Dependence Logic: Investigations into Higher-Order Semantics Defined on Teams

Ville Nurmi
Department of Mathematics and Statistics
Faculty of Science
University of Helsinki

Academic dissertation

To be presented, with the permission of the Faculty of Science of the University of Helsinki, for public criticism in Hall 4, Metsätalo (Unioninkatu 40), on August 22nd, 2009, at 10 a.m.

http://ethesis.helsinki.fi
ISBN 978-952-92-5944-1 (paperback)
ISBN 978-952-10-5679-6 (PDF)
Helsinki University Print
Helsinki 2009

Contents

Acknowledgements

I would like to thank Wilfrid Hodges, Alessio Guglielmi, Kai Brünnler and Tero Tulenheimo for helpful correspondence.

Wilfrid Hodges gave me inspiration and told me about type shifting. This led me to 1-semantics. Alessio Guglielmi and his students found the beautiful proof formalism called calculus of structures and have done a good job in making it accessible also to non-proof theorists.

I wish to express my thanks to the organisers and participants of ESSLLI, the annual European Summer School in Logic, Language and Information, and especially its Student Session. ESSLLI is a good place to meet both students and senior people in these disciplines.

I thank my advisor, Jouko Väänänen, for introducing me to dependence logic and independence friendly logic, for sharing ideas about them and for giving helpful comments on a preliminary version of this thesis. I also thank Ryan Siders, Juha Kontinen and other colleagues at the math department for interesting discussions. Thanks to Marcus Frösén for showing what mathematics is really about. Finally, I thank my friends, family and especially Цвете for support and encouragement. Each of you has influenced me, and this thesis is one of the outcomes.

The work behind this thesis was made possible by funding from the University of Helsinki Department of Mathematics and Statistics and the MALJA Graduate School in Mathematical Logic.

Chapter 1

Introduction

This thesis is a study of a rather new logic called dependence logic. In this chapter I will briefly review dependence logic in a historical context and then proceed to the goals and outline of the thesis.

1.1 History

Dependence logic is best introduced in context of its history; it is the latest entry in the following historical timeline of logics that relate to partially ordered quantification.

1959, Henkin quantifiers [10] Also known as partially ordered quantifers, Henkin quantifiers allow partial ordering of first order quantifiers. The classical example is

$$\begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} \phi(x, y, z, w). \tag{1.1}$$

Its semantics is defined by resorting to second order quantifiers:

$$\exists f_y \exists f_w \forall x \forall z \phi(x, f_y x, z, f_w z).$$

In a sense, each row of quantifiers in (1.1) is read like in first order logic, and different rows of quantifiers work independently of each other.

1987, Independence friendly logic with game theoretic semantics [13, 12] The syntax of independence friendly logic differs from Henkin quantifiers so that first order quantifiers are always written linearly on one line. The partial order of quantifiers is specified with optional slashes after quantifiers. After the slash one lists the quantifiers that in Henkin's syntax would appear on different rows. An equivalent formula to (1.1) is

$$\forall x \exists y \forall z \exists w / \forall x \phi(x, y, z, w). \tag{1.2}$$

Its semantics is defined by resorting to game theoretic semantics such that when choosing a value for w, a player does not know the value of x. The semantics is defined only for sentences.

1997, Independence friendly logic with trump semantics [14, 15] Trump semantics expresses game theoretic semantics in an algebraic way. It is the return back to the Tarskian definition of satisfaction from which game theoretic semantics originally stepped away. Satisfaction of a formula is defined with respect to a model and a *set* of assignments, as opposed to a model and a single assignment as is the case in all conventional logics. It was later shown by a counting argument that it is not possible to define Tarskian semantics for independence friendly logic using only single assignments [5]. Trump semantics applied to the first quantifier in (1.2) simulates the semantic game by extending each assignment in the set by all possible values for x. This imitates the way truth is defined in game theoretic semantics—the other player who chooses values for the existential quantifiers must have a winning strategy and thus we must check that strategy against all possible moves of the opponent. Trump semantics applied to the fourth, slashed quantifier in (1.2) extends each assignment in the set by one value for w. This value is thought to be given by the winning strategy we are testing. Furthermore, to simulate the requirement of not knowing the value of x, the way the values are chosen must be *uniform*, i.e. the choices must be describable as a function on the set of assignments that does not use the values of x. Trump semantics is defined for all formulas.

2007, Dependence logic and team logic [19] Dependence logic derives from trump semantics by changing two things. Firstly, it replaces "independence" by "dependence", that is, it moves from a negative expression to a positive expression. Secondly, it introduces a new form of atomic formulas that is dedicated to expressing dependence. Consequently, the need of adding slashes to quantifiers disappears. Henkin's example formula (1.1) can be equivalently expressed in dependence logic as

$$\forall x \exists y \forall z \exists w \big(=\!(z, w) \land \phi(x, y, z, w)\big). \tag{1.3}$$

The semantics are based on sets of assignments, as in trump semantics. Such sets are now called *teams*. As with trump semantics, the first quantifier in

(1.3) is interpreted by extending all assignments in the team by all possible values for x. The fourth quantifier is interpreted by extending each assignment by some value for w, without demands for uniformity. The check for uniformity happens not until the new atomic formula, $=(z, w)$, which holds for a team only if the team "contains" a function from the values of z to the values of w.

Team logic is obtained from dependence logic as its closure under classical negation.

All the logics in this timeline are in a way very similar. Each of them is able to express the existence of a function, be it either as a winning strategy of a player of the semantic game, or as dressed in a uniformity condition, or as isolated to a new kind of atomic formula. Furthermore, all the logics are able to express that the function has a restricted arity in a sense; the function is allowed to use the values of only certain previously quantified variables. This is what one can do in existential second order logic. Indeed, given a sentence in any of the logics, one can translate it into an equivalent sentence in any other of the logics.

Despite circling around the same key notion, the logics are also very different. Each logic in the above timeline can be seen as an improvement to the previous logic in ease of notation, in ease of technical definition, or in addition of desirable logical properties. For these reasons, the focus of study in this thesis is dependence logic. Furthermore, most things I state about dependence logic can be formulated in the other logics as well.

1.2 Goals of the Thesis

This thesis revolves around two goals. The first is to find out basic properties of dependence logic. One has to learn the basics in order to gain intuition, routine and general understanding which in turn are needed for finding and proving deeper statements about the logic. I hope to shed some light for others who might then be able to reach further in this process.

Dependence logic is still new and thus it is missing much of this necessary groundwork. Much of the research of logics related to dependence logic seems to concentrate on independence friendly logic and its game theoretic semantics. There are indeed interesting philosophical concerns related to what is independence, how semantics games can and should be interpreted, and how should one classify these logics. It is the more concrete and technical side of logic that seems to have been left with less attention.

The second goal of this thesis is to understand where dependence logic comes from in mathematical terms. The answer is not the historical roots that lead to trump semantics. However, the question itself leads there.

On the one hand, trump semantics contains the novel idea of expressing semantics in terms of sets of assignments. This I take as a solid concept. On the other hand, trump semantics contains operations on sets of assignments, one operation for each logical connective and quantifier. My question is, why exactly these operations? The answer seems to be that, at the time of conception of trump semantics, there was no clear view as to what alternative operations there are or how to compare these alternatives. After all, trump semantics does achieve the most important goal it was created for; it is a compositional semantics for independence friendly logic. But now that research in independence friendly logic has reached that goal, it is possible to look around for alternatives and compare them.

When starting from a simple concept, there may be many equivalent ways to define things. When generalising the simple concept to a more complex one, these definitions that were equivalent in the simple case may prove to generalise into definitions that are not equivalent anymore. In fact, some of these generalised definitions may prove to be impractical or lack properties that some of the other generalised definitions may hold. The step from first order logic to independence friendly logic may contain many such cases, and my humble advice to researchers in this area is to keep their eyes open for alternative definitions for the sake of finding better tools to work with.

1.3 Outline of the Thesis

Chapter 2 contains definitions of the logics and notational conventions used in this thesis and reviews some relevant facts.

Chapters 3 to 6 are in the field of the first goal of the thesis, to understand the basics of dependence logic and to gain intuition and general understanding.

Chapter 3 investigates the question under which conditions can the places of two consequtive quantifiers be swapped while preserving the meaning of the formula.

Chapter 4 studies the concept of logical equivalence of two models in both first order logic and dependence logic. I present an Ehrenfeucht-Fraïssé game for logical equivalence in first order logic where the depth or strength of equivalence is described in terms of dependence logic.

Chapter 5 is about translating formulas from one logic to another. I formulate in a general setting what is a translation between two logics that have incompatible definitions for the meaning of a formula. I present several new translations as well as a detailed transcription of the well-known but never precisely stated Enderton-Walkoe translation in dependence logic.

Chapter 6 enters proof theory in dependence logic. The goal is to find a nontrivial fragment of dependence logic such that there is an effectively axiomatisable deductive system for the fragment. A sound proof system is presented for a modest fragment and I conjecture the system to be complete for the fragment.

Chapter 7 presents a new semantics for the syntax of dependence logic. Several key properties of the new semantics are shown as well as a translation to the logic from existential second order formulas. This chapter is the result of research into the second goal of the thesis.

Chapter 2

Preliminaries

In this chapter the reader can familiarise himself with the conventions this thesis has adopted. I also revise the well-known definitions of first order logic and second order logic for the sake of being precise and complete. I also reproduce the basics of dependence logic and team logic with some changes to how they were originally presented by Väänänen [19].

I present each of the four logics separately, with references to the other logics only for concepts that are exactly the same. This is meant to make each logic stand on its own, to minimise induced preference to any particular logic, and also to make it easier to compare the logics in precise terms.

2.1 General Definitions

I use common abbreviations of mathematical expressions. For example, *s.t.* is short for *such that*, and *iff* is short for *if and only if.* When defining a mathematical symbol by an equality, I use $:=$ in place of $=$. The set of natural numbers I denote by $\omega := \{0, 1, 2, \dots\}$. By $n < \omega$ I mean that n is a natural number. When I state something "for all $i \leq n$", it either means for all $i \in \{0, 1, \dots, n\}$ or for all $i \in \{1, \dots, n\}$, depending on the context. This should not cause confusion. Powerset, the set of all subsets of some set A, is denoted $\mathcal{P}A$.

A *language* L is a set of relation symbols and function symbols of various arities. A nullary function symbol is called a *constant*. Language is usually left implicit in this thesis but it is assumed to always contain at least the binary relation symbol of identity, $=$.

A *model* is a tuple $\mathcal{M} = (M, R^{\mathcal{M}}, f^{\mathcal{M}})_{R,f \in L}$ that satisfies the following conditions. M is any nonempty set, called the *universe* of the model. For each relation symbol $R \in L$ with arity n, $R^{\mathcal{M}}$ is a relation $R^{\mathcal{M}} \subseteq M^n$, called the *interpretation* of R in $\mathcal{M}$. The interpretation of the binary relation symbol $=$ is always taken to be the usual identity relation on M, $=^{\mathcal{M}}$ is $\{(a, a) : a \in M\}$. Similarly, for each function symbol $f \in L$ with arity n, $f^{\mathcal{M}}$ is a function $f^{\mathcal{M}} \colon M^n \to M$, called the *interpretation* of f in $\mathcal{M}$.

A *logic* consists of a set of formulas and a semantics that assigns an interpretation to each pair of formula and model. A formula is a string of symbols, more precisely defined separately for each logic. An interpretation of a formula is a set of semantic objects. The definition of a semantic object is done separately for each logic. I often identify a logic with its set of formulas. However, in cases where more than one semantics is defined for the same set of formulas, we must pay attention to which semantics to use.

If L is a logic and $\phi \in L$, then a *subformula* of ϕ is an occurrence of a substring ψ in ϕ such that $\psi \in L$. In other words, a subformula is a triple consisting of the formula ψ, the greater formula ϕ, and the location of ψ in ϕ. For example, there are two different instances of ϕ in the formula $\phi \wedge \phi$ even though both instances are the same when considered as mere formulas. Most often there is no need to refer explicitly to the location of a subformula in a greater formula. When I say that ψ is a subformula of ϕ, I think of ψ both as a formula and as a triple defining the substring occurrence. Let $\psi \leq \phi$ denote that ψ is a subformula of ϕ. Being a subformula is a partial order, given my notational abuse. When ϕ and ψ are the same as formulas, I write $\phi = \psi$. Please note that $\phi = \psi$ and $\psi \leq \theta$ do not imply $\phi \leq \theta$.

In logical formulas, I will use these symbols as connectives: $\neg$ (negation), $\vee$ (disjunction), $\wedge$ (conjunction), $\sim$ (strong negation), $\otimes$ (tensor), and $\oplus$ (sum), and I will use these symbols as quantifiers: $\exists$ (existential quantifier), $\forall$ (universal quantifier), and ! (shriek quantifier).

All instances of a variable x in a formula ϕ are said to be *free* except if the instance lies in a subformula that is of the form $Qx\psi$, where Q is a quantifier. These instances of x are *bound* by the outermost quantifier in the deepest such subformula. The *free variables* of formula ϕ, denoted $\mathrm{FV}(\phi)$, is the set of variables that have free instances in ϕ. If $\mathrm{FV}(\phi) = \emptyset$, then ϕ is called a *sentence*.

As general convention over all logics considered in this thesis, by $\phi(\alpha \mapsto \beta)$ I denote the formula obtained from formula ϕ by replacing α by β. I use this notation rather freely; α and β may be terms or subformulas. If α is an instance of one of these, the replacement is done only on that instance. Otherwise the replacement is performed on all instances of α in ϕ. This should be clear from context in each case. The point of such a convention is to provide lightweight notation for things that easily become cumbersome both in written English and in exact formulation.

We say that ψ is an *immediate subformula* of ϕ if ψ is not ϕ and the only subformulas of ϕ of which ψ is a subformula are ϕ and ψ.

We will be dealing with assignments and teams which I call by the common name *semantic objects*. A semantic object is an object (function or set) is in a sense a generalisation of truth value. The satisfaction of a formula of some logic is defined relative to a model and a semantic object. In propositional logic, a semantic object is a truth value. Thus, given a model and a truth value, we say that a formula either has or has not the truth value in the model. Generalising this, given a model and an assignment of values to free variables, we say that a formula of predicate logic either is or is not satisfied by the assignment in the model. The *interpretation* of a formula ϕ in a model $\mathcal{M}$ and semantics S is the collection of semantic objects that satisfy the formula in the model, denoted $[\![\phi]\!]^S_{\mathcal{M}}$. I may leave out the superscript and the subscript if they are clear from the context.

For formulas ϕ and ψ, we say that ψ is a *logical consequence* of ϕ, or that ϕ *entails* ψ, denoted $\phi \Rightarrow \psi$, if $[\![\phi]\!]_{\mathcal{M}} \subseteq [\![\psi]\!]_{\mathcal{M}}$ for all models $\mathcal{M}$. We say that formulas ϕ and ψ are *logically equivalent*, denoted $\phi \equiv \psi$, if the formulas have the same interpretation for all models, or equivalently, if they are logical consequences of each other.

A *fragment* of a logic L consists of a subset of the formulas of L with the semantics of L.

2.2 First Order Logic (FO)

We define *first order logic* as follows. There is a countably infinite set of variable symbols, or *variables*, $\{v_0, v_1, v_2, \dots\}$. Instead of speaking directly of variables v_n, I use expessions like x and y_k as meta-variables that stand for some actual variables v_n. A *term* (in an implicit language L) is a string of symbols built according to the following rules.

1. A variable x is a term.

2. For an n-ary function symbol $f \in L$ and terms $t_1, \dots, t_n$, also $ft_1 \dots t_n$ is a term.

The set of first order formulas in language L, denoted FO (with the choice of L left implicit), is the set of strings of symbols built according to the following rules.

1. The symbols $\top$ and $\bot$ are first order formulas.

$$
\begin{array}{lcl}
\mathcal{M}, s \models \bot & & \text{never} \\
\mathcal{M}, s \models \top & & \text{always} \\
\mathcal{M}, s \models Rt_1 \dots t_n & \iff & \big(s(t_1), \dots, s(t_n)\big) \in R^{\mathcal{M}} \\
\mathcal{M}, s \models \neg\phi & \iff & \mathcal{M}, s \not\models \phi \\
\mathcal{M}, s \models \phi \vee \psi & \iff & \mathcal{M}, s \models \phi \text{ or } \mathcal{M}, s \models \psi \\
\mathcal{M}, s \models \phi \wedge \psi & \iff & \mathcal{M}, s \models \phi \text{ and } \mathcal{M}, s \models \psi \\
\mathcal{M}, s \models \exists x\phi & \iff & \text{there is } a \in M \text{ s.t. } \mathcal{M}, s(x \mapsto a) \models \phi \\
\mathcal{M}, s \models \forall x\phi & \iff & \text{for all } a \in M,\ \mathcal{M}, s(x \mapsto a) \models \phi
\end{array}
$$

Figure 2.1: Semantics of first order logic

2. For a relation symbol $R \in L$ with arity n and terms $t_1, \dots, t_n$, the string $Rt_1 \dots t_n$ is a first order formula. For binary relation symbols we may use the shorthand xRy for the formula Rxy.

3. If ϕ and ψ are first order formulas and x is a variable, then the following strings are first order formulas: $\neg\phi$, $\phi \vee \psi$, $\phi \wedge \psi$, $\exists x\phi$, $\forall x\phi$.

We call formulas built according to rules 1 and 2 *atomic formulas*. A formula built according to rule 3 is a *compound formula*. We use $\phi \to \psi$ as shorthand notation for $\neg\phi \vee \psi$ for $\phi, \psi \in \text{FO}$.

An *assignment* for a model $\mathcal{M}$ is a function that maps some variables to the model, $s\colon V \to M$ for some $V \subseteq \{v_0, v_1, v_2, \dots\}$. We denote the *domain* of s by $\text{Dom}(s) := V$. In this thesis it is often left to the reader to determine from the context which model an assignment maps to. By $s(x \mapsto a)$ for some $a \in M$ we mean the assignment that maps the variable x to the element a and all other variables like s does. By writing $(x \mapsto a)$ we mean the assignment $\emptyset(x \mapsto a)$.

The *interpretation* of a term t by an assignment s mapping to a model $\mathcal{M}$ I denote with slight abuse of notation by $s(t)$. If t is a variable, the expression $s(t)$ is defined above. In other cases, $s(t)$ is to be read as

$$s(ft_1 \dots t_n) := f^{\mathcal{M}}\big(s(t_1), \dots, s(t_n)\big).$$

Definition 2.2.1. Let $\phi \in \text{FO}$, let $\mathcal{M}$ be a model in the same language as ϕ, and let $s\colon V \to M$ be an assignment for some $V \supseteq \text{FV}(\phi)$. We define *satisfaction* of ϕ in $\mathcal{M}$ by s, denoted $\mathcal{M}, s \models \phi$, or simply $s \models \phi$, as in Figure 2.1.

A first order sentence ϕ is *true* in a model $\mathcal{M}$, denoted $\mathcal{M} \models \phi$, if ϕ is satisfied in $\mathcal{M}$ by the empty assignment. Otherwise ϕ is *false* in $\mathcal{M}$.

A first order formula is *propositional* if it is quantifier-free and all its relation symbols are nullary. Thus, all propositional formulas are sentences.

Two important normal forms for first order formulas are the negation normal form and the conjunctive normal form (or equivalently the disjunctive normal form).

Theorem 2.2.2 (Negation normal form)**.** *For a formula $\phi \in$ FO there is a logically equivalent formula $\psi \in$ FO such that negation appears only in front of atomic formulas. This formula is said to be in* negation normal form*.*

Sketch of proof. We obtain ψ from ϕ by pressing negation down to atomic formulas using the following logical equivalences: $\neg\neg\phi \equiv \phi$, $\neg(\phi \vee \psi) \equiv \neg\phi \wedge \neg\psi$, $\neg(\phi \wedge \psi) \equiv \neg\phi \vee \neg\psi$, $\neg\exists x\phi \equiv \forall x\neg\phi$, $\neg\forall x\phi \equiv \exists x\neg\phi$. □

Theorem 2.2.3 (Conjunctive normal form)**.** *For a quantifier-free formula $\phi \in$ FO there is a logically equivalent quantifier-free first order formula of the form $\bigwedge_i \bigvee_j \psi^i_j$ where each ψ^i_j is either an atomic formula or the negation of one. This formula is said to be in* conjunctive normal form*.*

Sketch of proof. Assuming ϕ is in negation normal form, we obtain a formula in conjunctive normal form from ϕ by pressing disjunction below conjunction using the following logical equivalence: $\phi \vee (\psi \wedge \theta) \equiv (\phi \vee \psi) \wedge (\phi \vee \theta)$. □

2.3 Second Order Logic (SO)

Second order logic extends first order logic by allowing quantification over relations and functions. Formally we define it as follows. The set of variable symbols, or variables, is $\{v_n : n < \omega\} \cup \{V^{\text{rel}}_{k,n} : k, n < \omega\} \cup \{V^{\text{fun}}_{k,n} : k, n < \omega\}$. We may distinguish between the variables by calling v_n *element variables*, $V^{\text{rel}}_{k,n}$ *relation variables* of arity k, and $V^{\text{fun}}_{k,n}$ *function variables* of arity k. Instead of speaking directly of variables v_n, we use expressions like x and y_k as meta-variables that stand for some actual variables v_n. Similarly, we use expressions like R_i and S^i_j as meta-variables for variables $V^{\text{rel}}_{k,n}$, and expressions like f_i and g^i_j as meta-variables for variables $V^{\text{fun}}_{k,n}$. The arity of such meta-variables is left implicit and can usually be inferred from the formula in which they are used.

A *term* (in an implicit language L) is a string of symbols built according to the following rules.

1. An element variable x is a term.

2. For an n-ary function symbol $f \in L$ and terms $t_1, \dots, t_n$, also $ft_1 \dots t_n$ is a term.

3. For an n-ary function variable f and terms $t_1, \dots, t_n$, also $ft_1 \dots t_n$ is a term.

In particular, all terms of first order logic are terms of second order logic.

The set of second order formulas in language L, denoted SO (with the choice of L left implicit), is the set of strings of symbols built according to the following rules.

1. The symbols $\top$ and $\bot$ are second order formulas.

2. For a relation symbol $R \in L$ with arity n and terms $t_1, \dots, t_n$, the string $Rt_1 \dots t_n$ is a second order formula. For binary relation symbols we may use the shorthand xRy for the formula Rxy.

3. For a relation variable R with arity n and terms $t_1, \dots, t_n$, the string $Rt_1 \dots t_n$ is a second order formula.

4. If ϕ and ψ are second order formulas, x is an element variable, R is a relation variable, and f is a function variable, then the following strings are second order formulas: $\neg\phi$, $\phi \vee \psi$, $\phi \wedge \psi$, $\exists x\phi$, $\forall x\phi$, $\exists R\phi$, $\forall R\phi$, $\exists f\phi$, $\forall f\phi$.

In particular, all first order formulas are also second order formulas. We call formulas built according to rules 1, 2 and 3 *atomic formulas*. A formula built according to rule 4 is a *compound formula*. We use $\phi \to \psi$ as shorthand notation for $\neg\phi \vee \psi$ for $\phi, \psi \in$ FO.

An *assignment* for a model $\mathcal{M}$ is a function $s\colon V \to M \cup \{R : R \subseteq M^n$ for some $n < \omega\} \cup \{f : f\colon M^n \to M\}$ for some $V \subseteq \{v_n : n < \omega\} \cup \{V^{\text{rel}}_{k,n} : k, n < \omega\} \cup \{V^{\text{fun}}_{k,n} : k, n < \omega\}$ such that if $x \in V$ then $s(x) \in M$, if $R \in V$ then $s(R) \subseteq M^n$, where n is the arity of R, and if $f \in V$ then $s(f)\colon M^n \to M$, where n is the arity of f. In this thesis it is often left to the reader to determine from the context which model an assignment maps to. For variables x, R, f, where R and f are n-ary, and $a \in M$, $S \subseteq M^n$ and $g\colon M^n \to M$, we denote by $s(x \mapsto a)$, $s(R \mapsto S)$ and $s(f \mapsto g)$ the assignments that map the variable x to a, R to S, and f to g, respectively, and all other variables as s does. This is a simple generalisation of assignment in first order logic. Unsurprisingly, a first order assignment is also a second order assignment.

The *interpretation* of a term t by an assignment s mapping to a model $\mathcal{M}$ we denote with slight abuse of notation by $s(t)$. If t is an element variable,

$$
\begin{array}{lll}
\mathcal{M}, s \models \bot & & \text{never} \\
\mathcal{M}, s \models \top & & \text{always} \\
\mathcal{M}, s \models R\bar{t} & \iff & \big(s(t_1), \dots, s(t_n)\big) \in R^{\mathcal{M}}, \text{ for relation symbols } R \\
\mathcal{M}, s \models R\bar{t} & \iff & \big(s(t_1), \dots, s(t_n)\big) \in s(R), \text{ for relation variables } R \\
\mathcal{M}, s \models \neg\phi & \iff & \mathcal{M}, s \not\models \phi \\
\mathcal{M}, s \models \phi \vee \psi & \iff & \mathcal{M}, s \models \phi \text{ or } \mathcal{M}, s \models \psi \\
\mathcal{M}, s \models \phi \wedge \psi & \iff & \mathcal{M}, s \models \phi \text{ and } \mathcal{M}, s \models \psi \\
\mathcal{M}, s \models \exists x\phi & \iff & \text{there is } a \in M \text{ s.t. } \mathcal{M}, s(x \mapsto a) \models \phi \\
\mathcal{M}, s \models \forall x\phi & \iff & \text{for all } a \in M,\ \mathcal{M}, s(x \mapsto a) \models \phi \\
\mathcal{M}, s \models \exists R\phi & \iff & \text{there is } S \subseteq M^n \text{ s.t. } \mathcal{M}, s(R \mapsto S) \models \phi, \text{ for } n\text{-ary } R \\
\mathcal{M}, s \models \forall R\phi & \iff & \text{for all } S \subseteq M^n,\ \mathcal{M}, s(R \mapsto S) \models \phi, \text{ for } n\text{-ary } R \\
\mathcal{M}, s \models \exists f\phi & \iff & \text{there is } g\colon M^n \to M \text{ s.t. } \mathcal{M}, s(f \mapsto g) \models \phi, \text{ for } n\text{-ary } f \\
\mathcal{M}, s \models \forall f\phi & \iff & \text{for all } g\colon M^n \to M,\ \mathcal{M}, s(f \mapsto g) \models \phi, \text{ for } n\text{-ary } f
\end{array}
$$

Figure 2.2: Semantics of second order logic

the expression $s(t)$ is defined above. In other cases, $s(t)$ is to be read either as $s(ft_1 \dots t_n) := f^{\mathcal{M}}\big(s(t_1), \dots, s(t_n)\big)$, where $f \in L$ is a function symbol, or as $s(ft_1 \dots t_n) := s(f)\big(s(t_1), \dots, s(t_n)\big)$, where f is a function variable.

Definition 2.3.1. Let $\phi \in \mathrm{SO}$, let $\mathcal{M}$ be a model in the same language as ϕ, and let s be an assignment for $\mathcal{M}$, defined on some set V such that $V \supseteq \mathrm{FV}(\phi)$. We define *satisfaction* of ϕ in $\mathcal{M}$ by s, denoted $\mathcal{M}, s \models \phi$ or simply $s \models \phi$, as in Figure 2.2, where $\bar{t}$ stands for $t_1 \dots t_n$.

In particular, first order formulas are satisfied in the first order sense if and only if they are satisfied in the second order sense.

A second order sentence ϕ is *true* in a model $\mathcal{M}$, denoted $\mathcal{M} \models \phi$, if ϕ is satisfied in $\mathcal{M}$ by the empty assignment. Otherwise ϕ is *false* in $\mathcal{M}$.

The abilities to quantify both over relations and over functions is luxury. We can manage with only one.

Theorem 2.3.2. *For a formula $\phi \in \mathrm{SO}$ there is a logically equivalent formula $\psi \in \mathrm{SO}$ such that and ψ does not contain subformulas of the forms $\exists R\theta$ and $\forall R\theta$.*

Sketch of proof. We obtain ψ by repeatedly replacing subformulas in ϕ. Assume that $\exists R\theta$ appears as a subformula in ϕ and R is n-ary. Then we replace

$\exists R\theta$ by $\exists f\exists c\theta'$, where the n-ary f and nullary c are function variables that do not appear in ϕ, and θ' is obtained from θ by replacing the subformulas $Rt_1 \dots t_n$, where R is bound by the quantifier in question, by $ft_1 \dots t_n = c$. In effect, f and c encode the characteristic function of R. Similarly, we can replace $\forall R\theta$ with $\forall f\forall c\theta'$, where θ' is obtained from θ as above. This needs at least two elements in the universe of the model. For the cases of singleton models, we can add an additional subformula. □

Theorem 2.3.3. *For a formula $\phi \in$ SO there is a logically equivalent formula $\psi \in$ SO such that ψ does not contain subformulas of the forms $\exists f\theta$ and $\forall f\theta$.*

Sketch of proof. We obtain ψ by repeatedly replacing subformulas in ϕ. Assume that $\exists f\theta$ appears as a subformula in ϕ and f is n-ary. Then we replace quantification over functions by quantification over relations with the additional assertion that the relation acts like a function, namely we replace $\exists f\theta$ by

$$\exists R\big(\forall x_1 \dots \forall x_n \exists y_1 \forall y_2 (Rx_1 \dots x_n y_1 \wedge (Rx_1 \dots x_n y_2 \rightarrow y_1 = y_2)) \wedge \theta'\big),$$

where θ' is obtained from θ by the following replacements. If $ft_1 \dots t_n$ a term occurrence in ϕ, where f is bound by the quantifier in question, and χ is the least subformula of ϕ that contains this term occurrence, then replace χ by

$$\exists z(Rt_1 \dots t_n z \wedge \chi(ft_1 \dots t_n \mapsto z)).$$ □

An important normal form for second order formulas is the (generalised) Skolem normal form.

Theorem 2.3.4 (Skolem normal form)**.** *For a formula $\phi \in$ SO there is a logically equivalent formula $\psi \in$ SO such that*

$$\psi := \bigexists_{i \le n_1} f_i^1 \bigforall_{i \le n_2} f_i^2 \dots \bigexists_{i \le n_p} f_i^p \bigforall_{i \le m} x_i \theta,$$

where θ is a quantifier-free second order formula. We say that ψ is in Skolem normal form*.*

Sketch of proof. We can obtain ψ from ϕ by repeatedly replacing subformulas by logically equivalent ones. □

We define *existential second order logic* as the fragment of second order logic where universal quantification over relations and sets is disallowed. Formally, the set of formulas of existential second order logic, ESO, contains formulas built according to the following rules.

