Eötvös Loránd Tudományegyetem
Faculty of Science

Péter Kuti

Logical systems and definability properties in First-order logic – and beyond

Mathematics BSc
Applied Mathematics major
Thesis

Supervisor:

Ildikó Sain

Alfréd Rényi Institute of Mathematics
Department of Algebraic Logic,
Hungarian Academy of Sciences

Budapest, 2015

Contents

Acknowledgements

I hereby would like to express my gratitude towards my supervisor, Ildikó Sain, who guided my investigation through a field of mathematics that was formerly unknown for me. Her exactitude, attentiveness, her advice and teaching meant great help, and are highly appreciated. I have learned very much from her.

I truly am indebted to my family, mostly my mother and grandfather, who have not only given me motivation and encouragement, but also stood by my side, ready to help.

Furthermore, I am very thankful for "many people", more specifically for every human being for whom at least one of the two conditions below hold.

- He/she has helped me getting acquainted with mathematics, mainly with mathematical logic and made me endear to this topic.
- He/she has helped me in my personal life in this period, providing assistance to the completion of this thesis. Special thanks to all, who have made me laugh every once in a while.

Though it was challenging, I have taken delight in working on this paper.

Köszönetnyilvánítás

Ezúton szeretném kifejezni hálámat témavezetőmnek, Sain Ildikónak, aki a matematika egy számomra korábban ismeretlen területének tanulmányozása során vezette utam. Precizitása, figyelmessége, tanácsai és tanításai rengeteg segítséget jelentettek. Sokat tanultam tőle.

Köszönettel tartozom a családomnak is, kiváltképp édesanyámnak és nagypapámnak, akik nem csak motiváltak és biztattak, de végig mellettem is álltak.

Továbbá szintén köszönettel tartozom "sok embernek", némileg specifikusabban minden emberi lénynek, akikre az alábbi két feltétel közül legalább az egyik teljesül.

- Segített megismerkedni a matematika világával, ezen belül is főként a matematikai logikával, valamint megszerettette velem ezeket.
- A magánéletem során segítséget nyújtott az utóbbi időben, ezzel is elősegítve a dolgozat elkészültét. Külön köszönet azoknak, akik néhanapján nevetésre bírtak.

Bár komoly kihívást jelentett, örömmel dolgoztam.

Introduction

Throughout the history of our culture, starting, say, from the Greek culture to which our science, technology, civilization is based on, the concept of a logical system went through a long development. In accordance with Aristotle's view, the first motivation was to determine the rules of *"correct reasoning"*. This motivation lead to the concept of a logical system called *propositional logic* today. Later it turned out that there are several aspects of "correct reasoning", therefore several versions of logical systems have been developed. For example, the system we call *first order logic* today, can be considered to be a refined version of propositional logic in that it analyses the structure of the atomic statements as well (we have *terms* as building blocks and we have *quantifiers*), while propositional logic analyses only the structure of the compound statements (how they can be built up from simpler statements).

Another basic motivation in defining logical systems is the study of *consequence*. The emphasis on the fact whether one can prove (or derive) a proposition from other ones using formal logical rules. This happens on the syntactical level. After a while a method, called the *model method* was discovered / introduced, for proving non-provability. The idea here is to construct a so called model which intends to be a mathematical model of the world in which our propositions have their meanings. It is the model method from which the semantical part of logic grew out later.

A third, and most recent motivation comes from theoretical computer science. When programming computers, we write texts that have some logical structures. Then we interpret these texts in the "real world" in the sense that we *execute* them on computers. When scientists began programming computers, they thought that descriptions of the meanings of programs in natural language will work. But bitter experiences showed that this was not the case. For example, a Martian expedition failed only because the semantics of its software contained trivial errors. More knowledge about the semantics of programs had to be developed.

The three motivations above seem to culminate in the following idea. **We would like to build a *mathematical model* of the logical aspects of languages** – be them natural languages (like English, Hungarian, German) or artificial ones (like programming languages, interfaces). We will call such a mathematical model a *logical system* or *logic* for short. This is the topic of chapter 1, which is based on [1], [2], [3], [11], [13].

Precision is essential in scientific thinking, whenever we reason about something, we need to specify the notions we use. Definitions are the perfect tools for this. As a part of mathematical logic, called *definability theory*, a theory about structuring knowledge and about concept formation, also developed. About definitions, we have many questions coming to our minds, for instance we would like to know if they are mere abbreviations, tools for making communication more simple or if they have a role in concept formation itself. Another very important question is that in a certain logical system, what a definition is and when do we say (and when can we say) that something is defined indeed. Are there more ways of defining the same thing? If there are, is there any connection between the different kinds of definitions?

Speaking about definitions, a fundamental question of humanity comes up, as well: what is the concept of truth, can truth be defined? An outstanding result of research on this topic was when Alfred Tarski proved that arithmetical truth can not be defined within the arithmetic itself, which tells us how important the metalanguage is, that is, we always need some kind of background information when defining truth.

Naturally, when examining definability, we would like to know about the similarities and differences of the definability properties of different logical systems. We will show the process of algebraization of a logic, which helps us to be able to generalize definability properties and see a comprehensive picture of definability in logical systems.

Issues concerning definability can be found in chapter 2. The main bulk of this chapter is based on [4]. Our chapter 5 connects parts I and II in that we look into the concept of definability in the framework given in chapter 1. Our main references here are: [2], [3], [13].

Part I

General concept of a logical system

Chapter 1

A logical system in general

In part I, first we recall a general concept of *logical systems* (in chapter 1) from publications by Hajnal Andréka, István Németi and Ildikó Sain, see e.g. [2], [3], [13]. Then, in chapter 2 we will give some concrete examples of logical systems in the sense of 1. We start out from the basics, sentential logic (also called propositional logic or propositional calculus). It is a successful, widely known and used logical system. It incorporates all the important parts of a logical system: complex propositions are built up recursively from atomic propositions, using the Boolean connectives; its semantics is very transparent in that the algebras that can be associated to its models form a finitely axiomatizable variety; the syntactical and semantical consequence relations coincide (completeness).Then we take a small step of expanding sentential logic to be able to even express modalities, like the possibility of a proposition being true. After that we turn to First-order logic (FOL, for short), a more complex system. First we define FOL the classical way (in section 2.3), then we will define fragments of FOL; FOL with n variables (section 2.4) and the rank-free formulation (section 2.5). To give a hint on applications, in section 2.6 a *logic of programs* is defined, which is a many-sorted FOL. This logic was deeply investigated (under the name "dynamic logic with nonstandard model theory") by Andréka, Németi, Sain, see for example [1] and [12]. In chapter 3, we will introduce some basic properties of logical systems, and see if the examples shown before have those or not. This will lead up to the concept of a strongly nice logic, which concept we will use in chapter 5 of part II.

As we explained in the introduction, we would like to build a mathematical model of the logical aspects of a language. We assume that, basically, a logic must include a set called *formulas*, this is the mathematical model of the texts that can be "said" in the language. We also want to include a mathematical model of the world that we can speak about. Further, we want to have a mathematical model of the connection between our texts and the world. Intuitively, this connection tells us what is the meaning of a text in the world. Therefore we will call the mathematical model of this connection *meaning*.

When building such a mathematical model, some ideas from the *methodology of science* are borrowed. An example to this is the following. Clearly, a logic or language must not reflect (much from) our knowledge about the world it speaks about. This is why we can speak about, e.g., doings of unicorns, or the Loch Ness monster, independently whether they exist or not. Instead, a mathematical model of a logic must reflect the *lack of knowledge* of the speaker. According to a principle in methodology of science, lack of knowledge can be expressed via considering *all the possibilities*. Therefore, our mathematical model of the world will consist of the collection of all "*possible worlds*", and the mathematical model of the meaning will be a collection of functions associating things in connection with the possible worlds (not necessarily elements of the possible worlds) to formulas.[1] These functions, called

[1]Ignoring this principle lead to the incompleteness of classical higher order logics, see e.g. [11].

meaning functions, are defined via some schema, reflecting our vision that the meaning of a formula must be defined in similar ways in each possible world.

Convention 1.1.1. Throughout this document, we will use the Zermelo–Fraenkel set theory with the axiom of choice (ZFC for short) as our basic background theory. A detailed introduction on using this theory as our foundation can be found e.g., in [2, Chap.1]. ◁

In what follows, we recall an abstract notion of logical systems (logic for short) from [2], [3] and [13].

Definition 1.1.2. (Logic) A logic $\mathcal{L}$ is defined to be an ordered quadruple

$$\mathcal{L} = \langle F_{\mathcal{L}}, M_{\mathcal{L}}, mng_{\mathcal{L}}, \models_{\mathcal{L}} \rangle,$$

where

i) $F_{\mathcal{L}}$ (called the set of *formulas*) is a set of finite sequences over some set X (the alphabet of $\mathcal{L}$);

ii) $M_{\mathcal{L}}$ is a class, called the class of all *models* of $\mathcal{L}$ (the possible worlds);

iii) $mng_{\mathcal{L}}$ is a function with domain $F_{\mathcal{L}} \times M_{\mathcal{L}}$, called the *meaning function* of $\mathcal{L}$, hence, by the usual convention of our foundation of mathematics,

$$mng_{\mathcal{L}} : F_{\mathcal{L}} \times M_{\mathcal{L}} \longrightarrow \text{"}Sets\text{"}$$

where "*Sets*" is the class of all sets;

iv) $\models_{\mathcal{L}}$ is a binary relation, $\models_{\mathcal{L}} \subseteq M_{\mathcal{L}} \times F_{\mathcal{L}}$, called the *validity relation* of $\mathcal{L}$ (tells if the sentences are true or false in our world);

v) there is a connection between $\models_{\mathcal{L}}$ and $mng_{\mathcal{L}}$: for all $\varphi, \psi \in F_{\mathcal{L}}$ and $\mathfrak{M} \in M_{\mathcal{L}}$ we assume

$$\Big(mng_{\mathcal{L}}(\varphi, \mathfrak{M}) = mng_{\mathcal{L}}(\psi, \mathfrak{M}) \text{ and } \mathfrak{M} \models_{\mathcal{L}} \varphi \Big) \Longrightarrow \mathfrak{M} \models_{\mathcal{L}} \psi \ . \qquad \triangleleft$$

Remark 1.1.3. a) In this definition we sketch a slightly modified form of a logic, the more complete definition is a five-tuple, namely $\mathcal{L} = \langle F_{\mathcal{L}}, \vdash_{\mathcal{L}}, M_{\mathcal{L}}, mng_{\mathcal{L}}, \models_{\mathcal{L}} \rangle$, where $\vdash_{\mathcal{L}}$ is called the *provability relation* of $\mathcal{L}$ (a relation between sets of formulas and formulas, that is: $\vdash_{\mathcal{L}} \subseteq \mathcal{P}(F) \times F$). In this thesis we will not investigate the provability relation, therefore the above quadruple is enough for us. Other thoughts concerning the omission of $\vdash_{\mathcal{L}}$ from our model can be found for example in [2, section 3.1].
b) Later, in section 3.2 we will see, that in case of "well-constructed" logics (compositional and has the filter property), condition v) automatically holds.
c) Sometimes not all parts of a logic is given, for example we may only have $\langle F, \vdash \rangle$ and we are searching for semantics $\langle M, \models \rangle$ such that $\langle F, \vdash, M, \models \rangle$ is complete[2]. $\langle F, \vdash \rangle$ is sometimes called a "syntactical logic", and $\langle F, M, \models \rangle$ a "semantical logic". (In case of a complete logic $\mathcal{L} = \langle F_{\mathcal{L}}, \vdash_{\mathcal{L}}, M_{\mathcal{L}}, mng_{\mathcal{L}}, \models_{\mathcal{L}} \rangle$, $\langle F, \models \rangle$ is the same as $\langle F, \vdash_{\mathcal{L}} \rangle$, thus the semantical and the syntactical logics coincide.) ◁

A great motivation for building up logical systems is to find answers for what we can tell about the "truth" of some statement, and what can we know about the consequences of the information we have. The next definitions are about the validity of formulas and that of sets of formulas in certain worlds (models).

[2]Though in this writing we are not engaged in investigating completeness, the intuitive meaning of that in this particular sense, is that a logic is called complete if every true formula can be derived using the logic's inference system.

Definition 1.1.4. (Semantical consequence, valid formulas) Let $\mathcal{L} = \langle F_\mathcal{L}, M_\mathcal{L}, mng_\mathcal{L}, \models_\mathcal{L}\rangle$ be a logic. For every $\mathfrak{M} \in M_\mathcal{L}$ and $\Sigma \subseteq F_\mathcal{L}$,

$$\mathfrak{M} \models_\mathcal{L} \Sigma \overset{\text{def}}{\Longleftrightarrow} (\forall\varphi \in \Sigma)\, \mathfrak{M} \models_\mathcal{L} \varphi,$$

$$Mod_\mathcal{L}(\Sigma) \overset{\text{def}}{=} \{\mathfrak{M} \in M_\mathcal{L} : \mathfrak{M} \models_\mathcal{L} \Sigma\}$$

$Mod_\mathcal{L}(\Sigma)$ is called the class of all models of Σ.

A formula φ is *valid* (in symbols $\models_\mathcal{L} \varphi$) if and only if $Mod_\mathcal{L}(\{\varphi\}) = M_\mathcal{L}$.

For any $\Sigma \cup \{\varphi\} \subseteq F_\mathcal{L}$,

$$\Sigma \models_\mathcal{L} \varphi \overset{\text{def}}{\Longleftrightarrow} Mod_\mathcal{L}(\Sigma) \subseteq Mod_\mathcal{L}(\{\varphi\}),$$

$$Csq_\mathcal{L}(\Sigma) \overset{\text{def}}{\Longleftrightarrow} \{\varphi \in F_\mathcal{L} : \Sigma \models_\mathcal{L} \varphi\}.$$

If $\varphi \in Csq_\mathcal{L}(\Sigma)$ then we say that φ is a *semantical consequence* of Σ (in logic $\mathcal{L}$). $\triangleleft$

Definition 1.1.5. (theory, the set of validities) Let $\mathcal{L} = \langle F_\mathcal{L}, M_\mathcal{L}, mng_\mathcal{L}, \models_\mathcal{L}\rangle$ be a logic. For any $K \subseteq M_\mathcal{L}$ let the *theory* of K in $\mathcal{L}$ be defined as

$$Th_\mathcal{L}(K) \overset{\text{def}}{=} \{\varphi \in F_\mathcal{L} : (\forall\mathfrak{M} \in K)\, \mathfrak{M} \models_\mathcal{L} \varphi\}.$$

The set $Th_\mathcal{L}(M_\mathcal{L})$ is called the *set of validities* of $\mathcal{L}$. $\triangleleft$

Now we introduce the notion of the rank or arity of a relation R, which notion we will strongly rely on in the following chapters.

