. This property is called soundness of  $R$ . We call  $(r + 1)$  the arity of the rule  $R$  and write  $\text{ar}(R) = r + 1$ .

- A Frege rule in which  $r = 0$  is called a Frege axiom scheme.
- For any substitution  $\sigma$  we say the propositional formula  $\sigma(\varphi_0)$  is inferred from propositional formulas  $\sigma(\varphi_1), \dots, \sigma(\varphi_r)$  by  $R$ .
- A finite collection of Frege rules  $\mathcal{F}$  is called a Frege refutation system.
- A Frege refutation from a collection of Frege rules ( $\mathcal{F}$ -refutation) of a set of propositional formulas  $\Phi$  is a finite sequence  $(\psi_0, \dots, \psi_s)$  of propositional formulas such that for  $0 \leq i < s$  either
  - $\psi_i$  is in the set  $\Phi$ , or
  - there are  $\psi_{j_1}, \dots, \psi_{j_r}$  ( $0 \leq j_1, \dots, j_r < i$ ) such that  $\psi_i$  is inferred from  $\psi_{j_1}, \dots, \psi_{j_r}$  by a rule  $R$  in  $\mathcal{F}$  with  $\text{ar}(R) = r + 1$ ,

and  $\psi_s = \perp$ .

- If  $\sigma = (\psi_0, \dots, \psi_s)$  is a Frege refutation, then we say  $(s + 1)$  is the length of  $\sigma$  and write  $\|\sigma\| = s + 1$ .
- The size of a Frege refutation is the sum of all sizes of propositional formulas occurring in the Frege refutation. We write  $\text{size}_{\text{Refut}}(\sigma)$ .

## The Frege refutation systems $\mathcal{F}(d)$

In this chapter we focus on the commonly used Frege refutation systems  $\mathcal{F}(d)$  which we define below.

**Definition.** We define the following three rules.

Let  $d \in \mathbb{N}$  with  $d \geq 1$ .

- Weakening rule: Let  $\varphi$  and  $\psi$  be  $\Sigma_d^{prop}$ -formulas, then the weakening rule for  $\Sigma_d^{prop}$ -formulas is:

$$\frac{\varphi}{\varphi \vee \psi}$$

- Cut rule: Let  $\varphi_1, \varphi_2$  be  $\Sigma_d^{prop}$ -formulas and let  $\psi$  be a  $\Pi_{d-1}^{prop}$ -formula, then the cut rule for  $\Sigma_d^{prop}$ -formulas is:

$$\frac{\varphi_1 \vee \psi \quad \varphi_2 \vee \neg\psi}{\varphi_1 \vee \varphi_2}$$

Let  $d \in \mathbb{N}$  with  $d \geq 2$ .

- $\wedge$ -introduction rule: Let  $\varphi_1$  and  $\varphi_2$  be  $\Sigma_d^{prop}$ -formulas and let  $\psi_1$  and  $\psi_2$  be  $\Pi_{d-1}^{prop}$ -formulas which are not disjunctions. The  $\wedge$ -introduction rule for  $\Sigma_d^{prop}$ -formulas is:

$$\frac{\varphi_1 \vee \psi_1 \quad \varphi_2 \vee \psi_2}{\varphi_1 \vee \varphi_2 \vee (\psi_1 \wedge \psi_2)}$$

**Definition.** Let  $d \in \mathbb{N}$  with  $d \geq 1$ . We define the Frege refutation systems  $\mathcal{F}(d)$  for  $\Sigma_d^{prop}$ -formulas as follows.

- $\mathcal{F}(1)$  consists of
  - the weakening rule for  $d = 1$ , and
  - the cut rule for  $d = 1$ .

- $\mathcal{F}(d)$  for  $d \geq 2$  consists of
  - the weakening rule,
  - the cut rule, and
  - the  $\wedge$ -introduction rule.

**Definition.**

- The Frege refutation system  $\mathcal{F}(1)$  is also called Resolution. We write  $\mathcal{R}$ .
- We call  $\mathcal{F}(2)$  also DNF-Resolution. We write  $\mathcal{R}_{\text{DNF}}$ .
- Let  $k \geq 1$  be an integer. We define  $\mathcal{R}_{\text{DNF}_k}$  as the Frege refutation system consisting of the following three rules.

- Weakening rule: Let  $\varphi$  and  $\psi$  be  $k$ -DNF's, then the weakening rule for  $k$ -DNF's is:

$$\frac{\varphi}{\varphi \vee \psi}$$

- Cut rule: Let  $\varphi_1, \varphi_2$  be  $k$ -DNF's and let  $\psi$  be a  $\Pi_1^{\text{prop}}$ -formula such that  $\varphi_1 \vee \psi$  and  $\varphi_2 \vee \neg\psi$  are  $k$ -DNF's, then the cut rule for  $k$ -DNF's is:

$$\frac{\varphi_1 \vee \psi \quad \varphi_2 \vee \neg\psi}{\varphi_1 \vee \varphi_2}$$

- $\wedge$ -introduction rule: Let  $\varphi_1$  and  $\varphi_2$  be  $k$ -DNF's and let  $\psi_1$  and  $\psi_2$  be  $\Pi_1^{\text{prop}}$ -formulas such that  $\psi_1 \wedge \psi_2$  is a  $k$ -DNF's. The  $\wedge$ -introduction rule for  $k$ -DNF's is:

$$\frac{\varphi_1 \vee \psi_1 \quad \varphi_2 \vee \psi_2}{\varphi_1 \vee \varphi_2 \vee (\psi_1 \wedge \psi_2)}$$

**Note.** The refutation systems defined above are complete in the following sense: There is an  $\mathcal{F}(d)$ -refutation of a set of  $\Sigma_d^{\text{prop}}$ -formulas  $\Phi$  if and only

if  $\Phi$  is unsatisfiable.

Note that the same applies to  $k$ -DNF-Resolution and a set of  $k$ -DNFs  $\Phi$ .

**Note.** The rules introduced above are sound, i.e.,  $\mathcal{F}(d)$  are sound Frege refutation systems for integers  $d \geq 1$ .

### 3.3 Propositional translation

In this section we show how to translate first-order formulas in a language bigger than the language of arithmetic to propositional formulas.

**Definition.** A relation  $R \subseteq \mathbb{N}^r$  is definable in the standard model, if there is a formula  $\psi_R$  in the language of  $\mathfrak{N}$  such that for  $a_1, \dots, a_r \in \mathbb{N}$ , we have  $R(a_1, \dots, a_r)$  if and only if  $\mathfrak{N} \models \psi_R(a_1, \dots, a_r)$ .

A function  $f : \mathbb{N}^r \rightarrow \mathbb{N}$  is definable in the standard model, if there is a formula  $\psi_f$  in the language of  $\mathfrak{N}$  such that for  $a_0, \dots, a_r \in \mathbb{N}$ , we have  $f(a_1, \dots, a_r) = a_0$  if and only if  $\mathfrak{N} \models \psi_f(a_0, \dots, a_r)$ .

**Note.** Every  $\text{FP}_i$ -function is definable in the standard model ([4]).

**Notation.** Let  $\mathcal{L}_{\mathbb{N}}$  be the language of all functions and relations definable in the standard model. For the rest of the chapter we fix a finite language  $\mathcal{L}'$  and let  $\mathcal{L} = \mathcal{L}_{\mathbb{N}} \cup \mathcal{L}'$ .

**Definition.** For an  $\mathcal{L}$ -formula  $\theta(w_1, \dots, w_r)$  we define  $\theta^{<x}(w_1, \dots, w_r)$  recursively:

- If  $\theta(w_1, \dots, w_r)$  is atomic, then  $\theta^{<x}(w_1, \dots, w_r) = \theta(w_1, \dots, w_r)$ .
- If  $\theta(w_1, \dots, w_r) = \neg\psi(w_1, \dots, w_r)$  where  $\psi$  is an  $\mathcal{L}$ -formula, then  $\theta^{<x} = \neg\psi^{<x}(w_1, \dots, w_r)$ .
- If  $\theta(w_1, \dots, w_r) = \theta_1(w_1, \dots, w_r) \wedge \theta_2(w_1, \dots, w_r)$  where  $\theta_1$  and  $\theta_2$  are  $\mathcal{L}$ -formulas, then  $\theta^{<x}(w_1, \dots, w_r) = \theta_1^{<x}(w_1, \dots, w_r) \wedge \theta_2^{<x}(w_1, \dots, w_r)$ .

- If  $\theta(w_1, \dots, w_r) = \forall y \psi(y, w_1, \dots, w_r)$  where  $\psi(y, w_1, \dots, w_r)$  is an  $\mathcal{L}$ -formula, then  $\theta^{<x}(w_1, \dots, w_r) = \forall y < x \psi^{<x}(y, w_1, \dots, w_r)$ .

**Note.** Keep in mind that if we use the abbreviations  $\forall y \leq t$  and  $\exists y \leq t$  for a term  $t$  in  $\theta$ , we obtain the following:

- If  $\theta = \forall y \leq t \psi(y)$  with  $\psi(y)$  an  $\mathcal{L}$ -formula, then  $\theta = \forall y (y \leq t \rightarrow \psi(y))$  and hence,  $\theta^{<x} = \forall y < x (y \leq t \rightarrow \psi^{<x}(y))$ .
- If  $\theta = \exists y \leq t \psi(y)$  with  $\psi(y)$  an  $\mathcal{L}$ -formula, then  $\theta = \exists y (y \leq t \wedge \psi(y))$  and hence,  $\theta^{<x} = \exists y < x (y \leq t \wedge \psi^{<x}(y))$ .

**Notation.** We write  $\text{ar}(f)$  or  $\text{ar}(R)$  for the arity of a function  $f$  or a relation  $R$  respectively.

**Notation.** For every relation  $R \in \mathcal{L}'$  and every function  $f \in \mathcal{L}'$ , we introduce propositional variables  $R_{a_1, \dots, a_{\text{ar}(R)}}$  for every tuple  $(a_1, \dots, a_{\text{ar}(R)}) \in \mathbb{N}^{\text{ar}(R)}$  and  $f_{a_1, \dots, a_{\text{ar}(f)+1}}$  for every tuple  $(a_1, \dots, a_{\text{ar}(f)+1}) \in \mathbb{N}^{\text{ar}(f)+1}$ .

For the rest of the chapter we fix  $\mathbb{V}_{\mathcal{L}'}$  to be the set of all the propositional variables introduced above. We will work with  $\Sigma_d^{\text{prop}}$ - and  $\Pi_d^{\text{prop}}$ -formulas built from this particular  $\mathbb{V}_{\mathcal{L}'}$ .

Next we want to translate  $\mathcal{L}$ -formulas to propositional formulas built from propositional variables in  $\mathbb{V}_{\mathcal{L}'}$ .

**Notation.** An  $\mathcal{L}$ -formula is *unnested*, if all occurring first-order atoms are of the form  $x = y$ ,  $f(x_1, \dots, x_{\text{ar}(f)}) = x_{\text{ar}(f)+1}$  or  $R(x_1, \dots, x_{\text{ar}(R)})$  for a function  $f \in \mathcal{L}$  or a relation  $R \in \mathcal{L}$ .

**Definition.** Let  $\theta(x_1, \dots, x_r)$  be an unnested  $\mathcal{L}$ -formula and let  $m$  be an integer. We define the propositional translation  $\langle \theta(a_1, \dots, a_r) \rangle_{(m)}$  of  $\theta(a_1, \dots, a_r)$  for  $a_1, \dots, a_r \in \mathbb{N}$  by induction on the structure of  $\theta$ .

(T1) If  $\theta(x_1, \dots, x_r)$  is a formula in the language  $\mathcal{L}_{\mathbb{N}}$ , then define the propositional translation as

$$\langle \theta(a_1, \dots, a_r) \rangle_{(m)} = \begin{cases} \top & \text{if } \mathfrak{N} \models \theta^{<m}(a_1, \dots, a_r) \\ \perp & \text{if } \mathfrak{N} \not\models \theta^{<m}(a_1, \dots, a_r). \end{cases}$$

(T2) If  $\theta(x_1, \dots, x_{\text{ar}(R)})$  is an atomic  $\mathcal{L}'$ -formula  $R(x_1, \dots, x_{\text{ar}(R)})$  where  $R$  is a relation in  $\mathcal{L}'$ , then define the propositional translation as

$$\langle \theta(a_1, \dots, a_{\text{ar}(R)}) \rangle_{(m)} = R_{a_1, \dots, a_{\text{ar}(R)}}.$$

(T3) If  $\theta(x_1, \dots, x_{\text{ar}(f)+1})$  is an atomic  $\mathcal{L}'$ -formula  $f(x_1, \dots, x_{\text{ar}(f)}) = x_{\text{ar}(f)+1}$  for a function symbol  $f \in \mathcal{L}'$ , then define the propositional translation as

$$\langle \theta(a_1, \dots, a_{\text{ar}(f)+1}) \rangle_{(m)} = f_{a_1, \dots, a_{\text{ar}(f)+1}}.$$

(T4) If  $\theta(x_1, \dots, x_r)$  is an  $\mathcal{L}$ -formula  $\neg\theta'(x_1, \dots, x_r)$ , then define the propositional translation as

$$\langle \theta(a_1, \dots, a_r) \rangle_{(m)} = \neg \langle \theta'(a_1, \dots, a_r) \rangle_{(m)}.$$

(T5) If  $\theta(x_1, \dots, x_r)$  is an  $\mathcal{L}$ -formula  $\theta_1(x_1, \dots, x_r) \wedge \theta_2(x_1, \dots, x_r)$ , then define the propositional translation as

$$\langle \theta(a_1, \dots, a_r) \rangle_{(m)} = \langle \theta_1(a_1, \dots, a_r) \rangle_{(m)} \wedge \langle \theta_2(a_1, \dots, a_r) \rangle_{(m)}.$$

(T6) If  $\theta(x_1, \dots, x_r)$  is an  $\mathcal{L}$ -formula  $\forall y \theta'(x_1, \dots, x_r, y)$ , then define the propositional translation as

$$\langle \theta(a_1, \dots, a_r) \rangle_{(m)} = \bigwedge_{0 \leq i < m} \langle \theta'(a_1, \dots, a_r, i) \rangle_{(m)}.$$



**Note.** The propositional conjunctions in (T5) and (T6) are as in the notational comment on page 36. Thus, a propositional translation always yields a  $\Sigma_d^{prop}$ - or  $\Pi_d^{prop}$ -formula for some  $d \in \mathbb{N}$ .

For the rest of the section we show how to bound the size of a propositional translation.

**Lemma 3.2.** *Let  $m$  be an integer and  $\varphi, \varphi', \varphi_0, \dots, \varphi_{m-1}$  be propositional formulas. Then the following holds.*

$$(a) \text{ size}_{\text{Fml}}(\neg\varphi) \leq 2 \cdot \text{size}_{\text{Fml}}(\varphi)$$

$$(b) \text{ size}_{\text{Fml}}(\bigwedge_{0 \leq i < m} \varphi_i) \leq 1 + m \cdot \max\{\text{size}_{\text{Fml}}(\varphi_i) \mid 0 \leq i < m\}$$

*Proof.* (a): We show this result by induction on the structure of  $\varphi$ . If  $\varphi$  is a propositional variable  $p$ , then  $\neg\varphi = \neg p$  and thus,  $\text{size}_{\text{Fml}}(\neg\varphi) = 2 \leq 2 \cdot \text{size}_{\text{Fml}}(\varphi)$ . If  $\varphi = \neg p$ , then  $\neg\varphi = p$  and thus,  $\text{size}_{\text{Fml}}(\neg\varphi) = 1 \leq 2 \cdot \text{size}_{\text{Fml}}(\varphi)$ . Now assume that  $\varphi = \bigwedge \Phi$  or  $\varphi = \bigvee \Phi$  for a finite set of propositional formulas  $\Phi$  and assume that the claim holds for all  $\psi$  in  $\Phi$ . Then the size of  $\neg\varphi$  is

$$\begin{aligned} \text{size}_{\text{Fml}}(\neg\varphi) &= 1 + \sum_{\psi \in \Phi} \text{size}_{\text{Fml}}(\neg\psi) \leq 1 + \sum_{\psi \in \Phi} 2 \cdot \text{size}_{\text{Fml}}(\psi) \leq \\ &2 \cdot (1 + \sum_{\psi \in \Phi} \text{size}_{\text{Fml}}(\psi)) = 2 \cdot \text{size}_{\text{Fml}}(\varphi). \end{aligned}$$

(b): This can be shown in a straightforward way:

$$\text{size}_{\text{Fml}}\left(\bigwedge_{0 \leq i < m} \varphi_i\right) \leq 1 + \sum_{i=0}^{m-1} \text{size}_{\text{Fml}}(\varphi_i) \leq 1 + m \cdot \max\{\text{size}_{\text{Fml}}(\varphi_i) \mid 0 \leq i < m\}$$

Note that this is also true for  $m = 0$  since  $\text{size}_{\text{Fml}}(\top) = 1$  and  $m = 1$ .  $\square$

**Proposition 3.3.** *Let  $\theta(x_1, \dots, x_r)$  be an  $\mathcal{L}$ -formula and let  $m$  be a natural number. Then there is an  $l \in \mathbb{N}$  such that for every  $(a_1, \dots, a_r) \in \mathbb{N}^r$  and for*

every  $m \in \mathbb{N}$

$$\text{size}_{\text{Fml}}(\langle \theta(a_1, \dots, a_r) \rangle_{(m)}) \leq (m + 2)^l.$$

*Proof.* We prove this by induction on the structure of  $\theta$ .

- $\mathcal{L}_{\mathbb{N}}$ -formulas:

Assume  $\theta(x_1, \dots, x_r)$  is an  $\mathcal{L}_{\mathbb{N}}$ -formula. Then for  $a_1, \dots, a_r \in \mathbb{N}$ , the propositional translation  $\langle \theta(a_1, \dots, a_r) \rangle_{(m)}$  is in  $\{\top, \perp\}$ . Therefore, we obtain  $\text{size}_{\text{Fml}}(\langle \theta(a_1, \dots, a_r) \rangle_{(m)}) = 1$ , thus  $l = 0$ .

- Atoms:

If  $\theta(x_1, \dots, x_{\text{ar}(R)}) = R(x_1, \dots, x_{\text{ar}(R)})$  for a relation  $R \in \mathcal{L}'$ , then for  $a_1, \dots, a_{\text{ar}(R)} \in \mathbb{N}$ , we have  $\langle \theta(a_1, \dots, a_{\text{ar}(R)}) \rangle_{(m)} = R_{a_1, \dots, a_{\text{ar}(R)}}$  and therefore,  $\text{size}_{\text{Fml}}(\langle \theta(a_1, \dots, a_{\text{ar}(R)}) \rangle_{(m)}) = 1$ . Hence,  $l = 0$ .