1. The symbols $\top$ and $\perp$ are existential second order formulas.
2. For a relation symbol $R \in L$ with arity n and terms $t_1, \ldots, t_n$, the string $Rt_1 \ldots t_n$ is an existential second order formula. For binary relation symbols we may use the shorthand xRy for the formula Rxy.
3. For a relation variable R with arity n and terms $t_1, \ldots, t_n$, the string $Rt_1 \ldots t_n$ is an existential second order formula.
4. If ϕ and ψ are existential second order formulas, x is an element variable, R is a relation variable, and f is a function variable, then the following strings are existential second order formulas built from connectives: $\neg\phi$, $\phi \vee \psi$, $\phi \wedge \psi$, and the following strings are existential second order formulas built from quantifiers: $\exists x\phi$, $\forall x\phi$, $\exists R\phi$, $\exists f\phi$.

Theorems 2.3.2, 2.3.3 and 2.3.4 are also true about existential second order formulas in the sense that we can replace SO by ESO everywhere in the claims and the theorems will still hold. In particular, the Skolem normal form for existential second order logic is

$$\exists f_1 \ldots \exists f_n \forall x_1 \ldots \forall x_m \theta,$$

where θ is a quantifier-free second order formula.

There is a *semantic game for second order logic*, denoted $\partial^{\mathrm{SO}}(\mathcal{M}, \phi)$, where $\mathcal{M}$ is a model and $\phi \in \mathrm{SO}$ is a sentence. The game is played by two players called player **I** (male) and player **II** (female). The game is a straightforward generalisation of the semantic game for first order logic; at an existential second order quantifier player **II** chooses how the quantified variable should be interpreted; at a universal second order quantifier the same choice is made by player **I**. A position of the game is a tuple (ψ, s, α), where ψ is a subformula of ϕ, s is a second order assignment, and $\alpha \in \{\mathbf{I}, \mathbf{II}\}$ denotes a player. Each play of the game ends up in one of the players winning. A *strategy* for a player in some game $\partial^{\mathrm{SO}}(\mathcal{M}, \phi)$ is a function that tells the player exactly how to move, based on the current position of play. If the player wins every play where he or she plays by a certain strategy, then that strategy is called a *winning strategy*. The semantic game characterises truth: for all models $\mathcal{M}$ and sentences $\phi \in \mathrm{SO}$, $\mathcal{M} \models \phi$ if and only if player **II** has a winning strategy in $\partial^{\mathrm{SO}}(\mathcal{M}, \phi)$.

2.4 Dependence Logic (FOD)

I define *dependence logic* as follows. The concept of a term is as in first order logic. The set of dependence formulas in language L, denoted FOD (with the choice of L left implicit), is the set of strings of symbols built according to the following rules.

1. The symbols $\top$ and $\bot$ are dependence formulas.

2. For a relation symbol $R \in L$ with arity n and terms $t_1, \ldots, t_n$, the string $Rt_1 \ldots t_n$ is a dependence formula. For binary relation symbols we may use the shorthand xRy for the formula Rxy.

3. For terms $t_1, \ldots, t_n, u$, the string $(t_1 \ldots t_n)\rightsquigarrow u$ is a dependence formula.

4. If ϕ and ψ are dependence formulas and x is a variable, then the following strings are dependence formulas: $\neg\phi$, $\phi \vee \psi$, $\phi \wedge \psi$, $\exists x\phi$, $\forall x\phi$.

In particular, a first order formula is a dependence formula. We call formulas built according to rules 1, 2 and 3 *atomic formulas*. A formula built according to rule 4 is a *compound formula*. A formula of the form 3 is called a *D-formula*. D stands for either dependence, as in "u depends on each t_i", or determination, as in "t_i together determine u".

A formula $\phi \in$ FOD is said to be in *strict negation normal form* if negation appears only in front of atomic formulas of the form $Rt_1 \ldots t_n$. For example, the formulas $\neg(Rxy \vee Pz)$ and $\neg(x, y)\rightsquigarrow z$ are not in strict negation normal form but the formula $\neg Pz \wedge (x, y)\rightsquigarrow z$ is.

Väänänen denotes D-formulas by $=(t_1, \ldots, t_n, u)$. I chose the notation $(t_1 \ldots t_n)\rightsquigarrow u$ instead to make D-formulas stick out in formulas more prominently and to help the reader not to confuse them with the identity relation. The wiggly arrow symbol in D-formulas is an arrow because it represents the direction of functional dependence—given the values on the left we can compute the value on the right. The arrow is wiggly so that the reader would not mistake it for implication (even though we do not use the implication arrow in FOD). This idea is borrowed from early logic books where the symbol of equivalence relation in logical formulas was chosen to be $\approx$ instead of $=$ in an attempt to emphasise that it is a symbol, not the relation itself. My choice of notation for D-formulas also rules out the degenerate case $=()$ which Väänänen puts to the role where I have the symbol $\top$.

The most important concept is that of a *team*. We define it as a set of assignments that have the same domain and map to the same model. For a nonempty team X, we define its *domain*, $\mathrm{Dom}(X)$, as the domain of any of the assignments in the team. We leave the domain of the empty team undefined; by $\mathrm{Dom}(\emptyset)$ we mean any set of variables, interpreted in a suitable way in each context separately.[1] For a model $\mathcal{M}$ and a set of variables V, let $X_V^{\mathcal{M}}$ denote the *full team* on V, i.e. the set of all assignments that have domain V and map to the model $\mathcal{M}$. Given $\phi \in \mathrm{FOD}$ and a model $\mathcal{M}$, the full team for these two is $X^{\mathcal{M}}_{\mathrm{FV}(\phi)}$.

Teams and relations have a close relationship. If X is a team such that $\mathrm{Dom}(X) = \{v_{i_1}, \ldots, v_{i_k}\}$, where $i_1 < \cdots < i_k$, we define the *corresponding relation* as

$$\mathrm{Rel}(X) = \big\{\big(s(v_{i_1}), \ldots, s(v_{i_k})\big) : s \in X\big\}.$$

Note that the order of the values of variables in the tuples in the resulting relation is always the one induced by the natural order of the indices of the variables. Conversely, for a n-ary relation R, we define the *corresponding team* with variable order $(x_1, \ldots, x_n)$ as

$$R_{(x_1,\ldots,x_n)} = \big\{(x_1 \mapsto a_1, \ldots, x_n \mapsto a_n) : (a_1, \ldots, a_n) \in R\big\}.$$

For a team X and a set of variables V, we define the *restriction* of X to V as

$$X{\upharpoonright}V := \{s{\upharpoonright}V : s \in X\},$$

where $s{\upharpoonright}V$ is the restriction of s to the domain V, i.e. the assignment that has domain $\mathrm{Dom}(s) \cap V$ and that maps like s does. Similarly, we define the *co-restriction* of X from V as

$$X{\downharpoonleft}V := X{\upharpoonright}(\mathrm{Dom}(X) \setminus V).$$

If $V = \{x\}$, we denote $X{\upharpoonright}V = X{\upharpoonright}x$ and $X{\downharpoonleft}V = X{\downharpoonleft}x$.

Let X be a team for some model $\mathcal{M}$, let $a \in M$, let $F\colon X{\upharpoonright}V \to M$ for some $V \subseteq \mathrm{Dom}(X)$, and let x be a variable. We define the following operations for extending a team.

$$\begin{aligned} X(x \mapsto a) &:= \{s(x \mapsto a) : s \in X\} \\ X(x \mapsto F) &:= \{s(x \mapsto F(s{\upharpoonright}V)) : s \in X\} \\ X(x \mapsto M) &:= \{s(x \mapsto b) : s \in X \text{ and } b \in M\} \end{aligned}$$

It might help one's intuition to note that the two first operations will not increase the cardinality of the team, whereas the third operation can potentially blow the team's cardinality skyhigh. By writing $(x \mapsto M)$ we mean $\{\emptyset\}(x \mapsto M)$, etc.

[1] For example, $\mathrm{Dom}(X) = \mathrm{Dom}(Y)$ is true also if one or both of the teams X and Y are empty.

I shall now define the semantics for dependence logic. There is a key difference to the way Väänänen defines the semantics [19, Definition 3.5], namely the treatment of negation. As is known and thoroughly demonstrated by Burgess [3], negation in dependence logic is not a semantic operation on formula interpretations.[2] Because of this, I define semantics only for dependence formulas in strict negation normal form. This way we avoid mixing the syntactic formula manipulation that negation represents into the semantics which is otherwise not about syntactic manipulation of formulas. I will then allow the free use of negation as shorthand notation only. This will in effect have the same result as Väänänen's semantics.[3]

Definition 2.4.1. Let $\phi \in \mathrm{FOD}$ be in strict negation normal form, let $\mathcal{M}$ be a model in the same language as ϕ, and let X be a team with $\mathrm{Dom}(X) \supseteq \mathrm{FV}(\phi)$. We define *satisfaction* of ϕ in $\mathcal{M}$ by X, denoted $\mathcal{M}, X \models^{\mathrm{FOD}} \phi$, or $\mathcal{M}, X \models \phi$, or simply $X \models \phi$, as in Figure 2.3. A dependence sentence ϕ is *true* in a model $\mathcal{M}$, denoted $\mathcal{M} \models \phi$, if ϕ is satisfied in $\mathcal{M}$ by the full team. Otherwise ϕ is *false* in $\mathcal{M}$.[4] There are no other truth values.

Definition 2.4.2. We allow the use of dependence formulas ϕ that are not in strict negation normal form in satisfaction statements $\mathcal{M}, X \models^{\mathrm{FOD}} \phi$ by reading negation as shorthand that is unravelled by the rules in Figure 2.4. The rule $\neg(t_1 \ldots t_n) \rightsquigarrow u \mapsto \bot$ is to be applied only when no other rule can be applied.

The requirement of applying the shorthand rule of negated D-formula last resolves an ambiguity caused by nested negations in front of D-formulas. For example, $\neg\neg(x) \rightsquigarrow y$ is shorthand for $(x) \rightsquigarrow y$ and not for $\top$. The other rules can be applied in any order without ambiguity.

Another kind of ambiguity emerges from the use of negation in a formula denoted by a symbol. For example, consider the formula $\neg\phi$, where $\phi :=$ $\neg(x) \rightsquigarrow y$. If we think of ϕ as a formula with the replacement rules applied, then $\neg\phi$ reads $\top$. If we think of ϕ as meta-notation and not as a concrete formula, we may read $\neg\phi$ as $(x) \rightsquigarrow y$. I do not provide a means to resolve this kind of ambiguities. As with all mathematical shorthand notation, the use of $\neg$ in dependence formulas must be done with consideration. The one who writes down the formula carries the responsibility of avoiding and resolving possible ambiguities caused by free use of negation.

---

[2] Burgess' paper concerns Henkin quantifiers, but the same treatment applies to independence friendly logic and dependence logic as well. See Theorem 2.4.6.

[3] In Chapter 7 I take another approach and define a semantics for dependence formulas where negation *is* a semantic operation.

[4] It is equivalent to define that ϕ is true in $\mathcal{M}$ if $\mathcal{M}, \{\emptyset\} \models \phi$.

$$
\begin{array}{lll}
\mathcal{M}, X \models \bot & \iff & X = \emptyset \\
\mathcal{M}, X \models \top & & \text{always} \\
\mathcal{M}, X \models Rt_1 \dots t_n & \iff & X \subseteq \big\{ s \in X^{\mathcal{M}}_{\mathrm{Dom}(X)} : \mathcal{M}, s \models Rt_1 \dots t_n \big\} \\
\mathcal{M}, X \models \neg Rt_1 \dots t_n & \iff & X \subseteq \big\{ s \in X^{\mathcal{M}}_{\mathrm{Dom}(X)} : \mathcal{M}, s \models \neg Rt_1 \dots t_n \big\} \\
\mathcal{M}, X \models (t_1 \dots t_n) \rightsquigarrow u & \iff & \text{there is } f \colon M^n \to M \text{ s.t.} \\
& & \text{for all } s \in X \colon s(u) = f\big(s(t_1), \dots, s(t_n)\big) \\
\mathcal{M}, X \models \phi \vee \psi & \iff & \text{there is } Y, Z \subseteq X \text{ s.t. } Y \cup Z = X \text{ and} \\
& & \mathcal{M}, Y \models \phi \text{ and } \mathcal{M}, Z \models \psi \\
\mathcal{M}, X \models \phi \wedge \psi & \iff & \mathcal{M}, X \models \phi \text{ and } \mathcal{M}, X \models \psi \\
\mathcal{M}, X \models \exists x \phi & \iff & \text{there is } F \colon X \to M \text{ s.t.} \\
& & \mathcal{M}, X(x \mapsto F) \models \phi \\
\mathcal{M}, X \models \forall x \phi & \iff & \mathcal{M}, X(x \mapsto M) \models \phi
\end{array}
$$

Figure 2.3: Semantics of dependence logic

$$
\begin{array}{rr}
\neg(\phi \vee \psi) \mapsto \neg\phi \wedge \neg\psi & \neg\bot \mapsto \top \\
\neg(\phi \wedge \psi) \mapsto \neg\phi \vee \neg\psi & \neg\top \mapsto \bot \\
\neg\exists x\phi \mapsto \forall x \neg\phi & \neg\neg\phi \mapsto \phi \\
\neg\forall x\phi \mapsto \exists x \neg\phi & \neg(t_1 \dots t_n) \rightsquigarrow u \mapsto \bot
\end{array}
$$

Figure 2.4: Syntactic unravelling of negation

Note that $\neg\phi$ for formulas ϕ not of the form $Rt_1 \ldots t_n$ is mere shorthand notation. Such formulas $\neg\phi$ have no part in the actual semantics of dependence logic, as I define it, and thus there is no ambiguity in Definition 2.4.1, the semantics. The ambiguity comes only from the additional syntactic tool that I give in Definition 2.4.2, the unrestricted use of negation which is provided as a convenience that should be used responsibly. Väänänen's semantics resolves this ambiguity. His solution is to define satisfaction, $\mathcal{M}, X \models \phi$, by referring to the *fundamental predicate* of $\mathcal{M}$, $\mathcal{T}_{\mathcal{M}}$, which consists of triples (ϕ, X, d), where $\phi \in \mathrm{FOD}$, X is a team that satisfies ϕ, and $d \in \{0, 1\}$ we could call the *mode* of satisfaction. Intuitively speaking, the mode of satisfaction keeps count of multiple negations. Because negation flips the mode of satisfaction instead of modifying the subformula, Väänänen's semantics never loses information stored in the syntax of the subformula. I have chosen to present the semantics without the fundamental predicate firstly to make the definition of semantics more direct and clear and secondly to explicitly state that negation is a purely syntactic operation except in front of relation symbols.

The semantics for FOD, as presented above, is equivalent to Väänänen's semantics in the sense that both semantics give the same interpretation to all dependence formulas.

Theorem 2.4.3. *For all $\phi \in \mathrm{FOD}$, $[\![\phi]\!]^{\mathrm{FOD}}_{\mathcal{M}} = [\![\phi]\!]^{\mathrm{V}}_{\mathcal{M}}$, where* V *stands for Väänänen's semantics for dependence logic for which we do not read negation as shorthand.*[5]

Proof. We prove the theorem by proving the stronger claim that

$$[\![\phi]\!]^{\mathrm{FOD}}_{\mathcal{M}} = [\![\phi]\!]^{\mathrm{V}}_{\mathcal{M}} \quad \text{and} \quad [\![\neg\phi]\!]^{\mathrm{FOD}}_{\mathcal{M}} = [\![\neg\phi]\!]^{\mathrm{V}}_{\mathcal{M}}. \tag{2.1}$$

The proof is by induction on formulas $\phi \in \mathrm{FOD}$ in strict negation normal form. We can easily see that (2.1) holds for the cases where ϕ is of the form $\top$, $\bot$ or $Rt_1 \ldots t_n$.

Case $(t_1 \ldots t_n) \rightsquigarrow u$**.** We have $\mathcal{M}, X \models^{\mathrm{FOD}} (t_1 \ldots t_n) \rightsquigarrow u$ iff there is $f \colon M^n \to M$ such that for all $s \in X$: $s(u) = f\big(s(t_1), \ldots, s(t_n)\big)$. This is equivalent with the condition that for all $s, s' \in X$: if $s(t_i) = s'(t_i)$ for all $i \leq n$, then $s(u) = s'(u)$, which is the definition for $\mathcal{M}, X \models^{\mathrm{V}} {=}(t_1, \ldots, t_n, u)$.

Reading negation as shorthand, we get $\neg(t_1 \ldots t_n) \rightsquigarrow u = \bot$. We have $\mathcal{M}, X \models^{\mathrm{FOD}} \bot$ iff $X = \emptyset$ iff $\mathcal{M}, X \models^{\mathrm{V}} \neg{=}(t_1, \ldots, t_n, u)$.

[5] Furthermore, we implicitly translate between the two syntaxes of D-formulas, $(t_1 \ldots t_n) \rightsquigarrow u$ and ${=}(t_1, \ldots, t_n, u)$, as necessary.

Case $\neg\phi$. We have $[\![\neg\phi]\!]^{\mathrm{FOD}}_{\mathcal{M}} = [\![\neg\phi]\!]^{\mathrm{V}}_{\mathcal{M}}$ because (2.1) holds for ϕ by the inductive hypothesis.

Reading negation as shorthand, we get $\neg\neg\phi = \phi$. We have $\mathcal{M}, X \models^{\mathrm{FOD}} \phi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \phi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \neg\neg\phi$. Thus $[\![\neg\neg\phi]\!]^{\mathrm{FOD}}_{\mathcal{M}} = [\![\neg\neg\phi]\!]^{\mathrm{V}}_{\mathcal{M}}$.

Case $\phi \vee \psi$. We have $\mathcal{M}, X \models^{\mathrm{FOD}} \phi \vee \psi$ iff there is $Y, Z \subseteq X$ such that $Y \cup Z = X$ and $\mathcal{M}, Y \models^{\mathrm{FOD}} \phi$ and $\mathcal{M}, Z \models^{\mathrm{FOD}} \psi$ iff there is $Y, Z \subseteq X$ such that $Y \cup Z = X$ and $\mathcal{M}, Y \models^{\mathrm{V}} \phi$ and $\mathcal{M}, Z \models^{\mathrm{V}} \psi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \phi \vee \psi$.

Reading negation as shorthand, we get $\neg(\phi \vee \psi) = \neg\phi \wedge \neg\psi$. We have $\mathcal{M}, X \models^{\mathrm{FOD}} \neg\phi \wedge \neg\psi$ iff $\mathcal{M}, X \models^{\mathrm{FOD}} \neg\phi$ and $\mathcal{M}, X \models^{\mathrm{FOD}} \neg\psi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \neg\phi$ and $\mathcal{M}, X \models^{\mathrm{V}} \neg\psi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \neg\phi \wedge \neg\psi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \neg(\phi \vee \psi)$.

Case $\phi \wedge \psi$. We have $\mathcal{M}, X \models^{\mathrm{FOD}} \phi \wedge \psi$ iff $\mathcal{M}, X \models^{\mathrm{FOD}} \phi$ and $\mathcal{M}, X \models^{\mathrm{FOD}} \psi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \phi$ and $\mathcal{M}, X \models^{\mathrm{V}} \psi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \phi \wedge \psi$.

Reading negation as shorthand, we get $\neg(\phi \wedge \psi) = \neg\phi \vee \neg\psi$. We have $\mathcal{M}, X \models^{\mathrm{FOD}} \neg\phi \vee \neg\psi$ iff there is $Y, Z \subseteq X$ such that $Y \cup Z = X$ and $\mathcal{M}, Y \models^{\mathrm{FOD}} \neg\phi$ and $\mathcal{M}, Z \models^{\mathrm{FOD}} \neg\psi$ iff there is $Y, Z \subseteq X$ such that $Y \cup Z = X$ and $\mathcal{M}, Y \models^{\mathrm{V}} \neg\phi$ and $\mathcal{M}, Z \models^{\mathrm{V}} \neg\psi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \neg\phi \vee \neg\psi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \neg(\phi \wedge \psi)$.

Case $\exists x\phi$. We have $\mathcal{M}, X \models^{\mathrm{FOD}} \exists x\phi$ iff there is $F \colon X \to M$ such that $\mathcal{M}, X(x \mapsto F) \models^{\mathrm{FOD}} \phi$ iff there is $F \colon X \to M$ such that $\mathcal{M}, X(x \mapsto F) \models^{\mathrm{V}} \phi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \exists x\phi$.

Reading negation as shorthand, we get $\neg\exists x\phi = \forall x\neg\phi$. We have $\mathcal{M}, X \models^{\mathrm{FOD}} \forall x\neg\phi$ iff $\mathcal{M}, X(x \mapsto M) \models^{\mathrm{FOD}} \neg\phi$ iff $\mathcal{M}, X(x \mapsto M) \models^{\mathrm{V}} \neg\phi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \forall x\neg\phi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \neg\exists x\phi$.

Case $\forall x\phi$. $\mathcal{M}, X \models^{\mathrm{FOD}} \forall x\phi$ iff $\mathcal{M}, X(x \mapsto M) \models^{\mathrm{FOD}} \phi$ iff $\mathcal{M}, X(x \mapsto M) \models^{\mathrm{V}} \phi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \forall x\phi$.

Reading negation as shorthand, we get $\neg\forall x\phi = \exists x\neg\phi$. We have $\mathcal{M}, X \models^{\mathrm{FOD}} \exists x\neg\phi$ iff there is $F \colon X \to M$ such that $\mathcal{M}, X(x \mapsto F) \models^{\mathrm{FOD}} \neg\phi$ iff there is $F \colon X \to M$ such that $\mathcal{M}, X(x \mapsto F) \models^{\mathrm{V}} \neg\phi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \exists x\neg\phi$ iff $\mathcal{M}, X \models^{\mathrm{V}} \neg\forall x\phi$. $\square$

Here are a few important properties of dependence logic.

Theorem 2.4.4. *Interpretations of dependence formulas are closed downward; for all formulas $\phi \in \mathrm{FOD}$, models $\mathcal{M}$ and teams X, Y it holds that if $\mathcal{M}, X \models \phi$ and $Y \subseteq X$ then $\mathcal{M}, Y \models \phi$.*

Interpretations of dependence formulas are nonempty; for all formulas $\phi \in \mathrm{FOD}$ and models $\mathcal{M}$ it holds that $\mathcal{M}, \emptyset \models \phi$.

Proof. Elementary, see Väänänen [19, Lemma 3.9 and Proposition 3.10]. □

Theorem 2.4.5 (Separation Theorem). *Let $\phi, \psi \in \mathrm{FOD}$ be sentences such that for all models $\mathcal{M}$ either $\mathcal{M} \not\models \phi$ or $\mathcal{M} \not\models \psi$. Then there is a first order sentence χ such that if $\mathcal{M} \models \phi$ then $\mathcal{M} \models \chi$, and if $\mathcal{M} \models \chi$ then $\mathcal{M} \not\models \psi$.*

Proof. See Väänänen [19, Theorem 6.7]. □

For the purposes of the next theorem, let $\mathrm{Mod}^{\mathrm{V}}(\phi)$, for dependence sentences ϕ, denote the class of models $\mathcal{M}$ such that the universe M has at least two elements and ϕ is true in $\mathcal{M}$ in Väänänen's semantics.

Theorem 2.4.6 (Burgess [3]). *Assume that $\phi, \psi \in \mathrm{FOD}$ are sentences such that $\mathrm{Mod}^{\mathrm{V}}(\phi) \cap \mathrm{Mod}^{\mathrm{V}}(\psi) = \emptyset$. Then there is a sentence $\theta \in \mathrm{FOD}$ such that $\mathrm{Mod}^{\mathrm{V}}(\theta) = \mathrm{Mod}^{\mathrm{V}}(\phi)$ and $\mathrm{Mod}^{\mathrm{V}}(\neg\theta) = \mathrm{Mod}^{\mathrm{V}}(\psi)$.*

Proof. The proof is divided into three cases.

Firstly, if $\mathrm{Mod}^{\mathrm{V}}(\phi) = \emptyset$ and $\mathrm{Mod}^{\mathrm{V}}(\psi) = \emptyset$, then let $\theta := \theta_0$, where

$$\theta_0 := \forall x{=}(x).$$

Then only singleton models satisfy θ, whence $\mathrm{Mod}^{\mathrm{V}}(\theta) = \emptyset$, and no models satisfy $\neg\theta$, whence $\mathrm{Mod}^{\mathrm{V}}(\neg\theta) = \emptyset$.

If $\mathrm{Mod}^{\mathrm{V}}(\phi) \neq \emptyset$ and $\mathrm{Mod}^{\mathrm{V}}(\psi) = \emptyset$, then let

$$\theta := \phi \vee \theta_0.$$

Then

$$\mathrm{Mod}^{\mathrm{V}}(\theta) = \mathrm{Mod}^{\mathrm{V}}(\phi) \cup \mathrm{Mod}^{\mathrm{V}}(\theta_0) = \mathrm{Mod}^{\mathrm{V}}(\phi),$$

and

$$\mathrm{Mod}^{\mathrm{V}}(\neg\theta) = \mathrm{Mod}^{\mathrm{V}}(\neg\phi \wedge \neg\theta_0) = \mathrm{Mod}^{\mathrm{V}}(\neg\phi) \cap \mathrm{Mod}^{\mathrm{V}}(\neg\theta_0) = \emptyset.$$

If $\mathrm{Mod}^{\mathrm{V}}(\phi) \neq \emptyset$ and $\mathrm{Mod}^{\mathrm{V}}(\psi) \neq \emptyset$, by the above case we can assume that $\mathrm{Mod}^{\mathrm{V}}(\neg\phi) = \emptyset$ and $\mathrm{Mod}^{\mathrm{V}}(\neg\psi) = \emptyset$. Then by Theorem 2.4.5 there is a sentence $\chi \in \mathrm{FO}$ such that $\mathrm{Mod}^{\mathrm{V}}(\phi) \subseteq \mathrm{Mod}^{\mathrm{V}}(\chi)$ and $\mathrm{Mod}^{\mathrm{V}}(\psi) \subseteq \mathrm{Mod}^{\mathrm{V}}(\neg\chi)$. Let

$$\theta := \phi \wedge (\neg\psi \vee \chi).$$

Then

$$\begin{aligned}\mathrm{Mod}^{\mathrm{V}}(\theta) &= \mathrm{Mod}^{\mathrm{V}}(\phi) \cap \big(\mathrm{Mod}^{\mathrm{V}}(\neg\psi) \cup \mathrm{Mod}^{\mathrm{V}}(\chi)\big)\\ &= \mathrm{Mod}^{\mathrm{V}}(\phi) \cap \mathrm{Mod}^{\mathrm{V}}(\chi)\\ &= \mathrm{Mod}^{\mathrm{V}}(\phi),\end{aligned}$$

and

$$\begin{aligned}\mathrm{Mod}^{\mathrm{V}}(\neg\theta) &= \mathrm{Mod}^{\mathrm{V}}\big(\neg\phi \vee (\psi \wedge \neg\chi)\big)\\ &= \mathrm{Mod}^{\mathrm{V}}(\neg\phi) \cup \big(\mathrm{Mod}^{\mathrm{V}}(\psi) \cap \mathrm{Mod}^{\mathrm{V}}(\neg\chi)\big)\\ &= \emptyset \cup \mathrm{Mod}^{\mathrm{V}}(\psi)\\ &= \mathrm{Mod}^{\mathrm{V}}(\psi).\end{aligned}$$ □

Burgess' theorem shows how utterly impossible it is to define negation in dependence logic as a semantic operation on formula interpretations. To see this, assume f is such an operation, i.e. that $f\big([\![\phi]\!]^{\mathrm{V}}_{\mathcal{M}}\big) = [\![\neg\phi]\!]^{\mathrm{V}}_{\mathcal{M}}$ for all $\phi \in \mathrm{FOD}$ and all models $\mathcal{M}$. Let ϕ be any dependence sentence that is false in some nonempty model class C. Let ψ_1 and ψ_2 be dependence sentences that define two different subclasses of C. Then, without loss of generality, there is a model $\mathcal{M} \in C$ such that $\mathcal{M} \models \psi_1$ and $\mathcal{M} \not\models \psi_2$. Theorem 2.4.6 gives us ϕ_1 and ϕ_2 such that $\mathrm{Mod}^{\mathrm{V}}(\phi_1) = \mathrm{Mod}^{\mathrm{V}}(\phi) = \mathrm{Mod}^{\mathrm{V}}(\phi_2)$ and $\mathrm{Mod}^{\mathrm{V}}(\neg\phi_1) = \psi_1$ and $\mathrm{Mod}^{\mathrm{V}}(\neg\phi_2) = \psi_2$. Now, $\{\emptyset\} \in [\![\neg\phi_1]\!]^{\mathrm{V}}_{\mathcal{M}} = f\big([\![\phi]\!]^{\mathrm{V}}_{\mathcal{M}}\big)$ and $\{\emptyset\} \notin [\![\neg\phi_2]\!]^{\mathrm{V}}_{\mathcal{M}} = f\big([\![\phi]\!]^{\mathrm{V}}_{\mathcal{M}}\big)$, which is a contradiction. The notable thing is that this is not only a single counterexample to the definability of negation in dependence logic as a semantic operation—this is a procedure that, by letting us choose the formulas ψ_i, lets us choose the interpretation of $\neg\phi_i$ in models in class C. This is in strong contrast to the supposed operation f determining the interpretation of $\neg\phi_i$ from the interpretation of ϕ_i which is the interpretation of ϕ.

Note that Theorem 2.4.6 relies on dependence formulas that are not in strict negation normal form. In its proof, negation is used as a switch that leaves one part of the formulas θ active in the positive case θ while activating the other part in the negative case $\neg\theta$. It is an open question if a similar construction is possible when formulas are required to be in strict negation normal form, as in Definition 2.4.1.

Sometimes it is said that dependence logic, like independence friendly logic, has three truth values and thus fails the law of excluded middle; a sentence can be true, it can be false, or it can be neither. In this thesis I have only defined two truth values for dependence sentences; truth and falsehood. One can recover the "third truth value" by considering pairs of sentences $(\phi, \neg\phi)$ instead of single sentences $\phi \in \mathrm{FOD}$. Then we can define truth as the case where ϕ is true and $\neg\phi$ is false, falsehood as the case where ϕ is false and $\neg\phi$ is true, and the third truth value as the case where ϕ is false and $\neg\phi$ is false. As Theorem 2.4.6 shows, there is no way to tell apart falsehood and the third truth value knowing only if ϕ is true. In other words, given the interpretation of a dependence sentence in some model, we can easily determine if the sentence is false but we have no way of telling if its negation is true or false, unless we know the exact syntax of the sentence. Because of this, the third truth value is artificial and is no more justified than taking any other kinds of tuples of formulas such as $(\phi, \forall v_0\phi, x = y \wedge \phi)$ and define more truth values in a similar way.