Definition 1.1.6. (The *rank* or *arity* of a relation). If $n \in \omega$ and R is a set of n-sequences , then we say that R is an *n-ary relation.* We refer to this n as the *rank* or *arity* of R. An n-ary relation R is a subset of the Cartesian product $U_1 \times U_2 \times \ldots \times U_n$ of the following sets U_i $(1 \leq i \leq n)$:

$$U_1 = \{u_1 : \langle u_1, u_2, \ldots, u_n\rangle \in R \text{ for some } u_2, \ldots u_n\},$$

$$U_2 = \{u_2 : \langle u_1, u_2, \ldots, u_n\rangle \in R \text{ for some } u_1, u_3 \ldots u_n\},$$

$$U_n = \{u_n : \langle u_1, u_2, \ldots, u_n\rangle \in R \text{ for some } u_1, \ldots u_{n-1}\},$$

Also, $R \subseteq {}^nU$ for $U = \bigcup\{U_i : 1 \leq i \leq n\}$. $\triangleleft$

Remark 1.1.7. If $n \in \omega$ and U is a set, then a function f with $Dom(f) = {}^nU$ and $Rng(f) \subseteq U$ is called an *n-ary function* (or *operation*) on U, and we say that the *rank* or *arity* of the function f is n. We often identify n-ary functions with $(n+1)$-ary relations, associating to an n-ary function f the $(n+1)$-ary relation $\big\{\langle u_1, \ldots, u_n, u_{n+1}\rangle : \big\langle\langle u_1, \ldots, u_n\rangle, u_{n+1}\big\rangle \in f\big\}$.

A synonym for "1-ary function (relation)" is "unary function (relation)", and similarly, for "2-ary" and "3-ary" functions (relations) we say "binary" and "ternary".

For any relation R, we use the notation "$rank(R) = n$" iff R is n-ary relation. $\triangleleft$

Chapter 2

Examples for different kinds of logics as special cases of the general concept

Naturally, the validity of formulas, the connection between a set of formulas and a single formula and many other properties are highly dependent of what kind of logic we use. In the following section we will give some examples of different kinds of logics and see how they fit into the general framework we defined in chapter 1. The first example is going to be sentential logic, which we will define in "modal logic style"; we define the meaning function and validity relation via an auxiliary relation $\Vdash$.

2.1 Sentential logic $\mathcal{L}_S$

Definition 2.1.1. (Propositional or sentential logic $\mathcal{L}_S$) By *propositional* (sentential) *logic* we mean a quadruple

$$\mathcal{L}_S \stackrel{\text{def}}{=} \langle F_S, M_S, mng_S, \models_S \rangle,$$

where its parts are defined as follows. Let P be a set, called the set of *atomic formulas* of $\mathcal{L}_S$, and let $Cn(\mathcal{L}_S) \stackrel{\text{def}}{=} \{\neg, \wedge\}$ be disjoint from P, called the set of *logical connectives* of $\mathcal{L}_S$, where $rank(\neg) = 1$ and $rank(\wedge) = 2$.

Now, we can define F_S. The **set of formulas** is defined to be the smallest set H satisfying the following three conditions:

i) $P \subseteq H$,

ii) $\varphi \in H \Longrightarrow (\neg\varphi) \in H$.

iii) $\varphi, \psi \in H \Longrightarrow (\varphi \wedge \psi) \in H$.

The definition of the **class of models** is:

$$M_S \stackrel{\text{def}}{=} \{\langle W, v\rangle : W \text{ is a non-empty set and } v : P \longrightarrow \mathcal{P}(W)\},$$

where $\mathcal{P}(W)$ denotes the power set (the set of all subset of) W. Intuitively, a model declares the meanings of the atomic formulas. Here W is called the set of *possible states* (or worlds or situations) of $\mathfrak{M}$.

Let us fix $\langle W, v\rangle \in M_S$, $w \in W$; and $\varphi \in F_S$. Our auxiliary binary relation $\Vdash_v$ is defined by recursion on the complexity of the formulas, as follows:

- if $p \in P$ then $\left(w \Vdash_v p \overset{\text{def}}{\Longleftrightarrow} w \in v(p)\right)$
- if $\psi_1, \psi_2 \in F_S$, then

$$w \Vdash_v \neg\psi_1 \overset{\text{def}}{\Longleftrightarrow} w \nVdash_v \psi_1 \text{ and}$$

$$w \Vdash_v (\psi_1 \wedge \psi_2) \overset{\text{def}}{\Longleftrightarrow} w \Vdash_v \psi_1 \text{ and } w \Vdash_v \psi_2$$

If $w \Vdash_v \varphi$ then we say φ is *true in* w, or w *forces* φ. We define the meaning function and the validity relation with the help of this relation.

meaning function: $mng_S(\varphi, \langle W, v\rangle) \overset{\text{def}}{=} \{w \in W : w \Vdash_v \varphi\}$

validity relation: $\langle W, v\rangle \models_S \varphi \overset{\text{def}}{\Longleftrightarrow}$ for every $w \in W, w \Vdash_v \varphi$ ◁

Remark 2.1.2. A significant fact is that the set P (the set of atomic formulas) is a parameter in the definition of $\mathcal{L}_S$. It is a fixed, but arbitrary set, so in a sense, $\mathcal{L}_S$ is a function of P. It usually has low influence on the behavior of $\mathcal{L}_S$, therefore we only rarely mention it as $\mathcal{L}_S(P)$, but the notation $\mathcal{L}_S$ implicitly carries that information. ◁

2.2 Modal logic $S5$

This logic is very similar to sentential logic, but still is somewhat different. In the definition below, the parts that we do not define are identical to the corresponding parts of definition 2.1.1.

Definition 2.2.1. (Modal logic $S5$) *Modal logic* $S5$ is basically sentential logic, but we add an extra connective to $Cn(\mathcal{L}_S)$, so that $Cn(S5) = \{\wedge, \neg, \Diamond\}$, where $rank(\Diamond) = 1$.

The **set of formulas** (F_{S5}) of $S5$ is defined just as in the case of sentential logic together with the following clause:

$$\varphi \in F_{S5} \Longrightarrow \Diamond\varphi \in F_{S5}$$

The **class of models** is defined the same ways as in sentential logic, so we let $M_{S5} \overset{\text{def}}{=} M_S$.

To have the definition of $w \Vdash_v \varphi$, we need to add an extra line, telling the case of $\Diamond$:

$$w \Vdash_v \Diamond\varphi \overset{\text{def}}{\Longleftrightarrow} (\exists w' \in W)\ w' \Vdash_v \varphi$$

$\Diamond$ is often called the possibility operator.

The **meaning function** is the same as it is in sentential logic (but using the expanded definition of $\Vdash_v$), namely: $mng_{S5}(\varphi) \overset{\text{def}}{=} \{w \in W : w \Vdash_v \varphi\}$.

The **validity relation** $\models_{S5}$ is defined as:

$$\langle W, v\rangle \models_{S5} \varphi \overset{\text{def}}{\Longleftrightarrow} (\forall w \in W)\ w \Vdash_v \varphi$$

With these defined, modal logic $S5$ is $S5 \overset{\text{def}}{=} \langle F_{S5}, M_{S5}, mng_{S5}, \models_{S5}\rangle$ ◁

Remark 2.2.2. a) Modal logics in general have a so called accessibility relation (R) in their models. The intuitive meaning of wRw' (or we could write $R(w, w')$) is that the world w' is accessible from w. In our definition for $S5$, the accessibility relation of a model $\langle W, v\rangle$ is $W \times W$. We have built this information in the (meta-logical) definition of $\Vdash_v$. If in a model not every other possible world is accessible from an arbitrary world, then we need to add a condition to the possibility operator, namely: $w \Vdash_v \Diamond\varphi \overset{\text{def}}{\Longleftrightarrow} (\exists w' \in W)\ R(w, w')$ and $w' \Vdash_v \varphi$. In the next section, we will show how that changes the logic through a concrete example.

b) The classical unary modalities have their dual modalities $\Box\varphi = \neg\big(\Diamond(\neg\varphi)\big)$, that is in our case $w \Vdash_v \Box\varphi \iff (\forall w' \in W)\ w' \Vdash_v \varphi$. The operator $\Box$ is called the necessity operator. $\triangleleft$

As mentioned before, the two logics above are very similar, thus the first question that comes to mind is what the differences are between those two. Naturally, since $S5$ is an expanded version of $\mathcal{L}_S$, we can reason about more things using that logic, or in other words, it has a bigger expressive power. In the next exemplification we will show a concrete model and some formulas, that help us understand the difference (and hopefully we give some intuition and clarification on how a logic is put into practice).

***Examplification*:** Let $\mathfrak{M} = \langle W, v\rangle$ ($\in M_S = M_{S5}$) be our model, where

$$W = \{living\ room, bedroom, bathroom, kitchen\}$$

(we make a model of our house). We want to talk about the colors of the rooms, let $p \in P$ mean "The room is blue" and $q \in P$ mean "The room is green". In sentential logic, we can examine the truth of statements like "The room is blue and green." (in symbols: $p \wedge q$), or the room is neither blue, nor green ($\neg p \wedge \neg q$). This, of course, can be interpreted when we fix a room (a world) in which we examine the truth of it. For example we can say that "The living room is blue and green" as: $living\ room \Vdash_v (p \wedge q)$, which statement is true iff[3] $living\ room \in v(p)$ and $living\ room \in v(q)$.

Up to this point, everything works the very same way in both of our logics. Now we would like to see what more we can say if using that extra connective of $S5$. The possibility operator lets us tell much more, statements using this operator might hold information about other worlds (rooms), as well. For instance, we can express that "There is a room that is green"; for any $w \in W$, $w \Vdash_v \Diamond q$ means that. As we can see, by knowing the truth of this formula, we have information about if the formula can be satisfied by any circumstances in our model. This can not be done in sentential logic. $\triangleleft$

2.3 First-order logic

The next example will be First-order logic (FOL), the one we will analyze in chapter 4 from the point of view of definability. In this section we recall the classical definition of First-order logic from [2] and then later, we define fragments of FOL, as well. We will show a restricted form of it, where only n variables are used, then we will introduce the so called rank-free formulation of this. In section 2.6, we will see a more complex version of a FOL, a special many-sorted FOL used for being a logic of programs. Then, in section 3 we will examine some properties of these logics, as well.

For any set X, X^* denotes the *set of all finite sequences over* X, that is:

$$X^* \stackrel{\text{def}}{=} \{f : Dom(f) \in \omega \text{ and } Rng(f) \subseteq X\} = \bigcup_{n\in\omega} ({}^{n}X)$$

We call X^* the *language over the alphabet* X and the elements of X^* are called *words* over X. If x is not a sequence we say that x is a *basic symbol.*

We fix a set $C = \{\neg, \wedge, \exists, \doteq\}$ consisting four distinct basic symbols; the set of logical symbols of FOL.

Definition 2.3.1. (Similarity type) We call a pair $t = \langle t_0, R\rangle$ a (one-sorted) *similarity type* (or vocabulary or signature) if and only if i)-iii) below hold:

[3] We use the phrase "iff" to abbreviate "if and only if".

i) t_0 is a function for which $Rng(t_0) \subseteq \omega$,

ii) $R \subseteq Dom(t_0)$,

iii) $Dom(t_0)$ is a set of basic symbols (disjoint from the logical symbols).

R is called the set of *relation symbols* and $Dom(t_0)\setminus R$ is called the set of *function symbols* (or operation symbols) of t. They are denoted as Rls_t and Fns_t, respectively. For any $f \in Dom(t_0)$, $t_0(f)$ is called the *rank* (or arity) of f. $\triangleleft$

Remark 2.3.2. If $c, f \in Fns_t$ and $t(c) = 0$ then c is called a *constant symbol* and if $t(s) = 1$ (or $t(s) = 2$) then we call f a *unary* (*binary*) *function symbol.* Sometimes $Dom(t)$ is called the set of all non-logical symbols (a set, which the user of FOL chooses). $\triangleleft$

Definition 2.3.3. (Terms and FOL-formulas of similarity type t) Let t be an arbitrary but fixed similarity type, V be a set of basic symbols satisfying that $V \cap (Dom(t) \cup C) = \emptyset$, but arbitrary otherwise. We say that V is a *set of variables* for t. We define the sets $Trm_t(V)$ (*terms* of the similarity type t) and $Fml_t(V)$ (*formulas* of similarity type t) with variables from V as follows.

- $Trm_t(V) \subseteq (V \cup Fns_t)^*$ is defined to be the smallest set satisfying i) and ii) below:
  - i) $V \subseteq Trm_t(V)$
  - ii) $\{f\tau_1, \ldots, \tau_n : f \in Fns_t,\ t(f) = n \text{ and } \tau_1, \ldots, \tau_n \in Trm_t(V)\} \subseteq Trm_t(V)$
- $Fml_t(V) \subseteq (V \cup Dom(t) \cup C)^*$ is defined to be the smallest set satisfying iii) and iv) below:
  - iii) $\{r\tau_1, \ldots, \tau_n : r \in Rls_t,\ t(r) = n \text{ and } \tau_1, \ldots, \tau_n \in Trm_t(V)\} \cup \{\tau \doteq \sigma : \tau, \sigma \in Trm_t(V)\} \subseteq Fml_t(V)$
  - iv) $\{\neg\varphi : \varphi \in Fml_t(V)\} \cup \{\wedge\varphi\psi : \varphi, \psi \in Fml_t(V)\} \cup \{\exists v\varphi : v \in V \text{ and } \varphi \in Fml_t(V)\} \subseteq Fml_t(V)$

The formulas in the set of the left-hand-side of "$\subseteq$" in iii) are called *atomic formulas.* A variable v occurring in φ is called free if it occurs outside of the scope of any quantifier $\exists v$. If $\varphi \in Fml_t$, then $Var(\varphi)$ denotes the set of variables occurring in φ, and $Fvar(\varphi)$ denotes the set of variables occurring freely in φ. $\triangleleft$

***Notation*:** The logical connectives $\neg, \wedge$ and $\exists v$ are called negation, conjunction and existential quantifier, respectively. For easier readability, we often use the notation $f(\tau_1, \ldots, \tau_n)$, $r(\tau_1, \ldots, \tau_n)$ and $\wedge(\varphi, \psi)$, by which we mean $f\tau_1, \ldots, \tau_n$, $r\tau_1, \ldots, \tau_n$ and $\wedge\varphi\phi$ (for the latter, we will use the notation $\varphi \wedge \psi$). We sometimes use $=$ instead of $\doteq$, as well. $\triangleleft$

Definition 2.3.4. (Model or structure of similarity type t) By a *model* (or structure) of similarity type t we mean a pair $\langle M, m\rangle$ satisfying i)-ii) below.

i) M is a non-empty set (called the universe of the model)

ii) m is a function with $Dom(m) = Dom(t)$ such that

- if $f \in Fns_t$ and $t(f) = n$ then $m(f) : {}^nM \longrightarrow M$;
- if $r \in Rls_t$ and $t(r) = n$ then $m(r) \subseteq {}^nM$.

The functions $m(f)$ and the relations $m(r)$ are called the interpretations of the symbols f and r in the model $\langle M, m\rangle$. The class of all models of a similarity type t is denoted by Mod_t. $\triangleleft$

***Notation*:** We usually denote models by the German capitals $\mathfrak{M}$, $\mathfrak{N}$, etc. When saying that "φ is a formula in the language of $\mathfrak{M}$", we mean that φ is a FOL formula of the same similarity type as that of $\mathfrak{M}$. If $\mathfrak{M} = \langle M, m\rangle$, $f \in Fns_t$ and $r \in Rls_t$, then $f^{\mathfrak{M}}$ and $r^{\mathfrak{M}}$ stand for $m(f)$ and $m(r)$, respectively. A usual notation for $\mathfrak{M}$ is:

$$\mathfrak{M} = \langle M, f^{\mathfrak{M}}, r^{\mathfrak{M}}\rangle_{f\in Fns_t, r\in Rls_t}. \qquad \lhd$$

Definition 2.3.5. (Valuation of variables and validity of formulas) Let $\mathfrak{M} \in Mod_t$ and let V be an arbitrary set of variables for t. A function $k : V \longrightarrow M$ is called a *valuation* of the variables from V in $\mathfrak{M}$.