If  $\theta(x_1, \dots, x_{\text{ar}(f)+1})$  is a formula  $f(x_1, \dots, x_{\text{ar}(f)}) = x_{\text{ar}(f)+1}$  for a function  $f \in \mathcal{L}'$ , then for  $a_1, \dots, a_{\text{ar}(f)+1} \in \mathbb{N}$  the propositional translation  $\langle \theta(a_1, \dots, a_{\text{ar}(f)+1}) \rangle_{(m)}$  equals  $f_{a_1, \dots, a_{\text{ar}(f)+1}}$  and therefore, we obtain  $\text{size}_{\text{Fml}}(\langle \theta(a_1, \dots, a_{\text{ar}(f)+1}) \rangle_{(m)}) = 1$ . Again,  $l = 0$ .

- Negations:

Let  $\theta(x_1, \dots, x_r) = \neg\theta'(x_1, \dots, x_r)$  where  $\theta'(x_1, \dots, x_r)$  is another  $\mathcal{L}$ -formula. Then for  $a_1, \dots, a_r \in \mathbb{N}$ , the translation  $\langle \theta(a_1, \dots, a_r) \rangle_{(m)}$  of  $\theta(a_1, \dots, a_r)$  is  $\neg\langle \theta'(a_1, \dots, a_r) \rangle_{(m)}$ . By the induction hypothesis there is an  $l' \in \mathbb{N}$  such that

$$\text{size}_{\text{Fml}}(\langle \theta'(a_1, \dots, a_r) \rangle_{(m)}) \leq (m + 2)^{l'}.$$

By Lemma 3.2 (a),  $\text{size}_{\text{Fml}}(\neg\varphi) \leq 2 \cdot \text{size}_{\text{Fml}}(\varphi)$  for any propositional formula  $\varphi$  and therefore,

$$\text{size}_{\text{Fml}}(\neg\langle \theta'(a_1, \dots, a_r) \rangle_{(m)}) \leq 2 \cdot \text{size}_{\text{Fml}}(\langle \theta'(a_1, \dots, a_r) \rangle_{(m)}).$$

Note that the following holds for all  $m \in \mathbb{N}$ :

$$2 \cdot (m + 2)^{l'} \leq (m + 2)^{l'+1}.$$

Putting those facts together, we obtain

$$\text{size}_{\text{Fml}}(\neg\langle\theta'(a_1, \dots, a_r)\rangle_{(m)}) \leq (m + 2)^{l'+1}$$

and therefore,  $l = l' + 1$ .

- Conjunctions:

Let  $\theta(x_1, \dots, x_r) = \theta_1(x_1, \dots, x_r) \wedge \theta_2(x_1, \dots, x_r)$  where  $\theta_1(x_1, \dots, x_r)$  and  $\theta_2(x_1, \dots, x_r)$  are  $\mathcal{L}$ -formulas. Then for integers  $a_1, \dots, a_r$ , we obtain

$$\langle\theta(a_1, \dots, a_r)\rangle_{(m)} = \langle\theta_1(a_1, \dots, a_r)\rangle_{(m)} \wedge \langle\theta_2(a_1, \dots, a_r)\rangle_{(m)}.$$

By the induction hypothesis there are natural numbers  $l_1$  and  $l_2$  such that

$$\text{size}_{\text{Fml}}(\langle\theta_i(a_1, \dots, a_r)\rangle_{(m)}) \leq (m + 2)^{l_i}$$

for  $i \in \{1, 2\}$ . Then by Lemma 3.2 (b),

$$\begin{aligned} \text{size}_{\text{Fml}}(\langle\theta(a_1, \dots, a_r)\rangle_{(m)}) &\leq \\ &1 + 2 \cdot \max \{ \text{size}_{\text{Fml}}(\langle\theta_i(a_1, \dots, a_r)\rangle_{(m)}) \mid i \in \{1, 2\} \}. \end{aligned}$$

Since  $\text{size}_{\text{Fml}}(\langle\theta_1(a_1, \dots, a_r)\rangle_{(m)})$  and  $\text{size}_{\text{Fml}}(\langle\theta_2(a_1, \dots, a_r)\rangle_{(m)})$  only differ in the exponent, we obtain for  $l' = \max\{l_1, l_2\}$  that

$$\text{size}_{\text{Fml}}(\langle\theta(a_1, \dots, a_r)\rangle_{(m)}) \leq 1 + 2 \cdot (m + 2)^{l'}.$$

Therefore, we have

$$\text{size}_{\text{Fml}}(\langle \theta(a_1, \dots, a_r) \rangle_{(m)}) \leq (m+2)^{l'+2}.$$

Thus,  $l = \max\{l_1, l_2\} + 2$ .

• Universal quantifications:

Assume  $\theta(x_1, \dots, x_r) = \forall y \theta'(x_1, \dots, x_r, y)$  where  $\theta'(x_1, \dots, x_r, y)$  is an  $\mathcal{L}$ -formula. For  $a_1, \dots, a_r \in \mathbb{N}$ , the translation is  $\langle \theta(a_1, \dots, a_r) \rangle_{(m)} = \bigwedge_{0 \leq i < m} \langle \theta'(a_1, \dots, a_r, i) \rangle_{(m)}$ . By the induction hypothesis, for every  $i < m$ , there is an  $l_i \in \mathbb{N}$  such that

$$\text{size}_{\text{Fml}}(\langle \theta'(a_1, \dots, a_r, i) \rangle_{(m)}) \leq (m+2)^{l_i}.$$

Applying Lemma 3.2 (b), we obtain

$$\begin{aligned} \text{size}_{\text{Fml}}(\langle \theta(a_1, \dots, a_r) \rangle_{(m)}) &\leq \\ &1 + m \cdot \max \{ \text{size}_{\text{Fml}}(\langle \theta'(a_1, \dots, a_r, i) \rangle_{(m)}) \mid 0 \leq i < m \}. \end{aligned}$$

By the same arguments as before, we can write max in the exponent and obtain

$$\text{size}_{\text{Fml}}(\langle \theta(a_1, \dots, a_r) \rangle_{(m)}) \leq 1 + m \cdot (m+2)^{\max\{l_i \mid 0 \leq i < m\}}.$$

Let  $l' = \max\{l_i \mid 0 \leq i < m\}$ . Thus, we have

$$\text{size}_{\text{Fml}}(\langle \theta(a_1, \dots, a_r) \rangle_{(m)}) \leq 1 + (m+2)^{l'+1},$$

and hence, we can conclude

$$\text{size}_{\text{Fml}}(\langle \theta(a_1, \dots, a_r) \rangle_{(m)}) \leq (m+2)^{l'+2}.$$

Therefore,  $l = \max\{l_i \mid 0 \leq i < m\} + 2$ .

□

### 3.4 Coding literals

Before we define the coding of literals, we show how we code sets.

**Notation.** Let  $\text{bit}(x, y) : \mathbb{N}^2 \rightarrow \{0, 1\}$  be the function in  $\mathcal{L}_{\mathbb{N}}$  mapping  $(x, y)$  to the  $x$ -th bit of the binary representation of  $y$  if it exists and to 0 otherwise.

**Notation.**

- We code a finite set  $X$  of natural numbers by the natural number  $a = \sum_{c \in X} 2^c$ . This means that  $\text{bit}(c, a) = 1$  if and only if  $c$  is in  $X$ . We write  $a = \ulcorner X \urcorner$  where  $\ulcorner \cdot \urcorner$  means "code of".
- We abbreviate " $\text{bit}(c, a) = 1$ " by writing " $c \in a$ ". Moreover, we write " $\exists x \in a \varphi$ " for " $\exists x (\text{bit}(x, a) = 1 \wedge \varphi)$ " as well as " $\forall x \in a \varphi$ " for " $\forall x (\text{bit}(x, a) = 1 \rightarrow \varphi)$ " for a first-order formula  $\varphi$ .
- Let  $X$  and  $Y$  be sets of natural numbers and let  $a = \ulcorner X \urcorner$  and  $b = \ulcorner Y \urcorner$ . Then  $X$  is a subset of  $Y$  if and only if for every  $c \in \mathbb{N}$ ,  $\text{bit}(c, a) \leq \text{bit}(c, b)$  holds. We write " $a \subseteq b$ " for " $\forall x \text{bit}(x, a) \leq \text{bit}(x, b)$ ".
- Let  $Z$  be the union of two finite sets of natural numbers  $X$  and  $Y$ . Further, let  $a = \ulcorner X \urcorner$  and  $b = \ulcorner Y \urcorner$ . Then  $c = \ulcorner Z \urcorner$  if and only if for all  $e \in \mathbb{N}$

$$\text{bit}(e, c) = 1 - (1 - \text{bit}(e, a)) \cdot (1 - \text{bit}(e, b)).$$

We use the binary  $\mathcal{L}_{\mathbb{N}}$ -function symbol " $\cup$ " which maps the codes of two sets of natural numbers to the code of their union, e.g., in the setting above  $c = a \cup b$ .

- Let  $Z$  be the set difference of two finite sets of natural numbers  $X$  and  $Y$ . Further let  $a = \ulcorner X \urcorner$  and  $b = \ulcorner Y \urcorner$ . Then  $c = \ulcorner Z \urcorner$  if and only if for all  $e \in \mathbb{N}$

$$\text{bit}(e, c) = \text{bit}(e, a) \cdot (\text{bit}(e, a) - \text{bit}(e, b)).$$

We use the binary  $\mathcal{L}_{\mathbb{N}}$ -function symbol " $\setminus$ " which maps the codes of two sets of natural numbers to the code of their set difference, e.g., in the setting above  $c = a \setminus b$ .

**Lemma 3.4.** *The following holds in the standard model.*

- (a)  $\mathfrak{N} \models \forall x, y, z (x \in y \cup z \leftrightarrow x \in y \vee x \in z)$ , and  
 (b)  $\mathfrak{N} \models \forall x, y, z (x \in y \setminus z \leftrightarrow x \in y \wedge \neg(x \in z))$ .

*Proof.* This follows from the definition above. □

**Note.** *The propositional translation  $\langle \exists x \in a \varphi(x) \rangle_{(m)}$  for integers  $a$  and  $m$  and an  $\mathcal{L}$ -formula  $\varphi(x)$  is*

$$\langle \exists x (\text{bit}(x, a) = 1 \wedge \varphi(x)) \rangle_{(m)} = \bigvee_{x < m} (\langle \text{bit}(x, a) = 1 \rangle_{(m)} \wedge \langle \varphi(x) \rangle_{(m)}).$$

*Note that  $\langle \text{bit}(x, a) = 1 \rangle_{(m)} = \langle x \in a \rangle_{(m)}$ .*

Now we turn to the coding of literals. First of all, we need to code relation and function symbols from the new language  $\mathcal{L}'$ .

**Notation.** *For every  $R \in \mathcal{L}'$  and every  $f \in \mathcal{L}'$  we associate a unique natural number  $c$  with  $R$  or  $f$  and write  $c = \ulcorner R \urcorner$  or  $c = \ulcorner f \urcorner$  respectively.*

**Notation.** *Let  $\text{Rel} \subseteq \mathbb{N}$  and  $\text{Fct} \subseteq \mathbb{N}$  be the sets such that the codes of all relation symbols or function symbols of  $\mathcal{L}'$  are in  $\text{Rel}$  or  $\text{Fct}$  respectively.*

**Definition.** *An integer  $c$  codes a literal  $\ell$  if and only if  $c = \langle c_1, c_2, c_3 \rangle$  for  $c_1, c_2, c_3 \in \mathbb{N}$  with the properties:*

- If  $\ell = R_{a_1, \dots, a_{\text{ar}(R)}}$  for a relation  $R \in \mathcal{L}'$ , then

$$c_1 = \ulcorner R \urcorner, c_2 = \langle a_1, \dots, a_{\text{ar}(R)} \rangle \text{ and } c_3 = 1.$$

- If  $\ell = \neg R_{a_1, \dots, a_{\text{ar}(R)}}$  for a relation  $R \in \mathcal{L}'$ , then

$$c_1 = \ulcorner R \urcorner, c_2 = \langle a_1, \dots, a_{\text{ar}(R)} \rangle \text{ and } c_3 = 0.$$

- If  $\ell = f_{a_1, \dots, a_{\text{ar}(f)+1}}$  for a function  $f \in \mathcal{L}'$ , then

$$c_1 = \ulcorner f \urcorner, c_2 = \langle a_1, \dots, a_{\text{ar}(f)+1} \rangle \text{ and } c_3 = 1.$$

- If  $\ell = \neg f_{a_1, \dots, a_{\text{ar}(f)+1}}$  for a function  $f \in \mathcal{L}'$ , then

$$c_1 = \ulcorner f \urcorner, c_2 = \langle a_1, \dots, a_{\text{ar}(f)+1} \rangle \text{ and } c_3 = 0.$$

We write  $c = \ulcorner \ell \urcorner$  if  $c$  codes the literal  $\ell$ .

**Definition.** We define the following formulas.

$$\begin{aligned} \text{Lit}(x) = \exists x_1, x_2, x_3 \left( x = \langle x_1, x_2, x_3 \rangle \wedge x_3 < 2 \wedge \right. \\ \left. \left( \bigvee_{\ulcorner R \urcorner \in \text{Rel}} [x_1 = \ulcorner R \urcorner \wedge \exists y_1, \dots, y_{\text{ar}(R)} x_2 = \langle y_1, \dots, y_{\text{ar}(R)} \rangle] \vee \right. \right. \\ \left. \left. \bigvee_{\ulcorner f \urcorner \in \text{Fct}} [x_1 = \ulcorner f \urcorner \wedge \exists y_1, \dots, y_{\text{ar}(f)+1} x_2 = \langle y_1, \dots, y_{\text{ar}(f)+1} \rangle] \right) \right) \end{aligned}$$

$$\begin{aligned} \text{True}_{\text{Rel}}(x_1, x_2, x_3) = \\ \bigvee_{\ulcorner R \urcorner \in \text{Rel}} \left( x_1 = \ulcorner R \urcorner \wedge \exists y_1, \dots, y_{\text{ar}(R)} [x_2 = \langle y_1, \dots, y_{\text{ar}(R)} \rangle \wedge \right. \\ \left. ((R(y_1, \dots, y_{\text{ar}(R)}) \wedge x_3 = 1) \vee (\neg R(y_1, \dots, y_{\text{ar}(R)}) \wedge x_3 = 0)) \right) \end{aligned}$$

$$\begin{aligned} \text{True}_{\text{Fct}}(x_1, x_2, x_3) = & \\ & \bigvee_{\ulcorner f \urcorner \in \text{Fct}} \left( x_1 = \ulcorner f \urcorner \wedge \exists y_1, \dots, y_{\text{ar}(f)+1} [x_2 = \langle y_1, \dots, y_{\text{ar}(f)+1} \rangle \wedge \right. \\ & \left. ((f(y_1, \dots, y_{\text{ar}(f)})) = y_{\text{ar}(f)+1} \wedge x_3 = 1) \vee \right. \\ & \left. (\neg f(y_1, \dots, y_{\text{ar}(f)}) = y_{\text{ar}(f)+1} \wedge x_3 = 0) \right) \end{aligned}$$

$$\text{True}_{\text{Lit}}(x) = \exists x_1, x_2, x_3 [x = \langle x_1, x_2, x_3 \rangle \wedge (\text{True}_{\text{Rel}}(x_1, x_2, x_3) \vee \text{True}_{\text{Fct}}(x_1, x_2, x_3))]$$

**Note.** The formula  $\text{Lit}(x)$  states "x codes a literal" and  $\text{True}_{\text{Lit}}(x)$  states "x codes a true literal". Note that  $\text{True}_{\text{Lit}}(x)$  implies  $\text{Lit}(x)$ . Further note that the formula  $\text{Lit}(x)$  is an  $\mathcal{L}_{\mathbb{N}}$ -formula but  $\text{True}_{\text{Lit}}(x)$  is not.

**Definition.** Let  $\varphi(x_1, \dots, x_r)$  be an unnested  $\mathcal{L}$ -literal.

We write  $T_\varphi : \mathbb{N}^{r+1} \rightarrow \mathbb{N}$  for the function in  $\mathcal{L}_{\mathbb{N}}$  that maps  $(m, a_1, \dots, a_r) \in \mathbb{N}^{r+1}$  to  $\ulcorner \langle \varphi(a_1, \dots, a_r) \rangle_{(m)} \urcorner$ . We use the notation  $T_\varphi(m; a_1, \dots, a_r)$  for better readability between the translation bound and the parameters in  $\varphi$ .

Next we show how to relate the truth of an  $\mathcal{L}$ -literal to the truth of its coded propositional correspondent.

**Proposition 3.5.** Let  $\mathcal{M}$  be a nonstandard model and  $\mathcal{N}$  an  $\mathcal{L}$ -expansion of  $\mathcal{M}$ . Further assume  $\theta(x_1, \dots, x_r)$  is an unnested  $\mathcal{L}'$ -literal.

For  $a_1, \dots, a_r, n \in \mathcal{N}$  the following holds

$$\mathcal{N} \models \theta(a_1, \dots, a_r) \text{ if and only if } \mathcal{N} \models \text{True}_{\text{Lit}}(T_\theta(n; a_1, \dots, a_r)).$$

*Proof.* Assume  $\theta(a_1, \dots, a_r) = R(a_1, \dots, a_r)$ . Intuitively the propositional translation  $\langle \theta(a_1, \dots, a_r) \rangle_{(n)}$  is equal to  $R_{a_1, \dots, a_{\text{ar}(R)}}$ . Since  $\mathcal{M}$  is a nonstandard model, the function  $T_\theta$  has an interpretation in  $\mathcal{M}$  such that

$$T_\theta^{\mathcal{M}}(n; a_1, \dots, a_{\text{ar}(R)}) = \langle \ulcorner R \urcorner, \langle a_1, \dots, a_{\text{ar}(R)} \rangle^{\mathcal{M}}, 1 \rangle^{\mathcal{M}}.$$



Next we determine if  $\mathcal{N} \models \text{True}_{\text{Lit}}(\text{T}_\theta(n; a_1, \dots, a_{\text{ar}(R)}))$ . As  $\mathcal{N}$  is an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  and  $\text{T}_\theta \in \mathcal{L}_{\mathbb{N}}$ , we have

$$\text{T}_\theta^{\mathcal{N}}(n; a_1, \dots, a_{\text{ar}(R)}) = \text{T}_\theta^{\mathcal{M}}(n; a_1, \dots, a_{\text{ar}(R)}).$$

Since there are  $c_1, c_2$  and  $c_3$  in  $\mathcal{N}$  such that

$$\text{T}_\theta^{\mathcal{N}}(n; a_1, \dots, a_{\text{ar}(R)}) = \langle c_1, c_2, c_3 \rangle^{\mathcal{N}},$$

namely  $\ulcorner R \urcorner, \langle a_1, \dots, a_{\text{ar}(R)} \rangle^{\mathcal{N}}$  and 1, we have  $\text{True}_{\text{Lit}}(\text{T}_\theta^{\mathcal{N}}(n; a_1, \dots, a_{\text{ar}(R)}))$  is true in  $\mathcal{N}$  if and only if

$$\mathcal{N} \models \text{True}_{\text{Rel}}(\ulcorner R \urcorner, \langle a_1, \dots, a_{\text{ar}(R)} \rangle, 1) \vee \text{True}_{\text{Fct}}(\ulcorner R \urcorner, \langle a_1, \dots, a_{\text{ar}(R)} \rangle, 1).$$

Since  $\ulcorner R \urcorner \in \text{Rel}$ , we obtain that  $\mathcal{N} \models \neg \text{True}_{\text{Fct}}(\ulcorner R \urcorner, \langle a_1, \dots, a_{\text{ar}(R)} \rangle, 1)$  and therefore,  $\text{True}_{\text{Lit}}(\text{T}_\theta^{\mathcal{N}}(n; a_1, \dots, a_{\text{ar}(R)}))$  holds in  $\mathcal{N}$  if and only if  $\mathcal{N} \models \text{True}_{\text{Rel}}(\ulcorner R \urcorner, \langle a_1, \dots, a_{\text{ar}(R)} \rangle, 1)$ . Since the first entry is  $\ulcorner R \urcorner$ , all disjuncts in  $\text{True}_{\text{Rel}}(\ulcorner R \urcorner, \langle a_1, \dots, a_{\text{ar}(R)} \rangle^{\mathcal{N}}, 1)$  except for one are false in  $\mathcal{N}$ . We obtain that  $\text{True}_{\text{Rel}}(\ulcorner R \urcorner, \langle a_1, \dots, a_{\text{ar}(R)} \rangle^{\mathcal{N}}, 1)$  is true in  $\mathcal{N}$  if and only if

$$\begin{aligned} \mathcal{N} \models \ulcorner R \urcorner = \ulcorner R \urcorner \wedge \exists y_1, \dots, y_{\text{ar}(R)} [\langle a_1, \dots, a_{\text{ar}(R)} \rangle = \langle y_1, \dots, y_{\text{ar}(R)} \rangle \wedge \\ ((R(y_1, \dots, y_{\text{ar}(R)}) \wedge 1 = 1) \vee (\neg R(y_1, \dots, y_{\text{ar}(R)}) \wedge 1 = 0)) \end{aligned}$$

in  $\mathcal{N}$ . Clearly, the above holds if and only if  $\mathcal{N} \models R(a_1, \dots, a_{\text{ar}(R)})$ , which proves the claim.