The reader may study Väänänen's book for other properties of dependence logic. One more property we present here with a proof, however, to point out the need of the axiom of choice. The following theorem states that satisfaction of a formula by a team does not depend on the values the team gives to variables that are not free in the formula [19, Lemma 3.27].

Theorem 2.4.7. *Satisfaction of a formula depends on the values of only those variables that occur free in the formula; for all $\phi \in \mathrm{FOD}$ and $V \supseteq \mathrm{FV}(\phi)$ it holds that $\mathcal{M}, X \models \phi$ if and only if $\mathcal{M}, X{\restriction}V \models \phi$.*

Proof. Proof by induction on ϕ.

Case atomic. The claim is clear for formulas $\top$ and $\bot$. The claim for $Rt_1 \dots t_n$ follows easily from the fact that $s \models Rt_1 \dots t_n$ if and only if $s{\restriction}V \models Rt_1 \dots t_n$. Similarly for $\neg Rt_1 \dots t_n$. The case for $(t_1 \dots t_n){\rightsquigarrow}u$ is just as straightforward because $s(t) = (s{\restriction}V)(t)$ for all the terms t that appear in the formula.

Case $\psi_1 \vee \psi_2$. First note that $(Y \cup Z){\restriction}V = Y{\restriction}V \cup Z{\restriction}V$ for any teams Y, Z with the same domain. Now, $X \models \phi$ iff there are $Y_1, Y_2 \subseteq X$ such that $Y_1 \cup Y_2 = X$ and $Y_i \models \psi_i$ for both $i = 1, 2$. Then by the induction hypothesis $Y_i{\restriction}V \models \psi_i$ for both $i = 1, 2$. Because $X{\restriction}V = (Y_1 \cup Y_2){\restriction}V = Y_1{\restriction}V \cup Y_2{\restriction}V$, we get $X{\restriction}V \models \phi$.

For the other direction, if $X{\restriction}V \models \phi$, then there are $Z_1, Z_2 \subseteq X{\restriction}V$ such that $Z_1 \cup Z_2 = X$ and $Z_i \models \psi_i$ for both $i = 1, 2$. Let $Y_i := \{s \in X : s{\restriction}V \in Z_i\}$ for both $i = 1, 2$. Then $Y_1 \cup Y_2 = X$. By the induction hypothesis and the fact that $Z_i = Y_i{\restriction}V$ we get $Y_i \models \phi$ for both $i = 1, 2$.

Case $\psi_1 \wedge \psi_2$. $X \models \phi$ iff $X \models \psi_i$ for both $i = 1, 2$ iff (by the induction hypothesis) $X{\restriction}V \models \psi_i$ for both $i = 1, 2$ iff $X{\restriction}V \models \phi$.

Case $\exists x\psi$**.** $X \models \phi$ iff there is $F\colon X \to M$ such that $X(x \mapsto F) \models \psi$. Then by the induction hypothesis $X(x \mapsto F)\upharpoonright(V \cup \{x\}) \models \psi$. Assuming the axiom of choice, we can pick a function $G\colon X\upharpoonright V \to M$ such that

$$G(s') \in \{F(s) : s \in X \text{ and } s \text{ extends } s'\}$$

for all $s' \in X\upharpoonright V$. Now $(X\upharpoonright V)(x \mapsto G) \subseteq X(x \mapsto F)\upharpoonright(V \cup \{x\})$, whence $(X\upharpoonright V)(x \mapsto G) \models \psi$ and further $X\upharpoonright V \models \phi$.

For the other direction, if $X\upharpoonright V \models \phi$, then there is $G\colon X\upharpoonright V \to M$ such that $(X\upharpoonright V)(x \mapsto G) \models \psi$. Define $F\colon X \to M$ by mapping

$$F(s) = G(s\upharpoonright V)$$

for all $s \in X$. Now $X(x \mapsto F)\upharpoonright(V \cup \{x\}) = (X\upharpoonright V)(x \mapsto G)$, whence by the induction hypothesis $X(x \mapsto F) \models \psi$ and further $X \models \phi$.

Case $\forall x\psi$**.** First note that $X(x \mapsto M)\upharpoonright(V \cup \{x\}) = (X\upharpoonright V)(x \mapsto M)$. Now, $X \models \phi$ iff $X(x \mapsto M) \models \psi$ iff (by the induction hypothesis) $X(x \mapsto M)\upharpoonright(V \cup \{x\}) \models \psi$ iff $(X\upharpoonright V)(x \mapsto M) \models \psi$ iff $X\upharpoonright V \models \phi$. □

The axiom of choice really is necessary for the previous theorem, as the following theorem shows.

Theorem 2.4.8. *Assuming the negation of the axiom of choice, there is some formula $\phi \in$ FOD, model $\mathcal{M}$ and team X such that $\mathcal{M}, X \models \phi$ but $\mathcal{M}, X\upharpoonright\mathrm{FV}(\phi) \not\models \phi$.*

Proof. Assuming the negation of the axiom of choice, there are some non-empty sets A_i for $i \in I$ for some index set I such that there is no function $f\colon \{A_i : i \in I\} \to \bigcup_{i\in I} A_i$ such that $f(A_i) \in A_i$ for all $i \in I$.

Let $\phi := \exists y Rxy$, let $\mathcal{M} := (M, R^{\mathcal{M}})$, where $M := I \cup \bigcup_{i\in I} A_i$ and

$$R^{\mathcal{M}} := \{(i, a) : i \in I \text{ and } a \in A_i\},$$

and let $X := R^{\mathcal{M}}_{(x,y)}$. Then we have $\mathcal{M}, X \models \phi$ because the function $F\colon X \to M$, $F(s) = s(y)$ for all $s \in X$, trivially satisfies $\mathcal{M}, X(y \mapsto F) \models Rxy$. However, if $\mathcal{M}, X\upharpoonright x \models \phi$ then there would be a function $G\colon X\upharpoonright x \to M$ such that $G(s') \in A_{s'(x)}$ for all $s' \in X\upharpoonright x$. Then we could define $f\colon \{A_i : i \in I\} \to \bigcup_{i\in I} A_i$, $f(A_i) = G(s'_i)$ for all $i \in I$, where $s'_i = (x \mapsto i)$, contradicting our assumption. □

2.5 Team Logic (TL)

We define *team logic* as follows. The concept of a term is as in first order logic. The set of team formulas in language L, denoted TL (with the choice of L left implicit), is the set of strings of symbols built according to the following rules.

1. The symbols $\top$, $\bot$, $\mathbf{0}$ and $\mathbf{1}$ are team formulas.
2. For a relation symbol $R \in L$ with arity n and terms $t_1 \dots t_n$, the string $Rt_1 \dots t_n$ is a team formula. For binary relation symbols we may use the shorthand xRy for the formula Rxy.
3. For terms $t_1, \dots, t_n, u$, the string $(t_1 \dots t_n) \rightsquigarrow u$ is a team formula.
4. If ϕ and ψ are dependence formulas and x is a variable, then the following strings are team formulas: $\neg\phi$, $\sim\phi$, $\phi \vee \psi$, $\phi \wedge \psi$, $\phi \otimes \psi$, $\phi \oplus \psi$, $\exists x\phi$, $\forall x\phi$, $!x\phi$.

In particular, a dependence formula is a team formula. Note, however, that the semantics of team logic give different meaning to dependence formulas than the semantics of dependence logic. In order to preserve their meaning in the semantics of team logic, dependence formulas must undergo a simple translation. We will return to this in Chapter 5.

We call formulas built according to rules 1, 2 and 3 *atomic formulas*. A formula built according to rule 4 is a *compound formula*. A formula of the form 3 is called a *D-formula*. D stands for either dependence, as in "u depends on each t_i", or determination, as in "t_i together determine u".

A formula $\phi \in \mathrm{TL}$ is said to be in *strict negation normal form* if negation ($\neg$) appears only in front of atomic formulas of the form $Rt_1 \dots t_n$. Strict negation normal form poses no restrictions for strong negation ($\sim$).

The concept of team is as in dependence logic.

Definition 2.5.1. Let $\phi \in \mathrm{TL}$ be in strict negation normal form, let $\mathcal{M}$ be a model in the same language as ϕ, and let X be a team with $\mathrm{Dom}(X) \supseteq \mathrm{FV}(\phi)$. We define *satisfaction* of ϕ in $\mathcal{M}$ by X, denoted $\mathcal{M}, X \models^{\mathrm{TL}} \phi$, or simply $\mathcal{M}, X \models \phi$ or $X \models \phi$, as in Figure 2.5.

We do not define satisfaction for formulas that are not in strict negation normal form. It is not clear how this should be done for all the cases, in particular for formulas of the form $\neg{\sim}\phi$.

$$
\begin{array}{lcl}
\mathcal{M}, X \models \bot & & \text{never} \\
\mathcal{M}, X \models \top & & \text{always} \\
\mathcal{M}, X \models \mathbf{0} & \iff & X = \emptyset \\
\mathcal{M}, X \models \mathbf{1} & \iff & X \neq \emptyset \\
\mathcal{M}, X \models Rt_1 \ldots t_n & \iff & X \subseteq \big\{ s \in X^{\mathcal{M}}_{\mathrm{Dom}(X)} : \mathcal{M}, s \models Rt_1 \ldots t_n \big\} \\
\mathcal{M}, X \models \neg Rt_1 \ldots t_n & \iff & X \subseteq \big\{ s \in X^{\mathcal{M}}_{\mathrm{Dom}(X)} : \mathcal{M}, s \models \neg Rt_1 \ldots t_n \big\} \\
\mathcal{M}, X \models (t_1 \ldots t_n) \rightsquigarrow u & \iff & \text{there is } f \colon M^n \to M \text{ s.t.} \\
 & & \text{for all } s \in X \colon s(u) = f\big(s(t_1), \ldots, s(t_n)\big) \\
\mathcal{M}, X \models {\sim}\phi & \iff & \mathcal{M}, X \not\models \phi \\
\mathcal{M}, X \models \phi \vee \psi & \iff & \mathcal{M}, X \models \phi \text{ or } \mathcal{M}, X \models \psi \\
\mathcal{M}, X \models \phi \wedge \psi & \iff & \mathcal{M}, X \models \phi \text{ and } \mathcal{M}, X \models \psi \\
\mathcal{M}, X \models \phi \otimes \psi & \iff & \text{there is } Y, Z \subseteq X \text{ s.t. } Y \cup Z = X \text{ and} \\
 & & \mathcal{M}, Y \models \phi \text{ and } \mathcal{M}, Z \models \psi \\
\mathcal{M}, X \models \phi \oplus \psi & \iff & \text{for all } Y, Z \subseteq X, \text{ if } Y \cup Z = X \text{ then} \\
 & & \mathcal{M}, Y \models \phi \text{ or } \mathcal{M}, Z \models \psi \\
\mathcal{M}, X \models \exists x \phi & \iff & \text{there is } F \colon X \restriction \mathrm{FV}(\phi) \to M \text{ s.t.} \\
 & & \mathcal{M}, X(x \mapsto F) \models \phi \\
\mathcal{M}, X \models \forall x \phi & \iff & \text{for all } F \colon X \restriction \mathrm{FV}(\phi) \to M, \\
 & & \mathcal{M}, X(x \mapsto F) \models \phi \\
\mathcal{M}, X \models !x \phi & \iff & \mathcal{M}, X(x \mapsto M) \models \phi
\end{array}
$$

Figure 2.5: Semantics of team logic

A team sentence ϕ is *true* in a model $\mathcal{M}$, denoted $\mathcal{M} \models \phi$, if ϕ is satisfied in $\mathcal{M}$ by the full team. Otherwise ϕ is *false* in $\mathcal{M}$.[6] There are no other truth values.

Readers who are more familiar with independence friendly logic and Hintikka's game theoretic semantics than team logic should note that the symbol that is commonly used as *game negation* in independence friendly logic, $\sim$, is used as classical, contradictory negation in team logic, and I call it by the name strong negation. Similarly, the symbol that is commonly used as classical negation in independence friendly logic, $\neg$, is used to denote the weaker kind of negation in team logic, and I simply call it negation.

The *semantic game for team logic*, $\Game^{\mathrm{TL}}(\mathcal{M}, \phi)$, for a model $\mathcal{M}$ and sentence $\phi \in \mathrm{TL}$ is a game played by two players, $\mathbf{I}$ and $\mathbf{II}$. Every play ends in one of the players winning and the other losing. A *position* in $\Game^{\mathrm{TL}}(\mathcal{M}, \phi)$ is a triple (ψ, X, α), where ψ is a subformula of ϕ, X is a team, and $\alpha \in \{\mathbf{I}, \mathbf{II}\}$ denotes a player. We write $\mathbf{I}^* := \mathbf{II}$ and $\mathbf{II}^* := \mathbf{I}$ for the opponent of a player. The initial position in $\Game^{\mathrm{TL}}(\mathcal{M}, \phi)$ is $(\phi, \{\emptyset\}, \mathbf{II})$. The rules of the game we define based on what ψ is in the current position (ψ, X, α), as follows.

Case $\top$. Player α wins.

Case $\bot$. Player α^* wins.

Case 0. Player α wins if $X = \emptyset$.

Case 1. Player α wins if $X \neq \emptyset$.

Case $Rt_1 \ldots t_n$. Player α wins if $\mathcal{M}, s \models \psi$ for all $s \in X$.

Case $\neg Rt_1 \ldots t_n$. Player α wins if $\mathcal{M}, s \models \psi$ for all $s \in X$.

Case $(t_1 \ldots t_n) \rightsquigarrow u$. Player α wins if there is a function $f \colon M^n \to M$ such that $s(u) = f\big(s(t_1), \ldots, s(t_n)\big)$ for all $s \in X$.

Case $\sim\theta$. Nobody chooses anything. The game continues from position (θ, X, α^*).

Case $\theta_1 \vee \theta_2$. Player α chooses $i \in \{1, 2\}$. The game continues from position (θ_i, X, α).

Case $\theta_1 \wedge \theta_2$. Player α^* chooses $i \in \{1, 2\}$. The game continues from position (θ_i, X, α).

[6] It is equivalent to define that ϕ is true in $\mathcal{M}$ if $\mathcal{M}, \{\emptyset\} \models \phi$.

Case $\theta_1 \otimes \theta_2$. Player α chooses a pair of teams (Y_1, Y_2) such that $X = Y_1 \cup Y_2$. The game proceeds to position $(\psi, (Y_1, Y_2), \alpha)$. Then player α^* chooses $i \in \{1, 2\}$, and the game continues from position (θ_i, Y_i, α).

Case $\theta_1 \oplus \theta_2$. Player α^* chooses a pair of teams (Y_1, Y_2) such that $X = Y_1 \cup Y_2$. The game proceeds to position $(\psi, (Y_1, Y_2), \alpha)$. Then player α chooses $i \in \{1, 2\}$, and the game continues from position (θ_i, Y_i, α).

Case $\exists x\theta$. Player α chooses a function $F\colon X \to M$. The game continues from position $(\theta, X(x \mapsto F), \alpha)$.

Case $\forall x\theta$. Player α^* chooses a function $F\colon X \to M$. The game continues from position $(\theta, X(x \mapsto F), \alpha)$.

Case $!x\theta$. Nobody chooses anything. The game continues from position $(\theta, X(x \mapsto M), \alpha)$.

Note that both players make a move in case of formulas of the forms $\theta_1 \otimes \theta_2$ and $\theta_1 \oplus \theta_2$.

A *strategy* of player α in $\Game^{\mathrm{TL}}(\mathcal{M}, \phi)$ is a function τ defined on game positions such that the following conditions hold:

1. $\tau(\theta_1 \vee \theta_2, X, \alpha) \in \{1, 2\}$;
2. $\tau(\theta_1 \wedge \theta_2, X, \alpha^*) \in \{1, 2\}$;
3. $\tau(\theta_1 \otimes \theta_2, X, \alpha) = (Y_1, Y_2)$ such that $X = Y_1 \cup Y_2$;
4. $\tau(\theta_1 \otimes \theta_2, (Y_1, Y_2), \alpha^*) \in \{1, 2\}$;
5. $\tau(\theta_1 \oplus \theta_2, X, \alpha^*) = (Y_1, Y_2)$ such that $X = Y_1 \cup Y_2$;
6. $\tau(\theta_1 \oplus \theta_2, (Y_1, Y_2), \alpha) \in \{1, 2\}$;
7. $\tau(\exists x_n\theta, X, \alpha)$ is a function $F\colon X \to M$;
8. $\tau(\forall x_n\theta, X, \alpha^*)$ is a function $F\colon X \to M$.

Player α *plays by* the strategy τ in a play of the semantic game if his or her choices are exactly the ones that τ gives in each position that is reached during the play. Player α's strategy τ is a *winning strategy* if α wins all the plays in which he or she plays by τ. If α has a winning strategy in $\Game^{\mathrm{TL}}(\mathcal{M}, \phi)$, we denote it by $\alpha \uparrow \Game^{\mathrm{TL}}(\mathcal{M}, \phi)$.

Theorem 2.5.2. *Let $\mathcal{M}$ be a model and let $\phi \in \mathrm{TL}$ be a sentence. Then $\mathcal{M} \models \phi$ if and only if $\mathbf{II} \uparrow \Game^{\mathrm{TL}}(\mathcal{M}, \phi)$.*

Proof. Assume $\mathcal{M} \models \phi$. We will define a strategy τ for $\mathbf{II}$ in $\Game^{\mathrm{TL}}(\mathcal{M}, \phi)$ and show that it is a winning strategy. The strategy τ is defined by maintaining the condition

$$\text{if } \alpha = \mathbf{II} \text{ then } \mathcal{M}, X \models \psi, \text{ and if } \alpha = \mathbf{I} \text{ then } \mathcal{M}, X \not\models \psi \tag{2.2}$$

for positions $P = (\psi, X, \alpha)$. In other words, if (2.2) holds and player $\mathbf{II}$ is to choose, then $\tau(P)$ is a choice for player $\mathbf{II}$ such that in the next resulting position (2.2) holds again. Notice that this condition deliberately skips the special half-move position of the tensor and sum moves. We now perform an induction on subformulas ψ of ϕ which will show that a τ with the above definition really exists. For each ψ we inspect an arbitrary position (ψ, X, α). As the base case we see that the initial position of $\Game^{\mathrm{TL}}(\mathcal{M}, \phi)$ satisfies (2.2). Our induction hypothesis is that (2.2) holds at position (ψ, X, α).

Case $\sim\theta$**.** The next position is (θ, X, α^*) and (2.2) still holds.

Case $\theta_1 \vee \theta_2$**.** If $\alpha = \mathbf{II}$, then $\mathcal{M}, X \models \theta_1 \vee \theta_2$ by the induction hypothesis, whence $\mathcal{M}, X \models \theta_i$ for some $i \in \{1, 2\}$. When player $\mathbf{II}$ chooses this i, (2.2) holds at the resulting position.

If $\alpha = \mathbf{I}$, then $\mathcal{M}, X \not\models \theta_1 \vee \theta_2$ by the induction hypothesis, whence $\mathcal{M}, X \not\models \theta_i$ for both $i \in \{1, 2\}$. Thus, whatever player $\mathbf{I}$ chooses, (2.2) still holds at the resulting position.

Case $\theta_1 \wedge \theta_2$**.** If $\alpha = \mathbf{II}$, then $\mathcal{M}, X \models \theta_1 \wedge \theta_2$ by the induction hypothesis, whence $\mathcal{M}, X \models \theta_i$ for both $i \in \{1, 2\}$. Thus, whatever player $\mathbf{I}$ chooses, (2.2) still holds at the resulting position.

If $\alpha = \mathbf{I}$, then $\mathcal{M}, X \not\models \theta_1 \wedge \theta_2$ by the induction hypothesis, whence $\mathcal{M}, X \not\models \theta_i$ for some $i \in \{1, 2\}$. When player $\mathbf{II}$ chooses this i, (2.2) holds at the resulting position.

Case $\theta_1 \otimes \theta_2$**.** If $\alpha = \mathbf{II}$, then $\mathcal{M}, X \models \theta_1 \otimes \theta_2$ by the induction hypothesis, whence there are Y_1, Y_2 such that $X = Y_1 \cup Y_2$ and $\mathcal{M}, Y_i \models \theta_i$ for both $i \in \{1, 2\}$. When player $\mathbf{II}$ chooses (Y_1, Y_2), whatever player $\mathbf{I}$ chooses, (2.2) still holds at the resulting position.

If $\alpha = \mathbf{I}$, then $\mathcal{M}, X \not\models \theta_1 \otimes \theta_2$ by the induction hypothesis, whence for all Y_1, Y_2, if $X = Y_1 \cup Y_2$, then $\mathcal{M}, Y_i \not\models \theta_i$ for some $i \in \{1, 2\}$. Whichever (Y_1, Y_2) player $\mathbf{I}$ chooses, player $\mathbf{II}$ can choose the corresponding i, and then (2.2) still holds at the resulting position.

Case $\theta_1 \oplus \theta_2$**.** If $\alpha = \mathbf{II}$, then $\mathcal{M}, X \models \theta_1 \oplus \theta_2$ by the induction hypothesis, whence for all Y_1, Y_2, if $X = Y_1 \cup Y_2$, then $\mathcal{M}, Y_i \models \theta_i$ for some $i \in$

$\{1,2\}$. Whichever (Y_1, Y_2) player **I** chooses, player **II** can choose the corresponding i, and then (2.2) still holds at the resulting position.

If $\alpha = \mathbf{I}$, then $\mathcal{M}, X \not\models \theta_1 \oplus \theta_2$ by the induction hypothesis, whence there are Y_1, Y_2 such that $X = Y_1 \cup Y_2$ and $\mathcal{M}, Y_i \not\models \theta_i$ for both $i \in \{1,2\}$. When player **II** chooses (Y_1, Y_2), whatever player **I** chooses, (2.2) still holds at the resulting position.

Case $\exists x\theta$**.** If $\alpha = \mathbf{II}$, then $\mathcal{M}, X \models \exists x\theta$ by the induction hypothesis, whence there is some function $F\colon X \to M$ such that $\mathcal{M}, X(x \mapsto F) \models \theta$. When player **II** chooses this F, (2.2) still holds at the resulting position.

If $\alpha = \mathbf{I}$, then $\mathcal{M}, X \not\models \exists x\theta$ by the induction hypothesis, whence for all functions $F\colon X \to M$ we have $\mathcal{M}, X(x \mapsto F) \not\models \theta$. Thus, whatever player **I** chooses, (2.2) still holds at the resulting position.

Case $\forall x\theta$**.** If $\alpha = \mathbf{II}$, then $\mathcal{M}, X \models \forall x\theta$ by the induction hypothesis, whence $\mathcal{M}, X(x \mapsto F) \models \theta$ for all functions $F\colon X \to M$. Thus, whatever player **I** chooses, (2.2) still holds at the resulting position.

If $\alpha = \mathbf{I}$, then $\mathcal{M}, X \not\models \forall x\theta$ by the induction hypothesis, whence there is some function $F\colon X \to M$ such that $\mathcal{M}, X(x \mapsto F) \not\models \theta$. When player **II** chooses this F, (2.2) still holds at the resulting position.

Case $!x\theta$**.** The next position is $(\theta, X(x \mapsto M), \alpha)$ and (2.2) still holds.

We have now shown that τ exists. When player **II** plays by τ and the game ends at an atomic formula, player **II** wins the play by (2.2). Thus τ is a winning strategy.

For the other direction, assume that τ is a winning strategy for player **II** in $\eth^{\mathrm{TL}}(\mathcal{M}, \phi)$. We want to show that $\mathcal{M} \models \phi$. We prove this by proving by induction on subformulas ψ of ϕ the claim that when player **II** plays by τ and the game reaches position $P = (\psi, X, \alpha)$, then (2.2) holds for the position. The induction hypothesis is that the claim holds for positions $P' := (\theta, Y, \alpha')$, where θ is an immediate subformula of ψ and P' can be reached when player **II** plays by τ.

Case atomic. The claim holds by definition.

Case $\sim\theta$**.** The induction hypothesis applied to (θ, X, α^*) gives us (2.2) for positions $(\sim\theta, X, \alpha)$.

Case $\theta_1 \vee \theta_2$**.** If $\alpha = \mathbf{II}$, the induction hypothesis applied to $(\theta_{\tau(P)}, X, \mathbf{II})$ yields $\mathcal{M}, X \models \theta_{\tau(P)}$, whence $\mathcal{M}, X \models \psi$. If $\alpha = \mathbf{I}$ instead, we apply the induction hypothesis to $(\theta_i, X, \mathbf{I})$ and get $\mathcal{M}, X \not\models \theta_i$ for both $i \in \{1,2\}$, whence $\mathcal{M}, X \not\models \psi$.

Case $\theta_1 \wedge \theta_2$. If $\alpha = \mathbf{II}$, the induction hypothesis applied to $(\theta_i, X, \mathbf{II})$ yields $\mathcal{M}, X \models \theta_i$ for both $i \in \{1, 2\}$, whence $\mathcal{M}, X \models \psi$. If $\alpha = \mathbf{I}$ instead, we apply the induction hypothesis to $(\theta_{\tau(P)}, X, \mathbf{I})$ and get $\mathcal{M}, X \not\models \theta_{\tau(P)}$, whence $\mathcal{M}, X \not\models \psi$.

Case $\theta_1 \otimes \theta_2$. If $\alpha = \mathbf{II}$, the induction hypothesis applied to $(\theta_i, \tau(P), \mathbf{II})$ yields $\mathcal{M}, Y_i \models \theta_i$ for both $i \in \{1, 2\}$, where $\tau(P) = (Y_1, Y_2)$. Because $Y_1 \cup Y_2 = X$, we get $\mathcal{M}, X \models \psi$. If $\alpha = \mathbf{I}$ instead, we apply the induction hypothesis to $(\theta_{\tau(P')}, Y_{\tau(P')}, \mathbf{I})$ for all (Y_1, Y_2) with $Y_1 \cup Y_2 = X$, where $P' := (\psi, (Y_1, Y_2), \alpha)$. We get $\mathcal{M}, Y_{\tau(P')} \not\models \theta_{\tau(P')}$ for all (Y_1, Y_2) with $Y_1 \cup Y_2 = X$. Thus $\mathcal{M}, X \not\models \psi$.

Case $\theta_1 \oplus \theta_2$. If $\alpha = \mathbf{II}$, then we apply the induction hypothesis to position $(\theta_{\tau(P')}, Y_{\tau(P')}, \mathbf{II})$ for all (Y_1, Y_2) with $Y_1 \cup Y_2 = X$, where $P' := (\psi, (Y_1, Y_2), \alpha)$. We get $\mathcal{M}, Y_{\tau(P')} \models \theta_{\tau(P')}$ for all (Y_1, Y_2) with $Y_1 \cup Y_2 = X$. Thus $\mathcal{M}, X \models \psi$. If $\alpha = \mathbf{I}$ instead, the induction hypothesis applied to $(\theta_i, \tau(P), \mathbf{I})$ yields $\mathcal{M}, Y_i \not\models \theta_i$ for both $i \in \{1, 2\}$, where $\tau(P) = (Y_1, Y_2)$. Because $Y_1 \cup Y_2 = X$, we get $\mathcal{M}, X \not\models \psi$.

Case $\exists x\theta$. If $\alpha = \mathbf{II}$, then the induction hypothesis applied to position $(\theta, X(x \mapsto \tau(P)), \mathbf{II})$ yields $\mathcal{M}, X(x \mapsto \tau(P)) \models \theta$, whence $\mathcal{M}, X \models \psi$. If $\alpha = \mathbf{I}$ instead, we apply the induction hypothesis to $(\theta, X(x \mapsto F), \mathbf{I})$ for all functions $F\colon X \to M$. We get $\mathcal{M}, X(x \mapsto F) \not\models \theta$ for all $F\colon X \to M$, whence $\mathcal{M}, X \not\models \psi$.

Case $\forall x\theta$. If $\alpha = \mathbf{II}$, we apply the induction hypothesis to $(\theta, X(x \mapsto F), \mathbf{II})$ for all functions $F\colon X \to M$. We get $\mathcal{M}, X(x \mapsto F) \models \theta$ for all $F\colon X \to M$, whence $\mathcal{M}, X \models \psi$. If $\alpha = \mathbf{I}$ instead, the induction hypothesis applied to $(\theta, X(x \mapsto \tau(P)), \mathbf{I})$ yields $\mathcal{M}, X(x \mapsto \tau(P)) \not\models \theta$, whence $\mathcal{M}, X \not\models \psi$.

Case $!x\theta$. If $\alpha = \mathbf{I}$, the induction hypothesis applied to $(\theta, X(x \mapsto M), \mathbf{I})$ gives us $\mathcal{M}, X(x \mapsto M) \models \theta$, whence $\mathcal{M}, X \models \psi$. If $\alpha = \mathbf{II}$, the induction hypothesis applied to $(\theta, X(x \mapsto M), \mathbf{II})$ gives us $\mathcal{M}, X(x \mapsto M) \not\models \theta$, whence $\mathcal{M}, X \not\models \psi$.

Now the induction is complete. Applying the shown claim for the initial position $(\phi, \{\emptyset\}, \mathbf{II})$ we get $\mathcal{M} \models \phi$. □

We can obtain the semantic game for dependence logic, denoted $\eth^{\mathrm{FOD}}$, by limiting the semantic game for team logic to dependence formulas and changing notation so that formulas of the form $\theta_1 \vee \theta_2$ are played like $\theta_1 \otimes \theta_2$ and formulas of the form $\forall x\theta$ are played like $!x\theta$. The semantic game for dependence logic was first presented by Väänänen [19, Definition 5.5].

Chapter 3

Swapping Quantifiers

In this chapter I study the question under which conditions can the places of two consequtive quantifiers be swapped while preserving the meaning of the formula. We will see a natural characterising condition based on D-formulas; two quantifiers can be swapped when the variables they quantify do not depend on each other.

Quantifier swapping is well understood in first order logic; swapping places of two consequtive existential quantifiers or two consequtive universal quantifiers has no effect on the interpretation of a formula. On the contrary, swapping a universal and an existential quantifier changes the formula essentially. A manifestation of this is that one common measure of formula complexity in first order logic is to count the number of alternating quantifier blocks—each step from a sequence of existential quantifiers to a sequence of universal quantifiers, and vice versa, is a step to the next level of complexity.