Let k be an arbitrary but fixed valuation of the variables in $\mathfrak{M}$. We define when a formula φ is *true in* $\mathfrak{M}$ *at the valuation* k *of the variables* (which is written in symbols as: $\mathfrak{M} \models \varphi[k]$) by recursion on the complexity of formulas as follows.

i) First, we define the value of $\tau^{\mathfrak{M}}[k]$ for any τ term:

- $v^{\mathfrak{M}}[k] \stackrel{\text{def}}{=} k(v)$ if $v \in V$
- $\big(f(\tau_1,\ldots,\tau_n)\big)^{\mathfrak{M}}[k] \stackrel{\text{def}}{=} f^{\mathfrak{M}}(\tau_1^{\mathfrak{M}}[k],\ldots,\tau_n^{\mathfrak{M}}[k])$ if $f \in Fns_t$, $t(f) = n$, $\tau_1,\ldots,\tau_n \in Trm_t$.

ii) Now we define the validity of formulas:

- For atomic formulas of the form $r(\tau_1,\ldots,\tau_n)$:

$$\mathfrak{M} \models r(\tau_1,\ldots,\tau_n)[k] \stackrel{\text{def}}{\iff} \langle\tau_1^{\mathfrak{M}}[k],\ldots,\tau_n^{\mathfrak{M}}[k]\rangle \in r^{\mathfrak{M}},$$

for atomic formulas of the form $\tau \doteq \sigma$:

$$\mathfrak{M} \models (\tau \doteq \sigma)[k] \stackrel{\text{def}}{\iff} \tau^{\mathfrak{M}}[k] = \sigma^{\mathfrak{M}}[k].$$

- For negated formulas:

$$\mathfrak{M} \models \neg\varphi[k] \stackrel{\text{def}}{\iff} \text{ not } \mathfrak{M} \models \varphi[k],$$

- for conjunctions:

$$\mathfrak{M} \models (\varphi\wedge\psi)[k] \stackrel{\text{def}}{\iff} \mathfrak{M} \models \varphi[k] \text{ and } \mathfrak{M} \models \psi[k],$$

- for quantified formulas:

$$\mathfrak{M} \models \exists v\varphi[k] \stackrel{\text{def}}{\iff} \mathfrak{M} \models \varphi[k'] \text{ for some valuation } k' \text{ such}^4 \text{ that } k|_{(V\setminus\{v\})} = k'|_{(V\setminus\{v\})}.$$

By these, $\mathfrak{M} \models \varphi[k]$ has been defined for any $\varphi \in Fml_t$.

We say that φ is *valid* in $\mathfrak{M}$ (or $\mathfrak{M}$ is a *model* of φ) iff $\mathfrak{M} \models \varphi[k]$ for every valuation $k : V \longrightarrow M$ (in symbols: $\mathfrak{M} \models \varphi$). A formula φ is (logically) *valid* iff for all $\mathfrak{M} \in Mod_t$, $\mathfrak{M} \models \varphi$ (in symbols: $\models \varphi$). $\lhd$

Definition 2.3.6. (Meaning function) By the *meaning* of a formula φ in a model $\mathfrak{M}$ we mean:

$$mng(\varphi,\mathfrak{M}) = \{k : k \text{ is a valuation of V and } \mathfrak{M} \models \varphi[k]\} \qquad \lhd$$

[4] Throughout the document, we will use the notation $f|_C$ for the *restriction* of f to C, that is for a function $f : A \longrightarrow B$ and a set $C \subseteq A$: $f|_C \stackrel{\text{def}}{=} \{\langle x, y\rangle \in f : x \in C\}$.

Intuition behind FOL: A model $\mathfrak{M}$ is a possible world for the language Fml_t, the terms are phrases referring to elements of the world $\mathfrak{M}$. Formulas are statements about $\mathfrak{M}$, hence they might be valid or not in $\mathfrak{M}$. The variables play the role of pronouns, like "this", "that", "he" and so on. These can be evaluated to denote any element of $\mathfrak{M}$.

By these, we defined First-order logic.

Remark 2.3.7. A very important part of using logical systems is examining what can be proven by using the logics inference system. Formulas are built up from simple characters, that do not have meanings unless we give them any. An interesting question is if we can know anything about the truth of certain statements in general, only by using this inference system, without connecting the symbols to their meanings. The answer is positive and there is more to be mentioned. The 20^{th} century brought celebrated results of the research in logics, for instance Gödel's completeness theorem stated that in First-order logic, what can be proven "on paper with pencil", only by using the symbols and the inference system, that is true in semantical sense, as well. In this thesis we just tangentially deal with this topic, but the interested reader can find a huge amount of studies about this. $\triangleleft$

2.4 The n-variable fragment of First-order logic

Definition 2.4.1. (First-order logic with n variables $\mathcal{L}_n$)

We fix a set $Cn_{\mathcal{L}_n} = \{\neg, \wedge, \{\exists v_i : i \leq n\}\}$ (called the *logical connectives*), where $rank(\neg) = rank(\exists v_i) = 1$, for all $i \leq n$ and $rank(\wedge) = 2$. Let t be an arbitrary but fixed similarity type. Let $V \stackrel{\text{def}}{=} \{v_0, \ldots, v_{n-1}\}$ be a set of basic symbols, for which the only prescription is that $V \cap (Dom(t) \cup C) = \emptyset$. V is a *set of variables* of $\mathcal{L}_n$. Let the set P of atomic formulas of $\mathcal{L}_n$ be defined as $P \stackrel{\text{def}}{=} \{r_i(v_0, \ldots, v_{n-1}) : i \in I\}$ for some set I.

i) The *set of formulas* (F_n) is the smallest set H satisfying:

- $P \subseteq H$
- $v_i = v_j \in H$, for all $i, j < n$
- $\varphi, \psi \in H, v_i \in V \Longrightarrow (\varphi \wedge \psi), \neg\varphi, \exists v_i \varphi \in H$

ii) The *class of models* (M_n) of $\mathcal{L}_n$ is:

$$M_n \stackrel{\text{def}}{=} \{\langle M, R_i \rangle_{i \in I} : M \text{ is a non-empty set and for all } i \in I, R_i \subseteq {}^n M\}.$$

M is called the *universe* of $\mathfrak{M}$, if $\mathfrak{M} = \langle M, R_i \rangle_{i \in I} \in M_n$.

iii) To define the meaning function we introduce a ternary relation $\models \subseteq (M_n \times F_n \times {}^n M)$. Let $\mathfrak{M} = \langle M, R_i \rangle_{i \in I} \in M_n$, $\varphi \in F_n$ and $q \in {}^n M$. We use the notation $\mathfrak{M} \models \varphi[q]$ (the evaluation q satisfies φ in the model $\mathfrak{M}$) and define this relation by recursion on the complexity of φ as follows:

- $\mathfrak{M} \models r_i(v_0, \ldots, v_{n-1})[q] \stackrel{\text{def}}{\Longleftrightarrow} q \in R_i (i \in I)$
- $\mathfrak{M} \models (v_i = v_j)[q] \stackrel{\text{def}}{\Longleftrightarrow} q_i = q_j \ (i, j < n)$
- let $\psi_1, \psi_2 \in F_n$,
  - $\mathfrak{M} \models \neg\psi_1[q] \stackrel{\text{def}}{\Longleftrightarrow}$ not $\mathfrak{M} \models \psi_1[q]$
  - $\mathfrak{M} \models (\psi_1 \wedge \psi_2)[q] \stackrel{\text{def}}{\Longleftrightarrow} \mathfrak{M} \models \psi_1[q]$ and $\mathfrak{M} \models \psi_2[q]$

$\circ\ \mathfrak{M} \models \exists v_i \psi_1[q] \overset{\text{def}}{\iff} (\exists q' \in {}^nM)(\forall j < n)\Big(j \neq i \Rightarrow (q'_j = q_j \text{ and } \mathfrak{M} \models \psi_1[q'])\Big)$

Now we define the *meaning function* as:

$$mng_n(\varphi, \mathfrak{M}) \overset{\text{def}}{=} \{q \in {}^nM : \mathfrak{M} \models \varphi[q]\}$$

iv) The *validity relation* is defined by

$$\mathfrak{M} \models_n \varphi \overset{\text{def}}{\iff} (\forall q \in {}^nM)\ \mathfrak{M} \models \varphi[q].$$

With these four items defined, *First-order logic with n variables*:

$$\mathcal{L}_n \overset{\text{def}}{=} \langle F_n, M_n, mng_n, \models_n \rangle$$ ◁

Remark 2.4.2. $\mathcal{L}_n$ is a little unusual, because the substitution of variables in atomic formulas is not allowed $\big(r_i(v_0, \dots, v_j)\big)$. This causes no harm, though, since this substitution is expressible by using quantifiers and equality. ◁

Now, we define $\mathcal{L}_{FOL}$, the rank-free formulation of the above. The definition is almost the same, but to clarify the notations, the whole definition is shown below, even the parts that are the same as in the restricted form.

2.5 First-order logic, rank-free formulation

Definition 2.5.1. (First-order logic $\mathcal{L}_{FOL}$, rank-free formulation)

The set of natural numbers is denoted by ω. Let $V \overset{\text{def}}{=} \{v_i : i \in \omega\}$ be a set, called the *set of variables* of $\mathcal{L}_{FOL}$. Let P be an arbitrary set, the set of *atomic formulas.* (Now, we think of atomic formulas as relation symbols, therefore we prefer using the notation R for elements of P, to the notation p.)

i) The *set of formulas* (F_{FOL}) is the smallest set H satisfying:

- $P \subseteq H$
- $v_i = v_j \in H$, for all $i, j \in \omega$
- $\varphi, \psi \in H, i \in \omega \Longrightarrow (\varphi \wedge \psi), \neg\varphi, \exists v_i \varphi \in H$

ii) The *class of models* (M_{FOL}) of $\mathcal{L}_{FOL}$ is:

$$M_{FOL} \overset{\text{def}}{=} \{\mathfrak{M} : \mathfrak{M} = \langle M, R^{\mathfrak{M}} \rangle_{R \in P}, M \text{ is a non-empty set and } \forall R \in P, R^{\mathfrak{M}} \subseteq {}^nM \text{ for some } n \in \omega\}.$$

If $\mathfrak{M} \in M_{FOL}$ then M denotes the universe of $\mathfrak{M}$. Further, for $R \in P$, $R^{\mathfrak{M}}$ denotes the meaning of R in $\mathfrak{M}$.

iii) Just as in the case of $\mathcal{L}_n$, to define the meaning function we introduce a ternary relation $\models$, such that $\models \subseteq (M_{FOL} \times F_{FOL} \times {}^{\omega}M)$. Let $\mathfrak{M} = \langle M, R_{\mathfrak{M}} \rangle_{R \in P} \in M_{FOL}$, $\varphi \in F_{FOL}$, $q \in {}^{\omega}M$ and $R \in P$. We define this relation by recursion on the complexity of φ as follows:

- $\mathfrak{M} \models R[q] \overset{\text{def}}{\iff} \langle q_0, \dots, q_{n-1} \rangle \in R^{\mathfrak{M}}$ for some $n \in \omega$
- $\mathfrak{M} \models (v_i = v_j)[q] \overset{\text{def}}{\iff} q_i = q_j\ (i, j \in \omega)$
- let $\psi_1, \psi_2 \in F_{FOL}$,

- $\circ\ \mathfrak{M} \models \neg\psi_1[q] \overset{\text{def}}{\iff}$ not $\mathfrak{M} \models \psi_1[q]$
- $\circ\ \mathfrak{M} \models (\psi_1 \wedge \psi_2)[q] \overset{\text{def}}{\iff} \mathfrak{M} \models \psi_1[q]$ and $\mathfrak{M} \models \psi_2[q]$
- $\circ\ \mathfrak{M} \models \exists v_i \psi_1[q] \overset{\text{def}}{\iff} (\exists q' \in {}^{\omega}M)(\forall j \in \omega) \Big(j \neq i \Rightarrow (q'_j = q_j \text{ and } \mathfrak{M} \models \psi_1[q']) \Big)$

We define the *meaning function* as:

$$mng_{FOL}(\varphi, \mathfrak{M}) \overset{\text{def}}{=} \{q \in {}^{\omega}M : \mathfrak{M} \models \varphi[q]\}$$

iv) The *validity relation* is defined by

$$\mathfrak{M} \models_{FOL} \varphi \overset{\text{def}}{\iff} (\forall q \in {}^{\omega}M)\ \mathfrak{M} \models \varphi[q].$$

First-order logic (rank-free form) is:

$$\mathcal{L}_n \overset{\text{def}}{=} \langle F_{FOL}, M_{FOL}, mng_{FOL}, \models_{FOL} \rangle \qquad \triangleleft$$

Remark 2.5.2. Unlike in $\mathcal{L}_{S5}$, where the "basic semantical units" were the possible situations ($w \in W$), in FOL, these are the evaluations of individual variables into models $\mathfrak{M}$. $\triangleleft$

2.6 A logic of programs: the First-order 3-sorted logic $\mathcal{L}_{td}$

To show a logical system that is less widely known, than the ones introduced so far, in this section we recall a logic from [1], that is used in *theoretical computer science* for expressing (and proving) statements about programs. This logic is a special many-sorted logic.

Many-sorted FOL is similar to classical FOL. The motivation is the following. When using classical FOL, the things we talk about can often be sorted into several groups, according to certain properties. We might say that this is not a problem, because we can add new unary relation symbols to the similarity type of the logic, which would do the sorting.

Another solution assumes that the universe of a model must consist of subuniverses, according to the properties we want to consider. In our example, we want to speak about *data, time, computation.* According to these, a model will contain a *data structure*, a *time scale* and a *set of functions.* As the expression "data *structure*" indicates, the fundamental operations and relations of a model will include functions and relations "functioning" inside the data universe. Similarly, we may want to assume that the time scale consists of discrete points, plus a "the next time point" function (like successor). Functions may be chosen to be a bare set. They are, intuitively, the computations themselves. We may also need expressions naming – intuitively – the value of a computation (function) at a certain time point. This will have to be an operation working between several universes: $value(f, t)$ will be an element of the data universe, attached to a computation f at the time point t.

The same ideas carry over to "*logics of actions*" investigated in philosophical logic. In this case, the actions correspond to computations (or programs). A statement like "After doing p, it will be the case that φ" contains p from the function universe, while φ is a property of data (a classical FOL statement about data items).