The other cases can be proven analogously.  $\square$

We end this section with results about the propositional translation of the  $\text{True}_{\text{Lit}}$ -formula.

**Proposition 3.6.** *The propositional translation  $\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)}$  of the formula  $\text{True}_{\text{Lit}}(c)$  for integers  $c$  and  $m$  is either a literal or equal to  $\perp$ .*

*Proof.* We translate  $\text{True}_{\text{Lit}}(c)$  to  $\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)}$  for integers  $c$  and  $m$  and enumerate the translations for convenient referral later on.

$$\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} = \bigvee_{x_1 < m} \bigvee_{x_2 < m} \bigvee_{x_3 < m} [\langle c = \langle x_1, x_2, x_3 \rangle \rangle_{(m)}^{(1)} \wedge \\ (\langle \text{True}_{\text{Rel}}(x_1, x_2, x_3) \rangle_{(m)}^{(2)} \vee \langle \text{True}_{\text{Fct}}(x_1, x_2, x_3) \rangle_{(m)}^{(3)})]$$

$$\langle \text{True}_{\text{Rel}}(x_1, x_2, x_3) \rangle_{(m)}^{(2)} = \bigvee_{\ulcorner R \urcorner \in \text{Rel}} (\langle x_1 = \ulcorner R \urcorner \rangle_{(m)}^{(4)} \wedge \\ \bigvee_{y_1 < m} \cdots \bigvee_{y_{\text{ar}(R)} < m} [\langle x_2 = \langle y_1, \dots, y_{\text{ar}(R)} \rangle \rangle_{(m)}^{(5)} \wedge ((\langle R(y_1, \dots, y_{\text{ar}(R)}) \rangle_{(m)}^{(6)} \wedge \\ \langle x_3 = 1 \rangle_{(m)}^{(7)}) \vee (\langle \neg R(y_1, \dots, y_{\text{ar}(R)}) \rangle_{(m)}^{(8)} \wedge \langle x_3 = 0 \rangle_{(m)}^{(9)})])])$$

$$\langle \text{True}_{\text{Fct}}(x_1, x_2, x_3) \rangle_{(m)}^{(3)} = \bigvee_{\ulcorner f \urcorner \in \text{Fct}} (\langle x_1 = \ulcorner f \urcorner \rangle_{(m)}^{(10)} \wedge \\ \bigvee_{y_1 < m} \cdots \bigvee_{y_{\text{ar}(f)+1} < m} [\langle x_2 = \langle y_1, \dots, y_{\text{ar}(f)+1} \rangle \rangle_{(m)}^{(11)} \wedge \\ ((\langle f(y_1, \dots, y_{\text{ar}(f)}) = y_{\text{ar}(f)+1} \rangle_{(m)}^{(12)} \wedge \langle x_3 = 1 \rangle_{(m)}^{(13)}) \vee \\ (\langle \neg f(y_1, \dots, y_{\text{ar}(f)}) = y_{\text{ar}(f)+1} \rangle_{(m)}^{(14)} \wedge \langle x_3 = 0 \rangle_{(m)}^{(15)})])])$$

Most of the subformulas of  $\text{True}_{\text{Lit}}(c)$  do not involve  $\mathcal{L}'$ -symbols and are thus translated to a truth value according to whether they are true or false in the standard model. Hence, we obtain a propositional formula with numerous occurrences of  $\top$  and  $\perp$  which are eliminated in the translation process.

Assume  $c$  does not code a literal: If  $c$  does not code a triple, i.e., there are no  $c_1, c_2, c_3$  such that  $c = \langle c_1, c_2, c_3 \rangle$ , then translation (1) is equal to  $\perp$ . If there are  $c_1, c_2, c_3$  such that  $c = \langle c_1, c_2, c_3 \rangle$  but there is no  $R \in \mathcal{L}'$  and no  $f \in \mathcal{L}'$  such that  $c_1 = \ulcorner R \urcorner$  or  $c_1 = \ulcorner f \urcorner$ , then translation (4) is equal

to  $\perp$  and translation (10) is equal to  $\perp$ . If there is such an  $R$  or  $f$  in  $\mathcal{L}'$ , but  $c_2$  does not code a tuple of length  $\text{ar}(R)$  or  $\text{ar}(f) + 1$  respectively, then translations (5) and (11) are equal to  $\perp$ . Lastly, if  $c_3 \geq 2$ , then translations (7), (9), (13) and (15) are equal to  $\perp$ . In all of those cases we obtain that  $\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} = \perp$ .

Assume  $c$  codes a literal, but one of  $c_1, c_2$  and  $c_3$  is bigger than  $m$  or equal to  $m$  where  $c = \langle c_1, c_2, c_3 \rangle$ . Then translation (1) is equal to  $\perp$ , making  $\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} = \perp$ .

Now assume  $c$  codes a literal and  $c_1, c_2$  and  $c_3$  are smaller than  $m$ . Without loss of generality let  $c = \langle c_1, c_2, 1 \rangle$  where  $c_1 = \ulcorner R \urcorner, c_2 = \langle b_1, \dots, b_{\text{ar}(R)} \rangle$  for a relation  $R' \in \mathcal{L}'$  and  $b_1, \dots, b_{\text{ar}(R')} \in \mathbb{N}$ .

Translation (1) will only be evaluated to  $\top$  in the disjunct where  $x_1 = c_1, x_2 = c_2$  and  $x_3 = 1$ . Thus, all other disjuncts are equal to  $\perp$ , since then translation (1) is equal to  $\perp$  and therefore, the whole conjunction is equal to  $\perp$ . Hence,

$$\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} = \langle \text{True}_{\text{Rel}}(c_1, c_2, c_3) \rangle_{(m)}^{(2)} \vee \langle \text{True}_{\text{Fct}}(c_1, c_2, c_3) \rangle_{(m)}^{(3)}.$$

Since  $c_1$  codes a relation, translation (3) is equal to  $\perp$ . We obtain that  $\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} = \langle \text{True}_{\text{Rel}}(c_1, c_2, c_3) \rangle_{(m)}$ .

Only one of the translations  $\langle c_1 = \ulcorner R \urcorner \rangle_{(m)}$  for  $\ulcorner R \urcorner \in \text{Rel}$  in translation (2) is equal to  $\top$ , namely if  $R = R'$ . We obtain that

$$\begin{aligned} \langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} &= \top \wedge \bigvee_{y_1 < m} \dots \bigvee_{y_{\text{ar}(R)} < m} [\langle c_2 = \langle y_1, \dots, y_{\text{ar}(R)} \rangle \rangle_{(m)}^{(5)} \wedge \\ &\quad ((\langle R'(y_1, \dots, y_{\text{ar}(R')}) \rangle_{(m)}^{(6)} \wedge \langle c_3 = 1 \rangle_{(m)}^{(7)}) \vee \\ &\quad (\langle \neg R'(y_1, \dots, y_{\text{ar}(R')}) \rangle_{(m)}^{(8)} \wedge \langle c_3 = 0 \rangle_{(m)}^{(9)})] \end{aligned}$$

Translation (5) is only equal to  $\top$  if  $y_i = b_i$  for  $1 \leq i \leq \text{ar}(R')$ . If the

$b_i$ 's are not smaller than  $m$ , then translation (5) is equal to  $\perp$  and hence,  $\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} = \perp$ . Assume otherwise. Then only one disjunct, namely the one involving  $y_i = b_i$  for  $1 \leq i \leq \text{ar}(R')$ , is not equal to  $\perp$ . We obtain that

$$\begin{aligned} \langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} = \top \wedge & \left( \langle \langle R'(b_1, \dots, b_{\text{ar}(R')}) \rangle_{(m)} \rangle_{(m)}^{(6)} \wedge \langle c_3 = 1 \rangle_{(m)}^{(7)} \right) \vee \\ & \left( \langle \neg R'(b_1, \dots, b_{\text{ar}(R')}) \rangle_{(m)}^{(8)} \wedge \langle c_3 = 0 \rangle_{(m)}^{(9)} \right). \end{aligned}$$

Since  $c_3 = 1$ , translation (7) is equal to  $\top$  and translation (9) is equal to  $\perp$ . It follows that

$$\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} = \left( \langle R'(b_1, \dots, b_{\text{ar}(R')}) \rangle_{(m)}^{(6)} \wedge \top \right) \vee \perp$$

and therefore,  $\langle \text{True}_{\text{Lit}}(c) \rangle_{(m)} = R'_{b_1, \dots, b_{\text{ar}(R')}}$ . The other cases can be proven analogously.  $\square$

**Corollary 3.7.** *Let  $\mathcal{M}$  be a nonstandard model. Then*

$$\mathcal{M} \models \forall x, y \left( \text{Lit}(\top_{\text{True}_{\text{Lit}}}(x; y)) \vee \top_{\text{True}_{\text{Lit}}}(x; y) = \ulcorner \perp \urcorner \right)$$

*Proof.* This follows from the elementary equivalence of  $\mathcal{M}$  to  $\mathfrak{N}$  where this statement holds by Proposition 3.6 and the definition of  $\top_{\text{True}_{\text{Lit}}}$ .  $\square$

## 3.5 Coding propositional formulas

**Definition.** *We define the code  $\ulcorner \varphi \urcorner$  of  $\Sigma_d^{\text{prop}}$ - and  $\Pi_d^{\text{prop}}$ -formulas  $\varphi$  for  $d \in \mathbb{N}$  recursively.*

*An integer  $c$  codes a  $\Sigma_d^{\text{prop}}$ - or  $\Pi_d^{\text{prop}}$ -formula  $\varphi$  if and only if  $c$  codes a literal or  $d \geq 1$  and  $c = \langle c_1, c_2 \rangle$  for  $c_1, c_2 \in \mathbb{N}$  with the properties:*

- *If  $\varphi$  is a  $\Sigma_d^{\text{prop}}$ -formula  $\bigvee \Phi$  where  $\Phi$  is a finite set of  $\Pi_{d-1}^{\text{prop}}$ -formulas, then*

$$- \quad c_1 = 0, \text{ and}$$

- $c_2$  codes the set of codes of all  $\psi$  in  $\Phi$ , i.e.,  $c_2 = \ulcorner \{ \ulcorner \psi \urcorner \mid \psi \in \Phi \} \urcorner$ .
- If  $\varphi$  is a  $\Pi_d^{prop}$ -formula  $\bigwedge \Phi$  where  $\Phi$  is a finite set of  $\Sigma_{d-1}^{prop}$ -formulas, then
  - $c_1 = 1$ , and
  - $c_2$  codes the set of codes of all  $\psi$  in  $\Phi$ , i.e.,  $c_2 = \ulcorner \{ \ulcorner \psi \urcorner \mid \psi \in \Phi \} \urcorner$ .

**Definition.** We define the formulas  $\text{Fml}_{\Sigma_d^{prop}}(x)$ ,  $\text{Fml}_{\Pi_d^{prop}}(x)$ ,  $\text{Fml}_{\text{DNF}}(x)$ ,  $\text{Fml}_{\text{DNF}_k}(x)$  and the formula  $\text{Fml}_{\text{CNF}_k}(x)$  for  $d, k \in \mathbb{N}$  recursively on  $d$ .

$$\begin{aligned} \text{Fml}_{\Sigma_0^{prop}}(x) &= \text{Fml}_{\Pi_0^{prop}}(x) = \text{Lit}(x) \\ \text{Fml}_{\Sigma_d^{prop}}(x) &= \text{Fml}_{\Pi_{d-1}^{prop}}(x) \vee \exists y \left( x = \langle 0, y \rangle \wedge \forall z \in y \left[ \text{Lit}(z) \vee \right. \right. \\ &\quad \left. \left. \exists w \left( z = \langle 1, w \rangle \wedge \text{Fml}_{\Pi_{d-1}^{prop}}(z) \right) \right] \right) \\ \text{Fml}_{\Pi_d^{prop}}(x) &= \text{Fml}_{\Sigma_{d-1}^{prop}}(x) \vee \exists y \left( x = \langle 1, y \rangle \wedge \forall z \in y \left[ \text{Lit}(z) \vee \right. \right. \\ &\quad \left. \left. \exists w \left( z = \langle 0, w \rangle \wedge \text{Fml}_{\Sigma_{d-1}^{prop}}(z) \right) \right] \right) \\ \text{Fml}_{\text{DNF}}(x) &= \text{Fml}_{\Sigma_2^{prop}}(x) \\ \text{Fml}_{\text{DNF}_k}(x) &= \text{Fml}_{\Pi_1^{prop}}(x) \vee \exists y \left( x = \langle 0, y \rangle \wedge \forall z \in y \left[ \text{Lit}(z) \vee \right. \right. \\ &\quad \left. \left. \exists w \left( z = \langle 1, w \rangle \wedge \text{Fml}_{\Pi_1^{prop}}(z) \wedge \text{card}(w) \leq k \right) \right] \right) \\ \text{Fml}_{\text{CNF}_k}(x) &= \text{Fml}_{\Sigma_1^{prop}}(x) \vee \exists y \left( x = \langle 1, y \rangle \wedge \forall z \in y \left[ \text{Lit}(z) \vee \right. \right. \\ &\quad \left. \left. \exists w \left( z = \langle 0, w \rangle \wedge \text{Fml}_{\Sigma_1^{prop}}(z) \wedge \text{card}(w) \leq k \right) \right] \right) \end{aligned}$$

The function  $\text{card}(x) \in \mathcal{L}_{\mathbb{N}}$  maps  $x$  to the cardinality of the set coded by  $x$ .

**Note.** All formulas defined above are  $\mathcal{L}_{\mathbb{N}}$ -formulas since  $\text{Lit}(x)$  is an  $\mathcal{L}_{\mathbb{N}}$ -formula and all other occurring symbols are in  $\mathcal{L}_{\mathbb{N}}$ .

With the formulas  $\text{Fml}_{\Sigma_d^{prop}}(x)$  and  $\text{Fml}_{\Pi_d^{prop}}(x)$  we can state "x codes a  $\Sigma_d^{prop}$ - or  $\Pi_d^{prop}$ -formula".

**Definition.** We define the functions  $\text{Disj} : \mathbb{N}^2 \rightarrow \mathbb{N}$  and  $\text{Conj} : \mathbb{N}^2 \rightarrow \mathbb{N}$  by

$$\text{Disj}(x, y) = \begin{cases} y & \text{if } x = \ulcorner \perp \urcorner \\ x & \text{if } y = \ulcorner \perp \urcorner \\ \langle 0, u \cup v \rangle & \text{if } \exists u, v \ (u \neq \ulcorner \emptyset \urcorner \wedge v \neq \ulcorner \emptyset \urcorner \wedge \\ & \quad x = \langle 0, u \rangle \wedge y = \langle 0, v \rangle) \\ \langle 0, u \cup \ulcorner \{y\} \urcorner \rangle & \text{if } \exists u \forall v \ (u \neq \ulcorner \emptyset \urcorner \wedge x = \langle 0, u \rangle \wedge y \neq \langle 0, v \rangle) \\ \langle 0, \ulcorner \{x\} \urcorner \cup v \rangle & \text{if } \forall u \exists v \ (v \neq \ulcorner \emptyset \urcorner \wedge x \neq \langle 0, u \rangle \wedge y = \langle 0, v \rangle) \\ \langle 0, \ulcorner \{x, y\} \urcorner \rangle & \text{if } \forall u, v \ (x \neq \langle 0, u \rangle \wedge y \neq \langle 0, v \rangle) \end{cases}$$

$$\text{Conj}(x, y) = \begin{cases} y & \text{if } x = \ulcorner \top \urcorner \\ x & \text{if } y = \ulcorner \top \urcorner \\ \langle 1, u \cup v \rangle & \text{if } \exists u, v \ (u \neq \ulcorner \emptyset \urcorner \wedge v \neq \ulcorner \emptyset \urcorner \wedge \\ & \quad x = \langle 1, u \rangle \wedge y = \langle 1, v \rangle) \\ \langle 1, u \cup \ulcorner \{y\} \urcorner \rangle & \text{if } \exists u \forall v \ (u \neq \ulcorner \emptyset \urcorner \wedge x = \langle 1, u \rangle \wedge y \neq \langle 1, v \rangle) \\ \langle 1, \ulcorner \{x\} \urcorner \cup v \rangle & \text{if } \forall u \exists v \ (v \neq \ulcorner \emptyset \urcorner \wedge x \neq \langle 1, u \rangle \wedge y = \langle 1, v \rangle) \\ \langle 1, \ulcorner \{x, y\} \urcorner \rangle & \text{if } \forall u, v \ (x \neq \langle 1, u \rangle \wedge y \neq \langle 1, v \rangle) \end{cases}$$

**Note.** The functions  $\text{Disj}$  and  $\text{Conj}$  are in  $\mathcal{L}_{\mathbb{N}}$ . We use them to map  $\ulcorner \varphi \urcorner$  and  $\ulcorner \psi \urcorner$  to  $\ulcorner \varphi \vee \psi \urcorner$  or  $\ulcorner \varphi \wedge \psi \urcorner$  respectively where  $\varphi$  and  $\psi$  are  $\Sigma_d^{\text{prop}}$ - or  $\Pi_d^{\text{prop}}$ -formulas for a  $d \in \mathbb{N}$ .