In second order logic, quantifier swapping is possible but usually involves changing the arity of the quantified relation or function in order to preserve the interpretation of the formula.

Also Caicedo, Dechesne and Janssen investigate interchange of quantifiers among other quantifier rules [4].

3.1 Definitions

We shall start with a couple of definitions. Let X be a team, let $V \subseteq \mathrm{Dom}(X)$, and let $F\colon X \to M$. The *deep restriction* of F to V is the function $F{\restriction\!\restriction}V\colon X{\restriction}V \to M$ that maps $(F{\restriction\!\restriction}V)(s{\restriction}V) = F(s)$ for all $s \in X$. Of course deep restriction cannot always be well defined because it has to map elements like the original function but using less information.

Lemma 3.1.1. *The following properties are equivalent.*

1. *Deep restriction $F \Vert V$ is well defined.*

2. *If $s, s' \in X$ and $s \neq s'$ and $F(s) \neq F(s')$ then $s \restriction V \neq s' \restriction V$.*

3. *If $s, s' \in X$ and $s \neq s'$ and $s \restriction V = s' \restriction V$ then $F(s) = F(s')$.*

Proof. Elementary. □

We write $X(xy \mapsto MF)$ as shorthand for $X(x \mapsto M)(y \mapsto F)$, and likewise for other similar cases.

I adopt the convention in this chapter that x and y are not the same variable. By swapping quantifiers I mean modifying a team formula $Q_1 x Q_2 y \phi$, where Q_1 and Q_2 are quantifiers (one of $\exists$, $\forall$ or !), into a logically equivalent formula $Q_2 y Q_1 x \phi$. As we set to investigate under which conditions can two consequtive quantifiers be swapped in a formula of team logic, we first need to make clear certain technicalities about the operations the semantics of the quantifiers perform on teams.

Let X be a team and let V be a set of variables. We say that y is *determined by V in X* if $X \models (V) \rightsquigarrow y$. We say that y is *independent of x in X* if y is determined by some set V of variables such that $x, y \notin V$.

Lemma 3.1.2. *Assume that $x, y \notin \mathrm{Dom}(X)$. Then we have $X(xy \mapsto \alpha_0\alpha_1) = X(yx \mapsto \alpha_1\alpha_0)$, where both α_i can be the universe, $\alpha_i = M$, or a function, $\alpha_i \colon X \restriction V \to M$ for some $V \subseteq \mathrm{Dom}(X)$, including any combination of these.*

3.2 Swapping Quantifiers in Team Logic

Lemma 3.1.2 gives us a few straightforward results.

Lemma 3.2.1. *For all $\phi \in \mathrm{TL}$ we have $\exists y ! x \phi \Rightarrow ! x \exists y \phi$.*

Proof. Assuming $X \models \exists y ! x \phi$ we get $X(yx \mapsto FM) \models \phi$ for some $F \colon X \to M$. Lemma 3.1.2 gives $X(xy \mapsto MF) \models \phi$, from which we get $X \models ! x \exists y \phi$. □

Lemma 3.2.2. *For all $\phi \in \mathrm{TL}$ we have $! x \forall y \phi \Rightarrow \forall y ! x \phi$.*

Proof. By Lemma 3.2.1 and duality we have

$$! x \forall y \phi \equiv {\sim} ! x \exists y {\sim} \phi \Rightarrow {\sim} \exists y ! x {\sim} \phi \equiv \forall y ! x \phi.$$ □

Note that even when $x \neq y$ and $x, y \notin \mathrm{Dom}(X)$, we do not have for example $X(xy \mapsto FG) = X(yx \mapsto GF)$ for $F\colon X \to M$ and $G\colon X(x \mapsto F) \to M$. This is so because of technical reasons, namely $X(y \mapsto G)$ is meaningless because G is not defined on any restricted team $X{\restriction}V$. However, in this case we can write $X(xy \mapsto FG) = X(xy \mapsto FG') = X(yx \mapsto G'F)$, where $G' := G{\upharpoonright\!\upharpoonright}\mathrm{Dom}(X)$, and get what we want. The deep restriction is well defined in this case. To see this from Lemma 3.1.1, assume $s_1, s_2 \in X(x \mapsto F)$, $s_1 \neq s_2$ and $G(s_1) \neq G(s_2)$. Then we can write $s_i = z_i(x \mapsto F(z_i))$ for some $z_1, z_2 \in X$. In fact $z_i = s_i{\restriction}\mathrm{Dom}(X)$. From $s_1 \neq s_2$ we get that either $z_1 \neq z_2$, as we want, or $F(z_1) \neq F(z_2)$. From the latter we get $z_1 \neq z_2$. In general we have the following lemma.

Lemma 3.2.3. *Let X be a team, $F\colon X \to M$, $V \subseteq \mathrm{Dom}(X)$ and $y \notin \mathrm{Dom}(X)$ a variable. If $F' = F{\upharpoonright\!\upharpoonright}V$ is well defined then $X(y \mapsto F) = X(y \mapsto F')$.*

Proof. For each $s \in X$, because $F'(s) = F(s)$, we have $X(y \mapsto F) \ni s(y \mapsto F(s)) = s(y \mapsto F'(s)) \in X(y \mapsto F')$. □

We can now state a few more straightforward results.

Lemma 3.2.4. *For all $\phi \in \mathrm{TL}$,*

1. $!x!y\phi \equiv\, !y!x\phi$;
2. $\exists x\exists y\phi \equiv \exists y\exists x\phi$;
3. $\forall x\forall y\phi \equiv \forall y\forall x\phi$.

Proof. To see the first equivalence, we use Lemma 3.1.2 and get

$$X \models\, !x!y\phi \Rightarrow X(xy \mapsto MM) \models \phi \Rightarrow X(yx \mapsto MM) \models \phi \Rightarrow X \models\, !y!x\phi.$$

For the second equivalence we use Lemma 3.2.3 that we just proved. When $X \models \exists x\exists y\phi$, then $X(xy \mapsto FG) \models \phi$ for some $F\colon X \to M$ and $G\colon X(x \mapsto F) \to M$. For any $s, s' \in X(x \mapsto F)$, if $s{\restriction}\mathrm{Dom}(X) = s'{\restriction}\mathrm{Dom}(X)$, then $F(s) = F(s')$, so $s = s'$. By Lemma 3.1.1, $G' := G{\upharpoonright\!\upharpoonright}\mathrm{Dom}(X)$ is well defined, so we get $X(xy \mapsto FG) = X(yx \mapsto G'F)$, from which we get $X(yx \mapsto G'F) \models \phi$ and finally $X \models \exists y\exists x\phi$.

The third equivalence follows from the previous by duality:

$$\forall x\forall y\phi \equiv {\sim}\exists x\exists y{\sim}\phi \equiv {\sim}\exists y\exists x{\sim}\phi \equiv \forall y\forall x\phi. \qquad \square$$

Here is another example to illustrate that the switch from $X(xy \mapsto \alpha_0\alpha_1)$ to $X(yx \mapsto \alpha_1\alpha_0)$ cannot always be taken for granted. Assume again $x \neq y$ and $x, y \notin \mathrm{Dom}(X)$. We do not have $X(xy \mapsto MF) = X(yx \mapsto FM)$ for $F\colon X(x \mapsto M) \to M$. As above, the expression $X(y \mapsto F)$ is not defined. But worse than above, we cannot save the situation by choosing $F' := F{\upharpoonright\!\upharpoonright}\mathrm{Dom}(X)$. The reason is that in general the deep restriction is not well defined. For a concrete example, consider $X = \big\{\{x \mapsto 0\}\big\}$ and $M = \{0,1\}$. Then $X(y \mapsto M) = \big\{\{x \mapsto 0, y \mapsto 0\}, \{x \mapsto 0, y \mapsto 1\}\big\}$. If furthermore $F\colon X(x \mapsto M) \to M$ has $F(0,0) = 0$ and $F(0,1) = 1$, then we should but cannot have $F'(0) = 0$ and $F'(0) = 1$.

Lemma 3.2.5. *Assume x and y are variables, $y \notin \mathrm{Dom}(X)$, and let $F\colon X \to M$. Then y is independent of x in $X(y \mapsto F)$ if and only if $F{\upharpoonright\!\upharpoonright}V$ is well defined for some $V \subseteq \mathrm{Dom}(X) \setminus \{x\}$.*

Proof. Assume first that y is independent of x in $X(y \mapsto F)$. Then $X(y \mapsto F) \models (z_1, \ldots, z_n){\rightsquigarrow}y$ for some variables $z_1, \ldots, z_n$ that do not include x and y, whence we get a function $f\colon M^n \to M$ such that, for all $s \in X(y \mapsto F)$, $f(s(z_1), \ldots, s(z_n)) = s(y) = F(s)$. Notice that $X(y \mapsto F){\upharpoonright}V = X{\upharpoonright}V$. If $s, s' \in X$ such that $s \neq s'$ and $s{\upharpoonright}\{z_1, \ldots, z_n\} = s'{\upharpoonright}\{z_1, \ldots, z_n\}$, then $F(s) = f(s(z_1), \ldots, s(z_n)) = f(s'(z_1), \ldots, s'(z_n)) = F(s')$. Thus $F{\upharpoonright\!\upharpoonright}\{z_1, \ldots, z_n\}$ is well defined.

Assume then that $V \subseteq \mathrm{Dom}(X) \setminus \{x\}$ and $F' := F{\upharpoonright\!\upharpoonright}V$ is well defined. Then $x, y \notin V$. To see that V determines y in $X(y \mapsto F)$, let $s, s' \in X(y \mapsto F)$ with $s{\upharpoonright}V = s'{\upharpoonright}V$. By Lemma 3.2.3 we get $X(y \mapsto F) = X(y \mapsto F')$, so $s(y) = F'(s{\upharpoonright}V) = F'(s'{\upharpoonright}V) = s'(y)$. Thus y is independent of x in $X(y \mapsto F)$. □

Lemma 3.2.6. *Let $\phi \in \mathrm{TL}$ and let V be a set of variables with $x, y \notin V$. Then $!x\exists y\big((V){\rightsquigarrow}y \wedge \phi\big) \Rightarrow \exists y!x\big((V){\rightsquigarrow}y \wedge \phi\big)$.*

Proof. Let $\mathcal{M}$ be a model and X be a team. Assume $X \models !x\exists y((V){\rightsquigarrow}y \wedge \phi)$. Then $X(xy \mapsto MF) \models (V){\rightsquigarrow}y$ and $X(xy \mapsto MF) \models \phi$, where $F\colon X(x \mapsto M) \to M$. As y is independent of x in $X(xy \mapsto MF)$, we get by Lemma 3.2.5 that $F' := F{\upharpoonright\!\upharpoonright}V$ is well defined for some $V \subseteq \mathrm{Dom}(X(x \mapsto M)) \setminus \{x\}$. Then $V \subseteq \mathrm{Dom}(X)$. Thus $X(xy \mapsto MF) = X(xy \mapsto MF') = X(yx \mapsto F'M)$, using Lemma 3.1.2. We get $X \models \exists y!x((V){\rightsquigarrow}y \wedge \phi)$. □

Theorem 3.2.7. *For all $\phi \in \mathrm{TL}$,*

$$!x\exists y\phi \Rightarrow \exists y!x\phi \quad \textit{iff} \quad !x\exists y\phi \Rightarrow !x\exists y\big((V){\rightsquigarrow}y \wedge \phi\big) \quad \textit{for some } V \textit{ with } x, y \notin V. \tag{3.1}$$

Proof. First assume the right side of (3.1). Then we have by Lemma 3.2.6 that

$$!x\exists y\phi \Rightarrow\ !x\exists y\big((V)\rightsquigarrow y \wedge \phi\big) \Rightarrow \exists y!x\big((V)\rightsquigarrow y \wedge \phi\big) \Rightarrow \exists y!x\phi.$$

Assume then the left side of (3.1). Let X be a team. We may assume that $x, y \notin \mathrm{Dom}(X)$ because x and y are not free in the formulas in (3.1). From $X \models\ !x\exists y\phi$ we get $X \models \exists y!x\phi$, whence $X(yx \mapsto FM) \models \phi$ for some $F\colon X \to M$. Let $V = \mathrm{Dom}(X)$. Then $x, y \notin V$. We want to show $X(yx \mapsto FM) \models (V)\rightsquigarrow y$, so let $s, s' \in X(yx \mapsto FM)$ with $s\restriction V = s'\restriction V$. Then there are some $z, z' \in X$ such that $s = z(yx \mapsto FM)$ and $s' = z'(yx \mapsto FM)$, and in fact $z = s\restriction V = s'\restriction V = z'$. Now $s(y) = F(z) = F(z') = s'(y)$. Thus $X(yx \mapsto FM) \models (V)\rightsquigarrow y \wedge \phi$, so we get $\exists y!x\phi \Rightarrow \exists y!x\big((V)\rightsquigarrow y \wedge \phi\big) \Rightarrow\ !x\exists y\big((V)\rightsquigarrow y \wedge \phi\big)$. □

Note that when we assume $!x\exists y\phi \Rightarrow \exists y!x\phi$, we do not in general have $\phi \Rightarrow (V)\rightsquigarrow y \wedge \phi$ for any V with $x, y \notin V$, which would be a stronger claim than the one in Theorem 3.2.7. We get an easy counterexample by choosing ϕ to be $y = y$. Even if we require the quantifiers $!x$ and $\exists y$ not to be redundant, we get a counterexample by choosing ϕ to be $x = x \wedge y = y$.

There is an immediate consequence by duality.

Theorem 3.2.8. *For all $\phi \in \mathrm{TL}$,*

$$\forall y!x\phi \Rightarrow\ !x\forall y\phi \text{ if and only if } !x\forall y\big({\sim}(V)\rightsquigarrow y \vee \phi\big) \Rightarrow\ !x\forall y\phi$$
$$\text{for some } V \text{ with } x, y \notin V.$$

Proof. By Theorem 3.2.7 we get the following.

$$\begin{aligned}
&\forall y!x\phi \Rightarrow\ !x\forall y\phi\\
\text{iff }&{\sim}\exists y!x{\sim}\phi \Rightarrow {\sim}!x\exists y{\sim}\phi\\
\text{iff }&!x\exists y{\sim}\phi \Rightarrow \exists y!x{\sim}\phi\\
\text{iff }&!x\exists y{\sim}\phi \Rightarrow\ !x\exists y\big((V)\rightsquigarrow y \wedge {\sim}\phi\big) \text{ for some } V \text{ with } x, y \notin V.\\
\text{iff }&{\sim}!x\exists y\big((V)\rightsquigarrow y \wedge {\sim}\phi\big) \Rightarrow {\sim}!x\exists y{\sim}\phi \text{ for some } V \text{ with } x, y \notin V.\\
\text{iff }&!x\forall y\big({\sim}(V)\rightsquigarrow y \vee \phi\big) \Rightarrow\ !x\forall y\phi \text{ for some } V \text{ with } x, y \notin V.
\end{aligned}$$
□

By Lemma 3.2.1 and Lemma 3.2.2 we get the following corollaries of Theorem 3.2.7 and Theorem 3.2.8.

Corollary 3.2.9. *For all $\phi \in \mathrm{TL}$,*

1. $!x\exists y\phi \equiv \exists y!x\phi$ *iff* $!x\exists y\phi \Rightarrow\ !x\exists y\big((V)\rightsquigarrow y \wedge \phi\big)$ *for any* $x, y \notin V$

2. *$!x\forall y\phi \equiv \forall y!x\phi$ iff $!x\forall y\big(\sim(V)\rightsquigarrow y \vee \phi\big) \Rightarrow\, !x\forall y\phi$ for any $x, y \notin V$*

Corollary 3.2.10. *For all $\phi \in \mathrm{TL}$,*

1. *$!x\exists y\big((V)\rightsquigarrow y \wedge \phi\big) \equiv \exists y!x\big((V)\rightsquigarrow y \wedge \phi\big)$ if $x, y \notin V$*

2. *$!x\forall y\big(\sim(V)\rightsquigarrow y \vee \phi\big) \equiv \forall y!x\big(\sim(V)\rightsquigarrow y \vee \phi\big)$ if $x, y \notin V$*

Example 3.2.11. With Corollary 3.2.9 we can show for example that for any $\phi \in \mathrm{TL}$ we have the following equivalences:

$$\begin{aligned}
&!x_0\exists x_1!x_2\exists x_3\big((x_0)\rightsquigarrow x_1 \wedge (x_2)\rightsquigarrow x_3 \wedge \phi\big)\\
\equiv\ &!x_0\exists x_1!x_2\big((x_0)\rightsquigarrow x_1 \wedge \exists x_3\big((x_2)\rightsquigarrow x_3 \wedge \phi\big)\big)\\
\equiv\ &!x_0!x_2\exists x_1\big((x_0)\rightsquigarrow x_1 \wedge \exists x_3\big((x_2)\rightsquigarrow x_3 \wedge \phi\big)\big)\\
\equiv\ &!x_0!x_2\exists x_1\exists x_3\big((x_0)\rightsquigarrow x_1 \wedge (x_2)\rightsquigarrow x_3 \wedge \phi\big)\\
\equiv\ &!x_2!x_0\exists x_3\exists x_1\big((x_0)\rightsquigarrow x_1 \wedge (x_2)\rightsquigarrow x_3 \wedge \phi\big)\\
\equiv\ &!x_2!x_0\exists x_3\big((x_2)\rightsquigarrow x_3 \wedge \exists x_1\big((x_0)\rightsquigarrow x_1 \wedge \phi\big)\big)\\
\equiv\ &!x_2\exists x_3!x_0\big((x_2)\rightsquigarrow x_3 \wedge \exists x_1\big((x_0)\rightsquigarrow x_1 \wedge \phi\big)\big)\\
\equiv\ &!x_2\exists x_3!x_0\exists x_1\big((x_0)\rightsquigarrow x_1 \wedge (x_2)\rightsquigarrow x_3 \wedge \phi\big).
\end{aligned}$$

We have now investigated all quantifier swaps except the case of $\exists x$ and $\forall x$. It turns out that there is no general way to swap these quantifiers.

Let us investigate some team sentence $\exists x\forall y\phi$. If it is true in some model $\mathcal{M}$ with universe, say, $M = \{0, 1, 2\}$, then ϕ is satisfied in $\mathcal{M}$ by the following teams, where a is some fixed element in M.

x	y
a	0

x	y
a	1

x	y
a	2

On the other hand, if the team sentence $\forall y\exists x\phi$ is true in the same model $\mathcal{M}$, then ϕ is satisfied in $\mathcal{M}$ by the following teams.

x	y
a_0	0

x	y
a_1	1

x	y
a_2	2

Here $a_0, a_1, a_2 \in M$ but they are not necessarily the same element. Furthermore, we cannot remedy the situation and force these two sets of teams to be the same by using D-formulas. The effect of D-formulas is limited to each team separately. In fact we have the following equivalences.

$$\begin{aligned}
\exists x\forall y\phi &\equiv \exists x\forall y\big(\phi \wedge ()\rightsquigarrow x \wedge ()\rightsquigarrow y\big)\\
\forall y\exists x\phi &\equiv \forall y\exists x\big(\phi \wedge ()\rightsquigarrow x \wedge ()\rightsquigarrow y\big)
\end{aligned}$$

This shows that there is no hope of swapping $\exists$ and $\forall$ except in special cases.

Chapter 4

FO vs. FOD in Logical Equivalence

In this chapter I study the concept of equivalence of two models in first order logic and dependence logic. In particular, I work with the concept of semiequivalence of two models, which means that every sentence that is true in the first model is also true in the other model. I define an Ehrenfeucht-Fraïssé game that characterises semiequivalence of first order logic up to a given dependence rank. I also give an effective conversion of a winning strategy in the EF-game for first order logic into a winning strategy in this new EF-game.

4.1 Definitions

We call atomic and a negated atomic formulas *basic formulas*.

The *first order rank* of $\phi \in \mathrm{FO}$, denoted $\mathrm{rank}^{\mathrm{FO}}(\phi)$, is the number of nested quantifiers in ϕ. The *dependence rank* of $\psi \in \mathrm{FOD}$, denoted $\mathrm{rank}^{\mathrm{FOD}}(\psi)$, counts the number of nested quantifiers and disjunctions. The precise definitions are as in Table 4.1.

Denote by FO_n the set of first order formulas up to first order rank n, denote by FOD_n the set of dependence formulas up to dependence rank n, and denote by $\mathrm{FO}_n^{\mathrm{FOD}}$ the set of first order formulas up to dependence rank n.

Let $\mathcal{M}$ and $\mathcal{N}$ be models, and let L and R be logics (either first order logic or dependence logic). We say that $\mathcal{M}$ and $\mathcal{N}$ are *L-semiequivalent up to R-rank n*, denoted $\mathcal{M} \Rrightarrow^{L}_{R,n} \mathcal{N}$, when $\mathcal{M} \models \phi$ implies $\mathcal{N} \models \phi$ for all L-sentences ϕ with $\mathrm{rank}^{R}(\phi) \leq n$. We say that $\mathcal{M}$ and $\mathcal{N}$ are *L-equivalent up to R-rank n*, denoted $\mathcal{M} \equiv^{L}_{R,n} \mathcal{N}$, when $\mathcal{M} \Rrightarrow^{L}_{R,n} \mathcal{N}$ and $\mathcal{M} \Lleftarrow^{L}_{R,n} \mathcal{N}$. If $L = R$,

formula	first order rank	dependence rank
atomic	0	0
$\neg\phi$	$\mathrm{rank}^{\mathrm{FO}}(\phi)$	$\mathrm{rank}^{\mathrm{FOD}}(\phi)$
$\phi_0 \vee \phi_1$	$\max\{\mathrm{rank}^{\mathrm{FO}}(\phi_i) : i < 2\}$	$\max\{\mathrm{rank}^{\mathrm{FOD}}(\phi_i) : i < 2\} + 1$
$\phi_0 \wedge \phi_1$	$\max\{\mathrm{rank}^{\mathrm{FO}}(\phi_i) : i < 2\}$	$\max\{\mathrm{rank}^{\mathrm{FOD}}(\phi_i) : i < 2\}$
$\exists x\phi$	$\mathrm{rank}^{\mathrm{FO}}(\phi) + 1$	$\mathrm{rank}^{\mathrm{FOD}}(\phi) + 1$
$\forall x\phi$	$\mathrm{rank}^{\mathrm{FO}}(\phi) + 1$	$\mathrm{rank}^{\mathrm{FOD}}(\phi) + 1$

Table 4.1: First order and dependence ranks

we can simply write $\Rrightarrow_n^L$ instead of $\Rrightarrow_{R,n}^L$. Similarly for $\equiv_n^L$. This is normally the case, but with dependence logic and first order logic we also have the combination $L = \mathrm{FOD}$, $R = \mathrm{FO}$ because dependence rank is also defined for first order sentences. By $\Rrightarrow_R^L$ we mean $\Rrightarrow_{R,n}^L$ for all $n < \omega$. Similarly for $\equiv_R^L$.

Lemma 4.1.1. *For all $\phi \in \mathrm{FO}$, $\mathrm{rank}^{\mathrm{FO}}(\phi) \leq \mathrm{rank}^{\mathrm{FOD}}(\phi)$.*

Proof. Induction on ϕ. For basic ϕ we have $\mathrm{rank}^{\mathrm{FO}}(\phi) = 0 = \mathrm{rank}^{\mathrm{FOD}}(\phi)$. The induction hypothesis gives the following inequalities.

$$\begin{aligned}
\mathrm{rank}^{\mathrm{FO}}(\phi_0 \vee \phi_1) = {} & \max\{\mathrm{rank}^{\mathrm{FO}}(\phi_i) : i < 2\} \leq \max\{\mathrm{rank}^{\mathrm{FOD}}(\phi_i) : i < 2\} < \\
& \max\{\mathrm{rank}^{\mathrm{FOD}}(\phi_i) : i < 2\} + 1 = \mathrm{rank}^{\mathrm{FOD}}(\phi_0 \vee \phi_1) \\
\mathrm{rank}^{\mathrm{FO}}(\phi_0 \wedge \phi_1) = {} & \max\{\mathrm{rank}^{\mathrm{FO}}(\phi_i) : i < 2\} \leq \max\{\mathrm{rank}^{\mathrm{FOD}}(\phi_i) : i < 2\} = \\
& \mathrm{rank}^{\mathrm{FOD}}(\phi_0 \wedge \phi_1) \\
\mathrm{rank}^{\mathrm{FO}}(\exists x\phi) = {} & \mathrm{rank}^{\mathrm{FO}}(\phi) + 1 \leq \mathrm{rank}^{\mathrm{FOD}}(\phi) + 1 = \mathrm{rank}^{\mathrm{FOD}}(\exists x\phi) \\
\mathrm{rank}^{\mathrm{FO}}(\forall x\phi) = {} & \mathrm{rank}^{\mathrm{FO}}(\phi) + 1 \leq \mathrm{rank}^{\mathrm{FOD}}(\phi) + 1 = \mathrm{rank}^{\mathrm{FOD}}(\forall x\phi)
\end{aligned}$$

□

I give definitions in my flavour for the known EF-games for first order logic and dependence logic, along with their known characterisations in terms of first order equivalence and dependence semiequivalence. The EF-game for dependence logic was originally presented by Väänänen [19].

When $\mathrm{EF}^L(\mathcal{M}, \mathcal{N})$ is some EF-game over the models $\mathcal{M}$ and $\mathcal{N}$, and P is a position in the EF-game, I denote by $\mathbf{II} \uparrow \mathrm{EF}^L(\mathcal{M}, \mathcal{N})$ @ P the phrase "player **II** has a winning strategy in $\mathrm{EF}^L(\mathcal{M}, \mathcal{N})$ at position P".

Definition 4.1.2. Let $\mathcal{M}$ and $\mathcal{N}$ be models, and $n < \omega$. The *n-move EF-game for* FO *over $\mathcal{M}$ and $\mathcal{N}$*, $\mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N})$, is played by players **I** and **II**. A position in the game is $(s, s')^k$, where s and s' are assignments such that

$\mathrm{Dom}(s) = \mathrm{Dom}(s')$, k is the number of moves left in the game, assignment s maps to model $\mathcal{M}$, and assignment s' maps to model $\mathcal{N}$.

The initial position in the game is $(\emptyset, \emptyset)^n$. If the game is in position $(s, s')^{k+1}$, then player **I** chooses an element $a \in M$ (or an element $b \in N$) and a variable x, after which player **II** chooses an element $b \in N$ (or an element $a \in M$). Then the game continues from position $\big(s(x \mapsto a), s'(x \mapsto b)\big)^k$.

If the game is in position $(s, s')^0$, then player **II** wins if it holds for all basic $\phi \in \mathrm{FO}$ with $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(s)$ that $\mathcal{M}, s \models \phi$ implies $\mathcal{N}, s' \models \phi$. Otherwise player **I** wins.

Definition 4.1.3. Let $\mathcal{M}$ and $\mathcal{N}$ be models, and $n < \omega$. The *n-move EF-game for* FOD *over $\mathcal{M}$ and $\mathcal{N}$*, $\mathrm{EF}_n^{\mathrm{FOD}}(\mathcal{M}, \mathcal{N})$, is played by players **I** and **II**. A position in the game is $(X, Y)^k$, where k is the number of moves left in the game, and X and Y are teams such that $\mathrm{Dom}(X) = \mathrm{Dom}(Y)$, team X maps to model $\mathcal{M}$ and team Y maps to model $\mathcal{N}$.

Initial position in the game is $(\{\emptyset\}, \{\emptyset\})^n$. If the game is in position $(X, Y)^{k+1}$, then player **I** can choose one of the following moves.

$\vee$-move. Player **I** splits $X = X_0 \cup X_1$. Then player **II** splits $Y = Y_0 \cup Y_1$. Finally player **I** chooses the game to continue either from position $(X_0, Y_0)^k$ or from position $(X_1, Y_1)^k$.

$\exists$-move. Player **I** chooses some function $F\colon X \to M$ and variable x. Then player **II** chooses some function $G\colon Y \to N$. The game then continues from position $\big(X(x \mapsto F), Y(x \mapsto G)\big)^k$.

$\forall$-move. Player **I** chooses some variable x. The game continues from position $\big(X(x \mapsto M), Y(x \mapsto N)\big)^k$.

If the game is in position $(X, Y)^0$, then player **II** wins if for all $\phi \in \mathrm{FOD}$ such that $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(X)$ it holds that $\mathcal{M}, X \models \phi$ implies $\mathcal{N}, Y \models \phi$. Otherwise player **I** wins.

Theorem 4.1.4. *Let $\mathcal{M}$ and $\mathcal{N}$ be models and $n < \omega$. Then player* **II** *has a winning strategy in* $\mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N})$ *if and only if* $\mathcal{M} \equiv_n^{\mathrm{FO}} \mathcal{N}$.

Theorem 4.1.5. *Let $\mathcal{M}$ and $\mathcal{N}$ be models and $n < \omega$. Then player* **II** *has a winning strategy in* $\mathrm{EF}_n^{\mathrm{FOD}}(\mathcal{M}, \mathcal{N})$ *if and only if* $\mathcal{M} \Rrightarrow_n^{\mathrm{FOD}} \mathcal{N}$.

Proof. See [19, Theorem 6.44]. □

4.2 Comparing Semiequivalences

I show some easy results to provide a context for further discussion.

Fact 4.2.1. *If* $\mathcal{M} \Rrightarrow_n^{\mathrm{FO}} \mathcal{N}$ *then* $\mathcal{M} \equiv_n^{\mathrm{FO}} \mathcal{N}$.

Proof. For any $\phi \in \mathrm{FO}$ with $\mathrm{rank}^{\mathrm{FO}}(\phi) \leq n$ we have $\mathrm{rank}^{\mathrm{FO}}(\neg\phi) \leq n$, and therefore

$$\mathcal{N} \models \phi \Rightarrow \mathcal{N} \not\models \neg\phi \Rightarrow \mathcal{M} \not\models \neg\phi \Rightarrow \mathcal{M} \models \phi. \qquad \square$$

Corollary 4.2.2. *If* $\mathcal{M} \Rrightarrow^{\mathrm{FO}} \mathcal{N}$ *then* $\mathcal{M} \equiv^{\mathrm{FO}} \mathcal{N}$.

Fact 4.2.3. $\mathcal{M} \Rrightarrow^{\mathrm{FOD}} \mathcal{N}$ *does not imply* $\mathcal{M} \equiv^{\mathrm{FOD}} \mathcal{N}$.