To define a many-sorted logic, first a *set of sorts* must be fixed, then a many-sorted similarity type will tell us not only the number of arguments of the symbols, but also the sorts of the arguments. That is, a many-sorted similarity type associates the "appropriate" sorts from a set S of sorts to a set Σ of relation symbols – that has a subset H of function symbols. Formally:

Definition 2.6.1. (Many-sorted similarity type) Let S be a set and let S^* – as before – denote the set of all finite sequences of the elements of S, that is $S^* \stackrel{\text{def}}{=} \bigcup\{{}^nS : n \in \omega\}$ (where ω is the set of natural numbers). By a *many-sorted similarity type* m we understand a triple $\langle S, H, m\rangle$, such that

- S is called the *set of sorts*
- m is a funcion $m : \Sigma \to S^*$ for some set Σ,
- $H \subseteq \Sigma$ (called the *function symbols*)
- $(\forall r \in \Sigma) m(r) \notin {}^0S$. $\triangleleft$

Given a many-sorted similarity type $\langle S, H, m\rangle$, by a *model* of this similarity type we understand a pair $\big\langle \langle U_s : s \in S\rangle, R\big\rangle$, where

- U_s is a non-empty set for every $s \in S$,
- R is a function with $Dom(R) = Dom(m) =: \Sigma$
- if $r \in \Sigma$ and $m(r) = \langle s_1, \dots, s_n\rangle$, then $R(r) \subseteq U_{s_1} \times \ldots \times U_{s_n}$
- of $r \in H$ in addition, then $R(r) : U_{s_1} \times \ldots \times U_{s_{n-1}} \longrightarrow U_{s_n}$.

U_s is called the *universe of sort s*.

In our example, the (usual) similarity type of arithmetic will be used (our intention is to make it the similarity type of the time structure). Therefore, let us remember that the similarity type t of arithmetic consists of the function symbols $+, \cdot, 0, 1$, with arities 2,2,0,0 and the binary relation symbol $\leq$. We imagine that we are given a (one-sorted) FOL similarity type d for the data structure, then "expand" this d to a 3-sorted similarity type td as follows.

Definition 2.6.2. (The 3-sorted similarity type td) Let t denote the similarity type of arithmetic and let d be an arbitrary (one-sorted) similarity type, for which we assume that $Dom(d) \cap Dom(t) = \emptyset$. To d, we associate a 3-sorted similarity type td as follows:

i) $S \stackrel{\text{def}}{=} \{t, d, i\}$, and so $|S| = 3$ (S is the *set of sorts* of td, where we call t, d, i *time, data* and *intensions*, respectively)
ii) $K \stackrel{\text{def}}{=} \{+, \cdot, 0, 1, ext\} \cup H$ (K is the *set of operation symbols* of td, the name of ext is *extension*)
iii) We have one relation symbol: $\leq$
iv) $td : \Big(Dom(t) \cup Dom(d) \cup ext\Big) \to S^*$, such that
  - $td(ext) = \langle i, t, d\rangle$
  - $td(+) = td(\cdot) = \langle t, t, t\rangle$
  - $td(\leq) = \langle t, t\rangle$
  - $td(0) = td(1) = \langle t\rangle$ $\triangleleft$

Now we can formulate the logic itself.

Definition 2.6.3. (The First-order 3-sorted logic $\mathcal{L}_{td}$ of similarity type td) Let d be an arbitrary similarity type. As we defined above, td is a 3-sorted similarity type with sorts $\{t, d, i\}$.

i) We define the set of First-order 3-sorted formulas (of type td, through variables and terms. Let $X \stackrel{\text{def}}{=} \{x_w : w \in \omega\}$, $Y \stackrel{\text{def}}{=} \{y_w : w \in \omega\}$ and $Z \stackrel{\text{def}}{=} \{z_w : w \in \omega\}$ be three mutually disjoint sets of distinct elements (that is, $w \neq j \in \omega$, then $x_w \neq x_j$, $y_w \neq y_j$ and $z_w \neq z_j$). X, Y, Z are the sets of variables of sorts d, i, t, respectively.

   F_t^Z (F_d^X) is the set of all First-order formulas of type t (d) with variables from Z (X) and Tm_t^Z is the set of all First-order terms of type t with variables from Z.

   The set $Tm_{td,d}$ of terms of type td and of sort d is defined to be the smallest set satisfying conditions 1)-3):

1) $X \subseteq Tm_{td,d}$,
2) $ext(y_w, \tau) \in Tm_{td,d}$, for any $\tau \in Tm_t^Z$ and $w \in \omega$,
3) $f(\tau_1, \ldots, \tau_n) \in Tm_{td,d}$ for any $f \in H$, where $d(f) = n + 1$ and $\tau_1, \ldots, \tau_n \in Tm_{td,d}$.

Now, we can define the set F_{td} of First-order *formulas* of type td; it is the smallest set satisfying 4)-8) below:

4) $(\tau_1 = \tau_2) \in F_{td}$ for any $\tau_1, \tau_2 \in Tm_{td,d}$,
5) $r(\tau_1, \ldots, \tau_n) \in F_{td}$ for any $\tau_1, \ldots, \tau_n \in Tm_{td,d}$ and for any $r \notin H$ if $d(r) = n$,
6) $(y_w = y_j) \in F_{td}$ for any $w, j \in \omega$,
7) $F_t^Z \subseteq F_{td}$,
8) if $\varphi, \psi \in F_{td}$, then

$$\{\neg\varphi, (\varphi \wedge \psi), (\exists z_w \varphi), (\exists x_w \varphi), (\exists y_w \varphi) : w \in \omega\} \subseteq F_{td}$$

ii) The *class of models* of $\mathcal{L}_{td}$ is defined as:

$$M_{td} \stackrel{\text{def}}{=} \Big\{\mathfrak{M} : \mathfrak{M} = \big\langle \langle U_t, U_d, U_i \rangle, R_r \big\rangle_{r \in \Sigma} \Big\}, \text{ where}$$

- $\langle U_t, R_r \rangle_{r \in Dom(t_1)} \in M_t$,
- $\langle U_d, R_r \rangle_{r \in Dom(d_1)} \in M_d$,
- $R_{ext} : U_i \times U_t \to U_d$

iii) The *meaning function* of this logic is very similar to what we are used to at one-sorted FOL. By an evaluation (of the variables) into $\mathfrak{M} \in M_{td}$ we understand a triple $v = \langle g, k, r \rangle$, such that $g \in {}^\omega U_t$, $k \in {}^\omega U_d$ and $r \in {}^\omega U_i$. The notation of the statement "the evaluation v satisfies φ in $\mathfrak{M}$" is denoted by $\mathfrak{M} \models \varphi[v]$ (or $\mathfrak{M} \models \varphi[g, k, r]$). We define the truth of this statement completely analogously of the one-sorted case, hence we will only show some examples:

- $\mathfrak{M} \models (x_i = x_j)[g, k, r] \iff k_i = k_j$ for $i, j, \in \omega$,
- $\mathfrak{M} \models \big(x_i = ext(y_j, z_l)\big)[g, k, r] \iff k_i = ext^{\mathfrak{M}}(r_j, g_l)$,
- $\mathfrak{M} \models \varphi[g, k, r] \iff \langle U_t, R_r \rangle_{r \in Dom(t_1)} \models \varphi[g]$ for $\varphi \in F_t^Z$,
- $\mathfrak{M} \models \varphi[g, k, r] \iff \langle U_d, R_r \rangle_{r \in Dom(d_1)} \models \varphi[k]$ for $\varphi \in F_d^X$,

We can now define the meaning function:

$$mng_{td}(\varphi, \mathfrak{M}) \stackrel{\text{def}}{=} \{v = \langle g, k, r \rangle : \mathfrak{M} \models \varphi[v]\}$$

iv) The validity relation $\models_{td}$ is also defined through the relation used above, namely:

$$\mathfrak{M} \models_{td} \varphi \iff (\forall g \in {}^\omega T)(\forall k \in {}^\omega D)(\forall r \in {}^\omega I)\ \mathfrak{M} \models \varphi[g, k, r]$$

With these defined, our 3-sorted logic is:

$$\mathcal{L}_{td} \stackrel{\text{def}}{=} \langle F_{td}, M_{td}, mng_{td}, \models_{td} \rangle \qquad \triangleleft$$

Remark 2.6.4. a) In condition 5) above, we implicitly assume, that $r \in Dom(d)$ (to be more accurate, $r \in Dom(d) \backslash H$).
b) An important fact, is that $F_d \subseteq F_{td}$.
c) Some notation in the definition might be declared earlier, for more information see definitions 2.3.1. and 2.6.1. $\triangleleft$

Chapter 3

New natural requirements on the general concept of logics

In this chapter, we will analyze the logical systems introduced above, by investigating if they have some special properties. Here we will give the explanation of why we defined First-order logic in two different ways – the classical concept does not satisfy a substitution property (to be introduced), while the other concept does. The reason we choose exactly these properties is that these play an essential role in algebraizing logics; this will be clarified in definition 5.1.1. on page 34.

3.1 Compositionality

Each of the logics introduced above were built up with respect to a set of *logical connectives*, which we did not really specify before. Now it is time to.

Definition 3.1.1. ($\mathcal{L}$ has logical connectives) $\mathcal{L}$ has logical connectives if and only if i)-ii) below hold.

i) set $Cn(\mathcal{L})$, the set of logical connectives of $\mathcal{L}$ is fixed, and every $c \in Cn(\mathcal{L})$ has a rank ($rank(c) \in \omega$).

ii) There is a set P, called the set of atomic formulas, such that $F_{\mathcal{L}}$ (the set of formulas in $_{\mathcal{L}}$) is the smallest set satisfying a)-b) below:

   a) $P \subseteq F$

   b) if $c \in Cn_k(\mathcal{L})$ and $\varphi_1, \dots, \varphi_k \in F_{\mathcal{L}}$, then $c(\varphi_1, \dots, \varphi_k) \in F_{\mathcal{L}}$. ◁

Remark 3.1.2. The set of all logical connectives of $Cn(\mathcal{L})$ for which $rank(c) = k$ is denoted by $Cn_k(\mathcal{L})$. The word-algebra generated by P using the logical connectives from $Cn(\mathcal{L})$ as algebraic operations is denoted by $\mathfrak{F}$, that is $\mathfrak{F} = \langle F, c\rangle_{c \in Cn(\mathcal{L})}$ (and $\mathfrak{F}$ is called the *formula algebra* of $\mathcal{L}$). ◁

It is easy to see, that all the logics defined above have logical connectives, namely:

- $Cn(\mathcal{L}_S) = \{\neg, \wedge\}$;
- $Cn(S5) = \{\neg, \wedge, \Diamond\}$;
- $Cn(FOL) = \{\neg, \wedge, \exists v, \doteq\}$,
- $Cn(\mathcal{L}_n) = Cn(\mathcal{L}_{FOL}) = Cn(\mathcal{L}_{td}) = \{\neg, \wedge, \exists v\}$,

where $rank(\neg) = rank(\Diamond) = rank(\exists v) = 1$, $rank(\wedge) = rank(\doteq) = 2$, (and the definitions of the logics implicate directly that condition ii) holds).

Definition 3.1.3. (Compositionality) A logic $\mathcal{L}$ is compositional iff it has logical connectives, and the meaning function ($mng_{\mathfrak{M}}$) is a homomorphism from $\mathfrak{F}$, for every $\mathfrak{M} \in M$ (so that it respects the logical relations). $\triangleleft$

An equivalent definition to compositionality is that for all k-ary connective $c \in Cn_k(\mathcal{L})$ and $\varphi_i, \psi_i \in F_{\mathcal{L}}$, $1 \leq i \leq k$:

$$\Big(\bigwedge_{i=1}^{k} mng_{\mathfrak{M}}(\varphi_i) = mng_{\mathfrak{M}}(\psi_i)\Big) \Longrightarrow mng_{\mathfrak{M}}\big(c(\varphi_1, \ldots, \varphi_k)\big) = mng_{\mathfrak{M}}\big(c(\psi_1, \ldots, \psi_k)\big).$$

Remark 3.1.4. Looking behind the formal definition, we see that the name 'compositionality' is neat, a logical system has this property if the meanings of all of its formulas are composed of the meanings of their subformulas. $\triangleleft$

Theorem 3.1.5. The logics $\mathcal{L}_S$, $S5$, $\mathcal{L}_n$, $\mathcal{L}_{FOL}$ and $\mathcal{L}_{td}$ defined above are compositional.

Proof. We examine the connection between the meaning function and the logical connectives. In this proof, φ, ψ are arbitrary formulas and $\mathfrak{M}$ is an arbitrary model of the corresponding logic. We want to prove that the meaning function is a homomorphism on $\mathfrak{F}$ ($= \langle F, c\rangle_{c \in Cn(\mathcal{L})}$), so we have to check if for every $c \in Cn(\mathcal{L}) : mng\big(c(\varphi_1, \ldots, \varphi_n), \mathfrak{M}\big) = c_{\mathfrak{M}}\big(mng(\varphi_1, \mathfrak{M}), \ldots, mng(\varphi_n, \mathfrak{M})\big)$, where $c_{\mathfrak{M}}$ is the corresponding operation to c in $Rng(mng)$.

i) $\mathcal{L}_S$: It is easy to see, that the corresponding operators to the logical connectives are taking the complement (in case of $\neg$), and the intersection (at $\wedge$).

   ($\neg$): $mng_S(\neg\varphi) = \{w \in W : w \Vdash_v \neg\varphi\} = W \backslash \{w \in W : w \Vdash_v \varphi\} = \overline{mng_S(\varphi)}$

   ($\wedge$): $mng_S(\varphi \wedge \psi) = \{w \in W : w \Vdash_v \varphi \text{ and } w \Vdash_v \psi\} = \{w \in W : w \Vdash_v \varphi\} \cap \{w \in W : w \Vdash_v \psi\} =$
   $= mng_S(\varphi) \cap mng_S(\psi)$

ii) $S5$: The same proof can be given for the two connectives above in case of $S5$, as well. We still have to see if $\Diamond$ satisfies the condition. The corresponding set operation to $\Diamond$ is $\Diamond_{\mathfrak{M}}(H) = \{w \in W : (\exists w' \in H)\big(R(w, w')\big)\}$ (for every $H \subseteq Rng(mng)$), where R is the accessibility relation of W (in this case, for every $w, z \in W : R(w, z)$ holds).

   ($\Diamond$): $mng_{S5}(\Diamond\varphi) = \{w \in W : (\exists w' \in W) w' \Vdash_v \varphi\} = \Diamond_{\mathfrak{M}}(mng_{S5}\varphi)$

iii) $\mathcal{L}_n$: The case of the first two logical connectives is analogous to what we have already seen at $\mathcal{L}_S$, but since the meaning functions are somewhat different, we show these, as well. [5]

   ($\neg$): $mng_n(\neg\varphi, \mathfrak{M}) = \{q \in {}^nM : \mathfrak{M} \models \neg\varphi[q]\} = {}^nM \backslash \{q \in {}^nM : q \models \varphi[q]\} = \overline{mng_n(\varphi, \mathfrak{M})}$

   ($\wedge$): $mng_n(\varphi \wedge \psi[q], \mathfrak{M}) = \{q \in {}^nM : \mathfrak{M} \models \varphi[q] \text{ and } \mathfrak{M} \models \psi[q]\} =$
   $= \{q \in {}^nM : \mathfrak{M} \models \varphi[q]\} \cap \{q \in {}^nM : \mathfrak{M} \models \psi[q]\} = mng_n(\varphi, \mathfrak{M}) \cap mng_S(\psi, \mathfrak{M})$

   ($\exists v_i$): $mng_n(\exists v_i \psi[q]) = \big\{q \in {}^nM : \exists q' \in {}^nM\ \forall j < n\big(j \neq i \Rightarrow (q'_j = q_j \text{ and } \mathfrak{M} \models \psi[q'])\big)\big\} =$
   $\exists_{\mathfrak{M}}(mng_n(\psi))$

iv) In case of $\mathcal{L}_{FOL}$, the proof is completely the same as for $\mathcal{L}_n$, and also in the case of $\mathcal{L}_{td}$, it is very similar, the handling of the different sorts causes no difficulties. *Q.E.D.*

[5] It is a "folklore" to say that the "basic semantical unit" of $\mathcal{L}_S$ is a state w, while the "basic semantical unit" of $\mathcal{L}_n$ is a valuation of the variables. The proofs for $\{\neg, \wedge\}$ in the cases of $\mathcal{L}_S$ and $\mathcal{L}_n$ differ only in that their basic semantical units are different.