**Definition.** We define the formulas  $\text{True}_{\Sigma_d^{\text{prop}}}(x)$ ,  $\text{True}_{\Pi_d^{\text{prop}}}(x)$ ,  $\text{True}_{\text{DNF}}(x)$  and  $\text{True}_{\text{DNF}_k}(x)$  for integers  $d$  and  $k \geq 1$  recursively on  $d$ .

$$\begin{aligned} \text{True}_{\Sigma_0^{\text{prop}}}(x) &= \text{True}_{\Pi_0^{\text{prop}}}(x) = \text{True}_{\text{Lit}}(x) \\ \text{True}_{\Sigma_d^{\text{prop}}}(x) &= \text{True}_{\Pi_{d-1}^{\text{prop}}}(x) \vee \exists y \ (x = \langle 0, y \rangle \wedge \exists z \in y \ \text{True}_{\Pi_{d-1}^{\text{prop}}}(z) \wedge \\ & \quad \neg \text{Fml}_{\Pi_{d-1}^{\text{prop}}}(x) \wedge \text{Fml}_{\Sigma_d^{\text{prop}}}(x)) \end{aligned}$$

$$\begin{aligned} \text{True}_{\Pi_d^{\text{prop}}}(x) &= \text{True}_{\Sigma_{d-1}^{\text{prop}}}(x) \vee \exists y (x = \langle 1, y \rangle \wedge \forall z \in y \text{ True}_{\Sigma_{d-1}^{\text{prop}}}(z) \wedge \\ &\quad \neg \text{Fml}_{\Sigma_{d-1}^{\text{prop}}}(x) \wedge \text{Fml}_{\Pi_d^{\text{prop}}}(x)) \end{aligned}$$

$$\text{True}_{\text{DNF}}(x) = \text{True}_{\Sigma_2^{\text{prop}}}(x)$$

$$\text{True}_{\text{DNF}_k}(x) = \text{Fml}_{\text{DNF}_k}(x) \wedge \text{True}_{\text{DNF}}(x)$$

**Note.** With the formulas  $\text{True}_{\Sigma_d^{\text{prop}}}(x)$  and  $\text{True}_{\Pi_d^{\text{prop}}}(x)$  we can state "x codes a true  $\Sigma_d^{\text{prop}}$ - or  $\Pi_d^{\text{prop}}$ -formula". By this we mean an extension of the  $\text{True}_{\text{Lit}}$ -formula to propositional formulas. Note that all of the defined formulas above are not  $\mathcal{L}_{\mathbb{N}}$ -formulas since  $\text{True}_{\text{Lit}}(x)$  is not.

**Lemma 3.8.** Let  $\mathcal{M}$  be a nonstandard model,  $\mathcal{N}$  an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  and  $d$  an integer.

$$(a) \mathcal{M} \models \forall x \left( \text{Fml}_{\Sigma_d^{\text{prop}}}(x) \rightarrow \text{Fml}_{\Pi_{d+1}^{\text{prop}}}(x) \right)$$

$$(b) \mathcal{M} \models \forall x \left( \text{Fml}_{\Pi_d^{\text{prop}}}(x) \rightarrow \text{Fml}_{\Sigma_{d+1}^{\text{prop}}}(x) \right)$$

$$(c) \mathcal{N} \models \forall x \left( \text{True}_{\Sigma_d^{\text{prop}}}(x) \rightarrow \text{True}_{\Pi_{d+1}^{\text{prop}}}(x) \right)$$

$$(d) \mathcal{N} \models \forall x \left( \text{True}_{\Pi_d^{\text{prop}}}(x) \rightarrow \text{True}_{\Sigma_{d+1}^{\text{prop}}}(x) \right)$$

*Proof.* This follows from the definitions of the formulas.  $\square$

**Definition.** We define the  $\mathcal{L}_{\mathbb{N}}$ -function  $\text{Neg} : \mathbb{N} \rightarrow \mathbb{N}$  by

$$\text{Neg}(x) = \begin{cases} \langle x_1, x_2, 1 - x_3 \rangle & \text{if } x = \langle x_1, x_2, x_3 \rangle \\ \langle 1 - x_1, x'_2 \rangle & \text{if } x = \langle x_1, x_2 \rangle \text{ where } x'_2 = \ulcorner \{\text{Neg}(y) \mid y \in x_2\} \urcorner \\ 0 & \text{otherwise.} \end{cases}$$

**Lemma 3.9.** Let  $\varphi$  be a propositional formula in  $\Sigma_d^{\text{prop}}$  or  $\Pi_d^{\text{prop}}$  for an integer  $d$ . Then we have  $\ulcorner \neg \varphi \urcorner = \text{Neg}(\ulcorner \varphi \urcorner)$ .

*Proof.* We show this by induction on  $d$ : Let  $\ell$  be a literal and assume that  $\ulcorner \ell \urcorner = \langle c_1, c_2, c_3 \rangle$  for  $c_1, c_2, c_3 \in \mathbb{N}$ . Then  $\ulcorner \neg \ell \urcorner = \langle c_1, c_2, 1 - c_3 \rangle$ , which is

$\text{Neg}(\ulcorner \ell \urcorner)$ .

Assume the claim holds for  $d-1$  and  $\varphi = \bigvee \Phi \in \Sigma_d^{\text{prop}} \setminus \Pi_{d-1}^{\text{prop}}$  or  $\varphi = \bigwedge \Phi \in \Pi_d^{\text{prop}} \setminus \Sigma_{d-1}^{\text{prop}}$ . Further let  $\Psi = \{\neg\psi \mid \psi \in \Phi\}$ . Then,

$$\begin{aligned}
 \ulcorner \neg\varphi \urcorner &\stackrel{(1)}{=} \begin{cases} \ulcorner \bigwedge \Psi \urcorner & \text{if } \varphi = \bigvee \Phi \\ \ulcorner \bigvee \Psi \urcorner & \text{if } \varphi = \bigwedge \Phi \end{cases} \\
 &\stackrel{(2)}{=} \begin{cases} \langle 1, \ulcorner \{\neg\psi \mid \psi \in \Phi\} \urcorner \rangle & \text{if } \varphi = \bigvee \Phi \\ \langle 0, \ulcorner \{\neg\psi \mid \psi \in \Phi\} \urcorner \rangle & \text{if } \varphi = \bigwedge \Phi \end{cases} \\
 &\stackrel{(3)}{=} \begin{cases} \langle 1, \ulcorner \{\text{Neg}(\ulcorner \psi \urcorner) \mid \psi \in \Phi\} \urcorner \rangle & \text{if } \varphi = \bigvee \Phi \\ \langle 0, \ulcorner \{\text{Neg}(\ulcorner \psi \urcorner) \mid \psi \in \Phi\} \urcorner \rangle & \text{if } \varphi = \bigwedge \Phi \end{cases} \\
 &\stackrel{(4)}{=} \begin{cases} \langle 1, \ulcorner \{\text{Neg}(c) \mid c \in \ulcorner \Phi \urcorner \urcorner \rangle & \text{if } \varphi = \bigvee \Phi \\ \langle 0, \ulcorner \{\text{Neg}(c) \mid c \in \ulcorner \Phi \urcorner \urcorner \rangle & \text{if } \varphi = \bigwedge \Phi \end{cases} \\
 &\stackrel{(5)}{=} \begin{cases} \text{Neg}(\langle 1, \ulcorner \{\neg\psi \mid \psi \in \Phi\} \urcorner \rangle) & \text{if } \varphi = \bigvee \Phi \\ \text{Neg}(\langle 0, \ulcorner \{\neg\psi \mid \psi \in \Phi\} \urcorner \rangle) & \text{if } \varphi = \bigwedge \Phi \end{cases} \\
 &\stackrel{(6)}{=} \text{Neg}(\ulcorner \varphi \urcorner).
 \end{aligned}$$

Equality (2) holds by definition of codes of propositional formulas. The third equality holds by the induction hypothesis and (4) holds again by definition. Note that " $c \in \ulcorner \Phi \urcorner$ " on the right-hand side of (4) is the abbreviation for " $\text{bit}(c, \ulcorner \Phi \urcorner) = 1$ ". Equality (5) is true by definition of  $\text{Neg}$  and so is equality (6).  $\square$

**Notation.** Let  $\varphi(x_1, \dots, x_r)$  be an unnested  $\mathcal{L}$ -formula. Just like before for  $\mathcal{L}$ -literals, we write  $T_\varphi : \mathbb{N}^{r+1} \rightarrow \mathbb{N}$  for the  $\mathcal{L}_{\mathbb{N}}$ -function that maps  $(m, a_1, \dots, a_r) \in \mathbb{N}^{r+1}$  to  $\ulcorner \langle \varphi(a_1, \dots, a_r) \rangle_{(m)} \urcorner$ . If  $T_\varphi(m; a_1, \dots, a_r) = \langle 0, c \rangle$  or  $T_\varphi(m; a_1, \dots, a_r) = \langle 1, c \rangle$  for a  $c \in \mathbb{N}$ , i.e., the output of  $T_\varphi(m; a_1, \dots, a_r)$  codes a conjunction or disjunction, then we write  $C_\varphi : \mathbb{N}^{r+1} \rightarrow \mathbb{N}$  for the  $\mathcal{L}_{\mathbb{N}}$ -function that maps  $(m, a_1, \dots, a_r) \in \mathbb{N}^{r+1}$  to that  $c$ . Again we write  $C_\varphi(m; a_1, \dots, a_r)$  for the reader's benefit.



**Corollary 3.10.** *Let  $\varphi(x_1, \dots, x_r)$  be an unnested  $\mathcal{L}$ -formula.*

$$\mathfrak{N} \models \forall x, x_1, \dots, x_r \quad T_{\neg\varphi}(x; x_1, \dots, x_r) = \text{Neg}(T_{\varphi}(x; x_1, \dots, x_r))$$

*Proof.* From rule (T4) of the definition of propositional translations, it follows that  $\langle \neg\varphi(a_1, \dots, a_r) \rangle_{(m)} = \neg\langle \varphi(a_1, \dots, a_r) \rangle_{(m)}$  for all  $a_1, \dots, a_r, m \in \mathbb{N}$ . From Lemma 3.9, we obtain that the claim holds.  $\square$

**Lemma 3.11.** *Let  $\mathcal{M}$  be a nonstandard model,  $\mathcal{N}$  an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  and  $d$  be an integer with  $d \geq 0$ .*

$$(a) \quad \mathcal{N} \models \forall x \left( \text{Fml}_{\Pi_d^{prop}}(x) \wedge \left( \text{True}_{\Sigma_d^{prop}}(\text{Neg}(x)) \leftrightarrow \neg \text{True}_{\Pi_d^{prop}}(x) \right) \right)$$

$$(b) \quad \mathcal{N} \models \forall x \left( \text{Fml}_{\Sigma_d^{prop}}(x) \wedge \left( \text{True}_{\Pi_d^{prop}}(\text{Neg}(x)) \leftrightarrow \neg \text{True}_{\Sigma_d^{prop}}(x) \right) \right)$$

*Proof.* Let  $c \in \mathcal{N}$  be such that  $\text{Fml}_{\Pi_d^{prop}}(c)$  is true in  $\mathcal{N}$ . We show these claims by induction on  $d$ . If  $\text{Lit}(c)$  is also true in  $\mathcal{N}$ , then the claim follows from the definitions of  $\text{Neg}$  and  $\text{True}_{\text{Lit}}$ . Suppose now that  $c$  codes a  $(\Pi_d^{prop} \setminus \Sigma_{d-1}^{prop})$ -formula in the sense of  $\mathcal{N}$ , i.e.,  $\text{Fml}_{\Pi_d^{prop}}(c) \wedge \neg \text{Fml}_{\Sigma_{d-1}^{prop}}(c)$  holds in  $\mathcal{N}$ . Then there is a  $c' \in \mathcal{N}$  such that  $c = \langle 1, c' \rangle^{\mathcal{N}}$  and a  $c'' \in \mathcal{N}$  such that  $\text{Neg}^{\mathcal{N}}(c) = \langle 0, c'' \rangle^{\mathcal{N}}$ . The statement  $\neg \text{True}_{\Pi_d^{prop}}(c)$  holds in  $\mathcal{N}$  by definition of  $\text{True}_{\Pi_d^{prop}}$  if and only if  $\neg \text{True}_{\Sigma_{d-1}^{prop}}(c) \wedge \exists y \in c' \neg \text{True}_{\Sigma_{d-1}^{prop}}(y)$  is true. Since  $\text{Fml}_{\Pi_d^{prop}}(c) \wedge \neg \text{Fml}_{\Sigma_{d-1}^{prop}}(c)$  holds in  $\mathcal{N}$ , we obtain that  $\text{True}_{\Sigma_{d-1}^{prop}}(c)$  cannot be true in  $\mathcal{N}$ . By the induction hypothesis, there is a  $c_0 \in \mathcal{N}$  with  $\mathcal{N} \models c_0 \in c'$  such that  $\neg \text{True}_{\Sigma_{d-1}^{prop}}(c_0)$  holds if and only if  $\text{True}_{\Pi_{d-1}^{prop}}(\text{Neg}^{\mathcal{N}}(c_0))$  is true in  $\mathcal{N}$ . By definition of  $\text{Neg}$  and elementary equivalence, we obtain that  $\mathcal{N} \models \text{Neg}(c_0) \in c''$ . Since  $\text{Neg}^{\mathcal{N}}(c) = \langle 0, c'' \rangle^{\mathcal{N}}$ , from the definition of  $\text{True}_{\Sigma_d^{prop}}$  we obtain that  $\mathcal{N} \models \neg \text{True}_{\Pi_d^{prop}}(c)$  if and only if  $\mathcal{N} \models \text{True}_{\Sigma_d^{prop}}(\text{Neg}(c))$ . Analogously, we can show the result for  $c \in \mathcal{N}$  such that  $c = \langle 0, c' \rangle$  for a  $c' \in \mathcal{N}$  which is result (b).  $\square$

We now combine the results just proven to show how to relate the truth of a first-order formula to the truth of its propositional translation.

**Proposition 3.12.** *Let  $\mathcal{M}$  be a nonstandard model,  $\mathcal{N}$  an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  and  $d, k$  be an integer with  $d, k \geq 1$ . Assume  $\varphi(x_1, \dots, x_r)$  is unnested an  $\mathcal{L}$ -sentence.*

(a) *If  $\mathcal{N} \models \forall x, x_1, \dots, x_r \text{ Fml}_{\Pi_d^{prop}}(\text{T}_\varphi(x; x_1, \dots, x_r))$ , then*

$$\mathcal{N} \models \forall x, x_1, \dots, x_r [\varphi^{<x}(x_1, \dots, x_r) \leftrightarrow \text{True}_{\Pi_d^{prop}}(\text{T}_\varphi(x; x_1, \dots, x_r))].$$

(b) *If  $\mathcal{N} \models \forall x, x_1, \dots, x_r \text{ Fml}_{\Sigma_d^{prop}}(\text{T}_\varphi(x; x_1, \dots, x_r))$ , then*

$$\mathcal{N} \models \forall x, x_1, \dots, x_r [\varphi^{<x}(x_1, \dots, x_r) \leftrightarrow \text{True}_{\Sigma_d^{prop}}(\text{T}_\varphi(x; x_1, \dots, x_r))].$$

(c) *If  $\mathcal{N} \models \forall x, x_1, \dots, x_r \text{ Fml}_{\text{DNF}_k}(\text{T}_\varphi(x; x_1, \dots, x_r))$ , then*

$$\mathcal{N} \models \forall x, x_1, \dots, x_r [\varphi^{<x}(x_1, \dots, x_r) \leftrightarrow \text{True}_{\text{DNF}_k}(\text{T}_\varphi(x; x_1, \dots, x_r))].$$

*Proof.* (a): This proof is done by induction on the structure of  $\mathcal{L}'$ -formulas.

Atomic formulas: If  $\varphi(x_1, \dots, x_r)$  is an  $\mathcal{L}'$ -atom, then apply Proposition 3.5. Assume now,  $\varphi(x_1, \dots, x_r)$  is an  $\mathcal{L}_{\mathbb{N}}$ -atom and  $a_1, \dots, a_r, m$  are in  $\mathbb{N}$ . Then by definition of propositional translation

$$\langle \varphi(a_1, \dots, a_r) \rangle_{(m)} = \begin{cases} \top & \text{if } \mathfrak{N} \models \varphi^{<m}(a_1, \dots, a_r) \\ \perp & \text{if } \mathfrak{N} \not\models \varphi^{<m}(a_1, \dots, a_r). \end{cases}$$

Since  $\mathfrak{N} \models \text{True}_{\Pi_d^{prop}}(\top) \wedge \neg \text{True}_{\Pi_d^{prop}}(\perp)$ , we obtain

$$\mathfrak{N} \models \forall x, x_1, \dots, x_r [\varphi^{<x}(x_1, \dots, x_r) \leftrightarrow \text{True}_{\Pi_d^{prop}}(\text{T}_\varphi(x; x_1, \dots, x_r))]$$

and by elementary equivalence this holds in  $\mathcal{N}$  too.

Negations: Assume  $\varphi(x_1, \dots, x_r) = \neg\psi(x_1, \dots, x_r)$  where  $\psi(x_1, \dots, x_r)$  is

an unnested  $\mathcal{L}$ -formula and let  $a_1, \dots, a_r, n \in \mathcal{N}$ .

$$\begin{aligned}
 \mathcal{N} \models \varphi^{<n}(a_1, \dots, a_r) & \text{ if and only if } \mathcal{N} \not\models \psi^{<n}(a_1, \dots, a_r) \\
 & \text{ if and only if } \mathcal{N} \models \neg \text{True}_{\Pi_d^{prop}}(\mathbb{T}_\psi(n; a_1, \dots, a_r)) \\
 & \text{ if and only if } \mathcal{N} \models \text{True}_{\Sigma_d^{prop}}(\text{Neg}(\mathbb{T}_\psi(n; a_1, \dots, a_r))) \\
 & \text{ if and only if } \mathcal{N} \models \text{True}_{\Sigma_d^{prop}}(\mathbb{T}_{\neg\psi}(n; a_1, \dots, a_r))
 \end{aligned}$$

The second equivalence holds by assumption. The third equivalence follows from Lemma 3.11 (a). The last one is obtained by applying Corollary 3.10. Note that this holds in  $\mathcal{N}$  since  $\mathcal{N}$  is an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  where it holds by elementary equivalence.