Proof. We get $(\mathbb{R}, \mathbb{N}) \Rrightarrow^{\mathrm{FOD}} (\mathbb{Q}, \mathbb{N})$ by Löwenheim-Skolem theorem. The other direction, $(\mathbb{R}, \mathbb{N}) \Lleftarrow^{\mathrm{FOD}} (\mathbb{Q}, \mathbb{N})$, does not hold because $(\mathbb{Q}, \mathbb{N}) \models \phi$, where ϕ is a dependence sentence expressing that there is a bijection between the universe and the predicate. Clearly $(\mathbb{R}, \mathbb{N}) \not\models \phi$. $\square$

Fact 4.2.4. *If* $\mathcal{M} \Rrightarrow_{\mathrm{FOD},n}^{\mathrm{FOD}} \mathcal{N}$, *then* $\mathcal{M} \Rrightarrow_{\mathrm{FOD},n}^{\mathrm{FO}} \mathcal{N}$.

Proof. Clear because first order formulas are dependence formulas. $\square$

Corollary 4.2.5. *If* $\mathcal{M} \Rrightarrow^{\mathrm{FOD}} \mathcal{N}$ *then* $\mathcal{M} \Rrightarrow^{\mathrm{FO}} \mathcal{N}$.

Fact 4.2.6. $\mathcal{M} \Rrightarrow^{\mathrm{FO}} \mathcal{N}$ *does not imply* $\mathcal{M} \Rrightarrow^{\mathrm{FOD}} \mathcal{N}$.

Proof. We have $(\mathbb{Q}, <) \Rrightarrow^{\mathrm{FO}} (\mathbb{R}, <)$. On the other hand, $(\mathbb{Q}, <) \models \phi$, where ϕ is a dependence sentence expressing that the order relation is incomplete, and $(\mathbb{R}, <) \not\models \phi$. $\square$

Corollary 4.2.7. $\mathcal{M} \Rrightarrow_{\mathrm{FO},n}^{\mathrm{FO}} \mathcal{N}$ *does not imply* $\mathcal{M} \Rrightarrow_{\mathrm{FOD},n}^{\mathrm{FOD}} \mathcal{N}$ *for all* n.

Corollary 4.2.8. $\mathcal{M} \Rrightarrow_{\mathrm{FOD},n}^{\mathrm{FO}} \mathcal{N}$ *does not imply* $\mathcal{M} \Rrightarrow_{\mathrm{FOD},n}^{\mathrm{FOD}} \mathcal{N}$ *for all* n.

Fact 4.2.9. *If* $\mathcal{M} \Rrightarrow_{\mathrm{FO},n}^{\mathrm{FO}} \mathcal{N}$, *then* $\mathcal{M} \Rrightarrow_{\mathrm{FOD},n}^{\mathrm{FO}} \mathcal{N}$.

Proof. Clear by Lemma 4.1.1. $\square$

Fact 4.2.10. $\mathcal{M} \Rrightarrow_{\mathrm{FOD},n}^{\mathrm{FOD}} \mathcal{N}$ *does not imply* $\mathcal{M} \Rrightarrow_{\mathrm{FO},n}^{\mathrm{FO}} \mathcal{N}$ *for all* n.

Proof. Let $\mathcal{M} = (\{0, 1, 2\}, \{0, 1\}, \{2\})$ and $\mathcal{N} = (\{0, 1, 2\}, \{0\}, \{2\})$ be models of the unary language $\{P, Q\}$. We consider rank 1. Clearly $\mathcal{M} \not\Rrightarrow_{\mathrm{FO},1}^{\mathrm{FO}} \mathcal{N}$ because $\mathcal{M} \models \forall x(Px \vee Qx)$ but $\mathcal{N} \not\models \forall x(Px \vee Qx)$. We can show $\mathcal{M} \Rrightarrow_{\mathrm{FOD},1}^{\mathrm{FOD}} \mathcal{N}$ by giving player **II** a winning strategy in the game $\mathrm{EF}_1^{\mathrm{FOD}}(\mathcal{M}, \mathcal{N})$.

To check the end condition of the game we need to know all basic dependence formulas of the considered language with zero or one free variables. They are the following.

$$\top \qquad \bot \qquad ()\rightsquigarrow x \qquad Px \qquad Qx \qquad \neg Px \qquad \neg Qx$$

If player **I** chooses a $\forall$-move, the game ends in position $(X, Y)^0$, where $X = \{\emptyset\}(x \mapsto M)$ and $Y = \{\emptyset\}(x \mapsto N)$. The only basic dependence formula that is satisfied by X is $\top$ which is trivially satisfied by any model and any team.

If player **I** chooses an $\exists$-move and $F\colon \emptyset \mapsto a \in M$, player **II** can choose $G\colon \emptyset \mapsto b \in N$ so that $a \in P^{\mathcal{M}}$ iff $b \in P^{\mathcal{N}}$, and $a \in Q^{\mathcal{M}}$ iff $b \in Q^{\mathcal{N}}$. Now the formulas Px, Qx, $\neg Px$, $\neg Qx$ are satisfied by X iff they are satisfied by Y. Also $()\rightsquigarrow x$ holds necessarily for both teams.

It is no good for player **I** to choose a $\vee$-move because he can only split $\{\emptyset\} = \emptyset \cup \{\emptyset\}$, and player **II** can imitate this, giving practically the same initial game position where $()\rightsquigarrow x$ holds for both teams.

In all these cases player **II** wins the game. Thus she has a winning strategy in $\mathrm{EF}_1^{\mathrm{FOD}}(\mathcal{M}, \mathcal{N})$. $\square$

Corollary 4.2.11. *$\mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FOD},n} \mathcal{N}$ does not imply $\mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FO},n} \mathcal{N}$ for all n.*

To sum up the previous facts, these are the implications between equivalence and semiequivalence of first order logic and dependence logic. All missing implications are false.

$$\mathcal{M} \equiv^{\mathrm{FOD}} \mathcal{N} \implies \mathcal{M} \Rrightarrow^{\mathrm{FOD}} \mathcal{N} \implies \mathcal{M} \Rrightarrow^{\mathrm{FO}} \mathcal{N} \iff \mathcal{M} \equiv^{\mathrm{FO}} \mathcal{N}$$

We also have the following implications for each $n < \omega$. All missing implications are false.

$$\begin{array}{ccc} \mathcal{M} \Rrightarrow^{\mathrm{FOD}}_{\mathrm{FOD},n} \mathcal{N} & \Longrightarrow & \mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FOD},n} \mathcal{N} \\ & & \Uparrow \\ \mathcal{M} \equiv^{\mathrm{FO}}_{\mathrm{FO},n} \mathcal{N} & \Longleftrightarrow & \mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FO},n} \mathcal{N} \end{array}$$

In the light of EF-game characterisations of $\equiv^{\mathrm{FO}}_n$ and $\Rrightarrow^{\mathrm{FOD}}_n$ by $\mathrm{EF}^{\mathrm{FO}}_n$ and $\mathrm{EF}^{\mathrm{FOD}}_n$, we see that Fact 4.2.10 says that even if player **II** has a winning strategy in $\mathrm{EF}^{\mathrm{FOD}}_n(\mathcal{M}, \mathcal{N})$, there is not necessarily any way to translate it into a winning strategy in $\mathrm{EF}^{\mathrm{FO}}_n(\mathcal{M}, \mathcal{N})$.

4.3 EF-game for FO in FOD-rank

The negative result that $\mathcal{M} \Rrightarrow^{\mathrm{FOD}}_{\mathrm{FOD},n} \mathcal{N}$ does not always imply $\mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FO},n} \mathcal{N}$ raises the question how to define an EF-game EF^* such that player **II** has a winning strategy in $\mathrm{EF}^*_n(\mathcal{M},\mathcal{N})$ iff $\mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FOD},n} \mathcal{N}$. As an answer to this, I propose the following game.

Definition 4.3.1. Let $\mathcal{M}$ and $\mathcal{N}$ be models, and $n < \omega$. The *EF-game for first order logic with dependence rank* n, $\mathrm{EF}^*_n(\mathcal{M},\mathcal{N})$, is played by players **I** and **II**. A position in the game is $(X,Y)^k$, where k is the number of moves left in the game, and X and Y are teams such that $\mathrm{Dom}(X) = \mathrm{Dom}(Y)$, team X maps to model $\mathcal{M}$ and team Y maps to model $\mathcal{N}$.

Initial position in the game is $(\{\emptyset\},\{\emptyset\})^n$. If the game is in position $(X,Y)^{k+1}$, then player **I** can choose one of the following moves.

$\vee$-move. Player **I** splits $X = X_0 \cup X_1$. Then player **II** splits $Y = Y_0 \cup Y_1$. Finally player **I** chooses the game to continue either from position $(X_0,Y_0)^k$ or from position $(X_1,Y_1)^k$.

$\exists$-move. Player **I** chooses some function $F\colon X \to M$ and variable x. Then player **II** chooses some function $G\colon Y \to N$. The game then continues from position $\big(X(x \mapsto F), Y(x \mapsto G)\big)^k$.

$\forall$-move. Player **I** chooses some variable x. The game continues from position $\big(X(x \mapsto M), Y(x \mapsto N)\big)^k$.

If the game is in position $(X,Y)^0$, then player **II** wins if for all basic $\phi \in \mathrm{FO}$ with $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(X)$ it holds that

$$\text{for all } s \in X\colon\ \mathcal{M}, s \models \phi \quad \Longrightarrow \quad \text{for all } s \in Y\colon\ \mathcal{N}, s \models \phi. \tag{4.1}$$

Otherwise player **I** wins.

Note that the game EF^* is highly similar to the EF-game of dependence logic, $\mathrm{EF}^{\mathrm{FOD}}$. The only difference is at the end of the game. In EF^* we inspect only basic first order formulas, whereas in $\mathrm{EF}^{\mathrm{FOD}}$ we inspect basic dependence formulas. In other parts the games EF^* and $\mathrm{EF}^{\mathrm{FOD}}$ are identical.

We will now prove that EF^* characterises the semiequivalence $\Rrightarrow^{\mathrm{FO}}_{\mathrm{FOD},n}$. The proof will be by induction on dependence rank.

Theorem 4.3.2. *Let $\mathcal{M}$ and $\mathcal{N}$ be models, and $n < \omega$. Player **II** has a winning strategy in $\mathrm{EF}^*_n(\mathcal{M},\mathcal{N})$ if and only if $\mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FOD},n} \mathcal{N}$.*

Proof. Let $n < \omega$. It suffices to prove the more general claim that for all positions $(X,Y)^k$, $k \leq n$, in the game $\mathrm{EF}^*_n(\mathcal{M},\mathcal{N})$ the following two conditions are equivalent.

(i) Player **II** has a winning strategy at $(X,Y)^k$.

(ii) Condition (4.1) holds for all $\phi \in \mathrm{FO}_k^{\mathrm{FOD}}$ with $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(X)$.

We prove this general claim by induction on k.

Basic case. Let $k = 0$. Assume first (i). The game is over, so player **II** won. Then (4.1) holds for all basic $\phi \in \mathrm{FO}$ with $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(X)$. In order to get condition (ii) to hold for all formulas $\phi \in \mathrm{FO}_0^{\mathrm{FOD}}$, we need to show (4.1) also for conjunctions of basic formulas. Let $\phi_1, \ldots, \phi_m$ each be basic. Because (4.1) holds for each of them, we get the following chain of equivalences.

$$\begin{aligned} & \text{for all } s \in X\text{: } \mathcal{M}, s \models \phi_1 \wedge \ldots \wedge \phi_m \\ \iff & \text{for all } s \in X \text{ and all } i \leq m\text{: } \mathcal{M}, s \models \phi_i \\ \iff & \text{for all } s \in Y \text{ and all } i \leq m\text{: } \mathcal{N}, s \models \phi_i \\ \iff & \text{for all } s \in Y\text{: } \mathcal{N}, s \models \phi_1 \wedge \ldots \wedge \phi_m \end{aligned}$$

Thus (ii) holds.

Assume then (ii). Because all basic $\phi \in \mathrm{FO}$ are also in $\mathrm{FO}_0^{\mathrm{FOD}}$, condition (i) follows.

For the inductive steps, let $0 < k \leq n$. Assume first (i). By the same chain of equivalences as we saw in the proof of (i) $\Rightarrow$ (ii) in case $k = 0$, we get condition (ii) when we prove (4.1) for formulas $\phi \vee \psi$, $\exists x\phi$, $\forall x\phi$ in $\mathrm{Fml}^{\mathrm{FO}}_{\mathrm{FOD},k}$. Let $\phi, \psi \in \mathrm{Fml}^{\mathrm{FO}}_{\mathrm{FOD},k-1}$ and let x be a variable symbol.

Case $\phi \vee \psi$. Assume $\mathrm{FV}(\phi \vee \psi) \subseteq \mathrm{Dom}(X)$ and, for all $s \in X$, $\mathcal{M}, s \models \phi \vee \psi$. Then for each $s \in X$ we have either $\mathcal{M}, s \models \phi$ or $\mathcal{M}, s \models \psi$. Therefore we can split $X = X_0 \cup X_1$ such that $\mathcal{M}, s \models \phi$ for all $s \in X_0$ and $\mathcal{M}, s \models \psi$ for all $s \in X_1$. Taking this as player **I**'s split in a $\vee$-move at game position $(X, Y)^k$, we get, by (i), a split $Y = Y_0 \cup Y_1$ from player **II**'s winning strategy so that player **II** still has a winning strategy at both positions $(X_0, Y_0)^{k-1}$ and $(X_1, Y_1)^{k-1}$, whichever player **I** chooses. By the induction hypothesis we get that $\mathcal{N}, s \models \phi$ for all $s \in Y_0$ and $\mathcal{N}, s \models \psi$ for all $s \in Y_1$. This gives $\mathcal{N}, s \models \phi \vee \psi$ for all $s \in Y$.

Case $\exists x\phi$. Assume $\mathrm{FV}(\exists x\phi) \subseteq \mathrm{Dom}(X)$ and $\mathcal{M}, s \models \exists x\phi$ for all $s \in X$. Then for each $s \in X$ there is an element $F(s) \in M$ such that $\mathcal{M}, s' \models \phi$, where $s' := s(x \mapsto F(s))$. That is, $\mathcal{M}, s \models \phi$ for each $s \in X(x \mapsto F)$. This function $F\colon X \to M$ and the variable x we can take as player **I**'s choices in an $\exists$-move at game position $(X, Y)^k$. By (i), from player **II**'s winning strategy we get a function $G\colon Y \to N$ so that player **II** still has a winning strategy at position $\big(X(x \mapsto F), Y(x \mapsto G)\big)^{k-1}$. By the induction hypothesis we get that $\mathcal{N}, s \models \phi$ for all $s \in Y(x \mapsto G)$. This gives $\mathcal{N}, s \models \exists x\phi$ for all $s \in Y$.

Case $\forall x\phi$**.** Assume $\mathrm{FV}(\forall x\phi) \subseteq \mathrm{Dom}(X)$ and $\mathcal{M}, s \models \forall x\phi$ for all $s \in X$. Then $\mathcal{M}, s' \models \phi$ for all $s \in X$ and $a \in M$, where $s' := s(x \mapsto a)$. In other words, $\mathcal{M}, s \models \phi$ for all $s \in X(x \mapsto M)$. By taking the variable x as player **I**'s move in a $\forall$-move at game position $(X, Y)^k$, we get, by (i), from player **II**'s winning strategy that she still has a winning strategy at position $\big(X(x \mapsto M), Y(x \mapsto N)\big)^{k-1}$. By the induction hypothesis we get that $\mathcal{N}, s \models \phi$ for all $s \in Y(x \mapsto N)$. This gives $\mathcal{N}, s \models \forall x\phi$ for all $s \in Y$.

Assume then (ii). We prove (i) for every possible move in the game.

Case $\vee$**-move.** Let player **I** split $X = X_0 \cup X_1$ in a $\vee$-move at game position $(X, Y)^k$. We know that $\mathrm{FO}^{\mathrm{FOD}}_{k-1}$ is a finite set of formulas. Therefore, letting, for $i = 0, 1$,

$$T_i := \{\phi \in \mathrm{FO}^{\mathrm{FOD}}_{k-1} : \mathcal{M}, s \models \phi \text{ for all } s \in X_i\},$$

we get $\mathcal{M}, s \models \bigwedge T_0 \vee \bigwedge T_1$ for all $s \in X$. Because $\bigwedge T_0 \vee \bigwedge T_1 \in \mathrm{FO}^{\mathrm{FOD}}_k$, from (ii) we get $\mathcal{N}, s \models \bigwedge T_0 \vee \bigwedge T_1$ for all $s \in Y$. Therefore, for each $s \in Y$, either $\mathcal{N}, s \models \bigwedge T_0$ or $\mathcal{N}, s \models \bigwedge T_1$, so we get a split $Y = Y_0 \cup Y_1$ such that $\mathcal{N}, s \models \bigwedge T_0$ for all $s \in Y_0$ and $\mathcal{N}, s \models \bigwedge T_1$ for all $s \in Y_1$. We take the split $Y = Y_0 \cup Y_1$ as player **II**'s choice in the $\vee$-move. Assume player **I** chooses the game to continue from position $(X_i, Y_i)^{k-1}$. To show that player **II** has a winning strategy from this position, we prove (ii) for $(X_i, Y_i)^{k-1}$ and then use the induction hypothesis. So, let $\phi \in \mathrm{FO}^{\mathrm{FOD}}_{k-1}$ and assume that $\mathcal{M}, s \models \phi$ for all $s \in X_i$. Then $\phi \in T_i$, so $\mathcal{N}, s \models \phi$ for all $s \in Y_i$. This is what we wanted.

Case $\exists$**-move.** Let player **I** choose a function $F\colon X \to M$ and a variable x in an $\exists$-move at game position $(X, Y)^k$. Let

$$T := \{\phi \in \mathrm{FO}^{\mathrm{FOD}}_{k-1} : \mathcal{M}, s \models \phi \text{ for all } s \in X(x \mapsto F)\}.$$

We get $\mathcal{M}, s \models \exists x \bigwedge T$ for all $s \in X$. Because $\exists x \bigwedge T \in \mathrm{FO}^{\mathrm{FOD}}_k$, from (ii) we get $\mathcal{N}, s \models \exists x \bigwedge T$ for all $s \in Y$. Therefore for each $s \in Y$ there is an element $G(s) \in N$ such that $\mathcal{N}, s' \models \bigwedge T$, where $s' := s(x \mapsto G(s))$, so we get the function $G\colon Y \to N$ such that $\mathcal{N}, s \models \bigwedge T$ for all $s \in Y(x \mapsto G)$. We take this function as player **II**'s choice in the $\exists$-move. To show that player **II** has a winning strategy from position $\big(X(x \mapsto F), Y(x \mapsto G)\big)^{k-1}$, we prove (ii) for this position and then use the induction hypothesis. So, let $\phi \in \mathrm{FO}^{\mathrm{FOD}}_{k-1}$ and assume $\mathcal{M}, s \models \phi$ for all $s \in X(x \mapsto F)$. Then $\phi \in T$, so $\mathcal{N}, s \models \phi$ for all $s \in Y(x \mapsto G)$. This is what we wanted.

Case $\forall$-move. Let player **I** choose a variable x in a $\forall$-move at game position $(X, Y)^k$. Let

$$T := \{\phi \in \mathrm{FO}^{\mathrm{FOD}}_{k-1} : \mathcal{M}, s \models \phi \text{ for all } s \in X(x \mapsto M)\}.$$

We get $\mathcal{M}, s \models \forall x \bigwedge T$ for all $s \in X$. Because $\forall x \bigwedge T \in \mathrm{FO}^{\mathrm{FOD}}_{k}$, from (ii) we get $\mathcal{N}, s \models \forall x \bigwedge T$ for all $s \in Y$. Therefore $\mathcal{N}, s' \models \bigwedge T$ for all $s \in Y$ and $a \in N$, where $s' := s(x \mapsto a)$, so we get $\mathcal{N}, s \models \bigwedge T$ for all $s \in Y(x \mapsto N)$. To show that player **II** has a winning strategy from position $\big(X(x \mapsto M), Y(x \mapsto N)\big)^{k-1}$, we prove (ii) for this position and use the induction hypothesis. So, let $\phi \in \mathrm{FO}^{\mathrm{FOD}}_{k-1}$ and assume $\mathcal{M}, s \models \phi$ for all $s \in X(x \mapsto M)$. Then $\phi \in T$, so $\mathcal{N}, s \models \phi$ for all $s \in Y(x \mapsto N)$. This is what we wanted. $\square$

4.4 Converting Winning Strategies

The result that $\mathcal{M} \Rrightarrow^{\mathrm{FOD}}_{\mathrm{FOD},n} \mathcal{N}$ implies $\mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FOD},n} \mathcal{N}$, together with the EF-characterisations of these semiequivalences, gives the result that if player **II** has a winning strategy in $\mathrm{EF}^{\mathrm{FOD}}_n(\mathcal{M}, \mathcal{N})$ then she has a winning strategy also in $\mathrm{EF}^{*}_n(\mathcal{M}, \mathcal{N})$. There is even a stronger link between these games. Namely, we can effectively convert a winning strategy in $\mathrm{EF}^{\mathrm{FOD}}_n(\mathcal{M}, \mathcal{N})$ into one in $\mathrm{EF}^{*}_n(\mathcal{M}, \mathcal{N})$. This is, however, trivial because the games have identical moves.

Because $\mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FO},n} \mathcal{N}$ implies $\mathcal{M} \Rrightarrow^{\mathrm{FO}}_{\mathrm{FOD},n} \mathcal{N}$, we get also that if player **II** has a winning strategy in $\mathrm{EF}^{\mathrm{FO}}_n(\mathcal{M}, \mathcal{N})$ then she has one in $\mathrm{EF}^{*}_n(\mathcal{M}, \mathcal{N})$. Also here we have the stronger result that the first strategy can be effectively converted into the second strategy. This time the conversion is not trivial.

Theorem 4.4.1. *Player **II**'s winning strategy in* $\mathrm{EF}^{\mathrm{FO}}_n(\mathcal{M}, \mathcal{N})$ *can be effectively converted into her winning strategy in* $\mathrm{EF}^{*}_n(\mathcal{M}, \mathcal{N})$.

Proof. Let $\mathcal{M}$ and $\mathcal{N}$ be models, and let $n < \omega$. We prove the theorem by proving by induction on k the following more general claim. For any teams X and Y, if

$$\text{for all } s' \in Y \text{ there is } s \in X \text{ such that } \mathbf{II} \uparrow \mathrm{EF}^{\mathrm{FO}}_n(\mathcal{M}, \mathcal{N}) \text{ @ } (s, s')^k, \quad (4.2)$$

then

$$\mathbf{II} \uparrow \mathrm{EF}_n^*(\mathcal{M}, \mathcal{N}) \ @ \ (X, Y)^k. \tag{4.3}$$

The general idea in player **II**'s winning strategy in $\mathrm{EF}_n^*(\mathcal{M}, \mathcal{N})$ is that she plays several games of $\mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N})$ at the same time, and reads from those games her winning strategy in $\mathrm{EF}_n^*(\mathcal{M}, \mathcal{N})$.

Basic case. Let $k = 0$. Assume (4.2). Let $\phi \in \mathrm{FO}$ be basic with $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(X)$ and assume $\mathcal{M}, s \models \phi$ for all $s \in X$. Let $s' \in Y$. By (4.2), there is $s \in X$ such that **II** wins $\mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N})$ at position $(s, s')^0$. This means $\mathcal{M}, s \models \phi$ iff $\mathcal{N}, s' \models \phi$, so $\mathcal{N}, s' \models \phi$. Hereby $\mathcal{N}, s' \models \phi$ for all $s' \in Y$. Therefore (4.3) holds.

For the inductive cases, let $0 < k \leq n$. Assume the induction hypothesis and (4.2). We construct a winning strategy for player **II** by cases on the three different moves that player **I** can choose.

Case $\vee$-move. Let player **I** split $X = X_0 \cup X_1$. Define, for $i < 2$, teams

$$Y_i := \{s' \in Y : \text{there is } s \in X_i \text{ such that } \mathbf{II} \uparrow \mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N}) \ @ \ (s, s')^k\}.$$

By (4.2), $Y = Y_0 \cup Y_1$. By definition of Y_i, for all $s' \in Y_i$ there is $s \in X_i$ such that $\mathbf{II} \uparrow \mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N}) \ @ \ (s, s')^k$, so also $\mathbf{II} \uparrow \mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N}) \ @ \ (s, s')^{k-1}$. By the induction hypothesis, $\mathbf{II} \uparrow \mathrm{EF}^*(\mathcal{M}, \mathcal{N}) \ @ \ (X_i, Y_i)^{k-1}$ for $i < 2$.

Case $\exists$-move. Let player **I** choose function $F\colon X \to M$ and variable x. Player **II** chooses her function $G\colon Y \to N$ as follows. Let $s' \in Y$. By (4.2), there is $s \in X$ such that $\mathbf{II} \uparrow \mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N}) \ @ \ (s, s')^k$. By making player **I** choose $F(s) \in M$ and x in $\mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N})$, player **II**'s winning strategy gives $b \in N$ such that $\mathbf{II} \uparrow \mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N}) \ @ \ \big(s(x \mapsto F(s)), s'(x \mapsto b)\big)^{k-1}$. Let $G(s')$ be this b. Now player **II** has made her choice in $\mathrm{EF}_n^*(\mathcal{M}, \mathcal{N})$, and it holds that for each $s \in X(x \mapsto F)$ there is $s' \in Y(x \mapsto G)$ such that $\mathbf{II} \uparrow \mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N}) \ @ \ (s, s')^{k-1}$. Thus, by the induction hypothesis, $\mathbf{II} \uparrow \mathrm{EF}_n^*(\mathcal{M}, \mathcal{N}) \ @ \ \big(X(x \mapsto F), Y(x \mapsto G)\big)^{k-1}$.

Case $\forall$-move. Let player **I** choose variable x. To show $\mathbf{II} \uparrow \mathrm{EF}_n^*(\mathcal{M}, \mathcal{N}) \ @ \ \big(X(x \mapsto M), Y(x \mapsto N)\big)^{k-1}$ by using the induction hypothesis, we must find for each $s'(x \mapsto b) \in Y(x \mapsto N)$ some $s(x \mapsto a) \in X(x \mapsto M)$ such that $\mathbf{II} \uparrow \mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N}) \ @ \ (s, s')^{k-1}$. So, let $s' \in Y$ and $b \in N$. By (4.2), there is $s \in X$ such that $\mathbf{II} \uparrow \mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N}) \ @ \ (s, s')^k$. By making player **I** choose $b \in N$ and x in $\mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N})$, player **II**'s winning strategy gives $a \in M$ such that $\mathbf{II} \uparrow \mathrm{EF}_n^{\mathrm{FO}}(\mathcal{M}, \mathcal{N}) \ @ \ \big(s(x \mapsto a), s'(x \mapsto b)\big)^{k-1}$. This is what we wanted.

As we succeeded in all the three cases to give player **II** a move that results in she still having a winning strategy after the move, (4.3) holds. This completes the proof. □

4.5 Further Points of Interest

Even though $\mathcal{M} \Rrightarrow_n^{\mathrm{FOD}} \mathcal{N}$ does not imply $\mathcal{M} \equiv_n^{\mathrm{FO}} \mathcal{N}$ for all n, it is reasonable to ask if there is a limit n from which on the implication holds. Perhaps this limit is 4 as from there on D-formulas give additional expressive power by limiting the domain of Skolem functions of existentially quantified variables.

More generally put, we can ask for which functions $f\colon \omega \to \omega$ does it hold that, for all models $\mathcal{M}$, $\mathcal{N}$ and all $n < \omega$, $\mathcal{M} \Rrightarrow_n^{\mathrm{FOD}} \mathcal{N}$ implies $\mathcal{M} \equiv_{f(n)}^{\mathrm{FO}} \mathcal{N}$. Trivially the constant function $f = 0$ is one such function but probably not the only one, and identity $f(n) = n$ is not such a function. For these functions we have the nontrivial problem of converting any given winning strategy of player **II** in $\mathrm{EF}_n^{\mathrm{FOD}}$ into player **II**'s winning strategy in $\mathrm{EF}_{f(n)}^{\mathrm{FO}}$.

In spite of the negative result $\mathcal{M} \Rrightarrow_{\mathrm{FOD},n}^{\mathrm{FO}} \mathcal{N} \not\Rightarrow \mathcal{M} \Rrightarrow_{\mathrm{FO},n}^{\mathrm{FO}} \mathcal{N}$, one could try to show that if $\mathcal{M} \Rrightarrow_{\mathrm{FOD},n}^{L} \mathcal{N}$ then $\mathcal{M} \Rrightarrow_{\mathrm{FO},n}^{L} \mathcal{N}$, where L is the fragment of FO that consists of disjunctions of FO-sentences where $\vee$ does not occur.

It was Ryan Siders who initially pointed out the problem of two different ranks for dependence logic. He also gave helpful ideas concerning the comparison of semiequivalences.

Chapter 5

Translating between Logics

In this chapter I investigate the question of translating formulas from one logic to another. I will first define translations in a general setting and revise some known results in this light. I will then present two detailed translations from second order logic to team logic, the first for sentences and the second for all formulas.

5.1 The General Setting

Recall that semantic objects in first order logic are first order assignments, in second order logic they are second order assignments, and in dependence logic and team logic they are teams. For a logic L and a model $\mathcal{M}$, define $I^L_{\mathcal{M}}$ to be the powerset of sets of all semantic objects for model $\mathcal{M}$ in logic L,

$$I^L_{\mathcal{M}} := \{\mathcal{S} : \mathcal{S} \text{ is a set of semantic objects for } \mathcal{M} \text{ in } L\}.$$

That is, $I^L_{\mathcal{M}}$ is the set of all possible interpretations that formulas in L may have on model $\mathcal{M}$.