3.2 Filter property

A derived connective is a scheme of terms in the language of the word-algebra $\mathfrak{F}$. In words, it is an abbreviation of a longer formula, for example the well-known $\vee$ is available as a derived connective in all our introduced logics as: $\varphi \vee \psi \stackrel{\text{def}}{=} \neg(\neg\varphi \wedge \neg\psi)$. These connectives will have an important role in the following property.

Definition 3.2.1. (Filter property) We say that $\mathcal{L}$ has the filter property iff there are derived connectives $\epsilon_0, \dots, \epsilon_{m-1}$ and $\delta_0, \dots, \delta_{m-1}$ (unary) and $\Delta_0, \dots, \Delta_{n-1}$ (binary) of $\mathcal{L}$ $(m, n \in \omega)$, such that the following two hold:

i) For every $\mathfrak{M} \in M$ and $\varphi, \psi \in F$,

$$mng_{\mathfrak{M}}(\varphi) = mng_{\mathfrak{M}}(\psi) \iff (\forall i < n)\ \mathfrak{M} \models \varphi \Delta_i \psi.$$

ii) For every $\mathfrak{M} \in M$ and $\varphi \in F$,

$$\mathfrak{M} \models \varphi \longleftrightarrow (\forall j < m)(\forall i < n)\ \mathfrak{M} \models \epsilon_j(\varphi)\Delta_i\delta_j(\varphi). \qquad \triangleleft$$

Before examining if in our logical systems we have this property or not, recall the definition of a logic (definition 1.1.2), especially condition v). In logical systems, that are "constructed well" (by which we mean it bears some properties) this condition automatically holds.

Statement 3.2.2. If a logic $\mathcal{L}$ is compositional and has the filter property, then 1.1.2. v) automatically holds for $\mathcal{L}$. $\triangleleft$

Proof. We assume that *a)* $mng_{\mathcal{L}}(\varphi, \mathfrak{M}) = mng_{\mathcal{L}}(\psi, \mathfrak{M})$ and that *b)* $\mathfrak{M} \models_{\mathcal{L}} \varphi$. From the filter property's ii) condition at *b)*, we know that $\mathfrak{M} \models_{\mathcal{L}} \epsilon(\varphi)\Delta\delta(\varphi)$, that implies (by filter property's condition i)), that $mng_{\mathcal{L}}(\epsilon(\varphi), \mathfrak{M}) = mng_{\mathcal{L}}(\delta(\varphi), \mathfrak{M})$. Using our assumptions of compositionality and the equality of the formulas' meanings, we have that

$$\begin{aligned} mng_{\mathcal{L}}(\epsilon\varphi, \mathfrak{M}) = \epsilon_{\mathfrak{M}} mng_{\mathcal{L}}(\varphi, \mathfrak{M}) = \epsilon_{\mathfrak{M}} mng(\psi, \mathfrak{M}) = mng_{\mathcal{L}}(\epsilon\psi, \mathfrak{M}) = mng_{\mathcal{L}}(\delta\psi, \mathfrak{M}) = \\ = \delta_{\mathfrak{M}} mng(\psi, \mathfrak{M}) = \delta_{\mathfrak{M}} mng(\varphi, \mathfrak{M}) = mng_{\mathcal{L}}(\delta\varphi, \mathfrak{M}) \end{aligned}$$

and since the two edges are known to be equal, all these are equal. By $mng_{\mathcal{L}}(\epsilon\psi, \mathfrak{M}) = mng_{\mathcal{L}}(\delta\psi, \mathfrak{M})$ (using filter property's i)) we know that $\mathfrak{M} \models_{\mathcal{L}} \epsilon\psi\Delta\delta\psi$, which means (according to filter property's ii)) that $\mathfrak{M} \models_{\mathcal{L}} \psi$. *Q.E.D.*

Now let us see if our logics have filter property. First, we introduce some derived connections, that are:

($\rightarrow$): $\varphi \rightarrow \psi \stackrel{\text{def}}{=} \neg(\varphi \wedge \neg\psi)$
($\leftrightarrow$): $\varphi \leftrightarrow \psi \stackrel{\text{def}}{=} (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi) = \big(\neg(\varphi \wedge \neg\psi)\big) \wedge \big(\neg(\psi \wedge \neg\varphi)\big)$
($\vee$): $\varphi \vee \psi \stackrel{\text{def}}{=} \neg(\neg\varphi \wedge \neg\psi)$
($True$): $True(\varphi) \stackrel{\text{def}}{=} \varphi \vee \neg\varphi = \neg(\varphi \wedge \neg\varphi)$
(Id): $Id(\varphi) \stackrel{\text{def}}{=} \varphi$

Theorem 3.2.3. The logics $\mathcal{L}_S$, $S5$, $\mathcal{L}_n$, $\mathcal{L}_{FOL}$ and $\mathcal{L}_{td}$ have the filter property.

Proof. In this proof we substitute the derived connectives defined above into the definition, so that $\epsilon\varphi = True(\varphi)$, $\Delta = \leftrightarrow$ and $\delta = Id$. By roman numerals we will mean the two conditions of filter property. We start with the case of $\mathcal{L}_S$.

i) Let us see what the right side of this condition means:

$$\begin{aligned}
&\mathfrak{M} \models_S \varphi \leftrightarrow \psi \\
&\iff \mathfrak{M} \models_S (\neg(\varphi \wedge \neg\psi)) \wedge (\neg(\psi \wedge \neg\varphi)) \\
&\iff \forall w \in W : w \Vdash_v \neg(\varphi \wedge \neg\psi) \text{ and } w \Vdash_v \neg(\neg\varphi \wedge \psi) \\
&\iff \forall w \in W : w \nVdash_v \varphi \wedge \neg\psi \text{ and } w \nVdash_v \neg\varphi \wedge \psi \\
&\iff \forall w \in W : (w \nVdash_v \varphi \text{ or } \underbrace{w \Vdash_v \psi}_{\Leftrightarrow w \Vdash_v \varphi}) \text{ and } (\underbrace{w \Vdash_v \varphi}_{\Leftrightarrow w \Vdash_v \psi} \text{ or } w \nVdash_v \psi)
\end{aligned}$$

If we assume that $mng_S(\varphi, \langle W, v\rangle) = mng_S(\psi, \langle W, v\rangle)$ ($\Longrightarrow$), then we can substitute the formulas written under the brackets, arriving to a statement that always holds. On the other hand, assuming that $\mathfrak{M} \models_S \varphi \leftrightarrow \psi$ ($\Longleftarrow$) we have that this can only hold if $mng_S(\varphi, \langle W, v\rangle) = mng_S(\psi, \langle W, v\rangle)$, since if there exists a $w \in W$ for which $w \Vdash_v \varphi$, but $w \nVdash_v \psi$, then the last line of the deduction above is not true, which would mean that our assumption doesn't hold.

ii) What we need is: $\mathfrak{M} \models_S \varphi \iff \mathfrak{M} \models_S True(\varphi) \leftrightarrow \varphi$. Once again, we examine the left side:

$$\begin{aligned}
&\mathfrak{M} \models_S True(\varphi) \leftrightarrow \varphi \\
&\iff \forall w \in W : w \Vdash_v (\varphi \vee \neg\varphi) \leftrightarrow \varphi \\
&\iff \forall w \in W : w \Vdash_v \underbrace{\neg\big[\underbrace{(\varphi \vee \neg\varphi) \wedge \neg\varphi}_{\Leftrightarrow \neg\varphi}\big]}_{\text{holds iff } \varphi \text{ holds}} \wedge \underbrace{\neg\big[\varphi \wedge \underbrace{\neg(\varphi \vee \neg\varphi)}_{\text{never holds}}\big]}_{\text{always holds}}
\end{aligned}$$

We got that the right side holds if and only if $\mathfrak{M} \models_S \varphi$, which proves the statement, thus $\mathcal{L}_S$ has the filter property.

For $S5$ the exact same proof can be told, and for $\mathcal{L}_n$, $\mathcal{L}_{FOL}$ it is completely analogous.

The case of $\mathcal{L}_{td}$ is also similar, but not identical. First, recall that $mng_{td}(\varphi) = mng_{td}(\psi)$ means that the very same evaluations $\langle g, k, r\rangle$ satisfy the formulas φ and ψ in the model $\mathfrak{M}$.

i) If φ, ψ are formulas of the same sort, than it is the case we discussed above. Let us choose formulas of two different sorts, for example let $\varphi \in F_t^Z$ and $\psi \in F_d^X$.

$$\begin{aligned}
&\mathfrak{M} \models_{td} (\varphi \leftrightarrow \psi) \\
&\iff \forall g \in {}^\omega T,\, \forall k \in {}^\omega D,\, \forall r \in {}^\omega I : \mathfrak{M} \models (\varphi \leftrightarrow \psi)[g,k,r] \\
&\iff \forall g \in {}^\omega T,\, \forall k \in {}^\omega D,\, \forall r \in {}^\omega I : \mathfrak{M} \models \neg(\varphi \wedge \neg\psi)[g,k,r] \text{ and } \mathfrak{M} \models \neg(\psi \wedge \neg\varphi)[g,k,r] \\
&\iff \forall g \in {}^\omega T,\, \forall k \in {}^\omega D,\, \forall r \in {}^\omega I : \big(\mathfrak{M} \not\models \varphi[g,k,r] \text{ or } \mathfrak{M} \models \psi[g,k,r]\big) \\
&\qquad\qquad \text{and } \big(\mathfrak{M} \not\models \psi[g,k,r] \text{ or } \mathfrak{M} \models \varphi[g,k,r]\big) \\
&\iff \forall g \in {}^\omega T,\, \forall k \in {}^\omega D : \big(\langle U_t, R_r\rangle_{r \in Dom(t_1)} \not\models \varphi[g] \text{ or } \langle U_d, R_r\rangle_{r \in Dom(d_1)} \models \psi[k]\big) \\
&\qquad\qquad \text{and } \big(\langle U_t, R_r\rangle_{r \in Dom(t_1)} \models \varphi[g] \text{ or } \langle U_d, R_r\rangle_{r \in Dom(d_1)} \not\models \psi[k]\big) \qquad (*)
\end{aligned}$$

If we assume that $mng_{td}(\varphi, \mathfrak{M}) = mng_{td}(\psi, \mathfrak{M})$, we know that when choosing a triplet $\langle g, k, r\rangle$, either both $\mathfrak{M} \models \varphi[g,k,r]$ and $\mathfrak{M} \models \psi[g,k,r]$ or both $\mathfrak{M} \not\models \varphi[g,k,r]$ and $\mathfrak{M} \not\models \psi[g,k,r]$ holds, thus (*) above holds. The other way round, if we assume that (*) holds, it is easy to see that we can not choose an evaluation for which φ and ψ are not simultaneously valid or invalid.

We still need to think through the case of formulas that have subformulas of different sorts. Let $\theta \in F_{td}$ be like that. While building θ, at some point of the construction we have to use the connective $\wedge$. From this point, we can separate the formula into formulas that has subformulas of only one sort, and the formula θ is valid if both of our separated parts are valid, hence filter property's condition i) works for these kind of formulas, as well.

ii) The second condition of filter property holds for $\mathcal{L}_{td}$, it can be proven the same way as we did in case of $\mathcal{L}_S$. *Q.E.D.*

Remark 3.2.4. As we have already seen, these logics are compositional, and all have the filter property, thus according to statement 3.2.2, the fifth condition in the definition of a logic automatically holds in all of them. ◁

A special case of the filter property and a somewhat stronger condition is if choosing $m = n = 1$, $\Delta_0 =\leftrightarrow$, $\epsilon_0(\varphi) = True$ and $\delta_0(\varphi) = \varphi$ (as we used in the proofs above), that is

(a) $mng_{\mathfrak{M}}(\varphi) = mng_{\mathfrak{M}}(\psi) \iff \mathfrak{M} \models \varphi \leftrightarrow \psi$

(b) $\mathfrak{M} \models \varphi \iff \mathfrak{M} \models True \leftrightarrow \varphi$

Statement 3.2.5. The following connection between $\models$ and mng is implied by (a) and (b) together.

$$(\forall \varphi, \psi \in F)\big((\models \varphi \text{ and } \models \psi) \Longrightarrow (\forall \mathfrak{M} \in M)\ mng_{\mathfrak{M}}(\varphi) = mng_{\mathfrak{M}}(\psi)\tag{3.1}$$

◁

Proof. Assume that $\varphi, \psi \in F$ such that $\models \varphi$ and $\models \psi$ and fix these formulas. Let $\mathfrak{M} \in M$ be arbitrary but fixed, then $\mathfrak{M} \models \varphi$ and $\mathfrak{M} \models \psi$. By condition (b) we know that $\mathfrak{M} \models True \leftrightarrow \varphi$ and $\mathfrak{M} \models True \leftrightarrow \psi$. By condition (a), $mng_{\mathfrak{M}}(\varphi) = mng_{\mathfrak{M}}(True) = mng_{\mathfrak{M}}(\psi)$. *Q.E.D.*

3.3 Syntactical and semantical substitution properties

In this section, we will introduce two substitution properties of logics.