Conjunctions: Assume  $\varphi(x_1, \dots, x_r) = \psi_1(x_1, \dots, x_r) \wedge \psi_2(x_1, \dots, x_r)$  such that  $\psi_1$  and  $\psi_2$  are unnested  $\mathcal{L}$ -formulas.

$$\begin{aligned}
 \mathcal{N} \models \varphi^{<n}(a_1, \dots, a_r) & \\
 \text{if and only if } \mathcal{N} \models \psi_1^{<n}(a_1, \dots, a_r) \text{ and } \mathcal{N} \models \psi_2^{<n}(a_1, \dots, a_r) & \\
 \text{if and only if } \mathcal{N} \models \text{True}_{\Pi_d^{prop}}(\mathbb{T}_{\psi_1}(n; a_1, \dots, a_r)) \text{ and} & \\
 \mathcal{N} \models \text{True}_{\Pi_d^{prop}}(\mathbb{T}_{\psi_2}(n; a_1, \dots, a_r)) & \\
 \text{if and only if } \mathcal{N} \models \text{True}_{\Pi_d^{prop}}(\mathbb{T}_{\psi_1}(n; a_1, \dots, a_r)) \wedge & \\
 \text{True}_{\Pi_d^{prop}}(\mathbb{T}_{\psi_2}(n; a_1, \dots, a_r)) & \\
 \text{if and only if } \mathcal{N} \models \text{True}_{\Pi_d^{prop}}(\mathbb{T}_{\psi_1 \wedge \psi_2}(n; a_1, \dots, a_r)) &
 \end{aligned}$$

Again, the second equivalence holds by assumption. Now we show the last equivalence. By definition of Conj, the statement

$$\begin{aligned}
 \forall x, x_1, \dots, x_r \quad \mathbb{T}_{\psi_1 \wedge \psi_2}(x; x_1, \dots, x_r) = \\
 \text{Conj}(\mathbb{T}_{\psi_1}(x; x_1, \dots, x_r), \mathbb{T}_{\psi_2}(x; x_1, \dots, x_r))
 \end{aligned}$$

holds in the standard model. Looking at the definition of Conj, one can

conclude that

$$\mathfrak{N} \models \forall x, x_1, \dots, x_r \forall z \in C_{\psi_1 \wedge \psi_2}(x; x_1, \dots, x_r) [z = T_{\psi_1}(x; x_1, \dots, x_r) \vee z = T_{\psi_2}(x; x_1, \dots, x_r) \vee z \in C_{\psi_1}(x; x_1, \dots, x_r) \vee z \in C_{\psi_2}(x; x_1, \dots, x_r)]. \quad (\star)$$

The same way we obtain that

$$\mathfrak{N} \models \forall x, x_1, \dots, x_r [T_{\psi_i}(x; x_1, \dots, x_r) \in C_{\psi_1 \wedge \psi_2}(x; x_1, \dots, x_r) \vee C_{\psi_i}(x; x_1, \dots, x_r) \subseteq C_{\psi_1 \wedge \psi_2}(x; x_1, \dots, x_r)]. \quad (\star\star)$$

for  $i \in \{1, 2\}$ . By elementary equivalence these statements hold in  $\mathcal{M}$  and therefore in  $\mathcal{N}$  as well.

If  $\mathcal{N} \models T_{\psi_1}(n; a_1, \dots, a_r) = \ulcorner \top \urcorner$ , then we have  $T_{\psi_1 \wedge \psi_2}^{\mathcal{N}}(n; a_1, \dots, a_r) = T_{\psi_2}^{\mathcal{N}}(n; a_1, \dots, a_r)$  and the claim clearly holds. Analogously, if we have  $\mathcal{N} \models T_{\psi_2}(n; a_1, \dots, a_r) = \ulcorner \top \urcorner$ .

Assume  $T_{\psi_i}(n; a_1, \dots, a_r) \neq \ulcorner \top \urcorner$  for both  $i \in \{1, 2\}$ .

If  $T_{\psi_i}^{\mathcal{N}}(n; a_1, \dots, a_r) = \langle 1, C_{\psi_i}(n; a_1, \dots, a_r) \rangle^{\mathcal{N}}$  where  $i \in \{1, 2\}$ , it follows that  $\mathcal{N} \models C_{\psi_i}(n; a_1, \dots, a_r) \subseteq C_{\psi_1 \wedge \psi_2}(n; a_1, \dots, a_r)$ . If  $T_{\psi_i}^{\mathcal{N}}(n; a_1, \dots, a_r) \neq \langle 1, C_{\psi_i}(n; a_1, \dots, a_r) \rangle^{\mathcal{N}}$ , then  $\mathcal{N} \models T_{\psi_i}(n; a_1, \dots, a_r) \in C_{\psi_1 \wedge \psi_2}(n; a_1, \dots, a_r)$ .

Assume now  $\text{True}_{\Pi_d^{\text{prop}}}(T_{\psi_i}(n; a_1, \dots, a_r))$  holds in  $\mathcal{N}$  for both  $i \in \{1, 2\}$ .

If  $T_{\psi_i}^{\mathcal{N}}(n; a_1, \dots, a_r) = \langle 1, C_{\psi_i}(n; a_1, \dots, a_r) \rangle^{\mathcal{N}}$ , then by definition of  $\text{True}_{\Pi_d^{\text{prop}}}$ , we obtain that  $\forall z \in C_{\psi_i}(n; a_1, \dots, a_r) \text{ True}_{\Sigma_{d-1}^{\text{prop}}}(z)$  is true in  $\mathcal{N}$ . If we have  $T_{\psi_i}^{\mathcal{N}}(n; a_1, \dots, a_r) \neq \langle 1, C_{\psi_i}(n; a_1, \dots, a_r) \rangle^{\mathcal{N}}$ , then we obtain that  $\mathcal{N} \models \text{True}_{\Sigma_{d-1}^{\text{prop}}}(T_{\psi_i}(n; a_1, \dots, a_r))$ .

Therefore, we obtain by  $(\star)$  that  $\mathcal{N} \models \forall z \in C_{\psi_1 \wedge \psi_2}(n; a_1, \dots, a_r) \text{ True}_{\Sigma_{d-1}^{\text{prop}}}(z)$  and by definition it follows that  $\text{True}_{\Pi_d^{\text{prop}}}(T_{\psi_1 \wedge \psi_2}(n; a_1, \dots, a_r))$  holds in  $\mathcal{N}$ .

Now assume,  $\mathcal{N} \models \text{True}_{\Pi_d^{\text{prop}}}(T_{\psi_1 \wedge \psi_2}(n; a_1, \dots, a_r))$ . Again, by definition

of  $\text{True}_{\Pi_d^{\text{prop}}}$ ,  $\mathcal{N} \models \forall z \in C_{\psi_1 \wedge \psi_2}(n; a_1, \dots, a_r) \text{ True}_{\Sigma_{d-1}^{\text{prop}}}(z)$ . Since by  $(\star\star)$ ,  $\mathcal{N}$  thinks that either  $C_{\psi_i}^{\mathcal{N}}(n; a_1, \dots, a_r)$  is a subset of  $C_{\psi_1 \wedge \psi_2}^{\mathcal{N}}(n; a_1, \dots, a_r)$  or  $T_{\psi_i}^{\mathcal{N}}(n; a_1, \dots, a_r)$  is in  $C_{\psi_1 \wedge \psi_2}^{\mathcal{N}}(n; a_1, \dots, a_r)$ .

Thus, it follows that  $\text{True}_{\Pi_d^{\text{prop}}}(T_{\psi_i}^{\mathcal{N}}(n; a_1, \dots, a_r))$  holds in  $\mathcal{N}$  for both  $i \in \{1, 2\}$ .

Universal quantifications: Let  $\varphi(x_1, \dots, x_r) = \forall y \psi(y, x_1, \dots, x_r)$  for an unnested  $\mathcal{L}$ -formula  $\psi(y, x_1, \dots, x_r)$ .

$$\begin{aligned} \mathcal{N} \models \varphi^{<n}(a_1, \dots, a_r) & \\ \text{if and only if } \mathcal{N} \models \psi^{<n}(c, a_1, \dots, a_r) & \text{ for all } c <^{\mathcal{N}} n \\ \text{if and only if } \mathcal{N} \models \text{True}_{\Pi_d^{\text{prop}}}(T_{\psi}(n; c, a_1, \dots, a_r)) & \text{ for all } c <^{\mathcal{N}} n \\ \text{if and only if } \mathcal{N} \models \forall y < n \text{ True}_{\Pi_d^{\text{prop}}}(T_{\psi}(n; y, a_1, \dots, a_r)) & \\ \text{if and only if } \mathcal{N} \models \text{True}_{\Pi_d^{\text{prop}}}(T_{\varphi}(n; a_1, \dots, a_r)) & \end{aligned}$$

As before, we obtain the second equivalence by assumption. We show the last equivalence.

First of all note that

$$\begin{aligned} \mathfrak{N} \models \forall x, x_1, \dots, x_r [T_{\varphi}(x; x_1, \dots, x_r) = \langle 1, C_{\varphi}(x; x_1, \dots, x_r) \rangle \vee \\ \exists y < x (T_{\varphi}(x; x_1, \dots, x_r) = T_{\psi}(x; y, x_1, \dots, x_r) \wedge \forall z < x (z \neq y \rightarrow \\ T_{\psi}(x; z, x_1, \dots, x_r) = \ulcorner \top \urcorner))] \end{aligned}$$

by definition of propositional translations and their coding functions.

If there is a  $c <^{\mathcal{N}} n$  such that  $T_{\varphi}^{\mathcal{N}}(n; a_1, \dots, a_r) = T_{\psi}^{\mathcal{N}}(n; c, a_1, \dots, a_r)$  and for all  $c' <^{\mathcal{N}} n$  with  $c' \neq c$ , we have  $\mathcal{N} \models T_{\psi}(n; c', a_1, \dots, a_r) = \ulcorner \top \urcorner$ , then the claim clearly holds.

Now assume otherwise, then  $T_{\varphi}^{\mathcal{N}}(n; a_1, \dots, a_r) = \langle 1, C_{\varphi}(n; a_1, \dots, a_r) \rangle^{\mathcal{N}}$ . Suppose  $\mathcal{N} \models \forall y < n \text{ True}_{\Pi_d^{\text{prop}}}(T_{\psi}(n; y, a_1, \dots, a_r))$ . Again by definition of

propositional translations, we obtain

$$\begin{aligned}
 \mathfrak{N} \models \forall x, x_1, \dots, x_r \forall z \in C_\varphi(x; x_1, \dots, x_r) \exists y < x \\
 [z = T_\psi(x; y, x_1, \dots, x_r) \vee \\
 (T_\psi(x; y, x_1, \dots, x_r) = \langle 1, C_\psi(x; y, x_1, \dots, x_r) \rangle \wedge \\
 z \in C_\psi(x; y, x_1, \dots, x_r))] \quad (\diamond)
 \end{aligned}$$

By elementary equivalence, this holds in  $\mathcal{M}$  and thus in  $\mathcal{N}$  too. Let  $c \in \mathcal{N}$  such that  $c <^{\mathcal{N}} n$ . If  $C_\psi(n; c, a_1, \dots, a_r) \subseteq^{\mathcal{N}} C_\varphi(n; a_1, \dots, a_r)$ , then by definition of  $\text{True}_{\Pi_d^{\text{prop}}}$ , we have  $\mathcal{N} \models \forall z \in C_\psi(n; c, a_1, \dots, a_r) \text{True}_{\Sigma_{d-1}^{\text{prop}}}(z)$ . Otherwise  $T_\psi^{\mathcal{N}}(n; c, a_1, \dots, a_r) \in^{\mathcal{N}} C_\varphi^{\mathcal{N}}(n; a_1, \dots, a_r)$ . Applying the statement  $(\diamond)$  as well as the assumption implies  $\mathcal{N} \models \forall z \in C_\varphi(n; a_1, \dots, a_r) \text{True}_{\Sigma_{d-1}^{\text{prop}}}(z)$ . We obtain  $\mathcal{N} \models \text{True}_{\Pi_d^{\text{prop}}}(T_\varphi(n; a_1, \dots, a_r))$ .

Assume now that  $\text{True}_{\Pi_d^{\text{prop}}}(T_\varphi(n; a_1, \dots, a_r))$  holds in  $\mathcal{N}$ . The following holds by definition of codings

$$\begin{aligned}
 \mathfrak{N} \models \forall x, x_1, \dots, x_r \forall y < x [T_\psi(x; y, x_1, \dots, x_r) \in C_\varphi(x; x_1, \dots, x_r) \vee \\
 (T_\psi(x; y, x_1, \dots, x_r) = \langle 1, C_\psi(x; y, x_1, \dots, x_r) \rangle \wedge \\
 C_\psi(x; y, x_1, \dots, x_r) \subseteq C_\varphi(x; x_1, \dots, x_r))] \quad (\diamond\diamond)
 \end{aligned}$$

Since  $\text{True}_{\Pi_d^{\text{prop}}}(T_\varphi(n; a_1, \dots, a_r))$  holds in  $\mathcal{N}$ , by definition of  $\text{True}_{\Pi_d^{\text{prop}}}$ , also  $\forall z \in C_\varphi(n; a_1, \dots, a_r) \text{True}_{\Sigma_{d-1}^{\text{prop}}}(z)$  is true in  $\mathcal{N}$ . Applying the statement  $(\diamond\diamond)$ , we obtain  $\mathcal{N} \models \forall y < n \text{True}_{\Pi_d^{\text{prop}}}(T_\psi(n; y, a_1, \dots, a_r))$ .  $\square$

Next we show how to classify the propositional translations of the True-formulas.

**Proposition 3.13.** *Let  $c, d, k$  and  $m$  be integers with  $k \geq 1$ .*

(a) *The propositional translation  $\langle \text{True}_{\Sigma_d^{\text{prop}}}(c) \rangle_{(m)}$  of  $\text{True}_{\Sigma_d^{\text{prop}}}(c)$  is a  $\Sigma_d^{\text{prop}}$ -formula or equal to  $\perp$ .*

- (b) The propositional translation  $\langle \text{True}_{\Pi_d^{prop}}(c) \rangle_{(m)}$  of  $\text{True}_{\Pi_d^{prop}}(c)$  is a  $\Pi_d^{prop}$ -formula or equal to  $\perp$ .
- (c) The propositional translation  $\langle \text{True}_{\text{DNF}_k}(c) \rangle_{(m)}$  of  $\text{True}_{\text{DNF}_k}(c)$  is a  $k$ -DNF.

*Proof.* We prove (a) and (b) by induction on  $d$ . If  $d = 0$ , then

$$\langle \text{True}_{\Sigma_0^{prop}}(c) \rangle_{(m)} = \langle \text{True}_{\Pi_0^{prop}}(c) \rangle_{(m)} = \langle \text{True}_{\text{Lit}}(c) \rangle_{(m)},$$

which is either a literal or equal to  $\perp$  by Proposition 3.6.

Assume now that the claim holds for  $d - 1$ .

(a): Translating  $\text{True}_{\Sigma_d^{prop}}(c)$  gives us

$$\begin{aligned} \langle \text{True}_{\Sigma_d^{prop}}(c) \rangle_{(m)} &= \langle \text{True}_{\Pi_{d-1}^{prop}}(c) \rangle_{(m)}^{(1)} \vee \bigvee_{y < m} \left[ \langle c = \langle 0, y \rangle \rangle_{(m)}^{(2)} \wedge \right. \\ &\left. \bigvee_{z < m} \left( \langle z \in y \rangle_{(m)}^{(3)} \wedge \langle \text{True}_{\Pi_{d-1}^{prop}}(z) \rangle_{(m)}^{(4)} \right) \wedge \langle \neg \text{Fml}_{\Pi_{d-1}^{prop}}(c) \rangle_{(m)}^{(5)} \wedge \langle \text{Fml}_{\Sigma_d^{prop}}(c) \rangle_{(m)}^{(6)} \right]. \end{aligned}$$

Translation (1) is equal to a  $\Pi_{d-1}^{prop}$ -formula or equal to  $\perp$  by the induction hypothesis.

If the former holds, then for some  $d' < d$ ,  $\langle \text{Fml}_{\Sigma_{d'}^{prop}}(c) \rangle_{(m)}$  or  $\langle \text{Fml}_{\Pi_{d'}^{prop}}(c) \rangle_{(m)}$  is equal to  $\top$ , making translation (5) equal to  $\perp$ . This follows from the fact that they are  $\mathcal{L}_{\mathbb{N}}$ -formulas which are translated to  $\top$  or  $\perp$  by definition. Therefore, the whole disjunction will be equal to  $\perp$  and  $\langle \text{True}_{\Sigma_d^{prop}}(c) \rangle_{(m)}$  is equal to translation (1). This means, the translation of  $\text{True}_{\Sigma_d^{prop}}(c)$  is equal to a  $\Pi_{d-1}^{prop}$ -formula and therefore, a  $\Sigma_d^{prop}$ -formula.

Assume the latter, i.e., translation (1) is equal to  $\perp$  and thus,

$$\begin{aligned} \langle \text{True}_{\Sigma_d^{prop}}(c) \rangle_{(m)} &= \bigvee_{y < m} \left[ \langle c = \langle 0, y \rangle \rangle_{(m)}^{(2)} \wedge \bigvee_{z < m} \left( \langle z \in y \rangle_{(m)}^{(3)} \wedge \right. \right. \\ &\left. \left. \langle \text{True}_{\Pi_{d-1}^{prop}}(z) \rangle_{(m)}^{(4)} \right) \wedge \langle \neg \text{Fml}_{\Pi_{d-1}^{prop}}(c) \rangle_{(m)}^{(5)} \wedge \langle \text{Fml}_{\Sigma_d^{prop}}(c) \rangle_{(m)}^{(6)} \right]. \end{aligned}$$

If translation (5) or (6) is equal to  $\perp$ , then again  $\langle \text{True}_{\Sigma_d^{prop}}(c) \rangle_{(m)} = \perp$ . Suppose now, translations (5) and (6) are equal to  $\top$ , which is equivalent to  $c$  coding a  $\Sigma_d^{prop} \setminus \Pi_{d-1}^{prop}$ -formula, and therefore,

$$\langle \text{True}_{\Sigma_d^{prop}}(c) \rangle_{(m)} = \bigvee_{y < m} \left[ \langle c = \langle 0, y \rangle \rangle_{(m)}^{(2)} \wedge \bigvee_{z < m} \left( \langle z \in y \rangle_{(m)}^{(3)} \wedge \langle \text{True}_{\Pi_{d-1}^{prop}}(z) \rangle_{(m)}^{(4)} \right) \right].$$

If there is an  $a \in \mathbb{N}$  with  $a < m$  such that  $c = \langle 0, a \rangle$ , then translation (2) is equal to  $\top$ . For all other  $a \in \mathbb{N}$ , the whole disjunct is equal to  $\perp$ . This means that if there is no such  $a$ , then  $\langle \text{True}_{\Pi_d^{prop}}(c) \rangle_{(m)} = \perp$ . Assume now that there is such an  $a \in \mathbb{N}$ . Then all other disjuncts are equal to  $\perp$ , making  $\langle \text{True}_{\Sigma_d^{prop}}(c) \rangle_{(m)}$  equal to

$$\bigvee_{z < m} \left( \langle z \in a \rangle_{(m)}^{(3)} \wedge \langle \text{True}_{\Pi_{d-1}^{prop}}(z) \rangle_{(m)}^{(4)} \right).$$

If translation (3) is equal to  $\perp$ , the disjunct is equal to  $\perp$ . If translation (3) is equal to  $\top$ , the conjunction is equal to translation (4). We obtain that

$$\langle \text{True}_{\Sigma_d^{prop}}(c) \rangle_{(m)} = \bigvee_{\substack{z \in a \\ z < m}} \langle \text{True}_{\Pi_{d-1}^{prop}}(z) \rangle_{(m)}. \quad (\star)$$

By the induction hypothesis,  $\langle \text{True}_{\Pi_{d-1}^{prop}}(z) \rangle_{(m)}$  is either a  $\Pi_{d-1}^{prop}$ -formula or equal to  $\perp$ . Therefore, either  $\langle \text{True}_{\Sigma_d^{prop}}(c) \rangle_{(m)}$  is a disjunction of  $\Pi_{d-1}^{prop}$ -formulas and therefore a  $\Sigma_d^{prop}$ -formula or  $\text{True}_{\Sigma_d^{prop}} = \perp$ .