Definition 5.1.1. Let L and R be logics. Define a *syntactic translation* of L to R to be a function $f\colon L \to R$ that maps L-formulas to R-formulas. Define a *semantic translation* of L to R to be a collection $\{g_{\mathcal{M}}\colon I^L_{\mathcal{M}} \to I^R_{\mathcal{M}} \mid \mathcal{M}$ is a model$\}$ of injective functions that map interpretations of L-formulas to interpretations of R-formulas. We define a *translation* of L to R to be a pair consisting of a syntactic translation $f\colon L \to R$ and a semantic translation $\{g_{\mathcal{M}}\colon I^L_{\mathcal{M}} \to I^R_{\mathcal{M}} \mid \mathcal{M}$ is a model$\}$ such that for all $\phi \in L$ and models $\mathcal{M}$,

$$[\![f(\phi)]\!]^R_{\mathcal{M}} = g_{\mathcal{M}}\big([\![\phi]\!]^L_{\mathcal{M}}\big). \tag{5.1}$$

A translation of L to R can also be a pair consisting of a syntactic translation $f\colon L \to R$ and a backward semantic translation $\{g_{\mathcal{M}}\colon I^R_{\mathcal{M}} \to I^L_{\mathcal{M}} : \mathcal{M}$ model$\}$

such that for all $\phi \in L$ and models $\mathcal{M}$,

$$g_{\mathcal{M}}\big(\llbracket f(\phi)\rrbracket^{R}_{\mathcal{M}}\big) = \llbracket \phi \rrbracket^{L}_{\mathcal{M}}. \tag{5.2}$$

In case both logics L and R have the same kind of semantic objects or consist only of sentences, the role of semantic translation becomes negligible and the functions $g_{\mathcal{M}}$ can be required to be identity functions. However, when translating between a logic whose semantic objects are assignments and a logic whose semantic objects are teams, a mere syntactic translation is meaningless. In such a case, a semantic translation is essential as it tells us how the meaning of a formula translates from one logic to another with the given syntactic translation.

Some semantic translations can be generated by simpler functions that map semantic objects of logic L to semantic objects of logic R. For example, let $\mathcal{M}$ be a model, let $S_{\mathcal{M}}$ denote the set of all second order assignments s on $\mathcal{M}$ such that $\mathrm{Dom}(s) = \{R\}$, let $\alpha_1(s) = s(R)_{(x_1,\ldots,x_n)}$ for all $s \in S_{\mathcal{M}}$, and let $\alpha_2(X) = (R \mapsto \mathrm{Rel}(X))$ for all teams X on $\mathcal{M}$. Functions α_1 and α_2 represent the natural and close relationship between teams and relations. They also generate the *natural semantic translations*

$$g^{\mathrm{Nat}}_{\mathcal{M}}(S) = \{\alpha_1(s) : s \in S\},$$

for all sets S of second order assignments s on $\mathcal{M}$ such that $\mathrm{Dom}(s) = \{R\}$, and

$$g^{\mathrm{Nat}}_{\mathcal{M}}(\mathcal{X}) = \{\alpha_2(X) : X \in \mathcal{X}\},$$

for all sets $\mathcal{X}$ of teams on $\mathcal{M}$.

Translating FO to FOD

Väänänen proves a translation of first order logic to dependence logic [19, Proposition 3.31] by the syntactic translation $\phi \mapsto \phi$, for all $\phi \in \mathrm{FO}$, and the semantic translation $\mathcal{X} \mapsto \bigcup \mathcal{X}$, for all sets $\mathcal{X}$ of teams on $\mathcal{M}$, or the backward semantic translation $X \mapsto \mathcal{P}X$, for all sets X of first order assignments. These semantic translations are apparently not generated by any functions on semantic objects. Equivalently, the translation can be stated as

$$\mathcal{M}, X \models \phi \text{ if and only if } \mathcal{M}, s \models \phi \text{ for all } s \in X$$

for all teams X and models $\mathcal{M}$.

Translating FOD to TL

There is a translation of FOD to TL. Its syntactic translation f is defined for all $\phi \in \text{FOD}$ in strict negation normal form as follows.

$$\begin{aligned} f(Rt_1 \dots t_n) &:= Rt_1 \dots t_n & f(\bot) &:= \mathbf{0} \\ f\big((t_1 \dots t_n) \rightarrowtail u\big) &:= (t_1 \dots t_n) \rightarrowtail u & f(\top) &:= \top \\ f(\phi \vee \psi) &:= f(\phi) \otimes f(\psi) & f(\exists x \phi) &:= \exists x f(\phi) \\ f(\phi \wedge \psi) &:= f(\phi) \wedge f(\psi) & f(\forall x \phi) &:= !x f(\phi) \\ f(\neg \phi) &:= \neg f(\phi) \end{aligned}$$

The semantic translation is the identity function. Alternatively, the translation can be formulated as

$$\mathcal{M}, X \models^{\text{TL}} f(\phi) \text{ if and only if } \mathcal{M}, X \models^{\text{FOD}} \phi$$

for all teams X and models $\mathcal{M}$.

Translating FOD to ESO

A straightforward translation of FOD to ESO is known; given $\phi(x_1, \dots, x_n) \in \text{FOD}$ we can write out the definition of satisfaction of ϕ as an ESO-formula $\psi(R)$ where R is an n-ary relation variable standing for the team that is supposed to satisfy ϕ. Väänänen has shown the syntactic translation in detail [19, Theorem 6.2]. The semantic translation is the natural one, consisting of functions $g^{\text{Nat}}_{\mathcal{M}}$. Equivalently, the translation can be stated as

$$\mathcal{M}, X \models \phi \text{ if and only if } \mathcal{M}, s_X \models f(\phi)$$

for all teams X and models $\mathcal{M}$, where $s_X := (R \mapsto \text{Rel}(X))$.

Translating TL to SO

The translation of TL to SO can be done with the same idea. It consists of functions $f \colon \text{TL} \to \text{SO}$, whose details Väänänen has given [19, Theorem 8.12], and $g_{\mathcal{M}}$ that are exactly as in the previous case of translating dependence logic to existential second order logic.[1]

In both the above translations of FOD to ESO and TL to SO, the functions $g_{\mathcal{M}}$ are generated by the following simple mappings of semantic objects: $s \mapsto s(R)_{(x_1,\dots,x_n)}$ and $X \mapsto (R \mapsto \text{Rel}(X))$. This is natural and based on the close relationship of teams and relations.

[1]Note that $I^{\text{ESO}}_{\mathcal{M}} = I^{\text{SO}}_{\mathcal{M}}$ and $I^{\text{FOD}}_{\mathcal{M}} = I^{\text{TL}}_{\mathcal{M}}$ for all models $\mathcal{M}$.

Translating ESO-Sentences to FOD

There is a well-known translation of ESO-sentences to FOD-sentences. This translation, known as the *Enderton-Walkoe translation*, is more involved and dates back almost three decades. Although this translation is cited in numerous papers (such as [12, pp. 62–63]), I have not found a detailed description of it let alone a proof that the translation works as it intuitively would seem to. The original independently written papers by Enderton [6] and Walkoe [20] settle for giving an example of the translation, each its own. Moreover, the translations are formulated in first order logic enhanced with Henkin quantifiers. Because of these reasons, I find it appropriate to give a detailed description of the translation, along with a detailed proof that it works, formulated in dependence logic. Section 5.2 is dedicated to this.

Translating Downward Closed ESO to FOD

A previous result by Kontinen and Väänänen shows that existential second order formulas with a free relation variable can be translated to dependence formulas if and only if the existential second order formula is closed downward with respect to the relation variable. [17]

Theorem 5.1.2. *Let $\phi(R) \in \mathrm{ESO}$, where R is an n-ary relation variable. Then the following conditions are equivalent:*

1. *There is $\psi(x_1, \ldots, x_n) \in \mathrm{FOD}$ such that for all models $\mathcal{M}$ and teams X, $\mathcal{M}, X \models \psi$ if and only if $\mathcal{M}, \mathrm{Rel}(X) \models \phi$;*
2. *If $\mathcal{M}, s \models \phi$ and $s'(R) \subseteq s(R)$, then $\mathcal{M}, s' \models \phi$, i.e. ϕ is* closed downward *with respect to R.*

This translation is based on the natural semantic translation $\{g_{\mathcal{M}}^{\mathrm{Nat}}\}$ and a complex syntactic translation, whose details I shall not reproduce here.

The requirement of ϕ being closed downward with respect to R is necessary with the natural semantic translation because dependence formula interpretations are always closed downward. It is an open question if some other semantic translation would allow translating all existential second order formulas to dependence formulas.

5.2 Translating ESO to FOD

In this section I give a detailed description of the well-known translation of existential second order sentences to dependence sentences. The syntactic translation is shown in the proof below. The semantic translation is

$g_{\mathcal{M}}\colon I^{\mathrm{FOD}}_{\mathcal{M}} \to I^{\mathrm{ESO}}_{\mathcal{M}}$ such that

$$g_{\mathcal{M}}(\mathcal{X}) = \begin{cases} \{\}, & \text{if } \mathcal{X} = \{\emptyset\}; \\ \{s : s \text{ is a second order assignment on } \mathcal{M}\}, & \text{otherwise,} \end{cases}$$

for all models $\mathcal{M}$, where $\mathcal{X}$ is a set of teams on model $\mathcal{M}$. The translation works also with the backward semantic translation $g_{\mathcal{M}}\colon I^{\mathrm{ESO}}_{\mathcal{M}} \to I^{\mathrm{FOD}}_{\mathcal{M}}$ such that

$$g_{\mathcal{M}}(S) = \begin{cases} \{\emptyset\}, & \text{if } S = \{\}; \\ \{X : X \text{ is a team on } \mathcal{M}\}, & \text{otherwise,} \end{cases}$$

for all models $\mathcal{M}$, where S is a set of second order assignments on model $\mathcal{M}$.

Theorem 5.2.1. *For every sentence $\phi \in$ ESO there is a sentence $\chi \in$ FOD such that, for all models $\mathcal{M}$, $\mathcal{M} \models \phi$ if and only if $\mathcal{M} \models \chi$.*

Proof. We can assume that ϕ is in Skolem normal form

$$\exists f_1 \ldots \exists f_n \forall x_1 \ldots \forall x_m \psi, \tag{5.3}$$

where ψ is a quantifier-free second order formula. We will perform some reductions on (5.3) to make it more suitable for finding χ. Remember that in general ψ can contain function variables as well as function symbols from the language. We repeat step 1 until it cannot be applied, after which we repeat step 2 until it cannot be applied. The result of each repetition is a formula in the form (5.3).

Step 1. First we ensure that function variables f occur in ψ only with element variables as arguments, and that each element variable occurs at most once as an argument of each occurrence of f. Assume that in ψ there is a term occurrence $ft_1 \ldots t_k$, where $t_1, \ldots, t_k$ are terms of which t_j either is the same element variable as t_i for some $i < j$ or is not an element variable at all. We write (5.3) as ϕ' as follows:

$$\phi' := \exists f_1 \ldots \exists f_n \forall x_1 \ldots \forall x_m \forall x_{m+1} \big(x_{m+1} = t_j \to \psi(t_j \mapsto x_{m+1})\big). \tag{5.4}$$

When the terms x_{m+1} and t_j have the same interpretation, they are interchangeable in ψ. Therefore we see that (5.3) and (5.4) are logically equivalent.

Step 2. Now we ensure that for each function variable there is at most one sequence of arguments with which it occurs. Assume that in ψ there are occurrences of terms $fu_1 \ldots u_k$ and $fv_1 \ldots v_k$, where f is a function variable and all u_i and v_i are element variables such that at some index i, u_i is a different variable than v_i. To replace $fv_1 \ldots v_k$, we introduce a new function variable and an exclusive sequence of element variables for it. More specifically, we write (5.3) as ϕ' as follows:

$$\phi' := \exists f_1 \ldots \exists f_n \exists f_{n+1} \forall x_1 \ldots \forall x_m \forall x_{m+1} \ldots \forall x_{m+k} (\theta_1 \wedge \theta_2), \tag{5.5}$$

where

$$\theta_1 := \bigwedge_{i \leq k} x_{m+i} = u_i \rightarrow fu_1 \ldots u_k = f_{n+1}x_{m+1} \ldots x_{m+k},$$
$$\theta_2 := \bigwedge_{i \leq k} x_{m+i} = v_i \rightarrow \psi(fv_1 \ldots v_k \mapsto f_{n+1}x_{m+1} \ldots x_{m+k}).$$

To see that (5.3) is equivalent to (5.5), let $\mathcal{M}$ be a model such that $\mathcal{M} \models \phi$. Then there are functions $g_1, \ldots, g_n$ such that for all $a_1, \ldots, a_m \in M$, $\mathcal{M}, s \models \psi$, where $s := (f_i \mapsto g_i)_{i \leq n}(x_i \mapsto a_i)_{i \leq m}$. Choose $g_{n+1} := s(f)$ and let $s' := s(f_{n+1} \mapsto g_{n+1})(x_{m+i} \mapsto a_{m+i})_{i \leq k}$, where $a_{m+1}, \ldots, a_{m+k} \in M$ are arbitrary. Then clearly $\mathcal{M}, s' \models \theta_1 \wedge \theta_2$, whence $\mathcal{M} \models \phi'$.

To see the other direction, let $\mathcal{M}$ be a model such that $\mathcal{M} \models \phi'$. Then there are functions $g_1, \ldots, g_{n+1}$ such that for all $a_1, \ldots, a_{m+k} \in M$, $\mathcal{M}, s \models \theta_1 \wedge \theta_2$, where $s := (f_i \mapsto g_i)_{i \leq n+1}(x_i \mapsto a_i)_{i \leq m+k}$. Varying over all values for all a_i we get from θ_1 that $s(f)$ and $s(f_{n+1})$ are the same function. Here it is essential that all variables $u_1, \ldots, u_k$ are distinct, as taken care of in step 1. Otherwise there might be some specific sequence of arguments $a_1, \ldots, a_k$ on which $s(f)$ and $s(f_{n+1})$ would disagree but we would not be able to express the sequence as $s(u_1), \ldots, s(u_k)$. Now, looking at θ_2 we see that $\mathcal{M}, s \models \psi$ by the fact that $s(f_{n+1}x_{m+1} \ldots x_{m+k}) = s(fv_1 \ldots v_k)$.

After these two steps, we have a sentence $\phi \in$ ESO in form (5.3) such that each function variable f occurs in ψ only in the term $fu_1^i \ldots u_{k_i}^i$, where each u_j^i is an element variable. Now we are able to express ϕ as the following sentence $\chi \in$ FOD:

$$\chi := \forall x_1 \cdots \forall x_m \exists x_{m+1} \cdots \exists x_{m+n}(\chi_1 \wedge \chi_2), \tag{5.6}$$

where

$$\chi_1 := \left(u_1^1 \ldots u_{k_1}^1\right) \rightsquigarrow x_{m+1} \wedge \cdots \wedge \left(u_1^n \ldots u_{k_n}^n\right) \rightsquigarrow x_{m+n},$$
$$\chi_2 := \psi\left(f_i u_1^i \ldots u_{k_i}^i \mapsto x_{m+i}\right)_{i \leq n}.$$

It is step 2 that enables us to replace each function quantifer in ϕ with one element quantifier in χ.[2] To see that (5.3) and (5.6) are equivalent, let $\mathcal{M}$ be a model such that $\mathcal{M} \models \phi$. Then there are functions $g_1, \ldots, g_n$ such that for all $a_1, \ldots, a_m \in M$, $\mathcal{M}, s \models \psi$, where $s := (f_i \mapsto g_i)_{i \leq n}(x_i \mapsto a_i)_{i \leq m}$. Now, let $X := (x_i \mapsto M)_{i \leq m}(x_{m+i} \mapsto F_i)_{i \leq n}$, where each $F_i \colon X \rightarrow M$ maps

[2]Actually, we could skip step 2 and compensate by replacing each function quantifier in ϕ with as many element quantifiers in χ as there are occurrences of the function variable in ϕ.

$F_i(s) = g_i\big(s(u^i_1), \ldots, s(u^i_{k_i})\big)$. Then $\mathcal{M}, X \models \chi_1$ because the values of each x_{m+i} depend only on variables used by F_i, i.e. $u^i_1, \ldots, u^i_{k_i}$. Also, because $\chi_2 \in$ FO, we have $\mathcal{M}, X \models \chi_2$ if $\mathcal{M}, s \models \chi_2$ for all $s \in X$, and this holds by assumption and the fact that $s(x_{m+i}) = s\big(f_i u^i_1 \ldots u^i_{k_i}\big)$ for all $s \in X$.

To see the other direction, let $\mathcal{M}$ be a model such that $\mathcal{M} \models \chi$. Then there are functions $F_j \colon X_j \to M$, where $X_j := (x_i \mapsto M)_{i \leq m}(x_{m+i} \mapsto F_i)_{i<j}$ for $j \leq n$, such that $\mathcal{M}, X_n \models \chi_1 \wedge \chi_2$. By $\mathcal{M}, X_n \models \chi_1$ there are functions $g_i \colon M^{k_i} \to M$ such that $s(x_{m+i}) = g_i\big(s(u^i_1), \ldots, s(u^i_{k_i})\big)$ for all $s \in X_i$ and all $i \leq n$. By $\mathcal{M}, X_n \models \chi_2$ and the fact that $s(x_{m+i}) = s'\big(f_i u^i_1 \ldots u^i_{k_1}\big)$ for all $i \leq n$, where $s' := s(f_i \mapsto g_i)_{i \leq n}$, we get that $\mathcal{M}, s' \models \psi$ for all $s \in X_n$. Therefore $\mathcal{M} \models \phi$. $\square$

5.3 Translating SO-Sentences to TL

In this section, I present an explicit translation of sentences in SO to sentences in TL. There is a similar result by Harel [9]. In place of Team Logic he used partially ordered quantifiers, or Henkin quantifiers, with his own semantics that provide a greater expressive power than Henkin's original semantics. Harel also provided a correspondence between the second order fragments Σ^1_n, Π^1_n and certain fragments of first order logic with Henkin quantifiers. Although it seems perfectly possible to formulate a similar correspondence between certain fragments of team logic and the mentioned fragments of the second order logic, the correspondence is rather complex to formulate and therefore I will not go as far as Harel and define it.

Definition 5.3.1. Let X be a team for some model $\mathcal{M}$, let $f \colon M^k \to M$ and $F \colon X \to M$ be functions, and let $y_1, \ldots, y_k \in \mathrm{Dom}(X)$ and $z \notin \mathrm{Dom}(X)$. We say that f *is similar to* F *via* $(y_1, \ldots, y_k)$ *in* X if $\mathcal{M}, X(z \mapsto F) \models (y_1 \ldots y_k) \rightsquigarrow z$ and $F(s) = f\big(s(y_1), \ldots, s(y_k)\big)$ for all $s \in X$.

Loosely speaking, f and F are similar if they are fundamentally the same function with the main difference that f maps sequences of elements whereas F maps sequences whose members are named by variables. Another difference is that F is defined only on sequences that appear as assignments in X whereas f is defined on all sequences. We will only use the concept of similarity of functions in the context of some team so that the difference in the domains of the functions will not become an issue. The corresponding team will usually be apparent from the context, so we will just say that f and F are similar via a sequence.

Lemma 5.3.2. *If $\mathcal{M}$ is a model and X is team on $\mathcal{M}$, then*

1. *Given a function* $f\colon M^k \to M$ *and variables* $y_1, \dots, y_k \in \mathrm{Dom}(X)$, *there is a function* $F\colon X \to M$ *that is similar to* f *via* $(y_1, \dots, y_k)$.

2. *Given a function* $F\colon X \to M$ *and variables* $y_1, \dots, y_k \in \mathrm{Dom}(X)$ *and* $z \notin \mathrm{Dom}(X)$ *such that* $\mathcal{M}, X(z \mapsto F) \models (y_1 \dots y_k) \rightsquigarrow z$, *there is a function* $f\colon M^k \to M$ *that is similar to* F *via* $(y_1, \dots, y_k)$.

Proof. We can see 1 by letting $F(s) = f\big(s(y_1), \dots, s(y_k)\big)$ for all $s \in X$. To see 2, let $f(a_1, \dots, a_k) = F(s_a)$, where $s_a := (y_i \mapsto a_i)_{i \leq k}$, when $s_a \in X$, and let f be defined arbitrarily elsewhere. $\square$

Now for the actual theorem.

Theorem 5.3.3. *For every sentence* $\phi \in \mathrm{SO}$ *there is* $\chi \in \mathrm{TL}$ *such that, for all models* $\mathcal{M}$, $\mathcal{M} \models \phi$ *if and only if* $\mathcal{M} \models \chi$.

Proof. Let $\phi \in \mathrm{SO}$ be in Skolem normal form,

$$\phi := \exists f_1^1 \dots \exists f_{n_1}^1 \forall f_1^2 \dots \forall f_{n_2}^2 \dots \exists f_1^p \dots \exists f_{n_p}^p \forall x_1 \dots \forall x_q \psi,$$

where ψ is quantifier-free and in negation normal form. We may assume that no variable is quantified twice in ϕ.[3]

Let $t_1, \dots, t_m$ enumerate all the occurrences of terms in ψ that start with instances of the quantified functions in ϕ. Each t_i, $i \leq m$, we write as

$$t_i := f_{n(i)}^{p(i)} t_1^i \dots t_{k(i)}^i.$$

Furthermore, we require that the enumeration $t_1, \dots, t_m$ satisfies the condition that $i < j$ implies that t_j does not occur as a subterm of t_i. Given $i \leq m$, we denote by $o(i)$ the least index that satisfies $p(i) = p(o(i))$ and $n(i) = n(o(i))$. In other words, $o(i)$ is the first index for which the term $t_{o(i)}$ begins with the symbol $f_{n(i)}^{p(i)}$.

By $\exists_{i \leq n} x_i$ we mean the quantifier block $\exists x_1 \dots \exists x_n$. Furthermore, if $\prec$ is a linear order on $\{1, \dots, n\}$, by $\exists_{i \leq n}^{\prec} x_i$ we mean the quantifier block $\exists x_{i_1} \dots \exists x_{i_n}$, where $i_1 \prec \dots \prec i_n$ and $\{i_1, \dots, i_n\} = \{1, \dots, n\}$. We will use the following ordering for $i, j \leq m$:

$$\begin{aligned} i \prec j \iff p(i) < p(j) \text{ or } &\big(p(i) = p(j) \text{ and } n(i) < n(j)\big) \\ \text{or } &\big(p(i) = p(j) \text{ and } n(i) = n(j) \text{ and } i < j\big). \end{aligned}$$

[3]We can get rid of doubly quantified variables by renaming variables suitably.

Now we are ready to define the sentence $\chi \in$ TL that is to be equivalent with ϕ:

$$\chi := \mathop{!}_{i\leq q} x_i \mathop{!}_{\substack{i\leq m\\ j\leq k(i)}} y^i_j \mathop{\exists^{\prec}}_{\substack{i\leq m\\ p(i)=1}} z_i \mathop{\forall^{\prec}}_{\substack{i\leq m\\ p(i)=2}} z_i \;\ldots\; \mathop{\exists^{\prec}}_{\substack{i\leq m\\ p(i)=p}} z_i \Bigg(\sim\Big(\bigwedge_{\substack{i\leq m\\ p(i) \text{ even}}} \chi^1_i \wedge \bigwedge_{\substack{i\leq m\\ p(i) \text{ even}}} \chi^2_i \Big) \vee$$
$$\Bigg(\bigwedge_{\substack{i\leq m\\ p(i) \text{ odd}}} \chi^1_i \wedge \bigwedge_{\substack{i\leq m\\ p(i) \text{ odd}}} \chi^2_i \wedge \Big(\bigotimes_{\substack{i\leq m\\ j\leq k(i)}} \chi^3_{i,j} \otimes \psi^*(t_i \mapsto z_i)_{i\leq m} \Big) \Bigg) \Bigg),$$

where ψ^* is obtained from ψ with the same syntactic mapping as in the translation of FOD to TL,[4] and, for $i \leq m$ and $j \leq k(i)$,

$$\begin{aligned} \chi^1_i &:= (y^i_1 \ldots y^i_{k(i)}) \rightarrowtail z_i; \\ \chi^2_i &:= \bigotimes_{j\leq k(i)} \neg\big(y^i_j = y^{o(i)}_j\big) \otimes z_i = z_{o(i)}; \\ \chi^3_{i,j} &:= \neg\big(y^i_j = t^i_j\big)(t_k \mapsto z_k)_{k\leq m}. \end{aligned}$$

All y^i_j and z_i are new variable symbols. Note that the variables z_i that correspond to the same function symbol's different occurrences appear consequtively quantified in χ. The quantifications of the variables z_i appear in the order given by $\prec$ on the indices i.

Here is an intuitive explanation of how to read χ. The quantifier block $!_{i\leq q}\, x_i$ is the team logic equivalent of the first order universal block $\forall_{i\leq q}\, x_i$ in ϕ. In addition to these q names for arbitrary elements, we are also provided with names y^i_j that refer to the elements that are the arguments to the various functions $f^{p(i)}_{n(i)}$ quantified in ϕ. Each quantifier block $\exists\, z_i$ and $\forall\, z_i$ in χ corresponds to a function quantifier block $\exists\, f^{p(i)}_{n(i)}$ in ϕ. Whereas $f^{p(i)}_{n(i)}$ is a name for a function, z_i is a name for the value that the function gives for a sequence of arguments. This is the fundamental difference in how second order logic and team logic handle second order objects. Each χ^1_i specifies which arguments y^i_j are needed for the function value z_i. Each χ^2_i ensures that the function value z_i is the same as $z_{o(i)}$ whenever the arguments for both are the same. We need this because for each occurrence of function variable $f^{p(i)}_{n(i)}$ we have a new function value z_i whose value does not *a priori* correlate with the corresponding values for other occurrences of the function

[4]In fact, ψ^* is not necessarily a formula of any logic. It may contain terms from second order logic and connectives and quantifiers from team logic. This is not a problem because ψ^* is nothing but an intermediate step in a series of syntactic transformations.

variable. Each $\chi^3_{i,j}$ eliminates one case of wrong values for the arguments y^i_j with respect to what the occurrences of $f^{p(i)}_{n(i)}$ in ψ expect.

To see why ϕ and χ are equivalent, let $\mathcal{M}$ be a model such that $\mathcal{M} \models \phi$. Then player $\mathbf{II}$ has a winning strategy in the semantic game $\eth^{\mathrm{SO}}(\mathcal{M}, \phi)$. Call this strategy σ^{SO}. We will describe a strategy σ^{TL} for player $\mathbf{II}$ in $\eth^{\mathrm{TL}}(\mathcal{M}, \chi)$ which will turn out to be a winning strategy. The idea in defining σ^{TL} is that player $\mathbf{II}$ plays $\eth^{\mathrm{SO}}(\mathcal{M}, \phi)$ using her winning strategy σ^{SO} and uses it as a source of useful information for her choices in $\eth^{\mathrm{TL}}(\mathcal{M}, \chi)$. We refer to the play of $\eth^{\mathrm{TL}}(\mathcal{M}, \chi)$ merely as *the play* and to the play of $\eth^{\mathrm{SO}}(\mathcal{M}, \phi)$ as *the shadowplay.* Denote $X' := (x_i \mapsto M)_{i \leq q}\big(y^i_j \mapsto M\big)_{i \leq m, j \leq k(i)}$.

In the beginning, while the play concerns quantifiers, player $\mathbf{II}$ maintains on her behalf that when the position in the play is of the form $(Qz_i\chi', X, \mathbf{II})$, where Q is either $\exists$ or $\forall$, then the following condition holds.

(C1) The shadowplay is at position $\big(Qf^{p(i)}_{n(i)}\phi', s, \mathbf{II}\big)$ or $(\phi', s, \mathbf{II})$, and $X = X'(z_j \mapsto F_j)_{j \prec i}$ such that for each $j \prec i$, $s\big(f^{p(j)}_{n(j)}\big)$ is similar to F_j via $\big(y^j_1, \ldots, y^j_{k(j)}\big)$.

However, condition (C1) can become false by a move of player $\mathbf{I}$. If this happens, player $\mathbf{II}$ will then have an easy victory ahead, as will be seen below.

The play of $\eth^{\mathrm{TL}}(\mathcal{M}, \chi)$ starts at the position $(\chi, \{\emptyset\}, \mathbf{II})$. First there are several shriek moves where neither player makes choices. After these the play is at position $(\exists z_i\chi', X', \mathbf{II})$, where $p(i) = 1$, $n(i) = 1$, and the shadowplay is still at the starting position $(\exists f^1_1\phi', \emptyset, \mathbf{II})$. We see that condition (C1) holds.

Assume then that the play is at position $P := (\exists z_i\chi', X, \mathbf{II})$ and condition (C1) holds. Then there is a subformula $\exists f^{p(i)}_{n(i)}\phi'$ of ϕ such that the shadowplay is at position P_1 or P_2, where

$$P_1 := \big(\exists f^{p(i)}_{n(i)}\phi', s, \mathbf{II}\big); \qquad\qquad P_2 := (\phi', s, \mathbf{II}).$$

If the shadowplay is at P_1, then we play one move in it by choosing for player $\mathbf{II}$ the $k(i)$-ary function $g := \sigma^{\mathrm{SO}}(P_1)$. Otherwise the shadowplay is at P_2, and we can define $g := s\big(f^{p(i)}_{n(i)}\big)$. In either case, let $F\colon X \to M$ be similar to g via $\big(y^i_1, \ldots, y^i_{k(i)}\big)$ and let $\sigma^{\mathrm{TL}}(P) := F$. The play proceeds now to position $\big(\chi', X(z_i \mapsto F), \mathbf{II}\big)$, and the shadowplay is at $\big(\phi', s\big(f^{p(i)}_{n(i)} \mapsto g\big), \mathbf{II}\big)$. We can see that condition (C1) holds.

Assume then that the play is at position $P := (\forall z_i\chi', X, \mathbf{II})$ and condition (C1) holds. Then there is a subformula $\forall f^{p(i)}_{n(i)}\phi'$ of ϕ such that the shadowplay is at position P_1 or P_2, where

$$P_1 := \big(\forall f^{p(i)}_{n(i)}\phi', s, \mathbf{II}\big); \qquad\qquad P_2 := (\phi', s, \mathbf{II}).$$

Let $F\colon X \to M$ be the function chosen by player **I** in the play. If $\mathcal{M}, X(z_i \mapsto F) \not\models (y^i_1 \dots y^i_{k(i)}) \rightsquigarrow z_i$, then player **I** has broken condition (C1) and thus player **I** can play arbitrarily all the remaining quantifier moves claiming a victory later at χ^1_i, see below. Otherwise, let $g\colon M^{k(i)} \to M$ be similar to F via $(y^i_1, \dots, y^i_{k(i)})$. If the shadowplay is at P_1, then we play one move in it by choosing g for player **I**. The play is then at $\big(\chi', X(z_i \mapsto F), \mathbf{II}\big)$, the shadowplay is at $\big(\phi', s(f^{p(i)}_{n(i)} \mapsto g), \mathbf{II}\big)$, and we can see that condition (C1) holds. Otherwise the shadowplay is at P_2. Then $s\big(f^{p(i)}_{n(i)}\big)$ is defined and similar to $F_{o(i)}$ via $\big(y^{o(i)}_1, \dots, y^{o(i)}_{k(o(i))}\big)$. If $s\big(f^{p(i)}_{n(i)}\big)$ is not similar to F via $(y^i_1, \dots, y^i_{k(i)})$, then player **I** has broken condition (C1), and thus player **II** can play arbitrarily all the remaining quantifier moves claiming a victory later at χ^2_i, see below. Otherwise the play is now at $\big(\chi', X(z_i \mapsto F), \mathbf{II}\big)$ and the shadowplay is at $\big(\phi', s, \mathbf{II}\big)$, and we can see that condition (C1) holds.