Definition 3.3.1. (Syntactical substitution property) $\mathcal{L}$ has the syntactical substitution property iff

$$(\forall \psi, \varphi_0, \dots, \varphi_k \in F_{\mathcal{L}})(\forall p_0, \dots, p_k \in P_{\mathcal{L}})\big(\models \psi(\overline{p}) \Longrightarrow \models \psi(\overline{p}/\overline{\varphi})\big),$$

where $\overline{p} = \langle p_0, \dots, p_k\rangle$, $\overline{\varphi} = \langle \varphi_0, \dots, \varphi_k\rangle$, and $\psi(\overline{p}/\overline{\varphi})$ denotes the formula that we get from ψ after simultaneously substituting φ_i for every occurrence of p_i $(i \le k)$ in ψ. ◁

Definition 3.3.2. (Semantical substitution property) $\mathcal{L}$ has the semantical substitution property iff

$$(\forall s \in {}^P F)(\forall \mathfrak{M} \in M)(\exists \mathfrak{N} \in M)(\forall \varphi(p_{i_0}, \dots, p_{i_k}) \in F)$$
$$mng_{\mathfrak{N}}(\varphi) = mng_{\mathfrak{M}}\big(\varphi(p_{i_0})/s(p_{i_0}, \dots, p_{i_k}/s(p_{i_k})\big)$$

holds. If $\hat{s} \in {}^F F$ is the natural extension of s to $\mathfrak{F}$, then this property says that

$$mng_{\mathfrak{N}}(\varphi) = mng_{\mathfrak{M}}(\hat{s}(\varphi)).$$

The model $\mathfrak{N}$ is the substituted version of $\mathfrak{M}$ along substitution s. ◁

Theorem 3.3.3. If $\mathcal{L}$ is a compositional logic, that has the filter property and the semantical substitution property, then $\mathcal{L}$ has the syntactical substitution property, as well. ◁

Proof. Assume that $\mathcal{L}$ is compositional, has the filter property and the semantical substitution property. Let $\varphi \in F$ be a formula and $s : F \longrightarrow F$ a substitution, both arbitrary but fixed. Let $\hat{s}$ denote the extension of s to a homomorphism from $\mathfrak{F}$ got $\mathfrak{F}$. Proving that

$$\models \varphi \Longrightarrow \models \hat{s}(\varphi)\tag{3.2}$$

is enough, so assume that $\models \varphi$. Let $\mathfrak{M} \in M$ be an arbitrary but fixed model. By the semantical substitution property, we have an $\mathfrak{N} \in M$ such that

$$(\forall \psi \in F)\ mng_{\mathfrak{N}}(\varphi) = mng_{\mathfrak{M}}\big(\hat{s}(\varphi)\big). \tag{3.3}$$

We know that $\mathfrak{N} \models \varphi$ (since $\models \varphi$).

$$\begin{array}{cr}
\mathfrak{N} \models \varphi & \\
\Updownarrow & \\
(\forall j < m)(\forall i < n)\ \mathfrak{N} \models \epsilon_j(\varphi)\Delta_i\delta_j(\varphi) & \text{(ii) of filter property)} \\
\Updownarrow & \\
(\forall j < m)\ mng_{\mathfrak{N}}\big(\epsilon_j(\varphi)\big) = mng_{\mathfrak{N}}\big(\delta_j(\varphi)\big) & \text{(i) of filter property)} \\
\Updownarrow & \\
(\forall j < m)\ mng_{\mathfrak{M}}\Big(\hat{s}\big(\epsilon_j(\varphi)\big)\Big) = mng_{\mathfrak{M}}\Big(\hat{s}\big(\delta_j(\varphi)\big)\Big) & (3.3) \\
\Updownarrow & \\
(\forall j < m)\ mng_{\mathfrak{M}}\Big(\epsilon_j\big(\hat{s}(\varphi)\big)\Big) = mng_{\mathfrak{M}}\Big(\delta_j\big(\hat{s}(\varphi)\big)\Big) & (\hat{s} \text{ is a homomorphism, } \mathcal{L} \text{ is compositional)} \\
\Updownarrow & \\
(\forall j < m)(\forall i < n)\ \mathfrak{M} \models \epsilon_j\big(\hat{s}(\varphi)\big)\Delta_i\delta_j\big(\hat{s}(\varphi)\big) & \text{(i) of filter property)} \\
\Updownarrow & \\
\mathfrak{M} \models \hat{s}(\varphi) &
\end{array}$$

Since $\mathfrak{M}$ was arbitrary, this proves (3.2). *Q.E.D.*

3.4 Nice and strongly nice logics

Definition 3.4.1. (Nice logic, strongly nice logic) We say that $\mathcal{L}$ is a *nice logic* iff it is compositional, has the filter property and has the syntactical substitution property. $\mathcal{L}$ is called a *strongly nice logic* iff it is nice, and it has the semantical substitution property, as well. ◁

Theorem 3.4.2. Sentential logic, modal logic $S5$, $\mathcal{L}_n$ and $\mathcal{L}_{FOL}$ are nice logics, furthermore, sentential logic, modal logic $S5$ and $\mathcal{L}_n$ are strongly nice logics. ◁

Part II

Definability

Chapter 4

Definability in First-order logic

It is clear (a glance at this thesis itself can be an example), that definitions have an essential role in modern, axiomatic thinking. What are definitions, what do we use them for and when do we say that our phrase indeed defined something? For example, if I say "Give me please the logic book from the shelf." might make sense at once, but an answer could easily be "There are more than one logic books on the shelf, please specify further which one you need.", or even "I do not see any logic books on the shelf, so I can not give you any". Our intuition tells us, that the definition is good if in our world, there is exactly one item, that satisfies the conditions, namely $"Mydef(v)" = Ontheshelf(v) \wedge Logicbook(v)$ is a good definition iff $myroom \Vdash \exists! v \; Mydef(v)$ (where $\exists!$ means that there exists one and only one). This might not be a good definition here, because there are no shelves in my room, but in the living room there is exactly one logic book on the shelf, which makes "$Mydef$" a perfect definition.

Definitions are basically tools for communication, since any defined symbol can be left out of our sentence, using only formulas of our logic without this newly defined symbol. For example we add a binary operation $\cap$ (intersection) to the language of ZFC set theory such that for any formula containing $\cap$ we can construct another formula of the language without our new symbol, and these two formulas mean exactly the same thing in $ZFC \cup \{\forall A, B \text{ sets } A \cap B = C \leftrightarrow C = \{x : x \in A \wedge x \in B\}\}$. The same way, we can define relations, as well. To do this, we say as much about a relation R that makes this unique.

A very important fact is that we rely on some background theory (or background knowledge), that we accept to hold, like Peano Arithmetics, or Zermelo–Fraenkel set theory (possibly expanded by the axiom of choice). In this section, if we do not mention else, we mean a FOL logic as defined before in section 2.3 (with the equation built into it). For now, we will concentrate on defining relation symbols (which is actually not a serious restriction: we can think of an n-place function symbol as a special, $n+1$-place relation symbol).

This chapter is based on the work of Hajnal Andréka and István Németi $\big([4]\big)$.

4.1 Preparation for examining definability

Ironically, first we give a definition for what a definition is. For this, we recall what a theory is from definition 1.1.5, on page 9.

Definition 4.1.1. (Definition of R in Th) Let $\mathcal{L}$ be a FOL language and Th a theory in $\mathcal{L}$, and let Σ be a theory in the language $\mathcal{L} \cup R$, where R is a new n-place relation symbol. Σ is a *definition of R in Th* iff for any model $\mathfrak{M}$ of Th there is exactly one $R \subseteq M^n$ for which $\langle \mathfrak{M}, R \rangle \models \Sigma$. ◁

When we define something, we give it a name (R), and a specification (an exact description $\Sigma(R)$ or sometimes we just write Σ) to it.

Example 4.1.2. In ZFC, we can define the binary function (or ternary relation) $\times$ for direct product as below:

$$\Sigma(\times) = \big\{ \times (A, B) = \{\langle a, b\rangle : a \in A, b \in B\}\big\}$$

This Σ defines the direct product in ZFC set theory, that is for every model $\mathfrak{M}$ of ZFC there is exactly one binary function (ternary relation) $f : M \longrightarrow M$, such that $\langle \mathfrak{M}, f\rangle \models \Sigma(f)$. $\lhd$

When there are some formulas in front of us (a description $\Sigma(R)$), it might not be obvious if it is a definition or not. For a kind of description, though, the form of the description ensures that it is a definition; these are called explicit definitions.

Definition 4.1.3. (Explicit definition of R) Let $\mathcal{L}$ be a FOL language, and let R be an n-place relation symbol ($R \notin \mathcal{L}$). Σ is an *explicit definition* of R (via φ) iff

$$\Sigma(R) = \big\{\forall x_1, \dots x_n \ \big(R(x_1, \dots x_n) \leftrightarrow \varphi\big)\big\}$$

for a $\varphi \in F_{\mathcal{L}}$ such that the free variables[6] of φ are from the set $\{x_1, \dots x_n\}$. $\lhd$

A definition that is not explicit is called *implicit*. Though implicit definitions are sometimes not as easy to understand, in some cases they are very useful, and even carry more information than explicit ones. Implicit definitions tells us what properties make R what it is, while an explicit definition tells how we can construct R.

An important and interesting question comes to mind when talking about implicit and explicit definitions, that is if every explicit definition has an equivalent implicit one or not (and of course the other way round). Now, we will search for the answer to this question in case of FOL languages, but to do so, we introduce some theorems that we will need later on.

Theorem 4.1.4. (Deduction theorem) Let Σ be a set of FOL formulas, and φ, ψ FOL formulas. Then, the following two are equivalent:

(i) $\Sigma \vdash \varphi \rightarrow \psi$

(ii) $\Sigma \cup \{\varphi\} \vdash \psi$ $\lhd$

Deduction theorem is a very useful auxiliary theorem, but even in itself is an interesting statement. Now we do not prove it, but for the inquisitive reader we suggest consulting [10].

Theorem 4.1.5. (Compactness theorem) Let Σ be a set of FOL formulas and φ be a formula. Then the two statements below are equivalent:

(1) $\Sigma \models \varphi$

(2) there is a finite $\Sigma_0 \subseteq \Sigma$, such that $\Sigma_0 \models \varphi$ $\lhd$

In words, this theorem states, that a formula is true in a set of formulas if and only if it is true in a finite subset of the set of formulas.

In the proof we rely on Gödel's completeness theorem, which is a fundamental statement in mathematical logic that establishes a correspondence between semantic truth and syntactic provability in First-order logic. Beside much of Kurt Gödel's invaluable work, this theorem and a proof for it can be found in [6].

[6]We recall that we say that a variable is free in φ if it occurs outside of the scope of any quantifier (e.g. $\exists v$ or $\forall v$) occurring in φ.

Proof. (1) ⇒ (2) Assume that $\Sigma \models \varphi$. By the completeness theorem we know that $\Sigma \vdash \varphi$. Let P be a deduction referring to this.

Let $\Sigma_0 = \Sigma \cap P$. Σ_0 is finite, and P guarantees that $\Sigma_0 \vdash \varphi$. Applying the completeness theorem again, we get that $\Sigma_0 \models \varphi$.

(2) ⇒ (1) This way is evident. *Q.E.D.*

Theorem 4.1.6. (Craig's interpolation theorem) Let φ, ψ be FOL formulas and assume that $\varphi \models \psi$. Then, in the common language of φ and ψ there is an interpoland ξ for which $\varphi \models \xi$ and $\xi \models \psi$. ◁

In [4, Theorem 1.3], we can read a proof for this theorem, and though in this proof the authors used higher level mathematics, later they show a more elementary proof, as well.

Remark 4.1.7. Craig's interpolation theorem is true for FOL without equality, but in this case, we need an extra formula $\bot = \bigwedge \emptyset$ (or sometimes it is called $False$, it is the dual formula of $True$, that we already used before) to the language. ◁

4.2 Examining definability

We made our preparations, and now we can go on to examining FOL languages from the aspect of definability, and of what properties do certain kinds of definitions have.

Theorem 4.2.1. (weak Beth definability theorem) Let $\mathcal{L}$ be a FOL language, let Th be a theory in $\mathcal{L}$ and let R ($\notin \mathcal{L}$) be a relation symbol. Every implicit definition of R is equivalent to an explicit definition of it (modulo Th). ◁

Proof. Assume that R is an n-place relation symbol and $\Sigma(R)$ defines R in Th. We know that

$$Th \cup \Sigma(R) \cup \Sigma(R') \models \forall \overline{x}\big(R(\overline{x}) \leftrightarrow R(\overline{x})\big),$$

where $\overline{x} = (x_0, \dots, x_{n-1})$, and R' is an n-place relation symbol, different from R. For any $\overline{c} = (c_0, \dots, c_{n-1})$, where $c_0, \dots, c_{n-1}$ are new constant symbols, we have

$$Th \cup \Sigma(R) \cup \Sigma(R') \models R(\overline{c}) \leftrightarrow R(\overline{c}),$$

By theorem 4.1.5. (the compactness theorem) we know that there are formulas $\xi \in \mathcal{L}$ and $\sigma(R)$ such that:

- $Th \models \xi$
- $\Sigma(R) \models \sigma(R)$
- $\xi \wedge \sigma(R) \wedge \sigma(R') \models R(\overline{c}) \leftrightarrow R'(\overline{c})$

If we separate R and R' to the two sides of $\models$ (using deduction theorem), we get

$$\xi \wedge \sigma(R) \wedge R(\overline{c}) \models \sigma(R') \to R'(\overline{c}) \tag{4.1}$$

Now we can use theorem 4.1.6 (Craig's interpolation theorem) to get an interpoland $\varphi(\overline{c})$ in the common language of $\mathcal{L}$ and the constant symbols (that is $\mathcal{L}$ expanded with the constants) such that

$$\xi \wedge \sigma(R) \wedge R(\overline{c}) \models \varphi(\overline{c}) \quad \text{and} \quad \varphi(\overline{c}) \models \sigma(R') \to R'(\overline{c}) \tag{4.2}$$

R' does not occur in φ, thus the second part of (4.2) is equivalent to

$$\varphi(\overline{c}) \models \sigma(R) \rightarrow R(\overline{c}) \tag{4.3}$$

Using the deduction theorem again on both (4.2) and (4.3), we get

$$\xi \wedge \sigma(R) \models R(\overline{c}) \leftrightarrow \varphi(\overline{c}) \tag{4.4}$$

Since $c_0, \ldots, c_{n-1}$ were arbitrary constant symbols, we know now that

$$\xi \wedge \sigma(R) \models \forall \overline{x}\big(R(\overline{x}) \leftrightarrow \varphi(\overline{x})\big). \tag{4.5}$$

By the properties of ξ and σ we have

$$Th \cup \Sigma(R) \models \forall \overline{x}\big(R(\overline{x}) \leftrightarrow \varphi(\overline{x})\big). \tag{4.6}$$

Using that in each model of Th there is a relation that satisfies Σ, (4.6) implies $Th \models \Sigma(R \backslash \varphi)$, thus

$$Th \cup \big\{\forall \overline{x}\big(R(\overline{x}) \leftrightarrow \varphi(\overline{x})\big)\big\} \models \Sigma(R). \tag{4.7}$$

Now we have (4.6) and (4.7), which state that $\Sigma(R)$ and $\forall \overline{x}\big(R(\overline{x}) \leftrightarrow \varphi(\overline{x})\big)$ are equivalent modulo Th. This is exactly what we were looking for, since the latter is an explicit definition. *Q.E.D.*

Remark 4.2.2. In the proof above, we used the deduction theorem, not even only once. This might be a bit confusing, since the deduction theorem says that if a formula ψ is deducible from a set of assumptions $\Sigma \cup \{\varphi\}$, then the implication $\varphi \rightarrow \psi$ is deducible from Σ, and in this proof, we used the theorem for semantical truth, not deduction. We can do this, though, without any problem, because the completeness theorem states that in FOL, semantical truth and provability coincide. $\triangleleft$

As we would expect, if there is a 'weak' definability theorem, then there is another one, that has a stronger statement. To understand that, we need to know what 'weak definitions' are.

Definition 4.2.3. (Weak definitions) $\Sigma(R)$ *weakly implicitly* defines R in Th if in each model of Th there is no more than one relation satisfying Σ.

We say that $\Sigma(R)$ *weakly explicitly* defines R in Th if there is a formula φ in the language of Th, such that $Th \cup \Sigma(R) \models R \leftrightarrow \varphi$. $\triangleleft$

Theorem 4.2.4. (Strong Beth definability theorem) $\Sigma(R)$ is a weak implicit definition of R if and only if there is a weak explicit definition equivalent to it. $\triangleleft$

This theorem we usually state as the original version, so if we say Beth definability theorem, we mean this one.