(b): Analogously, we can show result (b) for  $\text{True}_{\Pi_d^{prop}}(c)$ .

(c): The translation of  $\text{True}_{\text{DNF}_k}(c)$  is

$$\langle \text{True}_{\text{DNF}_k}(c) \rangle_{(m)} = \langle \text{Fml}_{\text{DNF}_k}(c) \rangle_{(m)} \wedge \langle \text{True}_{\text{DNF}}(c) \rangle_{(m)}.$$



Assume one of them is equal to  $\perp$ , then  $\langle \text{True}_{\text{DNF}_k}(c) \rangle_{(m)} = \perp$ .

If  $\langle \text{Fml}_{\text{DNF}_k}(c) \rangle_{(m)} = \top$  and  $\langle \text{True}_{\text{DNF}}(c) \rangle_{(m)} \neq \perp$ , then by (a), we obtain  $\langle \text{True}_{\text{DNF}}(c) \rangle_{(m)} \in \Sigma_2^{\text{prop}}$ . Moreover, equation  $(\star)$  holds in this case too with  $d = 2$ . Since  $\langle \text{Fml}_{\text{DNF}_k}(c) \rangle_{(m)} = \top$ , we can conclude that  $a$  in  $(\star)$  codes a set of  $k$ -many codes of  $\Pi_1^{\text{prop}}$ -formulas. Thus,  $\langle \text{True}_{\text{DNF}_k}(c) \rangle_{(m)}$  is a  $k$ -DNF as well.  $\square$

**Corollary 3.14.** *Let  $\mathcal{M}$  be a nonstandard model and  $d, k$  integers with  $k \geq 1$ .*

- (a)  $\mathcal{M} \models \forall x, y \left( \text{Fml}_{\Sigma_d^{\text{prop}}}(\text{T}_{\text{True}_{\Sigma_d^{\text{prop}}}}(x; y)) \vee \text{T}_{\text{True}_{\Sigma_d^{\text{prop}}}}(x; y) = \ulcorner \perp \urcorner \right)$ .
- (b)  $\mathcal{M} \models \forall x, y \left( \text{Fml}_{\Pi_d^{\text{prop}}}(\text{T}_{\text{True}_{\Pi_d^{\text{prop}}}}(x; y)) \vee \text{T}_{\text{True}_{\Pi_d^{\text{prop}}}}(x; y) = \ulcorner \perp \urcorner \right)$ .
- (c)  $\mathcal{M} \models \forall x, y \text{ Fml}_{\text{DNF}_k}(\text{T}_{\text{True}_{\text{DNF}_k}}(x; y))$

*Proof.* This follows from Proposition 3.13, the definition of  $\text{T}_\varphi$  and elementary equivalence since this holds in the standard model.  $\square$

**Corollary 3.15.** *Let  $c, d$  and  $m$  be integers.*

- (a) *The propositional translation  $\langle \neg \text{True}_{\Sigma_d^{\text{prop}}}(c) \rangle_{(m)}$  of  $\neg \text{True}_{\Sigma_d^{\text{prop}}}(c)$  is a  $\Pi_d^{\text{prop}}$ -formula or equal to  $\top$ .*
- (b) *The propositional translation  $\langle \neg \text{True}_{\Pi_d^{\text{prop}}}(c) \rangle_{(m)}$  of  $\neg \text{True}_{\Pi_d^{\text{prop}}}(c)$  is a  $\Sigma_d^{\text{prop}}$ -formula or equal to  $\top$ .*
- (c) *The propositional translation  $\langle \neg \text{True}_{\text{DNF}_k}(c) \rangle_{(m)}$  of  $\neg \text{True}_{\text{DNF}_k}(c)$  is a  $k$ -CNF.*

*Proof.* This follows from Proposition 3.13 and rule (T4) of the definition of propositional translations.  $\square$

**Corollary 3.16.** *Let  $\mathcal{M}$  be a nonstandard model and  $d, k$  integers with  $k \geq 1$*

- (a)  $\mathcal{M} \models \forall x, y \left( \text{Fml}_{\Pi_d^{\text{prop}}}(\text{T}_{\neg \text{True}_{\Sigma_d^{\text{prop}}}}(x; y)) \vee \text{T}_{\neg \text{True}_{\Sigma_d^{\text{prop}}}}(x; y) = \ulcorner \top \urcorner \right)$ .
- (b)  $\mathcal{M} \models \forall x, y \left( \text{Fml}_{\Sigma_d^{\text{prop}}}(\text{T}_{\neg \text{True}_{\Pi_d^{\text{prop}}}}(x; y)) \vee \text{T}_{\neg \text{True}_{\Pi_d^{\text{prop}}}}(x; y) = \ulcorner \top \urcorner \right)$ .

$$(c) \mathcal{M} \models \forall x, y \text{ Fml}_{\text{CNF}_k}(\text{T}_{\neg \text{True}_{\text{DNF}_k}}(x; y))$$

*Proof.* This follows from Corollary 3.15, the definition of  $\text{T}_\varphi$  and elementary equivalence since this holds in the standard model.  $\square$

### 3.6 Coding refutations

**Notation.** We use the  $\mathcal{L}_{\mathbb{N}}$ -formula  $\text{IsSequence}(x)$  stating "x codes a sequence".

Also we write  $\text{size}_{\text{Fml}}(x)$  for the  $\mathcal{L}_{\mathbb{N}}$ -function mapping  $x$  to the size of the propositional formula it codes or, if  $x$  codes a refutation, then  $\text{size}_{\text{Refut}}(x)$  maps  $x$  to the size of the refutation.

**Definition.** We define the formulas  $\text{Rule}_\vee$ ,  $\text{Rule}_{\text{cut}}$  and  $\text{Rule}_\wedge$ .

$$\text{Rule}_\vee(x, y, z) = (z = \text{Disj}(x, y))$$

$$\begin{aligned} \text{Rule}_{\text{cut}}(x, y, z) = \exists u, v, x', y' \ (x = \text{Disj}(x', u) \wedge y = \text{Disj}(y', v) \wedge \\ u = \text{Neg}(v) \wedge z = \text{Disj}(x', y')) \end{aligned}$$

$$\begin{aligned} \text{Rule}_\wedge(x, y, z) = \exists u, v, x', y', u', v' \ (x = \text{Disj}(x', u) \wedge y = \text{Disj}(y', v) \wedge \\ (\text{Lit}(u) \vee u = \langle 1, u' \rangle) \wedge (\text{Lit}(v) \vee v = \langle 1, v' \rangle) \wedge \\ z = \text{Disj}(x', \text{Disj}(y', \text{Conj}(u, v)))) \end{aligned}$$

We define the formulas  $\text{Refut}_{\mathcal{F}(d)}$ ,  $\text{Refut}_{\text{DNF}}$  and  $\text{Refut}_{\text{DNF}_k}$  for integers  $d, k \geq 1$ .

$$\begin{aligned} \text{Refut}_{\mathcal{F}(1)}(x, y) = \forall z \in x \ \text{Fml}_{\Sigma_1^{\text{prop}}}(z) \wedge \text{IsSequence}(y) \wedge (y)_{\|y\|-1} = \ulcorner \perp \urcorner \wedge \\ \forall u \leq \|y\| - 1 \left[ \text{Fml}_{\Sigma_1^{\text{prop}}}((y)_u) \wedge \left( (y)_u \in x \vee \exists v, w < u \right. \right. \\ \left. \left. (\text{Rule}_\vee((y)_v, (y)_w, (y)_u) \vee \text{Rule}_{\text{cut}}((y)_v, (y)_w, (y)_u)) \right) \right] \end{aligned}$$

$$\begin{aligned} \text{Refut}_{\mathcal{F}(d)}(x, y) = & \forall z \in x \text{ Fml}_{\Sigma_d^{\text{prop}}}(z) \wedge \text{IsSequence}(y) \wedge (y)_{\|y\|-1} = \ulcorner \perp \urcorner \wedge \\ & \forall u \leq \|y\| - 1 \left[ \text{Fml}_{\Sigma_d^{\text{prop}}}((y)_u) \wedge \left( (y)_u \in x \vee \exists v, w < u \right. \right. \\ & \left. \left. \left( \text{Rule}_{\vee}((y)_v, (y)_w, (y)_u) \vee \text{Rule}_{\text{cut}}((y)_v, (y)_w, (y)_u) \vee \right. \right. \right. \\ & \left. \left. \left. \text{Rule}_{\wedge}((y)_v, (y)_w, (y)_u) \right) \right) \right] \end{aligned}$$

$$\text{Refut}_{\text{DNF}}(x, y) = \text{Refut}_{\mathcal{F}(2)}(x, y)$$

$$\begin{aligned} \text{Refut}_{\text{DNF}_k}(x, y) = & \text{Refut}_{\text{DNF}}(x, y) \wedge \forall z \in x \text{ Fml}_{\text{DNF}_k}(z) \wedge \\ & \forall u \leq \|y\| - 1 \text{ Fml}_{\text{DNF}_k}((y)_u) \end{aligned}$$

**Note.** The  $\text{Refut}_{\mathcal{F}(d)}$ -formula states "y codes an  $\mathcal{F}(d)$ -refutation of the set of propositional formulas coded by x". The formulas  $\text{Refut}_{\text{DNF}}$  and  $\text{Refut}_{\text{DNF}_k}$  are defined analogously.

The formula  $\text{Rule}_{\vee}(x, y, z)$  states "the propositional formula coded by z is obtained from the propositional formulas coded by x and y by applying the weakening rule". In the same way we defined  $\text{Rule}_{\text{cut}}(x, y, z)$  and  $\text{Rule}_{\wedge}(x, y, z)$ . Note that we do not need to define a rule for  $\wedge$ -introduction of  $k$ -DNF's separately. The reason is that we have restricted all formulas in a  $k$ -DNF refutation to be  $k$ -DNF's and therefore, the formula obtained by  $\wedge$ -instruction has to be a  $k$ -DNF as well.

Further note that all formulas above are  $\mathcal{L}_{\mathbb{N}}$ -formulas since they are only defined by functions (including Neg, Disj and Conj) and  $\mathcal{L}_{\mathbb{N}}$ -formulas.

Now we show that the soundness of the rules of  $\mathcal{F}(d)$  also holds for coded refutations.

**Proposition 3.17.** *Let  $\mathcal{M}$  be a nonstandard model,  $\mathcal{N}$  an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  and  $d, k$  integers with  $d, k \geq 1$ . Then the following holds.*

$$(a) \mathcal{N} \models \forall x, y, z \left( \text{True}_{\Sigma_d^{\text{prop}}}(x) \wedge \text{Fml}_{\Sigma_d^{\text{prop}}}(y) \wedge \text{Rule}_{\vee}(x, y, z) \rightarrow \text{True}_{\Sigma_d^{\text{prop}}}(z) \right)$$

$$(b) \mathcal{N} \models \forall x, y, z \left( \text{True}_{\text{DNF}_k}(x) \wedge \text{Fml}_{\text{DNF}_k}(y) \wedge \text{Rule}_{\vee}(x, y, z) \rightarrow \text{True}_{\text{DNF}_k}(z) \right)$$

*Proof.* We argue in  $\mathcal{N}$ . Let  $a, b, c \in \mathcal{N}$  and assume

$$\mathcal{N} \models \text{True}_{\Sigma_d^{\text{prop}}}(a) \wedge \text{Fml}_{\Sigma_d^{\text{prop}}}(b) \wedge \text{Rule}_{\vee}(a, b, c).$$

(a): If  $\mathcal{N}$  thinks that  $a$  codes a disjunction, then there is an  $a' \in \mathcal{N}$  such that  $a = \langle 0, a' \rangle^{\mathcal{N}}$ ,  $a' \neq \ulcorner \emptyset \urcorner$  and there is an  $a_0 \in \mathcal{N}$  with  $a_0 \in^{\mathcal{N}} a'$  such that  $\text{True}_{\Pi_{d-1}^{\text{prop}}}(a_0)$  is true because  $\text{True}_{\Sigma_d^{\text{prop}}}(a)$  is true. By definition of Disj there is a  $c' \in \mathcal{N}$  such that  $c = \langle 0, c' \rangle^{\mathcal{N}}$ . By Lemma 3.4 (a),  $a_0 \in^{\mathcal{N}} c'$ . Therefore,  $\text{True}_{\Sigma_d^{\text{prop}}}(c)$  is true.

If  $\mathcal{N}$  thinks that  $a$  does not code a disjunction, then  $\text{True}_{\Pi_{d-1}^{\text{prop}}}(a)$  is also true. By definition of Disj and Lemma 3.4 (a), we obtain that either  $c = a$  or there is a  $c' \in \mathcal{N}$  such that  $c = \langle 0, c' \rangle^{\mathcal{N}}$  and  $a \in^{\mathcal{N}} c'$  and hence,  $\text{True}_{\Sigma_d^{\text{prop}}}(c)$  holds.

(b): This follows from (a). □

**Proposition 3.18.** *Let  $\mathcal{M}$  be a nonstandard model,  $\mathcal{N}$  an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  and  $d, k$  integers with  $d, k \geq 1$ . Then the following holds.*

$$(a) \quad \mathcal{N} \models \forall x, y, z \left( \text{True}_{\Sigma_d^{\text{prop}}}(x) \wedge \text{True}_{\Sigma_d^{\text{prop}}}(y) \wedge \text{Rule}_{\text{cut}}(x, y, z) \rightarrow \text{True}_{\Sigma_d^{\text{prop}}}(z) \right)$$

$$(b) \quad \mathcal{N} \models \forall x, y, z \left( \text{True}_{\text{DNF}_k}(x) \wedge \text{True}_{\text{DNF}_k}(y) \wedge \text{Rule}_{\text{cut}}(x, y, z) \rightarrow \text{True}_{\text{DNF}_k}(z) \right)$$

*Proof.* (a): Again, we argue in  $\mathcal{N}$ . Let  $a, b, c \in \mathcal{N}$  and assume  $\mathcal{N} \models (\text{True}_{\Sigma_d^{\text{prop}}}(a) \wedge \text{True}_{\Sigma_d^{\text{prop}}}(b) \wedge \text{Rule}_{\text{cut}}(a, b, c))$ .

Then there are integers  $a_1, b_1, e \in \mathcal{N}$  such that  $a = \text{Disj}^{\mathcal{N}}(a_1, e)$  and  $b = \text{Disj}^{\mathcal{N}}(b_1, \text{Neg}^{\mathcal{N}}(e))$ .

Case 1: Assume  $a = a_1$ . Then  $e = \ulcorner \perp \urcorner$ .

If  $\mathcal{N}$  thinks,  $a$  codes a disjunction, then there exists an  $a' \in \mathcal{N}$  such that  $a = \langle 0, a' \rangle^{\mathcal{N}}$  and there is an  $a_0 \in \mathcal{N}$  with  $a_0 \in^{\mathcal{N}} a'$  such that  $\text{True}_{\Pi_{d-1}^{\text{prop}}}(a_0)$  holds. By definition of Disj there is a  $c' \in \mathcal{N}$  such that  $c = \langle 0, c' \rangle^{\mathcal{N}}$  and

$a_0 \in^{\mathcal{N}} c'$ .

If  $\mathcal{N}$  thinks,  $a$  does not code a disjunction, then either  $a \in^{\mathcal{N}} c'$  or, if  $b_1 = \ulcorner \perp \urcorner$ , then  $a = c$  holds.

In both cases we obtain  $\text{True}_{\Sigma_d^{\text{prop}}}(c)$ .

Case 2: If  $a = e$ , then by Lemma 3.11 (a),  $\neg \text{True}_{\Pi_d^{\text{prop}}}(\text{Neg}^{\mathcal{N}}(e))$  is true and therefore,  $\text{True}_{\Sigma_d^{\text{prop}}}(b_1)$  has to be true. Analogously to above, we obtain that  $\text{True}_{\Sigma_d^{\text{prop}}}(c)$  holds. Note that in this case,  $\text{Fml}_{\Pi_{d-1}^{\text{prop}}}(e)$  has to be true in  $\mathcal{N}$  because if  $\mathcal{N} \models \text{Fml}_{\Sigma_d^{\text{prop}}}(e) \wedge \neg \text{Fml}_{\Pi_{d-1}^{\text{prop}}}(e)$ , the statement  $\text{Fml}_{\Pi_d^{\text{prop}}}(\text{Neg}^{\mathcal{N}}(e))$  would also be true in  $\mathcal{N}$  and thus,  $\text{Fml}_{\Sigma_d^{\text{prop}}}(\text{Disj}^{\mathcal{N}}(b_1, \text{Neg}^{\mathcal{N}}(e)))$  would be false in  $\mathcal{N}$ . Therefore,  $\text{True}_{\Sigma_d^{\text{prop}}}(b)$  would be false as well, which contradicts the assumption.

Case 3: Now assume  $a \neq a_1$  and  $a \neq e$ . Then there is an  $a' \in \mathcal{N}$  such that  $\text{Disj}^{\mathcal{N}}(a_1, e) = \langle 0, a' \rangle^{\mathcal{N}}$ .

Since  $\text{True}_{\Sigma_d^{\text{prop}}}(a)$  is true, there is an  $a_0 \in \mathcal{N}$  with  $a_0 \in^{\mathcal{N}} a'$  such that  $\text{True}_{\Pi_{d-1}^{\text{prop}}}(a_0)$  holds. Further, there exists a  $c'$  such that  $c = \langle 0, c' \rangle^{\mathcal{N}}$ .