Finally the play reaches position $P := (\chi_0, X, \mathbf{II})$ and the shadowplay reaches position

$$P_0 := (\forall x_1 \dots \forall x_q \psi, s_0, \mathbf{II}),$$

where $X = X'(z_i \mapsto F_i)_{i \le m}$ and, for each $i \le m$, $s_0\big(f^{p(i)}_{n(i)}\big)$ is similar to F_i via $\big(y^i_1, \dots, y^i_{k(i)}\big)$, and

$$\chi_0 := {\sim}\Big(\bigwedge_{\substack{i \le m \\ p(i)\text{ even}}} \chi^1_i \wedge \bigwedge_{\substack{i \le m \\ p(i)\text{ even}}} \chi^2_i\Big) \vee \Bigg(\bigwedge_{\substack{i \le m \\ p(i)\text{ odd}}} \chi^1_i \wedge \bigwedge_{\substack{i \le m \\ p(i)\text{ odd}}} \chi^2_i \wedge \Big(\bigotimes_{\substack{i \le m \\ j \le k(i)}} \chi^3_{i,j} \otimes \psi^*(t_i \mapsto z_i)_{i \le m}\Big)\Bigg).$$

If player **I** broke condition (C1) above, there is $i \le m$ such that $p(i)$ is even, and either (5.7) or (5.8) holds.

$$\mathcal{M}, X \not\models (y^i_1 \dots y^i_{k(i)}) \rightsquigarrow z_i \tag{5.7}$$

$$s_0\big(f^{p(i)}_{n(i)}\big) \text{ is similar to } F_{o(i)} \text{ via } \big(y^{o(i)}_1, \dots, y^{o(i)}_{k(o(i))}\big) \text{ but not similar to } F_i \text{ via } \big(y^i_1, \dots, y^i_{k(i)}\big) \tag{5.8}$$

In such a case, let $\sigma^{\mathrm{TL}}(P)$ be the left disjunct of χ_0, and let player **II** move accordingly. After the following complement move the play is at $P := \big(\bigwedge_j \chi^1_j \wedge \bigwedge_j \chi^2_j, X, \mathbf{I}\big)$. If (5.7) holds, let $\sigma^{\mathrm{TL}}(P')$ be the conjunct χ^1_i, whence the play proceeds to position $\big((y^i_1 \dots y^i_{k(i)}) \rightsquigarrow z_i, X, \mathbf{I}\big)$ where player **II** wins. If (5.8) holds, let $\sigma^{\mathrm{TL}}(P')$ be the conjunct χ^2_i, whence the play proceeds to position $P'' := \big(\bigotimes_j \neg\big(y^i_j = y^{o(i)}_j\big) \otimes z_i = z_{o(i)}, X, \mathbf{I}\big)$. Player **I** chooses some sequence $(X_j)_{j \le k(i)+1}$. If for some $j \le k(i)$ there is $s \in X_j$ such that $s\big(y^i_j\big) = s\big(y^{o(i)}_j\big)$, then let $\sigma^{\mathrm{TL}}(P'')$ be j, whence the play proceeds to

$\big(\neg\big(y^i_j = y^{o(i)}_j\big), X_j, \mathbf{I}\big)$, and player $\mathbf{II}$ wins. Otherwise, letting s be an assignment that fails the similarity of $s_0\big(f^{p(i)}_{n(i)}\big)$ to F_i, that is, $s\big(y^i_j\big) = s\big(y^{o(i)}_j\big)$ for all $j \leq k(i)$ and $s(z_i) \neq s(z_{o(i)})$, we have $s \in X_{k(i)+1}$. Let $\sigma^{\mathrm{TL}}(P'')$ be $k(i)+1$, whence the play proceeds to $(z_i = z_{o(i)}, X_{k(i)+1}, \mathbf{I})$, and player $\mathbf{II}$ wins.

If player $\mathbf{I}$ did not break condition (C1), let $\sigma^{\mathrm{TL}}(P)$ be the right disjunct of χ_0. The play is then at $\big(\bigwedge_i \chi^1_i \wedge \bigwedge_i \chi^2_i \wedge \big(\bigotimes_{i,j} \chi^3_{i,j} \otimes \psi^*(t_i \mapsto z_i)_{i\leq m}\big), X, \mathbf{II}\big)$ and it is for player $\mathbf{I}$ to choose a conjunct. If player $\mathbf{I}$ chooses χ^1_i for some $i \leq m$ with $p(i)$ even, the play proceeds to $\big((y^i_1 \dots y^i_{k(i)}) \rightarrowtail z_i, X, \mathbf{II}\big)$ and player $\mathbf{II}$ wins because of condition (C1). If player $\mathbf{I}$ chooses χ^2_i for some $i \leq m$ with $p(i)$ even, the play proceeds to

$$P' := \Big(\bigotimes_{j\leq k(i)} \neg\big(y^i_j = y^{o(i)}_j\big) \otimes z_i = z_{o(i)}, X, \mathbf{II}\Big).$$

Let then $\sigma^{\mathrm{TL}}(P')$ be $(X_j)_{j\leq k(i)+1}$, where each $X_j := \big\{s \in X : s\big(y^i_j\big) \neq s\big(y^{o(i)}_j\big)\big\}$, for $j \leq k(i)$, and $X_{k(i)+1} := X \setminus \big(\bigcup_{j\leq k(i)} X_j\big)$. Note that because of condition (C1), every $s \in X_{k(i)+1}$ has $s(z_i) = s\big(z_{o(i)}\big)$. If player $\mathbf{I}$ chooses some $j \leq k(i)$, the play proceeds to $\big(\neg\big(y^i_j = y^{o(i)}_j\big), X_j, \mathbf{II}\big)$, and player $\mathbf{II}$ wins. Otherwise the play proceeds to $\big(z_i = z_{o(i)}, X_{k(i)+1}, \mathbf{II}\big)$, and player $\mathbf{II}$ wins again.

If player $\mathbf{I}$ chooses the last remaining conjunct, the play proceeds to

$$P := \Big(\bigotimes_{\substack{i\leq m\\ j\leq k(i)}} \chi^3_{i,j} \otimes \psi^*(t_i \mapsto z_i)_{i\leq m}, X, \mathbf{II}\Big).$$

Let then $\sigma^{\mathrm{TL}}(P)$ be $(X^1_1, \dots, X^1_{k(1)},\ X^2_1, \dots, X^2_{k(2)},\ \dots,\ X^m_1, \dots, X^m_{k(m)},\ X_0)$, where each $X^i_j := \{s \in X : \mathcal{M}, s \models (y^i_j \neq t^i_j)(t_i \mapsto z_i)_{i\leq m}\}$ and $X_0 := X \setminus \big(\bigcup_{i\leq m, j\leq k(i)} X^i_j\big)$. If player $\mathbf{I}$ chooses any but the last index of the sequence, the play proceeds to position $\big(\neg\big(y^i_j = t^i_j\big)(t_i \mapsto z_i)_{i\leq m}, X^i_j, \mathbf{II}\big)$, for some $i \leq m$ and $j \leq k(i)$, and player $\mathbf{II}$ wins. Otherwise the play proceeds to

$$P := \big(\psi^*(t_i \mapsto z_i)_{i\leq m}, X_0, \mathbf{II}\big).$$

For the rest of the play, player $\mathbf{II}$ and her strategy σ^{TL} will maintain the following condition that will guide her to victory. If θ is a subformula of ψ and the play is at position $P' := \big(\theta^*(t_i \mapsto z_i)_{i\leq m}, Y, \mathbf{II}\big)$, then

(C2) Y is the set of assignments $s \in X_0$ such that the shadowplay can reach position $P'_s := \big(\theta, s_0(x_i \mapsto s(x_i))_{i\leq q}, \mathbf{II}\big)$ from P_0 when $\mathbf{II}$ plays by σ^{SO}.

At first, the play is at P and the shadowplay is at P_0, and we can see that (C2) holds.

Assume then that the play is at $P' := \big((\theta_1 \wedge \theta_2)^*(t_i \mapsto z_i)_{i\le m}, Y, \mathbf{II}\big)$ and (C2) holds. As $(\theta_1 \wedge \theta_2)^*(t_i \mapsto z_i)_{i\le m}$ is ${\theta_1}^*(t_i \mapsto z_i)_{i\le m} \wedge {\theta_2}^*(t_i \mapsto z_i)_{i\le m}$, the move is for player **I**, so assume he chooses $j \in \{1,2\}$. The play proceeds to $P'' := \big(\theta_j^*(t_i \mapsto z_i)_{i\le m}, Y, \mathbf{II}\big)$. For each $s \in Y$, the shadowplay can reach P'_s from P_0 when **II** plays by σ^{SO}, and from P'_s player **I** may choose the j'th conjunct, making the shadowplay proceed to P''_s. On the other hand, if the shadowplay can reach position P''_s from P_0 when **II** plays by σ^{SO}, for some s, then the shadowplay must have previously been in position P'_s, whence $s \in Y$. Therefore condition (C2) holds at P''.

Assume then that the play is at $P' := \big((\theta_1 \vee \theta_2)^*(t_i \mapsto z_i)_{i\le m}, Y, \mathbf{II}\big)$ and (C2) holds. As $(\theta_1 \vee \theta_2)^*(t_i \mapsto z_i)_{i\le m}$ is ${\theta_1}^*(t_i \mapsto z_i)_{i\le m} \otimes {\theta_2}^*(t_i \mapsto z_i)_{i\le m}$, the move is for player **II**. For each $s \in Y$, the shadowplay can reach P'_s from P_0 when **II** plays by σ^{SO}. P'_s yields a disjunction move for player **II**. Then $\sigma^{\mathrm{SO}}(P'_s) \in \{1,2\}$. Let $\sigma^{\mathrm{TL}}(P')$ be (Y_1, Y_2), where each $Y_j := \{s \in Y : \sigma^{\mathrm{SO}}(P'_s) = j\}$ for $j \in \{1,2\}$, and let player **II** move accordingly. Player **I** will then choose some $j \in \{1,2\}$ and the play proceeds to $P'' := \big(\theta_j^*(t_i \mapsto z_i)_{i\le m}, Y_j, \mathbf{II}\big)$. We can see that (C2) holds at P''.

Assume finally that the play is at $P' := \big(\theta^*(t_i \mapsto z_i)_{i\le m}, Y, \mathbf{II}\big)$, θ is an atomic or negated atomic formula, and (C2) holds. As θ^* is θ, also $\theta^*(t_i \mapsto z_i)_{i\le m}$ is atomic or negated atomic and the play ends. But who is the winner? For each $s \in Y$, the shadowplay can reach P'_s from P_0 when **II** plays by σ^{SO}. Player **II** wins these shadowplays, which gives $\mathcal{M}, s_0(x_i \mapsto s(x_i))_{i\le q} \models \theta$ for all $s \in Y$.

We prove by induction on terms t that occur in ϕ that for all $s \in Y$, $s_1(t) = s(t^*)$, where $t^* := t(t_i \mapsto z_i)_{i\le m}$ and $s_1 := s_0(x_i \mapsto s(x_i))_{i\le q}$. If t is a variable, it is some x_i. Then also $t^* = x_i$, and thus $s_1(t) = s(t^*)$. Function terms can be based either on a function symbol in the language L or on a function variable. If $f \in L$ and t is $fu_1 \dots u_k$ for some terms $u_1, \dots, u_k$, then $t^* = fu_1^* \dots u_k^*$ and we can assume that the claim holds for $u_1, \dots, u_k$. Then we have

$$\begin{aligned} s_1(t) &= f^{\mathcal{M}}\big(s_1(u_1), \dots, s_1(u_k)\big) \\ &= f^{\mathcal{M}}\big(s(u_1^*), \dots, s(u_k^*)\big) \\ &= s(t^*). \end{aligned}$$

If t is $fu_1 \dots u_k$, where f is a function variable, then (because we only consider terms that occur in ϕ) t is one of the terms t_i, $i \le m$, so $t^* = z_i$, $k = k(i)$, each u_j is t^i_j, and $s_1(f)$ is similar to F_i via $\big(y^i_1, \dots, y^i_k\big)$. By $Y \subseteq X_0$, $s(u_j^*) = s\big(y^i_j\big)$ for all $j \le k$. We can assume that the claim holds for $u_1, \dots, u_k$. Then we have

$$\begin{aligned}
s_1(t) &= s_1(f)\big(s_1(u_1),\dots,s_1(u_k)\big)\\
&= s_1(f)\big(s(u_1^*),\dots,s(u_k^*)\big)\\
&= s_1(f)\big(s(y_1^i),\dots,s(y_k^i)\big)\\
&= F_i(s)\\
&= s(z_i).
\end{aligned}$$

Thus $\mathcal{M}, s \models \theta^*(t_i \mapsto z_i)_{i\le m}$ for all $s \in Y$, player $\mathbf{II}$ wins the play, and σ^{TL} is a winning strategy.

For the other direction, assume $\mathcal{M} \models \chi$, that is, player $\mathbf{II}$ has a winning strategy in the semantic game $\Game^{\mathrm{TL}}(\mathcal{M},\chi)$. Call this strategy σ^{TL}. We will describe a strategy σ^{SO} for player $\mathbf{II}$ in $\Game^{\mathrm{SO}}(\mathcal{M},\phi)$ which will turn out to be a winning strategy. The idea behind defining σ^{SO} is that player $\mathbf{II}$ plays $\Game^{\mathrm{TL}}(\mathcal{M},\chi)$ by her winning strategy σ^{TL} and extracts useful information for her survival in $\Game^{\mathrm{SO}}(\mathcal{M},\phi)$. We call the play in $\Game^{\mathrm{SO}}(\mathcal{M},\phi)$ *the play* and the play in $\Game^{\mathrm{TL}}(\mathcal{M},\chi)$ *the shadowplay*. As before, let $X' := (x_i \mapsto M)_{i\le q}\big(y_j^i \mapsto M\big)_{i\le m, j\le k(i)}$. Let $o(i,j)$ be the $\prec$-least index that satisfies $p\big(o(i,j)\big) = i$ and $n\big(o(i,j)\big) = j$. In words, $z_{o(i,j)}$ is the outermost quantified variable in χ that corresponds to the function variable symbol f_j^i in ϕ. Denote $k(i,j) := k\big(o(i,j)\big)$, the arity of the function symbol f_j^i, for $i \le p$ and $j \le n_i$.

In the beginning of the game, while the play concerns quantifiers, player $\mathbf{II}$ maintains that when the position in the play is of the form $\big(Qf_j^i\phi', s, \mathbf{II}\big)$, where Q is either $\exists$ or $\forall$, then the following condition holds.

(D1) There is a subformula $Qz_{i_1}\dots Qz_{i_l}\chi'$ of χ, where $i_1 \prec \dots \prec i_l$ are all the indices $i' \le m$ for which $p(i') = i$ and $n(i') = j$ hold, and the shadowplay is at position $(Qz_{i_1}\dots Qz_{i_l}\chi', X, \mathbf{II})$ for some team X such that $X = X'(z_{i'} \mapsto F_{i'})_{i' \prec o(i,j)}$, where $s\big(f_{n(i')}^{p(i')}\big)$ is similar to $F_{i'}$ via $\big(y_1^{i'},\dots,y_{k(i')}^{i'}\big)$ for all $i' \prec o(i,j)$.

The play of $\Game^{\mathrm{SO}}(\mathcal{M},\phi)$ starts at position $(\exists f_1^1\phi', \emptyset, \mathbf{II})$. The shadowplay starts at $(\chi, \{\emptyset\}, \mathbf{II})$, and after the initial shriek moves the shadowplay is at $\big(\exists z_{o(1,1)}\chi', X', \mathbf{II}\big)$. We see that condition (D1) holds.

Assume then that the play is at position $P := \big(\exists f_j^i\phi', s, \mathbf{II}\big)$ and the condition (D1) holds. Then the shadowplay is at $P_1 := (\exists z_{i_1}\dots\exists z_{i_l}\chi', X, \mathbf{II})$, as stated in (D1). Note that $i_1 = o(i,j)$. Let player $\mathbf{II}$ make l moves in the shadowplay positions

$$P_{l'} := \big(\exists z_{i_{l'}}\dots\exists z_{i_l}\chi', X(z_{i_1} \mapsto F_{i_1})\cdots(z_{i_{l'-1}} \mapsto F_{i_{l'-1}}), \mathbf{II}\big),$$

for $l' \leq l$, by choosing $F_{i_{l'}} := \sigma^{\mathrm{TL}}(P_{l'})$ for each $l' \leq l$. Let g be a $k(i,j)$-ary function that is similar to F_{i_1} via $\big(y_1^{i_1}, \dots, y_{k(i_1)}^{i_1}\big)$. Define $\sigma^{\mathrm{SO}}(P)$ to be g. The play proceeds accordingly to $\big(\phi', s\big(f_j^i \mapsto g\big), \mathbf{II}\big)$ and the shadowplay is at $\big(\chi', X(z_{i_{l'}} \mapsto F_{i_{l'}})_{l' \leq l}, \mathbf{II}\big)$. Because σ^{TL} is a winning strategy, $s\big(f_j^i\big)$ is similar to $F_{i_{l'}}$ via $\big(y_1^{l'}, \dots, y_{k(l')}^{l'}\big)$ for all $l' \leq l$, as we will see below when the shadowplay reaches formulas $\chi_{i_{l'}}^1$ and $\chi_{i_{l'}}^2$. Thus we can see that condition (D1) holds.

Assume then that the play is at position $P := \big(\forall f_j^i \phi', s, \mathbf{II}\big)$ and condition (D1) holds. Then the shadowplay is at $P_1 := (\forall z_{i_1} \dots \forall z_{i_l} \chi', X, \mathbf{II})$, as stated in (D1). Note that $i_1 = o(i,j)$. Denote by g the function chosen by player $\mathbf{I}$ in the play. For each $l' \leq l$, let $F_{i_{l'}} : X \to M$ be similar to g via $\big(y_1^{i_{l'}}, \dots, y_{k(i,j)}^{i_{l'}}\big)$. In the shadowplay, let player $\mathbf{I}$ choose $F_{i_{l'}}$ at position

$$P_{l'} := \big(\forall z_{i_{l'}} \dots \forall z_{i_l} \chi', X(z_{i_1} \mapsto F_{i_1}) \cdots (z_{i_{l'-1}} \mapsto F_{i_{l'-1}}), \mathbf{II}\big)\,,$$

for each $l' \leq l$. The play is then at $\big(\phi', s(f_j^i \mapsto g), \emptyset, \mathbf{II}\big)$, the shadowplay is at $\big(\chi', X(z_{i_{l'}} \mapsto F_{i_{l'}})_{l' \leq l}, \mathbf{II}\big)$, and we can see that condition (D1) holds.

Finally the play reaches position $(\forall x_1 \dots \forall x_q \psi, s', \mathbf{II})$ and the shadowplay reaches position $(\chi_0, X, \mathbf{II})$, where $X = X'(z_i \mapsto F_i)_{i \leq m}$ and for all $i \leq m$, $s'\big(f_{n(i)}^{p(i)}\big)$ is similar to F_i via $\big(y_1^i, \dots, y_{k(i)}^i\big)$. After player $\mathbf{I}$ has made his q moves in the play, the position becomes

$$P := (\psi, s, \mathbf{II})$$

for some assignment s that extends s' to variables $x_1, \dots, x_q$.

In the shadowplay, player $\mathbf{II}$'s strategy σ^{TL} will choose the right side disjunct. This we see by noting that player $\mathbf{I}$ would win at any of the positions $(\chi_i^1, X, \mathbf{I})$ or $(\chi_i^2, X, \mathbf{I})$, when $i \leq m$ and $p(i)$ is even, due to the fact that $s\big(f_{n(i)}^{p(i)}\big)$ is similar to F_i via $\big(y_1^i, \dots, y_{k(i)}^i\big)$ for all $i \leq m$ where $p(i)$ is even. The shadowplay is then at $P_0 := \big(\bigwedge_i \chi_i^1 \wedge \bigwedge_i \chi_i^2 \wedge \big(\bigotimes_{i,j} \chi_{i,j}^3 \otimes \psi^*(t_i \mapsto z_i)_{i \leq m}\big), X, \mathbf{II}\big)$ and it is for player $\mathbf{I}$ to choose a conjunct. If player $\mathbf{I}$ chooses χ_i^1 for some $i \leq m$ with $p(i)$ odd, the play proceeds to $\big((y_1^i, \dots, y_{k(i)}^i\big) \rightsquigarrow z_i, X, \mathbf{II}\big)$, and player $\mathbf{II}$ wins by her winning strategy σ^{TL}. Therefore we have $\mathcal{M}, X \models \big(y_1^i \dots y_{k(i)}^i\big) \rightsquigarrow z_i$. If player $\mathbf{I}$ chooses χ_i^2 for some $i \leq m$ with $p(i)$ odd, the shadowplay proceeds to

$$P_1 := \Big(\bigotimes_{j \leq k(i)} \neg\big(y_j^i = y_j^{o(i)}\big) \otimes z_i = z_{o(i)}, X, \mathbf{II}\Big).$$

Let $Y := \big\{s \in X : s\big(y_j^i\big) = s\big(y_j^{o(i)}\big) \text{ for all } j \leq k(i)\big)\big\}$ and denote by $(X_j)_{j \leq k(i)+1}$ the move $\sigma^{\mathrm{TL}}(P_1)$ that player $\mathbf{II}$ makes. If player $\mathbf{I}$ chooses some index $j \leq k(i)$, the shadowplay proceeds to $\big(\neg\big(y^i_j = y^{o(i)}_j\big), X_j, \mathbf{II}\big)$, and player **II** wins by her winning strategy. This gives $Y \cap X_j = \emptyset$ and further $Y \subseteq X_{k(i)+1}$. If player **I** chooses the index $k(i)+1$, the shadowplay proceeds to $(z_i = z_{o(i)}, X_{k(i)+1}, \mathbf{II})$, and player **II** wins again. Because $Y \subseteq X_{k(i)+1}$, we get $\mathcal{M}, Y \models z_i = z_{o(i)}$. In other words, for all $i \leq m$, $s\big(f^{p(i)}_{n(i)}\big)$ is similar to F_i via $\big(y^i_1, \dots, y^i_{k(i)}\big)$, as we stated above.

If player **I** chooses the last remaining conjunct at P_0, the shadowplay proceeds to

$$P_1 := \Big(\bigotimes_{\substack{i \leq m \\ j \leq k(i)}} \chi^3_{i,j} \otimes \psi^*(t_i \mapsto z_i)_{i \leq m}, X, \mathbf{II}\Big).$$

Denote by $(X^1_1, \dots, X^1_{k(1)},\ X^2_1, \dots, X^2_{k(2)},\ \dots,\ X^m_1, \dots, X^m_{k(m)},\ X_0)$ the choice of player **II** at P_1, $\sigma^{\mathrm{TL}}(P_1)$. If player **I** chooses the last index of the sequence, the shadowplay proceeds to

$$P_2 := \big(\psi^*(t_i \mapsto z_i)_{i \leq m}, X_0, \mathbf{II}\big).$$

For the rest of the play, player **II** and her strategy σ^{SO} will maintain the following condition that will guide her to victory. If θ is a subformula of ψ and the play is at position $(\theta, s, \mathbf{II})$, then the following condition holds.

(D2) The shadowplay is at position $\big(\theta^*(t_i \mapsto z_i)_{i \leq m}, Z, \mathbf{II}\big)$ for some Z such that $s \restriction \{x_1, \dots, x_q\} \in Z \restriction \{x_1, \dots, x_q\}$.

At first, the play is at P and the shadowplay is at P_2, and we can see that (D2) holds.

Assume then that the play is at $(\theta_1 \wedge \theta_2, s, \mathbf{II})$ and (D2) holds. Then the shadowplay is at $\big((\theta_1 \wedge \theta_2)^*(t_i \mapsto z_i)_{i \leq m}, Z, \mathbf{II}\big)$, and as $(\theta_1 \wedge \theta_2)^*(t_i \mapsto z_i)_{i \leq m}$ is ${\theta_1}^*(t_i \mapsto z_i)_{i \leq m} \wedge {\theta_2}^*(t_i \mapsto z_i)_{i \leq m}$, the move is for player **I**. In the play, player **I** chooses some $j \in \{1, 2\}$ and the play proceeds to $\big(\theta_i, s, \mathbf{II}\big)$. In the shadowplay, let player **I** choose similarly ${\theta_i}^*(t_i \mapsto z_i)_{i \leq m}$, so that the shadowplay proceeds to $\big({\theta_i}^*(t_i \mapsto z_i)_{i \leq m}, Z, \mathbf{II}\big)$. We can see that (D2) still holds.

Assume then that the play is at $P' := (\theta_1 \vee \theta_2, s, \mathbf{II})$ and (D2) holds. Then the shadowplay is at position $P_3 := \big((\theta_1 \vee \theta_2)^*(t_i \mapsto z_i)_{i \leq m}, Z, \mathbf{II}\big)$, and as $(\theta_1 \vee \theta_2)^*(t_i \mapsto z_i)_{i \leq m}$ is ${\theta_1}^*(t_i \mapsto z_i)_{i \leq m} \otimes {\theta_2}^*(t_i \mapsto z_i)_{i \leq m}$, the move is for player **II**. Denote by (Z_1, Z_2) the choice of player **II** in the shadowplay, that is, $\sigma^{\mathrm{TL}}(P_3)$, and let player **I** choose the index $j \in \{1, 2\}$ that satisfies $s \in Z_j$. The shadowplay proceeds to $\big({\theta_j}^*(t_i \mapsto z_i)_{i \leq m}, Z_j, \mathbf{II}\big)$. Let $\sigma^{\mathrm{SO}}(P')$ be θ_j, and let player **II** play accordingly. The play proceeds to $(\theta_j, s, \mathbf{II})$. We can see that (D2) still holds.

Assume finally that the play is at $(\theta, s, \mathbf{II})$, θ is an atomic or negated atomic formula, and (D2) holds. Then the shadowplay is at some position $\big(\theta^*(t_i \mapsto z_i)_{i\leq m}, Z, \mathbf{II}\big)$. As θ^* is θ, also $\theta^*(t_i \mapsto z_i)_{i\leq m}$ is atomic or negated atomic, so both the play and the shadowplay end. Who wins the play? Because player $\mathbf{II}$ has played by her winning strategy in the shadowplay, $\mathcal{M}, s' \models \theta(t_i \mapsto z_i)_{i\leq m}$ for all $s' \in Z$. By (D2) there is some particular $s' \in Z$ such that $s\restriction\{x_1, \dots, x_q\} = s'\restriction\{x_1, \dots, x_q\}$. Note that $\theta^*(t_i \mapsto z_i)_{i\leq m}$ does not contain occurrences of variables y^i_j. Thus we have $\mathcal{M}, s'' \models \theta^*(t_i \mapsto z_i)_{i\leq m}$, where

$$s'' := s'\big(y^i_j \mapsto s'\big(t^i_j(t_i \mapsto z_i)_{i\leq m}\big)\big)_{\substack{i\leq m\\ j\leq k(i)}}.$$

We prove by induction on terms t that occur in ϕ that we have $s(t) = s''(t^*)$, where $t^* := t(t_i \mapsto z_i)_{i\leq m}$. If t is a variable, it is some x_i. Then also $t^* = x_i$, and thus $s(t) = s''(t^*)$. If f is a function symbol in the language of ϕ and t is $fu_1 \dots u_k$ for some terms $u_1, \dots, u_k$, then $t^* = fu^*_1 \dots u^*_k$ and we can assume that the claim holds for $u_1, \dots, u_k$. Then we have

$$\begin{aligned} s(t) &= f^{\mathcal{M}}\big(s(u_1), \dots, s(u_k)\big) \\ &= f^{\mathcal{M}}\big(s''(u^*_1), \dots, s''(u^*_k)\big) \\ &= s''(t^*). \end{aligned}$$

If t is $fu_1 \dots u_k$, where f is a function variable, then (because we only consider terms that occur in ϕ) t is one of the terms t_i, $i \leq m$, so $t^* = z_i$, $k = k(i)$, each u_j is t^i_j, and $s(f)$ is similar to F_i via $\big(y^i_1, \dots, y^i_k\big)$. By the choice of s'', $s''(u^*_j) = s''\big(y^i_j\big)$ for all $j \leq k$. We can assume that the claim holds for $u_1, \dots, u_k$. Then we have

$$\begin{aligned} s(t) &= s(f)\big(s(u_1), \dots, s(u_k)\big) \\ &= s(f)\big(s''(u^*_1), \dots, s''(u^*_k)\big) \\ &= s(f)\big(s''(y^i_1), \dots, s''(y^i_k)\big) \\ &= F_i(s'') \\ &= s''(z_i). \end{aligned}$$

Therefore $\mathcal{M}, s \models \theta$. Thus player $\mathbf{I}$ wins, and σ^{SO} is a winning strategy. □

We can simplify the previous theorem by assuming that the second order sentence is given in a nice form. The assumption on the form is based on known normal forms and is thus not a limitation of expressive power.

Corollary 5.3.4. *Let $\phi \in \mathrm{SO}$ be a sentence in Skolem normal form,*

$$\phi := \exists f^1_1 \dots \exists f^1_{n_1} \forall f^2_1 \dots \forall f^2_{n_2} \dots \exists f^p_1 \dots \exists f^p_{n_p} \forall x_1 \dots \forall x_q \psi,$$

such that each quantified function variable symbol f^i_j occurs in ϕ only as a unique term $t^i_j := f^i_j x^{i,j}_1 \ldots x^{i,j}_{k(i,j)}$. Then there is a sentence $\chi \in \mathrm{TL}$,

$$\chi := \mathop{!}_{i \leq q} x_i \mathop{\exists}_{j \leq n_1} z^1_j \mathop{\forall}_{j \leq n_2} z^2_j \ldots \mathop{\exists}_{j \leq n_p} z^p_j \left(\sim\Big(\bigwedge_{\substack{i \leq p \text{ even} \\ j \leq n_i}} \chi^i_j \Big) \vee \Big(\bigwedge_{\substack{i \leq p \text{ odd} \\ j \leq n_i}} \chi^i_j \wedge \psi^*(t^i_j \mapsto z^i_j)_{\substack{i \leq p \\ j \leq n_i}} \Big) \right),$$

where each $\chi^i_j := (x^{i,j}_1 \ldots x^{i,j}_{k(i,j)}) \rightsquigarrow z^i_j$, such that $\mathcal{M} \models \phi$ if and only if $\mathcal{M} \models \chi$ for all models $\mathcal{M}$.