Proof. The proof of this theorem is completely analogous to the proof of the weak Beth definability theorem. *Q.E.D.*

4.3 Undefinability of truth

Truth is one of the most important notions of our society. As we could see, it is a very important notion for logical systems, as well. Earlier in this thesis, we have defined some kinds of truth concepts (like $\Vdash_v$ in the case of sentential logic), and in these definitions, we always rely on some background language (e.g., using the word "and" in the definition of $\Vdash_v (\psi_1 \wedge \psi_2)$ in definition 2.1.1). This is called the metalanguage. As an outlook, in this section, we will discuss Tarski's theorem about the undefinability of truth using Tarski's book from the 50's, [14]. In words, this theorem states, that arithmetical truth can not be defined within arithmetic itself, which is a very strong statement that tells us about how important the metalanguage is and that a logic can never be built only using its language, we always need a metalanguage.

First we need to mention the formal definition of Convention T, which is Tarski's definition of the truth symbol "Tr" in metalanguage.

Definition 4.3.1. (Convention T). A symbol Tr formulated in metalanguage might be called an adequate definition of truth if it has the following consequences:

i) all sentences obtained from the expression "$x \in Tr$ iff p" by substituting a structural descriptive name of any sentence the language in question for the symbol "x", and the expression which forms the translation of this sentence into the metalanguage for the symbol "p".

ii) the sentence "for any x if $x \in Tr$ then $x \in X^*$". ◁

Now we can formulate Tarski's undefinability theorem.

Theorem 4.3.2. (Tarski's undefinability theorem)

1. In whatever way the symbol "Tr" denoting a class of expressions is defined in metatheory (background theory), it will be possible to derive the negation of one of the sentences, which were described in condition i) of Convention T, from it.

2. Assuming that the class of all provable sentences of the metatheory is consistent, it is impossible to construct an adequate definition of truth in the sense of Convention T on the basis of the metatheory. ◁

Remark 4.3.3. The formal language that Tarski used for investigating theory of truth was the language of general theory of classes in which we could express arithmetic, so we could say that the theorem relates to the language of arithmetic. ◁

4.4 Complexity of definitions

As mentioned before, although theorem 4.2.4 states, that in a FOL language, for every implicit definition there is an explicit equivalent to it (modulo theories), we still have very good use of implicit definitions. The theorem below tells us that an implicit definition can be much simpler than its explicit equivalent (if we measure simplicity by the number of variables used in the formulas).

Theorem 4.4.1. (No weak Beth property for $\mathcal{L}_n$) Let $n \geq 3$. There are a theory Th in the language of an n-place relation symbol R, a binary relation symbol s and a description $\Sigma(D)$ for a unary relation D, such that $\Sigma(D)$ is an implicit definition for D in Th, but there is no explicit definition for D in Th, that is in the language of Th for each n-variable formula φ we have

$$Th \cup \Sigma(D) \nvDash \forall v_0 \left(D(v_0) \leftrightarrow \varphi\right).$$ ◁

Remark 4.4.2. What does this theorem have to do with complexity of definitions? It states that while we can implicitly define a relation using only n variables, it is not sure that we can define the same relation explicitly, using at most n variables. ◁

4.5 Definitional extension

What makes two FOL languages different? Their vocabularies (similarity types). The vocabulary of a FOL language consists of the concepts we do not analyze further in the given language. We can refine or revise this choice of basic concepts by changing the language via the use of interpretations.

Recall from definition 1.1.4. that by a theory T in a language $\mathcal{L}$ we mean a set of sentences of the language, and by the set of its consequences we understand:

$$Csq_{\mathcal{L}}(T) \stackrel{\text{def}}{\Longleftrightarrow} \{\varphi \in \mathcal{L} : T \models \varphi\}$$

Definition 4.5.1. (Equivalent theories) We say that two theories T and T' in the same language are *equivalent* iff their consequences are the same set, in symbols:

$$T \equiv T' \stackrel{\text{def}}{\Longleftrightarrow} Csq_{\mathcal{L}}(T) = Csq_{\mathcal{L}}(T'). \qquad \triangleleft$$

What is a definitional extension of a language? What do we gain if using the extension? Technically, the definitional extension of a language is the language itself, together with some new relation symbols and explicit definitions of these. We introduce notations for ease of talk, thus while using the extended language, we can express ourselves in much shorter sentences.

Definition 4.5.2. (Definitional extension of a language) Let $\Sigma(\underline{R})$ be a set of explicit definitions for a sequence of relation symbols not in $\mathcal{L}(T)$ (where T is a theory of $\mathcal{L}$). We say that T' is a *definitional extension* of T (in symbols: $T \stackrel{\triangle}{\longrightarrow} T'$) if T' is equivalent to $T \cup \Sigma(\underline{R})$. $\triangleleft$

Remark 4.5.3. We always assume that only one definition is given for a specific relation symbol in Σ. $\triangleleft$

What is the connection between the definitional extension and the original language? We can consider the new sentences as abbreviations of old formulas. Now we define a translation function tr from the extended language to the original one that shows us that after having introduced the new relation symbols, we can leave them out of the language any time we want, without losing any capability of expressing. The idea is that we replace $R(v_0, \ldots, v_n)$ with its explicit definition, we replace the same atomic formula but with a different sequence of variables $R(x_0, \ldots, x_n)$ by the corresponding version of φ_R we get by using Tarski's substitution of variables, and otherwise we leave the formulas untouched.

Definition 4.5.4. (Translation function)

- $tr\big(R(v_0, \ldots, v_n)\big) = \varphi_R$ if $R(v_0, \ldots, v_n) \leftrightarrow \varphi_R$ is in Σ,

$tr\big(R(x_0, \ldots, x_n)\big)$ is the appropriate substituted version of φ_R

- $tr\big(S(x_1, \ldots, x_n)\big) = S(x_1, \ldots, x_n)$ if $S(x_1, \ldots, x_n) \in \mathcal{L}$,
- $tr(v_i = v_j) = (v_i = v_j)$,
- $tr(\neg\varphi) = \neg tr(\varphi)$, $tr(\varphi \wedge \psi) = tr(\varphi) \wedge tr(\psi)$, $tr(\exists v_i \varphi) = \exists v_i tr(\varphi)$. $\triangleleft$

Theorem 4.5.5. $\Sigma \models \varphi \leftrightarrow tr(\varphi)$ for every formula φ in the extended language and $tr(\psi) = \psi$ for every formula ψ in the unextended language. $\square$

Example 4.5.6. The language of ZFC set theory contains one binary relation symbol, the "elementhood" relation $\in$. When working in ZFC, we use many explicitly defined concepts, like we use $A \subseteq B$ instead of saying the longer formula $\forall z\,(z \in A \longrightarrow z \in B)$, or we use $\emptyset$ for the unique set that has no elements at all. These abbreviations make our job much easier. $\triangleleft$

4.6 Definitional equivalence

Definitionally equivalent theories have the same content, but different outfit, that is different symbols. They might even look very unlike sometimes. Technically, definitional equivalence is the symmetric and transitive closure of the notion of definitional extension, thus it preserves all the properties that definitional extension does preserve.

Definition 4.6.1. (Definitional equivalence) We say that two theories are *definitionally equivalent* (in symbols: $T \stackrel{\triangle}{\equiv} T'$) if there is a sequence $T_1, \ldots, T_n$ of theories, such that $T \equiv T_1$, $T' \equiv T_n$, and for all $1 \leq i < n$ either $T_i \stackrel{\triangle}{\longrightarrow} T_{i+1}$ or $T_{i+1} \stackrel{\triangle}{\longrightarrow} T_i$. ◁

Example 4.6.2. If we rename the relation symbols occurring in T to completely new symbols, we get a theory T' that is definitionally equivalent to T. It is easy to see that $T'' = T \cup T'$ is a definitional extension of both T and T'. ◁

There is a theorem that tells us about the length of the chain of theories leading from T to T'. It states that there always is a 5-long chain of definitional extensions between the definitionally equivalent theories. We will need to know that we call a function f from a language to another *structural* if f is the translation function associated to an explicit definition as in definition 4.5.4. Let $Mod(T)$ denote the class of all models of T.

Theorem 4.6.3. (Characterizations of definitional equivalence) Assume that T and T' have disjoint vocabularies. The following four are equivalent.

i) $T \stackrel{\triangle}{\equiv} T'$.

ii) T and T' have a common definitional extension, that is, there is a theory T'' such that

$$T \stackrel{\triangle}{\longrightarrow} T'' \stackrel{\triangle}{\longleftarrow} T'.$$

iii) There are structural translation functions $tr : \mathcal{L}(T) \to \mathcal{L}(T')$ and $tr' : \mathcal{L}(T') \to \mathcal{L}(T)$ that are inverses of each other with respect to T and T'. In symbols, for all $\varphi \in \mathcal{L}$ and $\psi \in \mathcal{L}'$ we have

$$T \models \varphi \leftrightarrow tr'\big(tr(\varphi)\big) \qquad \text{and} \qquad T' \models \psi \leftrightarrow tr\big(tr'(\psi)\big).$$

iv) There is a bijection β between $Mod(T)$ and $Mod(T')$ that is defined along the explicit definitions Σ and Σ' as: if $\mathfrak{M} \models T$ and $\mathfrak{M}' \models \beta(\mathfrak{M})$, then the universes of $\mathfrak{M}$ and $\mathfrak{M}'$ are the same, the relations in $\mathfrak{M}'$ are the ones defined in $\mathfrak{M}$ according to Σ and the other way round, the relations in $\mathfrak{M}$ are the ones defined in $\mathfrak{M}'$ according to Σ'. ◁

Chapter 5

Definability in general logics

In the second half of the 20$^{\text{th}}$ century, researchers of the topic realized, that while examining some properties of logical systems, certain patterns of proofs and ideas show up in the case of different systems, with only slight differences. Several schools (like Abstract Model Theory, Institutions Theory, Universal Logic) have formed to develop appropriate abstract levels of logic. Some of these benefited from using universal algebraic methods.

Naturally, while investigating logics, the topic of definability comes up, as well. In this chapter, we are going to study the basics of the algebraization of logics and then we will see how certain definability properties can be shown throughout the algebraic counterparts of logics. This chapter is based on the work of Ildikó Sain ([13]).

5.1 General and algebraizable logics

First, recall from section 3., that we call a logic strongly nice, iff it is compositional, has the filter property and has both (semantical and syntactical) substitution properties.

Definition 5.1.1. (Algebraizable semantical logic, structural logic) Let $\mathcal{L} = \langle F, M, mng, \models\rangle$ be a logic. We say that $\mathcal{L}$ is an *algebraizable semantical logic* iff $\mathcal{L}$ is strongly nice. We say that $\mathcal{L}$ is *structural* iff it is compositional and has the semantical substitution property. ◁

As mentioned in remark 2.1.2, the set P of atomic formulas is in most cases a parameter in the definition of a logic, thus whenever we write $\mathcal{L}^P$, we mean to make this dependence explicit.

Definition 5.1.2. (General logic) A *general logic* is defined to be a function $\mathbf{L} \stackrel{\text{def}}{=} \langle \mathcal{L}^P : P \text{ is a set}\rangle$, where for each set P, $\mathcal{L}^P = \langle F^P, \vdash^P, M^P, mng^P, \models^P\rangle$ is a logic in the sense of remark 1.1.3. We sometimes refer to the elements of M^P as *P-models.* ◁

All the properties of logics we introduced earlier extend to general logic, in a natural way, that is $\mathbf{L}$ has a logic property iff all its parameterized parts $\mathcal{L}^P$ have the property in the old sense. Though now we will not give a proof for this, it can be found in [3].

When investigating the property of having connectives and the filter property, it is the case that all the logics in $\mathbf{L}$ share the same set of connectives (Cn) and derived connectives ($\epsilon_i, \delta_i, \Delta_j$, respectively). In the case of substitution properties, different parameters P and Q might appear, as follows.

- $\mathbf{L}$ *has the syntactical substitution property* iff $\models^P \varphi$ implies $\models^Q \varphi\big(\overline{p}/s(\overline{p})\big)$ for all P, Q; $s : P \longrightarrow F^Q$, and $\varphi \in F^P$.

- **L** has the *semantical substitution property* iff for all sets P, Q; $s : P \longrightarrow F^Q$ and $\mathfrak{M} \in M^Q$ there exists $\mathfrak{N} \in M^P$ such that $mng^Q_{\mathfrak{M}} \circ \hat{s} = mng^P_{\mathfrak{N}}$.

Definition 5.1.3. (Algebraizable logic, structural logic) We say that **L** is an *algebraizable (general) logic* iff **L** is strongly nice (in the sense of a general logic). **L** is called *structural* iff it is compositional and has the semantical substitution property. ◁

Remark 5.1.4. A general logic **L** is algebraizable iff $\mathcal{L}^P$ is an algebraizable semantical logic for each P, the logical connectives and the derived connectives (for the filter property) are the same for each P, and

$$P \subseteq Q \Longrightarrow \{mng^P_{\mathfrak{M}} : \mathfrak{M} \in M^P\} = \{mng^Q_{\mathfrak{M}}\restriction_{F^P} : \mathfrak{M} \in M^Q\}$$

holds for each P, Q. Intuitively, this condition says that $\mathcal{L}^P$ is the natural restriction of $\mathcal{L}^Q$. ◁

5.2 Algebraization of a logic

Now we turn to introducing the algebraic method we intend to use for studying definability properties. We note that besides definability properties, this machinery is appropriate for examining many other logical purposes.

Definition 5.2.1. (Meaning algebra, Alg_m, Alg) Let $\mathcal{L} = \langle F, M, mng, \models\rangle$ be a compositional logic.

i) First we turn every model into an algebra. Compositionality of $mng_{\mathfrak{M}}$ ensures that we can easily define an algebra of type Cn on the set $\{mng_{\mathfrak{M}}(\varphi) : \varphi \in F\}$ of meanings. This algebra is called the *meaning algebra* of $\mathfrak{M}$ and it is denoted by $\mathfrak{Mng}(\mathfrak{M})$.