- $\mathcal{N}$  thinks,  $a_1$  codes a disjunction: Then there is an  $a'_1 \in \mathcal{N}$  such that  $a_1 = \langle 0, a'_1 \rangle^{\mathcal{N}}$ .
  - $a_0 \in^{\mathcal{N}} a'_1$ : Then by definition of  $\text{Disj}$ , also  $a_0 \in^{\mathcal{N}} c'$  and therefore,  $\text{True}_{\Sigma_d^{\text{prop}}}(c)$  is true.
  - $\mathcal{N}$  thinks,  $e = \langle 0, e' \rangle^{\mathcal{N}}$  codes a disjunction,  $a_0 \in^{\mathcal{N}} e'$ : By Lemma 3.11 (b), we obtain that  $\neg \text{True}_{\Pi_d^{\text{prop}}}(\text{Neg}^{\mathcal{N}}(e))$  holds as well. Thus,  $\text{True}_{\Sigma_d^{\text{prop}}}(b_1)$  holds. Then either  $b_1 \in^{\mathcal{N}} c'$  or there is a  $b'_1 \in \mathcal{N}$  such that  $b_1 = \langle 0, b'_1 \rangle^{\mathcal{N}}$  and a  $b_0 \in \mathcal{N}$  with  $b_0 \in^{\mathcal{N}} b'_1$  such that  $\text{True}_{\Pi_{d-1}^{\text{prop}}}(b_0)$  holds. If the latter applies, then by definition of  $\text{Disj}$ ,  $b_0 \in^{\mathcal{N}} c'$  and we obtain  $\mathcal{N} \models \text{True}_{\Sigma_d^{\text{prop}}}(c)$ .
  - $\mathcal{N}$  thinks,  $e$  does not code a disjunction,  $a_0 = e$ : Then by Lemma 3.11 (a),  $\neg \text{True}_{\Sigma_d^{\text{prop}}}(\text{Neg}^{\mathcal{N}}(e))$  is true. By the same argument as

before we have  $\mathcal{N} \models \text{True}_{\Sigma_d^{\text{prop}}}(c)$ .

•  $\mathcal{N}$  thinks,  $a_1$  does not code a disjunction:

- $a_0 = a_1$ : Then we obtain  $a_0 \in^{\mathcal{N}} c'$  and therefore,  $\text{True}_{\Sigma_d^{\text{prop}}}(c)$  is true.
- $a_0 \neq a_1$ : As before we obtain that  $\neg \text{True}_{\Sigma_d^{\text{prop}}}(\text{Neg}^{\mathcal{N}}(e))$  is true and thus,  $\text{True}_{\Sigma_d^{\text{prop}}}(c)$  holds.

(b): follows from (a). □

**Proposition 3.19.** *Let  $\mathcal{M}$  be a nonstandard model,  $\mathcal{N}$  an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  and  $d, k$  integers with  $d \geq 2$  and  $k \geq 1$ . Then the following holds.*

$$(a) \quad \mathcal{N} \models \forall x, y, z \left( (\text{True}_{\Sigma_d^{\text{prop}}}(x) \wedge \text{True}_{\Sigma_d^{\text{prop}}}(y) \wedge \text{Rule}_{\wedge}(x, y, z)) \rightarrow \text{True}_{\Sigma_d^{\text{prop}}}(z) \right)$$

$$(b) \quad \mathcal{N} \models \forall x, y, z \left( (\text{True}_{\text{DNF}_k}(x) \wedge \text{True}_{\text{DNF}_k}(y) \wedge \text{Rule}_{\wedge}(x, y, z) \wedge \text{Fml}_{\text{DNF}_k}(z)) \rightarrow \text{True}_{\text{DNF}_k}(z) \right)$$

*Proof.* We argue in  $\mathcal{N}$  as before and assume for  $a, b, c \in \mathcal{N}$  that

$$\text{True}_{\Sigma_d^{\text{prop}}}(a) \wedge \text{True}_{\Sigma_d^{\text{prop}}}(b) \wedge \text{Rule}_{\wedge}(a, b, c) \rightarrow \text{True}_{\Sigma_d^{\text{prop}}}(c)$$

holds. Note that there are  $a_1, b_1, e_1, e_2 \in \mathcal{N}$  such that  $a = \text{Disj}^{\mathcal{N}}(a_1, e_1)$  and  $b = \text{Disj}^{\mathcal{N}}(b_1, e_2)$ .

Assume  $\mathcal{N}$  thinks  $c$  does not code a disjunction. We obtain that  $a_1 = b_1 = \ulcorner \perp \urcorner$ . Then  $\text{True}_{\Pi_{d-1}^{\text{prop}}}(e_1)$  and  $\text{True}_{\Pi_{d-1}^{\text{prop}}}(e_2)$  have to hold. If  $\mathcal{N}$  thinks both  $e_1$  and  $e_2$  code conjunctions, there are  $e'_1$  and  $e'_2$  in  $\mathcal{N}$  such that  $e_1 = \langle 1, e'_1 \rangle^{\mathcal{N}}$  and  $e_2 = \langle 1, e'_2 \rangle^{\mathcal{N}}$ . Because both  $\text{True}_{\Pi_{d-1}^{\text{prop}}}(e_1)$  and  $\text{True}_{\Pi_{d-1}^{\text{prop}}}(e_2)$  hold, the statements  $\forall e''_1 \in e'_1 \text{ True}_{\Sigma_{d-2}^{\text{prop}}}(e''_1)$  and  $\forall e''_2 \in e'_2 \text{ True}_{\Sigma_{d-2}^{\text{prop}}}(e''_2)$  are true. By definition of  $\text{Conj}$  and by Lemma 3.4 (a), there exists an  $e' \in \mathcal{N}$  with  $\text{Conj}^{\mathcal{N}}(e_1, e_2) = \langle 1, e' \rangle^{\mathcal{N}}$  and

$$\forall e'' \left( (e'' \in e'_1 \vee e'' \in e'_2) \leftrightarrow e'' \in e' \right)$$

are true. Therefore,  $\text{True}_{\Pi_{d-1}^{prop}}(\text{Conj}^{\mathcal{N}}(e_1, e_2))$  holds. By Lemma 3.8 (d), we obtain that  $\text{True}_{\Sigma_d^{prop}}(c)$  is true. If  $\mathcal{N}$  thinks at least one of  $e_i$  for  $i \in \{1, 2\}$  does not code conjunction, we obtain that  $e_i \in e'$  and analogously that  $\text{True}_{\Pi_{d-1}^{prop}}(\text{Conj}^{\mathcal{N}}(e_1, e_2))$  holds.

Assume  $\mathcal{N}$  thinks that  $c$  codes a disjunction. Then there is a  $c' \in \mathcal{N}$  such that  $c = \langle 0, c' \rangle^{\mathcal{N}}$ .

Since  $\text{True}_{\Pi_d^{prop}}(a)$  holds,  $\text{True}_{\Sigma_d^{prop}}(a_1)$  or  $\text{True}_{\Sigma_d^{prop}}(e_1)$  also hold. If  $\mathcal{N}$  thinks  $a_1$  codes a disjunction, then there exists an  $a'_1 \in \mathcal{N}$  such that  $a_1 = \langle 0, a'_1 \rangle^{\mathcal{N}}$ .

Assume that  $\text{True}_{\Sigma_d^{prop}}(a_1)$  holds. Then, if  $\mathcal{N}$  thinks,  $a_1$  codes a disjunction, there is an  $a_0 \in \mathcal{N}$  such that  $a_0 \in^{\mathcal{N}} a'_1$  and  $\text{True}_{\Pi_{d-1}^{prop}}(a_0)$  holds. If  $\mathcal{N}$  thinks  $a_1$  does not code a disjunction, let  $a_0 = a_1$ . By definition of  $\text{Rule}_{\wedge}$  and of  $\text{Disj}$ , we obtain that  $\mathcal{N} \models a_0 \in c'$  and thus,  $\text{True}_{\Sigma_d^{prop}}(c)$  holds.

If  $\text{True}_{\Pi_d^{prop}}(a_1)$  does not hold, then  $\text{True}_{\Pi_{d-1}^{prop}}(e_1)$  has to be true.

If  $\mathcal{N}$  thinks,  $b_1$  codes a disjunction, then there is a  $b'_1 \in \mathcal{N}$  such that  $b_1 = \langle 0, b'_1 \rangle^{\mathcal{N}}$ . Assume now,  $\text{True}_{\Sigma_d^{prop}}(b_1)$  holds. Then, if  $\mathcal{N}$  thinks  $b_1$  codes a disjunction, there is a  $b_0 \in \mathcal{N}$  with  $b_0 \in^{\mathcal{N}} b'_1$  such that  $\text{True}_{\Pi_{d-1}^{prop}}(b_0)$  holds. If  $\mathcal{N}$  thinks  $b_1$  does not code a disjunction, let  $b_0 = b_1$ . As before, we obtain  $b_0 \in^{\mathcal{N}} c'$ . Assume now that neither of both cases apply. Then  $\text{True}_{\Pi_{d-1}^{prop}}(e_1)$  and  $\text{True}_{\Pi_{d-1}^{prop}}(e_2)$  are true. Analogously to above, we obtain that  $\text{True}_{\Sigma_d^{prop}}(c)$  holds.

(b): follows from (a). □

**Corollary 3.20.** *Let  $\mathcal{M}$  be a nonstandard model,  $\mathcal{N}$  an  $\mathcal{L}$ -expansion of  $\mathcal{M}$  and  $d \geq 2$  and  $k \geq 1$  be integers. Then the following holds.*

$$(a) \quad \mathcal{N} \models \forall x, y, z \left( \text{True}_{\Sigma_d^{prop}}(x) \wedge \text{True}_{\Sigma_d^{prop}}(y) \wedge (\text{Rule}_{\vee}(x, y, z) \vee \text{Rule}_{\text{cut}}(x, y, z) \vee \text{Rule}_{\wedge}(x, y, z)) \rightarrow \text{True}_{\Sigma_d^{prop}}(z) \right)$$

$$(b) \quad \mathcal{N} \models \forall x, y, z \left( \text{True}_{\Sigma_1^{prop}}(x) \wedge \text{True}_{\Sigma_1^{prop}}(y) \wedge (\text{Rule}_{\vee}(x, y, z) \vee \text{Rule}_{\text{cut}}(x, y, z)) \rightarrow \text{True}_{\Sigma_d^{prop}}(z) \right)$$

$$(c) \mathcal{N} \models \forall x, y, z \left( \text{True}_{\text{DNF}_k}(x) \wedge \text{True}_{\text{DNF}_k}(y) \wedge \text{Fml}_{\text{DNF}_k}(z) \wedge \right. \\ \left. (\text{Rule}_{\vee}(x, y, z) \vee \text{Rule}_{\text{cut}}(x, y, z) \vee \text{Rule}_{\wedge}(x, y, z)) \rightarrow \text{True}_{\text{DNF}_k}(z) \right)$$

*Proof.* This follows from Propositions 3.17, 3.18 and 3.19.  $\square$

We finish this section with a technical lemma.

**Lemma 3.21.** *The following holds in  $\mathfrak{N}$ :*

$$\forall x \|x\| \leq \text{size}_{\text{Refut}}(x).$$

*Proof.* This follows from the definitions of  $\text{size}_{\text{Refut}}(x)$  and  $\|x\|$  since every formula has at least size 1.  $\square$

### 3.7 Simulation of propositional refutations

In this section we want to combine all results of this chapter to show that there are no polynomial size  $\mathcal{F}(d)$ -refutations of the propositional translations of an unnested  $\mathcal{L}$ -sentence  $\varphi$  if there is an  $\mathcal{L}$ -expansion of a countable nonstandard model in which  $\varphi$  is true on an initial segment.

**Proposition 3.22.** *Assume that  $\varphi$  is an unnested universal  $\mathcal{L}$ -sentence  $\varphi = \forall x_1, \dots, x_r \varphi_0(x_1, \dots, x_r)$  where  $\varphi_0(x_1, \dots, x_r)$  is a quantifier-free  $\mathcal{L}$ -formula of the form  $\bigwedge_{i \in I} \bigvee_{j \in J_i} \varphi_{i,j}(x_1, \dots, x_r)$  for finite index sets  $I \neq \emptyset$  and  $J_i \neq \emptyset$  such that  $i \in I$  and  $\varphi_{i,j}(x_1, \dots, x_r)$  are  $\mathcal{L}$ -literals.*

(a) *The propositional translation  $\langle \varphi \rangle_{(m)}$  of  $\varphi$  is in conjunctive normal form for all  $m \in \mathbb{N}$ .*

(b)  $\mathfrak{N} \models \forall x \text{ Fml}_{\Pi_2^{\text{prop}}}(\text{T}_{\varphi}(x))$ .

*Proof.* (a): By the rules of propositional translation, we obtain

$$\langle \varphi \rangle_{(m)} = \bigwedge_{x_1 < m} \cdots \bigwedge_{x_r < m} \bigwedge_{i \in I} \bigvee_{j \in J_i} \langle \varphi_{i,j}(x_1, \dots, x_r) \rangle_{(m)}.$$



Because of the fact that  $\varphi_{i,j}(x_1, \dots, x_r)$  are unnested  $\mathcal{L}$ -literals, their propositional translation  $\langle \varphi_{i,j}(x_1, \dots, x_r) \rangle_{(m)}$  is either equal to  $\top$ ,  $\perp$  or a literal. Thus, we obtain that  $\langle \varphi \rangle_{(m)}$  is in conjunctive normal form. Note that by definition  $\top$  and  $\perp$  are also in conjunctive normal form and therefore, the claim also holds if the translation is equal to either of them.

(b): follows from (a) □

**Definition.** Let  $d \geq 1$  be an integer. We write  $\Lambda_d$  for the set of all unnested  $\mathcal{L}$ -formulas  $\varphi(x, w_1, \dots, w_r)$  such that for all  $a, b_1, \dots, b_r, m \in \mathbb{N}$ , their propositional translations  $\langle \varphi(a, b_1, \dots, b_r) \rangle_{(m)}$  are  $\Pi_d^{prop}$ -formulas.

**Note.** Note that for every  $\mathcal{L}$ -formula  $\varphi(x, w_1, \dots, w_r) \in \Lambda_d$  the propositional translation  $\langle \neg \varphi(a, b_1, \dots, b_r) \rangle_{(m)}$  of its negation is a propositional formula in  $\Sigma_d^{prop}$  for  $a, b_1, \dots, b_r, m \in \mathbb{N}$ .

Further note that

$$\mathfrak{N} \models \forall z, x, w_1, \dots, w_r [\text{Fml}_{\Pi_d^{prop}}(\top_\varphi(z; x, w_1, \dots, w_r)) \wedge \text{Fml}_{\Sigma_d^{prop}}(\top_{\neg\varphi}(z; x, w_1, \dots, w_r))]$$

holds for all  $\varphi(x, w_1, \dots, w_r) \in \Lambda_d$ .

**Definition.** The least number principle for a set of first-order formulas  $\Theta$  is the statement that for any  $\theta(x, w_1, \dots, w_r) \in \Theta$  the following holds:

$$\forall w_1, \dots, w_r \exists x [\theta(x, w_1, \dots, w_r) \rightarrow \exists y \leq x \forall z \leq x (\theta(y, w_1, \dots, w_r) \wedge (z < y \rightarrow \neg \theta(z, w_1, \dots, w_r)))]$$

We write  $\text{LNP}(\Theta)$ .

Let  $\mathcal{M}$  be a structure that interprets  $<$  as a linear order on its universe and let  $b_0 \in \mathcal{M}$ . For a set of first-order formulas  $\Theta$  the least number principle up to  $b_0$  is the statement that for any  $\theta(x, w_1, \dots, w_r) \in \Theta$  in  $\mathcal{M}$  the

following holds:

$$\forall w_1, \dots, w_r \exists x < b_0 [\theta(x, w_1, \dots, w_r) \rightarrow \exists y \leq x \forall z \leq x (\theta(y, w_1, \dots, w_r) \wedge (z < y \rightarrow \neg\theta(z, w_1, \dots, w_r)))]].$$

We write  $\text{LNP}^{<b_0}(\Theta)$ .

**Definition.** Let  $\vartheta_{\text{Tot}}(x)$  be the propositional conjunction

$$\bigwedge \left\{ \bigvee \left\{ f_{a_1, \dots, a_{\text{ar}(f)+1}} \mid a_{\text{ar}(f)+1} < x \right\} \mid \ulcorner f \urcorner \in \text{Fct}, a_1, \dots, a_{\text{ar}(f)} < x \right\}$$

and let  $\vartheta_{\text{Fct}}(x)$  be the propositional conjunction

$$\bigwedge \left\{ \bigvee \left\{ \neg f_{a_1, \dots, a_{\text{ar}(f)}, c}, \neg f_{a_1, \dots, a_{\text{ar}(f)}, d} \right\} \mid \ulcorner f \urcorner \in \text{Fct}, a_1, \dots, a_{\text{ar}(f)}, c, d < x, c \neq d \right\}$$

**Note.** Both  $\vartheta_{\text{Tot}}(x)$  and  $\vartheta_{\text{Fct}}(x)$  are conjunctions of  $\Sigma_1^{\text{prop}}$ -formulas.

**Definition.** Let  $C_{\text{Tot}}(x)$  be the  $\mathcal{L}_{\mathbb{N}}$ -function such that for any  $m \in \mathbb{N}$ , the conjunction  $\vartheta_{\text{Tot}}(m)$  is coded by  $\langle 1, C_{\text{Tot}}(m) \rangle$  and let  $C_{\text{Fct}}(x)$  be the  $\mathcal{L}_{\mathbb{N}}$ -function such that for any  $m \in \mathbb{N}$ , the conjunction  $\vartheta_{\text{Fct}}(m)$  is coded by  $\langle 1, C_{\text{Fct}}(m) \rangle$ .

**Lemma 3.23.** Let  $\mathcal{M}$  be a nonstandard model and  $\mathcal{N}$  an  $\mathcal{L}$ -expansion of  $\mathcal{M}$ . Then

$$\mathcal{N} \models \forall x \left( \text{True}_{\Pi_2^{\text{prop}}}(\langle 1, C_{\text{Tot}}(x) \rangle) \wedge \text{True}_{\Pi_2^{\text{prop}}}(\langle 1, C_{\text{Fct}}(x) \rangle) \right).$$

*Proof.* Assume for an  $n \in \mathcal{N}$ , the sentence  $\text{True}_{\Pi_2^{\text{prop}}}(\langle 1, C_{\text{Fct}}(n) \rangle)^{\mathcal{N}}$  is false in  $\mathcal{N}$ . Then there is an  $f \in \mathcal{L}'$  and  $a_1, \dots, a_{\text{ar}(f)}, c, d < n$  with  $c \neq d$ , such that  $\neg \text{True}_{\Sigma_2^{\text{prop}}}(\text{T}_{\neg f(a_1, \dots, a_{\text{ar}(f)})=c \vee \neg f(a_1, \dots, a_{\text{ar}(f)})=d})$  is true in  $\mathcal{M}$ . By definition of  $\text{True}_{\Sigma_d^{\text{prop}}}$  we obtain that this means that  $\text{True}_{\text{Lit}}(\text{T}_{f(a_1, \dots, a_{\text{ar}(f)})=c}) \wedge \text{True}_{\text{Lit}}(\text{T}_{f(a_1, \dots, a_{\text{ar}(f)})=d})$  holds in  $\mathcal{N}$ . By Proposition 3.5, it follows that

$\mathcal{N} \models f(a_1, \dots, a_r) = c \wedge f(a_1, \dots, a_r) = d$  contradicting that  $f$  is a function.

Analogously, the same holds for  $\text{True}_{\Pi_2^{\text{prop}}}(\langle 1, C_{\text{Tot}}(x) \rangle)$ .  $\square$

### 3.7.1 The main theorem

**Theorem 3.24.** *Let  $\mathcal{M}$  be a countable nonstandard model of arithmetic in the language  $\mathcal{L}_{\mathbb{N}}$  and  $d$  be an integer with  $d \geq 1$ .*

*Let  $\varphi$  be an unnested universal  $\mathcal{L}$ -sentence  $\varphi = \forall x_1, \dots, x_r \varphi_0(x_1, \dots, x_r)$  where  $\varphi_0(x_1, \dots, x_r)$  is a quantifier-free  $\mathcal{L}$ -formula of the form*

$$\varphi_0(x_1, \dots, x_r) = \bigwedge_{i \in I} \bigvee_{j \in J_i} \varphi_{i,j}(x_1, \dots, x_r)$$

*for finite index sets  $I \neq \emptyset$  and  $J_i \neq \emptyset$  such that  $i \in I$  and  $\varphi_{i,j}(x_1, \dots, x_r)$  are  $\mathcal{L}$ -literals.*

*If there are nonstandard integers  $n, b_0 \in \mathcal{M} \setminus \mathbb{N}$  with  $b_0 >^{\mathcal{M}} n^l$  for all  $l \in \mathbb{N}$  such that there exists an  $\mathcal{L}$ -expansion  $\mathcal{N}$  of  $\mathcal{M}$  with the properties  $\mathcal{N} \models \text{LNP}^{<b_0}(\Lambda_d)$  and  $\mathcal{N} \models \varphi^{<n}$ , then for all  $l \in \mathbb{N}$  there exists an  $m \in \mathbb{N}$  with  $m > 1$  such that the propositional formula  $\langle \varphi \rangle_{(m)} \wedge \vartheta_{\text{Tot}}(m) \wedge \vartheta_{\text{Fct}}(m)$  has no  $\mathcal{F}(d)$ -refutation of size  $\leq m^l$ .*

**Note.** *In Proposition 3.22 (a) we have shown that for  $m \in \mathbb{N}$  the propositional translation  $\langle \varphi \rangle_{(m)}$  of  $\varphi$  is a propositional formula in conjunctive normal form, i.e.,  $\langle \varphi \rangle_{(m)} = \bigwedge \Phi$  where  $\Phi$  is a set of  $\Sigma_1^{\text{prop}}$ -formulas.*

*Note that  $\vartheta_{\text{Tot}}(m)$  and  $\vartheta_{\text{Fct}}(m)$  are conjunctions of  $\Sigma_1^{\text{prop}}$ -formulas as well. Therefore,  $\langle \varphi \rangle_{(m)} \wedge \vartheta_{\text{Tot}}(m) \wedge \vartheta_{\text{Fct}}(m) = \bigwedge \Phi_m$  where  $\Phi_m$  is a set of  $\Sigma_1^{\text{prop}}$ -formulas. Then this means that regardless of the choice of  $d$  in the theorem,  $\Phi_m$  contains propositional formulas that we can work with in  $\mathcal{F}(d)$ .*

*Further note that for all  $m \in \mathbb{N}$  the set  $\Phi_m$  is coded by  $C_{\varphi}(m) \cup C_{\text{Tot}}(m) \cup C_{\text{Fct}}(m)$ .*

**Notation.** *By an  $\mathcal{F}(d)$ -refutation of  $\langle \varphi \rangle_{(m)} \wedge \vartheta_{\text{Tot}}(m) \wedge \vartheta_{\text{Fct}}(m) = \bigwedge \Phi_m$  for*

$m \in \mathbb{N}$ , we mean an  $\mathcal{F}(d)$ -refutation of the set  $\Phi_m$ . Since the propositional formula  $\langle \varphi \rangle_{(m)} \wedge \vartheta_{Tot}(m) \wedge \vartheta_{Fct}(m)$  is a conjunction, these mean the same: There is an  $\mathcal{F}(d)$ -refutation of  $\Phi_m$  if and only if  $\Phi_m$  is unsatisfiable.

*Proof of Theorem 3.24.* Assume towards a contradiction that there are non-standard integers  $n, b_0 \in \mathcal{M} \setminus \mathbb{N}$  with  $b_0 >^{\mathcal{M}} n^l$  for all  $l \in \mathbb{N}$  such that there is an  $\mathcal{L}$ -expansion  $\mathcal{N}$  of  $\mathcal{M}$  in which  $\varphi^{<n}$  and  $\text{LNP}^{<b_0}(\Lambda_d)$  are true and that there is an  $l_0 \in \mathbb{N}$  such that for every  $m \in \mathbb{N}$  with  $m > 1$  the propositional formula  $\langle \varphi \rangle_{(m)} \wedge \vartheta_{Tot}(m) \wedge \vartheta_{Fct}(m)$  has an  $\mathcal{F}(d)$ -refutation of size  $\leq m^{l_0}$ .

We can state the latter in the standard model as the sentence

$$\forall x > 1 \exists y (\text{Refut}_{\mathcal{F}(d)}(C_\varphi(x) \cup C_{Tot}(x) \cup C_{Fct}(x), y) \wedge \text{size}_{\text{Refut}}(y) \leq x^{l_0}).$$

Since this is true in the standard model, it is also true in  $\mathcal{M}$  and therefore in  $\mathcal{N}$ . Thus, for the nonstandard integer  $n$  there is a nonstandard integer  $\pi$  in  $\mathcal{N}$  such that the sentence is true in  $\mathcal{N}$  for  $x = n$  and  $y = \pi$ .

Intuitively,  $\pi$  codes a refutation of " $\langle \varphi \rangle_{(n)} \wedge \vartheta_{Tot}(n) \wedge \vartheta_{Fct}(n)$ ". Let  $s$  be the length of  $\pi$  in  $\mathcal{N}$ , i.e.,  $s = \|\pi\|^{\mathcal{N}}$ . By definition of  $\text{Refut}_{\mathcal{F}(d)}$ , the last entry  $(\pi)_{s-1}$  of  $\pi$  is  $\ulcorner \perp \urcorner$ . Hence,  $\mathcal{N} \models \neg \text{True}_{\Sigma_d^{prop}}((\pi)_{s-1})$ .

Define a "False-formula" which states "the  $y$ -th entry of the refutation coded by  $z$  is false":  $\text{False}(y, z) = \neg \text{True}_{\Sigma_d^{prop}}((z)_y)$ .

By Corollary 3.15,  $\langle \neg \text{True}_{\Sigma_d^{prop}}(a) \rangle_{(m)}$  is in  $\Pi_d^{prop}$  for all  $a, m \in \mathbb{N}$ . Note that the corollary states that the propositional translation could also be equal to  $\top$ . But since by assumption  $d \geq 1$  and  $\top \in \Pi_1^{prop}$ , this statement also holds for  $\top$ . Observe that for any  $a, b \in \mathbb{N}$ , the term  $(a)_b$  is also in  $\mathbb{N}$ . Thus, we obtain that  $\text{False}(y, z) \in \Lambda_d$ .

By assumption the least number principle is true in  $\mathcal{N}$  for  $\text{False}(y, \pi)$  up to  $b_0$ .

Since  $\mathcal{N} \models \text{False}(s-1, \pi)$ , because  $\text{size}_{\text{Refut}}(\pi) \leq^{\mathcal{N}} n^{l_0}$  holds and by Lemma 3.21, also  $s-1 <^{\mathcal{N}} b_0$  is true. Thus, we can apply the least number principle to  $\text{False}(y, \pi)$ . Hence there is a  $\leq^{\mathcal{N}}$ -minimal  $a_0 \in \mathcal{N}$  with  $a_0 \leq^{\mathcal{N}}$

$s - 1$  such that  $\text{False}(a_0, \pi)$  is true in  $\mathcal{N}$  and  $\text{False}(a, \pi)$  is false in  $\mathcal{N}$  for all  $a \in \mathcal{N}$  with  $a <^{\mathcal{N}} a_0$ . Thus, there are two cases:

- (a)  $a_0 = 0$ : By definition of  $\text{Refut}_{\mathcal{F}(d)}$ ,  $\mathcal{N} \models (\pi)_0 \in C_\varphi(n) \cup C_{\text{Tot}}(n) \cup C_{\text{Fct}}(n)$ . Since  $\mathcal{N} \models \neg \text{True}_{\Sigma_d^{\text{prop}}}((\pi)_0)$ , from the definition of  $\text{True}_{\Pi_d^{\text{prop}}}$  we obtain that

$$\begin{aligned} \mathcal{N} \models \neg \text{True}_{\Pi_{d+1}^{\text{prop}}}(\langle 1, C_\varphi(n) \rangle) \vee \neg \text{True}_{\Pi_{d+1}^{\text{prop}}}(\langle 1, C_{\text{Tot}}(n) \rangle) \vee \\ \neg \text{True}_{\Pi_{d+1}^{\text{prop}}}(\langle 1, C_{\text{Fct}}(n) \rangle). \end{aligned}$$

By Lemma 3.23, we can conclude that  $\mathcal{N} \models \neg \text{True}_{\Pi_{d+1}^{\text{prop}}}(\langle 1, C_\varphi(n) \rangle)$ . Since  $\text{T}_\varphi^{\mathcal{N}}(n) = \langle 1, C_\varphi^{\mathcal{N}}(n) \rangle^{\mathcal{N}}$ , this means  $\mathcal{N} \models \neg \text{True}_{\Pi_{d+1}^{\text{prop}}}(\text{T}_\varphi(n))$ . Intuitively, this is equivalent to saying  $\mathcal{N} \models \neg \text{True}_{\Pi_{d+1}^{\text{prop}}}(\ulcorner \langle \varphi \rangle_{(n)} \urcorner)$ .

By Proposition 3.12, we have  $\mathcal{N} \models \forall x (\varphi^{<x} \leftrightarrow \text{True}_{\Pi_{d+1}^{\text{prop}}}(\text{T}_\varphi(x)))$ . On one hand,  $\mathcal{N} \models \varphi^{<n}$ , but on the other hand  $\mathcal{N} \models \neg \text{True}_{\Pi_{d+1}^{\text{prop}}}(\langle 1, C_\varphi(n) \rangle)$ . This is a contradiction.

- (b)  $0 <^{\mathcal{N}} a_0 \leq^{\mathcal{N}} s - 1$ : This means that  $\text{True}_{\Sigma_d^{\text{prop}}}((\pi)_{a_0})$  is false in  $\mathcal{N}$ . By definition of  $\text{Refut}_{\mathcal{F}(d)}$ ,  $\mathcal{N}$  thinks that either  $(\pi)_{a_0} \in^{\mathcal{N}} C_\varphi^{\mathcal{N}}(n) \cup C_{\text{Tot}}^{\mathcal{N}}(n) \cup C_{\text{Fct}}^{\mathcal{N}}(n)$  or there are  $a_1, a_2 \in \mathcal{N}$  with  $a_1, a_2 <^{\mathcal{N}} a_0$  such that

- if  $d = 1$ ,

$$\mathcal{N} \models \text{Rule}_\vee((\pi)_{a_1}, (\pi)_{a_2}, (\pi)_{a_0}) \vee \text{Rule}_{\text{cut}}((\pi)_{a_1}, (\pi)_{a_2}, (\pi)_{a_0})$$

- or if  $d > 1$ ,

$$\begin{aligned} \mathcal{N} \models \text{Rule}_\vee((\pi)_{a_1}, (\pi)_{a_2}, (\pi)_{a_0}) \vee \text{Rule}_{\text{cut}}((\pi)_{a_1}, (\pi)_{a_2}, (\pi)_{a_0}) \vee \\ \text{Rule}_\wedge((\pi)_{a_1}, (\pi)_{a_2}, (\pi)_{a_0}). \end{aligned}$$

If  $\mathcal{N} \models (\pi)_{a_0} \in C_\varphi(n) \cup C_{\text{Tot}}(n) \cup C_{\text{Fct}}(n)$ , we obtain a contradiction analogously to case (a).

Assume the latter. Since  $a_1, a_2 <^{\mathcal{N}} a_0$ , by assumption we have  $\mathcal{N} \models \text{True}_{\Sigma_d^{\text{prop}}}((\pi)_{a_1})$  and  $\mathcal{N} \models \text{True}_{\Sigma_d^{\text{prop}}}((\pi)_{a_2})$ . Applying Corollary 3.20, we obtain a contradiction since  $\mathcal{N} \models \neg \text{True}_{\Sigma_d^{\text{prop}}}((\pi)_{a_0})$ .

□

### 3.7.2 The main theorem for other Frege refutation systems

**Notation.** Define  $\Lambda_{\text{CNF}}$  to be the set of all  $\mathcal{L}$ -sentences  $\varphi(x, y_1, \dots, y_r)$  such that their propositional translation  $\langle \varphi(a, b_1, \dots, b_r) \rangle_{(m)}$  is a CNF for all natural numbers  $a, b_1, \dots, b_r, m$ .

Let  $\Lambda_{\text{CNF}_k}$  the set of all  $\mathcal{L}$ -sentences  $\varphi(x, y_1, \dots, y_r)$  such that their propositional translation  $\langle \varphi(a, b_1, \dots, b_r) \rangle_{(m)}$  is a  $k$ -CNF for all  $a, b_1, \dots, b_r, m \in \mathbb{N}$ .

**Note.** Clearly,  $\Lambda_{\text{CNF}} = \Lambda_2$  and  $\Lambda_{\text{CNF}_k} \subseteq \Lambda_2$ .

**Corollary 3.25.** Let  $\mathcal{M}$  be a countable nonstandard model of arithmetic in the language  $\mathcal{L}_{\mathbb{N}}$ . Further let  $\mathcal{P}, \Lambda$  be one of the pairs (a), (b) or (c) (with  $k \in \mathbb{N}$  and  $k \geq 1$ ) shown in the table below.

Let  $\varphi$  be an unnested universal  $\mathcal{L}$ -sentence  $\varphi = \forall x_1, \dots, x_r \varphi_0(x_1, \dots, x_r)$  where  $\varphi_0(x_1, \dots, x_r)$  is a quantifier-free  $\mathcal{L}$ -formula of the form

$$\varphi_0(x_1, \dots, x_r) = \bigwedge_{i \in I} \bigvee_{j \in J_i} \varphi_{i,j}(x_1, \dots, x_r)$$

for finite index sets  $I \neq \emptyset$  and  $J_i \neq \emptyset$  such that  $i \in I$  and  $\varphi_{i,j}(x_1, \dots, x_r)$  are  $\mathcal{L}$ -literals.

If there are nonstandard integers  $n, b_0 \in \mathcal{M} \setminus \mathbb{N}$  with  $b_0 >^{\mathcal{M}} n^l$  for all  $l \in \mathbb{N}$  such that there exists an  $\mathcal{L}$ -expansion  $\mathcal{N}$  of  $\mathcal{M}$  with the properties  $\mathcal{N} \models \text{LNP}^{< b_0}(\Lambda)$  and  $\mathcal{N} \models \varphi^{< n}$ , then for all  $l \in \mathbb{N}$  there exists an  $m \in \mathbb{N}$  with  $m > 1$  such that the propositional formula  $\langle \varphi \rangle_{(m)} \wedge \vartheta_{\text{Tot}}(m) \wedge \vartheta_{\text{Fct}}(m)$  has no  $\mathcal{P}$ -refutation of size  $\leq m^l$ .

	Theorem 3.24	(a)	(b)	(c)
	$\Sigma_d^{prop}$	$\Sigma_1^{prop}$	DNFs (= $\Sigma_2^{prop}$ )	$k$ -DNFs
$\mathcal{P}$	$\mathcal{F}(d)$	$\mathcal{R}$ (= $\mathcal{F}(1)$ )	$\mathcal{R}_{\text{DNF}}$ (= $\mathcal{F}(2)$ )	$\mathcal{R}_{\text{DNF}_k}$
$\Lambda$	$\Lambda_d$	$\Lambda_1$	$\Lambda_{\text{CNF}}$ (= $\Lambda_2$ )	$\Lambda_{\text{CNF}_k}$

*Proof.* (a) and (b): For  $\mathcal{P} = \mathcal{R}$  let  $d = 1$  and for  $\mathcal{P} = \mathcal{R}_{\text{DNF}}$  let  $d = 2$  in Theorem 3.24.

(c): Let  $\mathcal{P} = \mathcal{R}_{\text{DNF}_k}$  for an integer  $k \geq 1$ . Since we have proven all results we used in the proof of Theorem 3.24 for  $k$ -DNFs as well, the theorem also holds for  $\mathcal{R}_{\text{DNF}_k}$ .  $\square$

# Bibliography

- [1] S. Arora and B. Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009.
- [2] J. Avigad. Saturated models of universal theories. *Annals of Pure and Applied Logic*, 118, 2002.
- [3] S. R. Buss. *Bounded Arithmetic*. Bibliopolis, 1986.
- [4] S. R. Buss. First-order proof theory of arithmetic. [5].
- [5] S. R. Buss, editor. *Handbook of Proof Theory*. Elsevier, 1998.
- [6] S. R. Buss. An introduction to proof theory. [5].
- [7] W. Hodges. *Model Theory*. Cambridge University Press, 1993.
- [8] J. Krajíček. *Bounded Arithmetic, Propositional Logic, and Complexity Theory*. Cambridge University Press, 1995.
- [9] M. Ziegler. *Mathematische Logik*. Birkhäuser, 2010.



# Curriculum Vitae

## Personal Data

Name            Thomas Imre  
Email            imretho@gmail.com

## Education

since 2012    Master studies in Mathematics at the University of Vienna with special focus on mathematical logic at the Kurt Gödel Research Center

July 2012     awarded Bachelor of Science (BSc.)

2008-2012    Bachelor studies in Mathematics at the University of Vienna

2002-2007    Höhere Technische Bundeslehranstalt Wien 10, Höhere Abteilung für Telekommunikation (Secondary school for technology, upper level, majored in communication engineering), graduated with distinction

1998-2002    BRG/ORG Wien 23, Anton-Krieger-Gasse (Secondary school, lower level)

1994-1998    Volksschule Prückelmayrgasse (Elementary school)

**Teaching**

- 2011-2014 Tutor at the Vienna University of Technology for "Mathematik 1 und 2 für Maschinenbau, Verfahrenstechnik und Bauingenieurwesen" (Math courses covering analysis and linear algebra for students of Civil Engineering and Management of Infrastructure, Mechanical Engineering and Chemical and Process Engineering)
- WS 12/13 Tutor at the University of Vienna for "Hilfsmittel der EDV" (Introductory course in  $\text{\LaTeX}$  and *Mathematica* for Math students)
- WS 2011 Tutor at the University of Natural Resources and Life Sciences Vienna for "Mathematik für Lebensmittel- und Biotechnologie" (Introductory math course for students of Food science and Technology and Biotechnology)
- Sept.-Nov. 2009 Substitute math teacher for the 4th grade of secondary school lower level at BRG/ORG Wien 23, Anton-Krieger-Gasse

**Scholarships**

- 2013 Performance Scholarship awarded by the University of Vienna
- 2010 Performance Scholarship awarded by the University of Vienna
- 2009 Performance Scholarship awarded by the University of Vienna

**Civilian service**

- 2007-2008 Care for students with disabilities at BRG/ORG Wien 23, Anton-Krieger-Gasse