   In more detail, for any logical connective c of arity k we define a k-ary function $c^{\mathfrak{M}}$ on the meanings of the formulas in $\mathfrak{M}$, by setting for all formulas $\varphi_1, \dots, \varphi_k$

   $$c^{\mathfrak{M}}\big(mng_{\mathfrak{M}}(\varphi_1), \dots, mng_{\mathfrak{M}}(\varphi_k)\big) \stackrel{\text{def}}{=} mng_{\mathfrak{M}}\big(c(\varphi_1, \dots, \varphi_k)\big), \text{ and}$$
   $$\mathfrak{Mng}(\mathfrak{M}) \stackrel{\text{def}}{=} \langle\{mng_{\mathfrak{M}}(\varphi) : \varphi \in F\}, c^{\mathfrak{M}}\rangle_{c\in Cn}.$$

ii) $\mathsf{Alg}_m(\mathcal{L})$ denotes the *class of all meaning algebras* of $\mathcal{L}$, that is

   $$\mathsf{Alg}_m(\mathcal{L}) \stackrel{\text{def}}{=} \{mng_{\mathfrak{M}}(\mathfrak{F}) : \mathfrak{M} \in M\} = \{\mathfrak{Mng}(\mathfrak{M}) : \mathfrak{M} \in M\}.$$

iii) Let $K \subseteq M$. For every $\varphi, \psi \in F$ we define

   $$\varphi \sim_K \varphi \stackrel{\text{def}}{\Longleftrightarrow} (\forall \mathfrak{M} \in K)\; mng_{\mathfrak{M}}(\varphi) = mng_{\mathfrak{M}}(\psi).$$

   Then $\sim_K$ is an equivalence relation and is a congruence on $\mathfrak{F}$ (since $\mathcal{L}$ is compositional). $\mathfrak{F}/\sim_K$ denotes the factor algebra of $\mathfrak{F}$. factorized by $\sim_K$. This is called the *(semantical) Lindenbaum-Tarski algebra* of K. $\mathsf{Alg}(\mathcal{L})$ is the class of isomorphic copies of the Lindenbaum-Tarski algebras of $\mathcal{L}$, that is:

   $$\mathsf{Alg}(\mathcal{L}) \stackrel{\text{def}}{=} \mathbf{I}\{\mathfrak{F}/\sim_K : K \subseteq M\}.$$

iv) Let $\mathbf{L} = \langle \mathcal{L}^P : P \text{ is a set}\rangle$ be a general logic. Then

   $$\mathsf{Alg}_m(\mathbf{L}) \stackrel{\text{def}}{=} \bigcup\{\mathsf{Alg}_m(\mathcal{L}^P) : P \text{ is a set}\}, \text{ and}$$

   $$\mathsf{Alg}(\mathbf{L}) \stackrel{\text{def}}{=} \bigcup\{\mathsf{Alg}(\mathcal{L}^P) : P \text{ is a set}\}.$$ ◁

Remark 5.2.2. a) The meaning algebra is the image of the formula algebra $\mathfrak{F} = \langle F, Cn\rangle$ under the meaning function, in symbols: $\mathfrak{Mng}(\mathfrak{M}) = mng_{\mathfrak{M}}(\mathfrak{F})$.
b) Congruence means here, that besides $\sim_K$ is an equivalence relation, for every c k-ary connective (of the formula algebra) if $\varphi_1 \sim_K \psi_1, \ldots, \varphi_k \sim_K \psi_k$, then $c(\varphi_1, \ldots, \varphi_k) = c(\psi_1, \ldots, \psi_k)$ (that is, it formulates "classes" of the equivalent formulas, and each element can represent the "class" it belongs to).
c) It is important that $\mathsf{Alg}_m(\mathcal{L})$ is not an abstract class in the sense that it is not closed under isomorphisms. The reason for this is that we need these algebras in their original state, substituting them with their isomorphic copies would lead to loss of information. $\triangleleft$

Notation:

- For a logic $\mathcal{L} = \langle F_{\mathcal{L}}, M_{\mathcal{L}}, mng_{\mathcal{L}}, \models_{\mathcal{L}}\rangle$, we let $Mng_{\mathcal{L}} \stackrel{\text{def}}{=} \{mng_{\mathfrak{M}} : \mathfrak{M} \in M_{\mathcal{L}}\}$, and similarly we let $Mng^P \stackrel{\text{def}}{=} Mng_{\mathcal{L}^P}$ if $\mathbf{L} = \langle \mathcal{L}^P : P \text{ is a set}\rangle$.
- If $\mathfrak{A}$ is an algebra and K is a class of algebras, then $Hom(\mathfrak{A}, \mathsf{K})$ denotes the class of all homomorphisms $h : \mathfrak{A} \longrightarrow \mathfrak{B}$ with $\mathfrak{B} \in \mathsf{K}$.
- $\mathfrak{F}^P$ denotes the formula algebra of parameter P. $\triangleleft$

Theorem 5.2.3. (Characterization of structural logics) Let $\mathcal{L}$ be a compositional logic and $\mathbf{L}$ be a compositional general logic. Then i)-ii) below hold.

i) $\mathcal{L}$ has the semantical substitution property iff $Mng_{\mathcal{L}} = Hom\big(\mathfrak{F}, \mathsf{Alg}_m(\mathcal{L})\big)$.

ii) $\mathbf{L}$ has the semantical substitution property iff $Mng^P = Hom\big(\mathfrak{F}^P, \mathsf{Alg}_m(\mathbf{L})\big)$, for all P. $\triangleleft$

In this thesis we concentrate on the semantical aspects of a logic, but it is important to mention that the syntactical part can also be algebraized. As we defined, both $\mathsf{Alg}_m(\mathbf{L})$ and $\mathsf{Alg}(\mathbf{L})$ correspond to the semantical part of a logic $\mathbf{L}$. From the definition of $\mathsf{Alg}(\mathbf{L})$ we can think of a natural generalization that applies to the syntactical part, as well. $\mathsf{Alg}(\langle F, \vdash\rangle)$ is the class of "syntactical Lindenbaum-Tarski algebras of $\langle F, \vdash\rangle$. While defining $\mathsf{Alg}(\langle F, \vdash\rangle)$, instead of the semantical congruence relation $\sim_K$, we we factor $\mathfrak{F}$ by the syntactical interderivability relation $\equiv_T \stackrel{\text{def}}{=} \{\langle \varphi, \psi\rangle : T, \varphi \vdash \psi \text{ and } T, \psi \vdash \varphi\}$ for all $T \subseteq F$. In this case it is a problem that $\equiv_T$ might not be a congruence relation, but this problem can be solved, moreover, there are (at least) two different (and equivalently good) ways for this. The first one insists on using $\vdash$ only as "local" derivation system, and the other way is using the largest congruence relation $\sim_T \subseteq \equiv_T$ contained in the equivalence relation $\equiv_T$. For more details about this topic see [3], [8], [7], and [5].

5.3 Definability

We would like to use the algebraic characterization of the logics to examine some definability properties. Although we have already defined some of the following notions, now we form the definitions so that they fit into the algebraic framework we have defined.

Definition 5.3.1. (Implicit and explicit definitions) Let $\mathbf{L} = \langle \mathcal{L}^P : P \text{ is a set}\rangle$ be a general logic, let $P \subseteq Q$ such that $F^P \neq \emptyset$, and let $R \stackrel{\text{def}}{=} Q \setminus P$.

i) The set Σ of formulas *defines R implicitly in Q* iff any P-model can be extended to a Q-model of Σ at most one way, that is iff

$$\big(\forall \mathfrak{M}, \mathfrak{N} \in Mod^Q(\Sigma)\big)\ \big(mng^Q_{\mathfrak{M}}\big|_{F^P} = mng^Q_{\mathfrak{N}}\big|_{F^P} \Longrightarrow mng^Q_{\mathfrak{M}} = mng^Q_{\mathfrak{N}}\big).$$

ii) Σ *defines R implicitly in Q in the strong sense* iff any P-model that can be extended to a Q-model of Σ can indeed be extended, and this extension is unique, that is iff Σ defines R implicitly in Q and in addition we have

$$\big(\forall \mathfrak{M} \in Mod^P(Th^Q Mod^Q(\Sigma) \cap F^P)\big)\ \big(\exists \mathfrak{N} \in Mod^Q(\Sigma)\big)\ mng^Q_{\mathfrak{N}}\big|_{F^P} = mng^P_{\mathfrak{M}}.$$

iii) Σ *defines R explicitly in Q* iff any element of R has an explicit definition (in the "old" sense) that works in all models of Σ, that is iff

$$(\forall r \in R)(\exists \varphi_r \in F^P)\,(\forall \mathfrak{M} \in Mod^Q(\Sigma))\ mng^Q_{\mathfrak{M}}(r) = mng^Q_{\mathfrak{M}}(\varphi_r).$$

iv) Σ *defines R local-explicitly in Q* iff the definition φ_r in iii) can vary from model to model, that is iff

$$(\forall r \in R)\,(\forall \mathfrak{M} \in Mod^Q(\Sigma))\,(\exists \varphi_r \in F^P)\ mng^Q_{\mathfrak{M}}(r) = mng^Q_{\mathfrak{M}}(\varphi_r). \qquad \triangleleft$$

Definition 5.3.2. (Beth definability properties) Let $\mathbf{L}$ be a general logic.

- $\mathbf{L}$ *has the weak Beth definability property* iff for all P, Q, R and Σ as in definition 5.3.1. above, we have that

$$\Sigma \text{ defines } R \text{ implicitly in } Q \text{ in the strong sense } \Longrightarrow \Sigma \text{ defines } R \text{ explicitly in } Q.$$

- $\mathbf{L}$ *has the (strong) Beth definability property* iff for all P, Q, R and Σ as in definition 5.3.1. above, we have that

$$\Sigma \text{ defines } R \text{ implicitly in } Q \Longrightarrow \Sigma \text{ defines } R \text{ explicitly in } Q.$$

- $\mathbf{L}$ *has the local Beth definability property* iff for all P, Q, R and Σ as in definition 5.3.1. above, we have that

$$\Sigma \text{ defines } R \text{ implicitly in } Q \Longrightarrow \Sigma \text{ defines } R \text{ local-explicitly in } Q. \qquad \triangleleft$$

Remark 5.3.3. Though now we only focus on the "infinite" Beth property, it is still important to note that a natural version of Beth definability property is obtained from this definition by requiring $|R| < \infty$. We say that this is the *finite Beth definability property.* $\triangleleft$

We now introduce the so called patchwork property, which is in words the property a general logic has iff any two "compatible" models have a common extension.

Definition 5.3.4. (Patchwork property) Let $\mathbf{L}$ be a general logic. $\mathbf{L}$ *has the patchwork property* iff for all sets P, Q and models $\mathfrak{M} \in M^P$, $\mathfrak{N} \in M^Q$ we have

$$\big(F^{P\cap Q} \neq \emptyset \text{ and } mng^{P\cap Q}_{\mathfrak{M}} = mng^{P\cap Q}_{\mathfrak{N}}\big) \Longrightarrow \big(\exists \mathfrak{B} \in M^{P\cap Q}\big)\ \big(mng^P_{\mathfrak{B}} = mng^P_{\mathfrak{M}} \text{ and } mng^Q_{\mathfrak{B}} = mng^Q_{\mathfrak{N}}\big). \quad \triangleleft$$

Before we state the characterization theorem of the Beth definability properties, we recall some notions that are essential for understanding the theorem itself. If K is a class of algebras, then by a *morphism* of K we mean a triple $\langle \mathfrak{A}, h, \mathfrak{B}\rangle$, where $\mathfrak{A}, \mathfrak{B} \in \mathsf{K}$ and $h : \mathfrak{A} \longrightarrow \mathfrak{B}$ is a homomorphism. A morphism $\langle \mathfrak{A}, h, \mathfrak{B}\rangle$ is an *epimorphism* of K iff for every $\mathfrak{C} \in \mathsf{K}$ and every pair of functions $f, k : \mathfrak{B} \longrightarrow \mathfrak{C}$ of homomorphisms we have that $(f \circ h = k \circ h) \Longrightarrow f = k$.

Let $\mathsf{K}_0 \subseteq \mathsf{K}$ be two classes of algebras and let $\langle \mathfrak{A}, h, \mathfrak{B}\rangle$ be a morphism of K. We say that h is K_0-*extensible* iff for every algebra $\mathfrak{C} \in \mathsf{K}_0$ and every surjective homomorphism $f : \mathfrak{A} \longrightarrow \mathfrak{C}$ there exists some $\mathfrak{N} \in \mathsf{K}_0$ and $g : \mathfrak{B} \longrightarrow \mathfrak{N}$ such that $\mathfrak{C} \subseteq \mathfrak{N}$ and $g \circ h = f$.

Remark 5.3.5. Typical examples of epimorphisms are surjections, though there are epimorphisms that are not surjective.

It is important to mention that in the case of K_0-extensibility $\mathfrak{C}$ is a concrete subalgebra of $\mathfrak{N}$ and not only is embeddable into $\mathfrak{N}$. ◁

Theorem 5.3.6. (Characterization of Beth definability property) Let **L** be an algebraizable general logic with the patchwork property.

i) **L** has the weak Beth definability property iff every $\mathsf{Alg}_m(\mathbf{L})$-extensible epimorphism of $\mathsf{Alg}(\mathbf{L})$ is surjective.

ii) **L** has the (strong) Beth definability property iff all the epimorphisms of $\mathsf{Alg}(\mathbf{L})$ are surjective.

iii) **L** has the local Beth definability property iff all the epimorphisms of $\mathsf{Alg}_m(\mathbf{L})$ are surjective. ◁

Remark 5.3.7. It is important that in theorem 5.3.6. above, $\mathsf{Alg}_m(\mathbf{L})$ is not an abstract class in the sense that it is not closed under isomorphism, since the definition of K-extensibility strongly differentiates isomorphic algebras:

$$\mathfrak{M} \cong \mathfrak{N} \;\underset{\not\Leftarrow}{\Rightarrow}\; \mathfrak{Mng}(\mathfrak{M}) \cong \mathfrak{Mng}(\mathfrak{N}).$$ ◁

Besides that this characterization theorem is interesting in itself, it has spectacular applications, as well. For example, in [9] Judit Madarász gave a concrete characterization of those varieties of cylindric algebras (and their reducts) in which epimorphisms are surjective. This, according to theorem 5.3.6. ii), gives an intrinsic characterization of those logics **L** which have the Beth definability property (under some conditions on **L**).

References

[1] H. Andréka, I. Németi, and I. Sain. A complete logic for reasoning about programs via nonstandard model theory, Parts I–II. *Theoretical Computer Science*, 17(2 and 3):193–212 and 259–278, 1982.

[2] H. Andréka, I. Németi, and I. Sain. *Universal Algebraic Logic (Dedicated to the Unity of Science)*. Birkhäuser, 1998. Manuscript.

[3] H. Andréka, I. Németi, and I. Sain. Algebraic Logic. In D. M. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic Vol. II, Second edition*, pages 133–247. Kluwer Academic Publishers, 2001. Preprint form available from the home page of the Rényi Institute.

[4] H. Andréka and Németi I. Definability theory course notes. Department of Logic, Eötvös Loránd Tudományegyetem, 2014. Also available in`www.math-inst.hu/pub/algebraic-logic/DefThNotes0828.pdf`.

[5] J. M. Font and R. Jansana. On the Sentential Logics associated with Strongly Nice and Semi-nice Logics. *Bulletin of the IGPL*, 2:55–76, 1994.

[6] Kurt Gödel. Kurt Gödel: Collected works: Volume i: Publications 1929-1936. volume 1. Oxford university press, 1986.

[7] L. Henkin, J. D. Monk, and A. Tarski. *Cylindric Algebras Parts I and II.* North–Holland, Amsterdam, 1971 and 1985.

[8] E. Hoogland. *Definability and Interpolation (Model theoretic investigations)*. PhD thesis, Institute for Logic, Language and Computation, University of Amsterdam, 2001. ILLC Dissertation Series DS-2001-05, xii+209 pp.

[9] J. X. Madarász. Surjectivity of epimorphisms in varieties of algebraic logic. Technical report, Rényi Institute of Mathematics, Budapest, 2007.

[10] Helena Rasiowa and Roman Sikorski. *The mathematics of metamathematics.* Pań stwowe Wydawn. Naukowe, 1963.

[11] I. Sain. There are general rules for specifying semantics: Observations on Abstract Model Theory. *CL&CL (Computational Linguistics and Computer Languages)*, XIII:195–250, 1979.

[12] I. Sain. *Dynamic logic with non-standard model theory*, 1986. Dissertation for candidate's degree, Hungarian Academy of Sciences, Budapest, In Hungarian.

[13] I. Sain. Definability issues in universal logic. In I. Németi H. Andréka and M. Ferenczi, editors, *Cylindric-like Algebras and Algebraic Logic*, Bolyai Society Mathematical Studies 22., pages 393–419. Springer–Verlag, Berlin, 2012.

[14] A. Tarski. The concept of truth in formalized languages. In J. Corcoran, editor, *Logics, semantics and mathematics*, pages 152–279. Clarendon Press, 1956.