Proof. The proof is similar to the proof of Theorem 5.3.3 with the exceptions that subformulas χ^2_i are not needed because different occurrences of a function variable always denote the same element in the universe, and subformulas $\chi^3_{i,j}$ are not needed because arguments for different occurrences of a function variable are uniquely determined by the function variable and are element variables. □

5.4 Translating SO-Formulas to TL

In this section, I show that team logic equals second order logic in expressive power over all formulas, generalising the result of the previous section. The result is joint work with Juha Kontinen [16].

The result of this section also generalises the previous result by Kontinen and Väänänen [17].

I present the result in less detail than the previous one in Section 5.3 because the structure of the proof is essentially the same.

The syntactic translation is presented in the proof. The semantic translation is the natural one.

Theorem 5.4.1. *For every $\phi(R) \in \mathrm{SO}$ with only the r-ary predicate symbol R free, there is $\psi(v_1, \ldots, v_r) \in \mathrm{TL}$ such that for all models $\mathcal{M}$ and teams X, $\mathcal{M}, X \models \psi$ if and only if $\mathcal{M}, s_X \models \phi$, where $s_X := (R \mapsto \mathrm{Rel}(X))$.*

Proof. We may assume that ϕ is in Skolem normal form

$$\exists f^1_1 \ldots \exists f^1_n \, \forall f^2_1 \ldots \forall f^2_n \; \ldots \; \exists f^p_1 \ldots \exists f^p_n \, \forall x_1 \ldots \forall x_n \, \theta', \tag{5.9}$$

where θ' is quantifier-free and in conjunctive normal form, the relation variable R occurs in θ' only in occurrences of the term $R\overline{x}$, where $\overline{x}$ denotes the sequence $x_1, \ldots, x_r$, and each function variable f^i_j occurs in θ' only in occurrences of the term $t^i_j := f^i_j u^{i,j}_1, \dots, u^{i,j}_{k(i,j)}$, where each $u^{i,j}_k$ is a variable. We can rewrite ϕ in the logically equivalent form

$$\phi := \exists_{j\le n} f^1_j \ \forall_{j\le n} f^2_j \ \dots \ \exists_{j\le n} f^p_j \ \forall_{j\le n} x_j \ \theta, \tag{5.10}$$

where n is possibly increased for the sake of obtaining two new function variables f^p_{n-1} and f^p_n which we shall simply call f_1 and f_2, and

$$\theta := \big(R\overline{x} \vee \neg(f_1\overline{x} = f_2\overline{x})\big) \wedge (\neg R\overline{x} \vee f_1\overline{x} = f_2\overline{x}) \wedge \theta',$$

where we replace the occurrences of $R\overline{x}$ in θ by $f_1\overline{x} = f_2\overline{x}$. Note that θ is in conjunctive normal form just like θ' and there are only one positive and one negative occurrence of $R\overline{x}$ in θ, either occurrence in its own conjunct.[5]

Let ψ be the team logic formula

$$\psi := \mathop{!}_{i\le n} x_i \ \exists_{j\le n} y^1_j \ \forall_{j\le n} y^2_j \ \dots \ \exists_{j\le n} y^p_j \Bigg(\sim\Big(\bigwedge_{\substack{i\le p \text{ even}\\ j\le n}} \chi^i_j\Big) \vee \Big(\bigwedge_{\substack{i\le p \text{ odd}\\ j\le n}} \chi^i_j \wedge \theta^*\Big)\Bigg),$$

where θ^* is obtained from θ with the same syntactic mapping as in the translation of FOD to TL and the following replacements of terms and subformulas;

$$\begin{aligned} t^i_j &\mapsto y^i_j \\ \neg R\overline{x} &\mapsto \bigotimes_{i\le r} \neg(v_i = x_i) \\ R\overline{x} &\mapsto \forall_{i\le r} z_i \Big(\bigvee_{i\le r} \sim() \rightsquigarrow z_i \vee \sim \bigotimes_{i\le r} \neg(v_i = z_i) \vee \bigotimes_{i\le r} \neg(x_i = z_i)\Big), \end{aligned}$$

where v_i and z_i, for $i \le r$, are new variables, and χ^i_j for $i \le p$ and $j \le n$ we define as

$$\chi^i_j := \big(u^{i,j}_1 \dots u^{i,j}_{k(i,j)}\big) \rightsquigarrow y^i_j.$$

We shall call y^p_{n-1} simply y_1 and y^p_n we call y_2, according to the notation of f_1 and f_2.

Let X be a team. Assume first that $\mathcal{M}, s_X \models \phi$, where $s_X := R \mapsto \mathrm{Rel}(X))$. We will show $\mathcal{M}, X \models \psi$. From $\mathcal{M}, s_X \models \phi$ we get that alternately for each odd $i \le p$ we can pick some particular sequence $g^i_1, \dots, g^i_n$ of functions

[5] A positive occurrence is one that has an even number of negations in front of it; a negative occurrence is one that has an odd number of negations in front of it. In this case, the positive occurrence has no negations and the negative occurrence has one.

such that whichever sequence we pick for each even $i \leq p$, it always results in $\mathcal{M}, s'_X \models \forall x_1 \dots \forall x_n \theta$, where $s'_X := s_X(f^i_j \mapsto g^i_j)_{i,j}$. We can translate this same function picking strategy to the side of team logic; for each i and j, let F^i_j map assignments like $F^i_j(s) = g^i_j\big(s(u^{i,j}_1), \dots, s(u^{i,j}_{k(i,j)})\big)$. If we can show that team Y satisfies the conditional subformula of ψ, where

$$Y := X(x_i \mapsto M)_{i \leq n}(y^i_j \mapsto F^i_j)_{i \leq p,\, j \leq n}, \tag{5.11}$$

we get that $\mathcal{M}, X \models \psi$. The team Y satisfies χ^i_j for each i and j. Therefore, in order for X to satisfy ψ, Y should satisfy θ^*, i.e. Y should satisfy each of the conjuncts in θ^*. Note that, for each $s \in Y$, $s\big(y^i_j\big) = s'\big(t^i_j\big)$, where $s' := s\big(f^i_j \mapsto g^i_j\big)$.

Note that in essence we are proving the powerful claim that, for arbitrary interpretations of the function variables f^i_j for all i and j, it holds that

$$\mathcal{M}, s'_X \models \forall x_1 \dots \forall x_n \theta \quad \text{if and only if} \quad \mathcal{M}, Y \models \theta^*. \tag{5.12}$$

There are three kinds of conjuncts in θ^*. We show that Y satisfies each kind. This is the implication from left to right in (5.12).

- To see that Y satisfies the formula that replaced $R\overline{x} \vee \neg(f_1\overline{x} = f_2\overline{x})$, consider any $\overline{a} \in Y$ and denote $s''_X := s'_X(x_i \mapsto a_i)_{i \leq n}$. From $\mathcal{M}, s''_X \models \theta$ we get either $\mathcal{M}, s''_X \models R\overline{x}$ or $\mathcal{M}, s''_X \models \neg(f_1\overline{x} = f_2\overline{x})$. We can then split $Y = Y_1 \cup Y_2$, where $Y_1 = \{s \in Y : (s(x_1), \dots, s(x_r)) \in \mathrm{Rel}(X)\}$ and $Y_2 = Y \setminus Y_1$ such that $s(y_1) \neq s(y_2)$ holds for all $s \in Y_2$. Thus $\mathcal{M}, Y_2 \models \neg(y_1 = y_2)$. We also have that, for all $s \in Y_1$ and all $(a_i)_{i \leq r} \in \mathrm{Rel}(X)$ there is some $s' \in Y_1$ such that $s'(v_i) = a_i$ for all $i \leq r$ and $s'(x_i) = s(x_i)$ for all $i \leq n$.

  Consider $Z := Y_1(z_i \mapsto F_i)_{i \leq r}$, where F_i are arbitrary. If any F_i is not a constant function, then $\mathcal{M}, Z \models {\sim}() {\rightsquigarrow} z_i$. Otherwise $Z = Y_1(z_i \mapsto a_i)_{i \leq r}$ for some $\overline{a} \in M$. If for all $s \in Z$ there is $i \leq r$ such that $s(x_i) \neq a_i$, then $\mathcal{M}, Z \models \bigotimes_{i \leq r} \neg(x_i = z_i)$. Otherwise there is $s \in Z$ such that $s(x_i) = a_i$ for all $i \leq r$. Then there is some $s' \in Z$, where $s'(v_i) = s(x_i)$ for all $i \leq r$ and $s'(z_i) = s(z_i)$ for all $i \leq r$. We have $s'(v_i) = s(x_i) = a_i = s'(z_i)$ for all $i \leq r$, whence $\mathcal{M}, Z \models {\sim} \bigotimes_{i \leq r} \neg(v_i = z_i)$. This shows that Y satisfies the conjunct.

- To see $\mathcal{M}, Y \models \bigotimes_{i \leq r} \neg(v_i = x_i) \otimes y_1 = y_2$, consider any $s \in Y$. As above, we get from $\mathcal{M}, s''_X \models \theta$ either $\mathcal{M}, s''_X \models \neg R\overline{x}$ or $\mathcal{M}, s''_X \models f_1\overline{x} = f_2\overline{x}$. We can then split $Y = Y_1 \cup Y_2$, where $Y_1 := \{s \in Y : (s(x_1), \dots, s(x_r)) \notin \mathrm{Rel}(X)\}$ and $Y_2 := \{s \in Y : s(y_1) = s(y_2)\}$. Then $\mathcal{M}, Y_1 \models \bigotimes_{i \leq r} \neg(v_i = x_i)$ because for each $s \in Y_1$, $(s(v_1), \dots, s(v_r)) \in \mathrm{Rel}(X)$. Clearly $\mathcal{M}, Y_2 \models y_1 = y_2$.

- To see $\mathcal{M}, Y \models \bigotimes_{i\le q} \alpha_i^*$, where each α_i is an atomic formula where R does not occur, simply split $Y = \bigcup_{i\le q} Y_i$ such that each $Y_i := \{s \in Y : \mathcal{M}, s''_X \models \alpha_i\}$. Then $\mathcal{M}, Y_i \models \alpha_i^*$ for each $i \le q$.

For the other direction, assume $\mathcal{M}, X \models \psi$. We will show $\mathcal{M}, \mathrm{Rel}(X) \models \phi$. From $\mathcal{M}, X \models \psi$ we get $\mathcal{M}, Y \models \theta^*$, where Y is as in (5.11) for certain sequences of functions F^i_j. We translate them into functions g^i_j by setting for each sequence of $a_1, \dots, a_{k(i,j)} \in M$, $g^i_j(a_1, \dots, a_{k(i,j)}) = F^i_j(s)$, where $s\big(u^{i,j}_k\big) = a_k$ for each $k \le k(i,j)$. Then each g^i_j is well defined because $\mathcal{M}, Y \models \chi^i_j$ for each i and j. If we can show that $\mathcal{M}, s'_X \models \forall x_1 \dots \forall x_n \theta$, we get $\mathcal{M}, s_X \models \phi$. To this end, let $\overline{a} \in M$ and denote $s''_X := s'_X(x_i \mapsto a_i)_{i\le n}$. Let $s \in Y$ such that $s(x_i) = a_i$ for all $i \le n$. Note that $s\big(y^i_j\big) = s''_X\big(t^i_j\big)$.

We show the implication from right to left in (5.12), i.e. we show it for the three kinds of conjuncts in θ for the arbitrary s.

- To see $\mathcal{M}, s''_X \models R\overline{x} \lor \neg(f_1\overline{x} = f_2\overline{x})$, assume $\mathcal{M}, s''_X \models f_1\overline{x} = f_2\overline{x}$. Then note that from $\mathcal{M}, Y \models \theta^*$ we get a split $Y = Y_1 \cup Y_2$ such that $\mathcal{M}, Y_1 \models (R\overline{x})^*$ and $\mathcal{M}, Y_2 \models \neg(y_1 = y_2)$. Because $s(y_1) = s(y_2)$, we have $s \in Y_1$. Now, for all $\overline{a} \in M$, if $a_i = s(x_i)$ for all $i \le r$, then there is some $s' \in Y_1$ such that $a_i = s'(v_i)$ for all $i \le r$. But we know that $(s'(v_1), \dots, s'(v_r)) \in \mathrm{Rel}(X)$, which is what we wanted.

- To see $\mathcal{M}, s''_X \models \neg R\overline{x} \lor f_1\overline{x} = f_2\overline{x}$, assume $\mathcal{M}, s''_X \models R\overline{x}$. Then note that from $\mathcal{M}, Y \models \theta^*$ we get a split $Y = Y_1 \cup Y_2$ such that $\mathcal{M}, Y_1 \models \bigotimes_{i\le r} \neg(v_i = x_i)$ and $\mathcal{M}, Y_2 \models y_1 = y_2$. Consider $s' := s\big(v_i \mapsto s(x_i)\big)_{i\le r}$. Then $s' \in Y$ because $s \in Y$ and $\big(s(x_1), \dots, s(x_r)\big) \in \mathrm{Rel}(X)$. Because $s'(v_i) = s(x_i)$ for all $i \le r$, we have $s' \in Y_2$, whence $s'(y_1) = s'(y_2)$, i.e. $\mathcal{M}, s''_X \models f_1\overline{x} = f_2\overline{x}$, as we wanted.

- It is left to show $\mathcal{M}, s''_X \models \bigvee_{i\le q} \alpha_i$, where no α_i mentions R. From $\mathcal{M}, Y \models \theta^*$ we get a split $Y = \bigcup_{i\le q} Y_i$ such that $\mathcal{M}, Y_i \models \alpha_i^*$ for each $i \le q$. Because $s \in Y_i$ for some i, we have $\mathcal{M}, s''_X \models \alpha_i$. □

5.5 Applications of Translations

We can use Theorem 5.4.1 and other translations to show the definability of many interesting classes of predicates in team logic. There are classes for which it is an open question whether a uniform, systematic definition exists, or if the definition varies wildly from predicate to another inside the class.

As an example, let us add a new connective $\hookrightarrow$ to team logic. We define

$$\begin{aligned} \mathcal{M}, X \models \phi \hookrightarrow \psi \iff & \text{ for all } Y \subseteq X\text{:} \\ & \text{if } Y \text{ is maximal w.r.t. } \mathcal{M}, Y \models \phi \text{ then } \mathcal{M}, Y \models \psi. \end{aligned}$$

We can express this connective in second order logic. Therefore we can translate any formula $\phi \hookrightarrow \psi$ into a team logic formula θ that does not use $\hookrightarrow$. The question is, is there a systematic way of doing this translation without resorting to the translation via second order logic? Is there a translation that leaves the subformulas ϕ and ψ intact?

Chapter 6

Axiomatising Fragments of FOD

In this chapter I investigate the question of finding a nontrivial fragment of dependence logic such that there is an effectively axiomatisable deductive system for the fragment. The basic requirements for such a deductive system are to be sound and complete with respect to the entailment relation of the fragment.

Any fragment that does not contain D-formulas is trivial as it can be axiomatised by any of the well-known proof systems of first order logic. On the other hand, we know that there is no effectively axiomatisable proof system for the whole dependence logic as its expressive power equals that of existential second order logic.

I denote finite conjunctions as $\bigwedge_{k \in K} \phi_k$ and similarly for disjunctions. Conjunction over the empty set denotes $\top$ and disjunction over the empty set denotes $\perp$.

For two subformulas in the same greater formula, $\phi_1, \phi_2 \leq \phi$, their *join*, $\mathrm{Join}(\phi_1, \phi_2)$ is the least subformula $\phi' \leq \phi$ for which $\phi_1, \phi_2 \leq \phi'$.

6.1 Calculus of Structures

There are many proof system formalisms of which sequent calculus, natural deduction, and semantic tableaux are best known. There is also a rather new formalism called the *calculus of structures* [7, 2]. It is based on a methodology called *deep inference* in which inference rules can be applied anywhere in a formula, not only at the root of the syntax tree of the formula. Calculus of structures has several advantages over other proof systems: Proof systems based on calculus of structures tend to have simple sets of inference rules. Proofs in calculus of structures are linear and short whereas proofs in other systems tend to be trees and even exponentially longer. Perhaps most

$$
{=}\frac{(\phi \vee \psi) \vee \gamma}{\phi \vee (\psi \vee \gamma)} \qquad {=}\frac{(\phi \wedge \psi) \wedge \gamma}{\phi \wedge (\psi \wedge \gamma)} \qquad {=}\frac{\phi \vee \psi}{\psi \vee \phi} \qquad {=}\frac{\bot \vee \phi}{\phi} \qquad {=}\frac{\top \wedge \phi}{\phi}
$$

$$
{=}\frac{\phi \vee (\psi \vee \gamma)}{(\phi \vee \psi) \vee \gamma} \qquad {=}\frac{\phi \wedge (\psi \wedge \gamma)}{(\phi \wedge \psi) \wedge \gamma} \qquad {=}\frac{\phi \wedge \psi}{\psi \wedge \phi} \qquad {=}\frac{\phi}{\bot \vee \phi} \qquad {=}\frac{\phi}{\top \wedge \phi}
$$

Figure 6.1: Identity rules

importantly, calculus of structures exposes symmetry in proofs on several levels; not only inference rules have symmetric duals but also proofs themselves show symmetry. There are also other interesting properties some of which I shall use.

As an introduction, I shall present two proof systems for classical propositional logic based on the calculus of structures. The systems were first presented by Brünnler and Tiu [2].

For purely syntactic purposes in this chapter, we extend the set of first order formulas by the atomic formula $\{\ \}$ called the *hole*. There is no semantics for the hole. Holes are used to mark places in formulas where subformulas can be plugged in. In practice, we only deal with formulas that have one or no holes. If formula χ contains a hole, we write it as $\chi\{\ \}$. By $\chi\{\psi\}$ we denote the formula obtained from $\chi\{\ \}$ by replacing the hole by ψ. We call χ the *context* of ψ.

An *inference rule* is a syntactic scheme of the form

$$
\rho\,\frac{\chi\{\phi\}}{\chi\{\psi\}},
$$

where ρ is the name, ψ is the *redex* and ϕ is the *contractum* of the rule. In all proof systems we allow the use of the identity rules in Figure 6.1

An *inference* of χ_2 from χ_1 according to inference rule ρ means that there are $\chi\{\ \}, \phi, \psi \in \mathrm{FO}$ such that $\chi_1 = \chi\{\phi\}$ and $\chi_2 = \chi\{\psi\}$. A *derivation* is a sequence of inferences, each applied to the result of the previous inference. We call the topmost formula in a derivation the *premise* and the bottommost formula the *conclusion* of the derivation. When writing down derivations, we may skip inference steps based on the identity rules. We may also group several inference steps into one by writing, for example, s^2 for two applications of the rule s. Let $\phi \vdash^S \psi$ denote that there is a derivation in proof system S with premise ϕ and conclusion ψ. If the derivation is called Δ, we write $\phi \vdash^S_\Delta \psi$. If the derivation is based on inferences only by rules $\rho_1, \ldots, \rho_n$, we write $\phi \vdash^{\rho_1,\ldots,\rho_n} \psi$.

$$\mathsf{i}\!\downarrow \frac{\chi\{\top\}}{\chi\{\phi \vee \neg\phi\}} \qquad\qquad \mathsf{i}\!\uparrow \frac{\chi\{\phi \wedge \neg\phi\}}{\chi\{\bot\}}$$

$$\mathsf{w}\!\downarrow \frac{\chi\{\bot\}}{\chi\{\phi\}} \qquad \mathsf{s} \frac{\chi\{\phi \wedge (\psi \vee \gamma)\}}{\chi\{(\phi \wedge \psi) \vee \gamma\}} \qquad \mathsf{w}\!\uparrow \frac{\chi\{\phi\}}{\chi\{\top\}}$$

$$\mathsf{c}\!\downarrow \frac{\chi\{\phi \vee \phi\}}{\chi\{\phi\}} \qquad\qquad \mathsf{c}\!\uparrow \frac{\chi\{\phi\}}{\chi\{\phi \wedge \phi\}}$$

Figure 6.2: Global proof system SKSg

$$\mathsf{ai}\!\downarrow \frac{\chi\{\top\}}{\chi\{\alpha \vee \neg\alpha\}} \qquad\qquad \mathsf{ai}\!\uparrow \frac{\chi\{\alpha \wedge \neg\alpha\}}{\chi\{\bot\}}$$

$$\mathsf{s} \frac{\chi\{\phi \wedge (\psi \vee \gamma)\}}{\chi\{(\phi \wedge \psi) \vee \gamma\}}$$

$$\mathsf{aw}\!\downarrow \frac{\chi\{\bot\}}{\chi\{\alpha\}} \qquad \mathsf{m} \frac{\chi\{(\phi \wedge \gamma) \vee (\psi \wedge \delta)\}}{\chi\{(\phi \vee \psi) \wedge (\gamma \vee \delta)\}} \qquad \mathsf{aw}\!\uparrow \frac{\chi\{\alpha\}}{\chi\{\top\}}$$

$$\mathsf{ac}\!\downarrow \frac{\chi\{\alpha \vee \alpha\}}{\chi\{\alpha\}} \qquad\qquad \mathsf{ac}\!\uparrow \frac{\chi\{\alpha\}}{\chi\{\alpha \wedge \alpha\}}$$

Figure 6.3: Local proof system SKS; α is an atomic formula.

The first proof system for classical propositional logic has rules of *global* nature; there are no restrictions on the redex and contractum of the rules. The proof system is called SKSg.[1] Its inference rules, shown in Figure 6.2, are called *switch* (s), *identity* ($\mathsf{i}\!\downarrow$), *cut* ($\mathsf{i}\!\uparrow$), *weakening* ($\mathsf{w}\!\downarrow$), *co-weakening* ($\mathsf{w}\!\uparrow$), *contraction* ($\mathsf{c}\!\downarrow$), and *co-contraction* ($\mathsf{c}\!\uparrow$). Of these, $\mathsf{i}\!\downarrow$, $\mathsf{w}\!\downarrow$ and $\mathsf{c}\!\downarrow$ are *down rules*, and $\mathsf{i}\!\uparrow$, $\mathsf{w}\!\uparrow$ and $\mathsf{c}\!\uparrow$ are *up rules*. The corresponding up and down rules are duals in the sense that one is obtained from the other by flipping the rule upside down and negating the redex and contractum.[2] Switch is a self-dual rule.

The second proof system for classical propositional logic has rules of *local* nature; the up and down rules can only be applied when the redex and contractum are atomic formulas. The proof system is called SKS and its inference rules are in Figure 6.3. Apart from switch and the new rule, *medial* (m), the inference rules of SKS are merely *atomic* counterparts of the corresponding rules in SKSg.

Both SKSg and SKS are sound and complete for classical propositional

[1] SKSg stands for "**s**ymmetric **k**lassisch (or classical) proof system in the calculus of **s**tructures, **g**lobal variant".

[2] The negation can be left out in some cases due to the implicit equivalence on relations in inferences.

logic [1].

Theorem 6.1.1. *If $\phi, \psi \in$ FO are propositional sentences then $\phi \Rightarrow \psi$ if and only if $\phi \vdash^{\mathrm{SKSg}} \psi$ if and only if $\phi \vdash^{\mathrm{SKS}} \psi$.*

The rules in SKS are clearly restrictions of the rules in SKSg. The restrictedness gives an additional property: the local system admits of a decomposition of derivations [1, Theorem 7.5.1].

Theorem 6.1.2. *If $\phi, \psi \in$ FO are propositional sentences and $\phi \vdash^{\mathrm{SKS}} \psi$, then $\phi \vdash^{\mathsf{ac}\uparrow} \phi_1 \vdash^{\mathsf{aw}\uparrow} \phi_2 \vdash^{\mathsf{ai}\downarrow} \phi_3 \vdash^{\mathsf{s,m}} \psi_3 \vdash^{\mathsf{ai}\uparrow} \psi_2 \vdash^{\mathsf{aw}\downarrow} \psi_1 \vdash^{\mathsf{ac}\downarrow} \psi$ for some $\phi_1, \phi_2, \phi_3, \psi_3, \psi_2, \psi_1 \in$ FO.*

For every derivation there is an atomic flow [8]. On the level of intuition, an atomic flow tracks the position of each atomic formula through the derivation. I give here only a geometric definition. A formulation in conventional terms of sets and functions is also possible but more difficult to read and write.

Given a derivation, we define its *atomic flow* as a graph drawn on top of the derivation, connecting certain occurrences of the same atomic subformulas. We use a shorthand notation for compound formulas; if $\phi, \psi \in$ FO are propositional sentences, $\phi = \gamma \vee \delta$ and $\psi = \gamma \wedge \delta$, then

$$\begin{array}{c}\phi \\ + \\ \phi\end{array} \quad \text{means} \quad \begin{array}{c}\gamma \vee \delta \\ +\!\!-\!\!-\!\!+ \\ \gamma \vee \delta\end{array}$$

and

$$\begin{array}{c}\psi \\ + \\ \psi\end{array} \quad \text{means} \quad \begin{array}{c}\gamma \wedge \delta \\ +\!\!-\!\!-\!\!+ \\ \gamma \wedge \delta\end{array} \quad .$$

The atomic flow for proof system SKSg is created rule by rule as shown in Figure 6.4. The atomic flow for proof system SKS is created as shown in Figure 6.5. If there is a derivation $\phi \vdash_\Delta \psi$ and subformulas $\phi' \leq \phi$ and $\psi' \leq \psi$, and the atomic flow of derivation Δ leads from ϕ' to ψ', we say that *ϕ' flows to ψ' in Δ*.

Atomic flows are usually used as a tool in normalisation of derivations and in the study of identity of derivations. I shall use them in defining a proof system for a fragment of dependence logic.

$$\mathsf{i}{\downarrow}\,\frac{\chi\{\top\}}{\chi\{\phi \vee \neg\phi\}} \qquad \mathsf{i}{\uparrow}\,\frac{\chi\{\phi \wedge \neg\phi\}}{\chi\{\bot\}}$$

$$\mathsf{w}{\downarrow}\,\frac{\chi\{\bot\}}{\chi\{\phi\}} \qquad \mathsf{s}\,\frac{\chi\{\phi \wedge (\psi \vee \gamma)\}}{\chi\{(\phi \wedge \psi) \vee \gamma\}} \qquad \mathsf{w}{\uparrow}\,\frac{\chi\{\phi\}}{\chi\{\top\}}$$

$$\mathsf{c}{\downarrow}\,\frac{\chi\{\phi \vee \phi\}}{\chi\{\phi\}} \qquad \mathsf{c}{\uparrow}\,\frac{\chi\{\phi\}}{\chi\{\phi \wedge \phi\}}$$

Figure 6.4: Atomic flow for SKSg

$$\mathsf{ai}{\downarrow}\,\frac{\chi\{\top\}}{\chi\{\alpha \vee \neg\alpha\}} \qquad \mathsf{ai}{\uparrow}\,\frac{\chi\{\alpha \wedge \neg\alpha\}}{\chi\{\bot\}}$$

$$\mathsf{s}\,\frac{\chi\{\phi \wedge (\psi \vee \gamma)\}}{\chi\{(\phi \wedge \psi) \vee \gamma\}}$$

$$\mathsf{aw}{\downarrow}\,\frac{\chi\{\bot\}}{\chi\{\alpha\}} \qquad \mathsf{aw}{\uparrow}\,\frac{\chi\{\alpha\}}{\chi\{\top\}}$$

$$\mathsf{m}\,\frac{\chi\{(\phi \wedge \gamma) \vee (\psi \wedge \delta)\}}{\chi\{(\phi \vee \psi) \wedge (\gamma \vee \delta)\}}$$

$$\mathsf{ac}{\downarrow}\,\frac{\chi\{\alpha \vee \alpha\}}{\chi\{\alpha\}} \qquad \mathsf{ac}{\uparrow}\,\frac{\chi\{\alpha\}}{\chi\{\alpha \wedge \alpha\}}$$

Figure 6.5: Atomic flow for SKS

6.2 A Proof System for a Fragment of FOD

I will consider a modest fragment of dependence logic that contains only the simplest forms of D-formulas, along with limited use of atomic formulas and connectives. For all $k < \omega$, denote

$$\theta_k := \exists y\big(()\rightsquigarrow y \wedge R_k xy\big).$$

Let F be the fragment of dependence logic that consists of the set of formulas that contains $\{\top, \bot\} \cup \{\theta_k : k < \omega\}$ and that is closed under $\vee$ and $\wedge$. In particular, there is no negation in F. We call the formulas θ_k *atoms*.

In a way, fragment F resembles classical propositional logic without negation; it is built by gluing atoms together with disjunction and conjunction. Therefore it is natural to start from a proof system of classical propositional logic and see how it works with fragment F. Choosing SKSg as the proof system, the first thing to note is that because F does not contain negation, the rules of identity and cut are not applicable. This does not pose a real problem at the moment. We will just leave these rules out of the proof system.

All applicable rules of SKSg are sound for F except for contraction. We can easily show that $\theta_1 \vee \theta_1 \not\Rightarrow \theta_1$. This time we seem to have a problem because there are instances of entailment in F that do not seem to be possible to derive without resorting to contraction. The simplest such example I have found is

$$\big((\phi \vee \psi) \wedge (\gamma \vee \delta)\big) \Rightarrow \Big((\phi \wedge \delta) \vee \Big(\big(\psi \vee (\phi \wedge \gamma)\big) \wedge \big(\gamma \vee (\psi \wedge \delta)\big)\Big)\Big),$$

where $\phi, \psi, \gamma, \delta \in F$. Figure 6.6 shows a derivation for this entailment. Because of these reasons, we will keep contraction as an inference rule but we will compensate by setting a condition on the use of contraction that derivations in fragment F must satisfy.

Definition 6.2.1. We define SKSgf as the proof system with the inference rules $\mathsf{w}\!\downarrow$, $\mathsf{c}\!\downarrow$, s, $\mathsf{w}\!\uparrow$, and $\mathsf{c}\!\uparrow$, and the additional requirement that a derivation in this SKSgf must satisfy the following *flow condition*;[3]

> The atomic flows of two distinct atom occurrences in the premise must not connect anywhere in the derivation.

Similarly, we can define SKSf as the proof system with the inference rules $\mathsf{aw}\!\downarrow$, $\mathsf{ac}\!\downarrow$, s, $\mathsf{aw}\!\uparrow$, and $\mathsf{ac}\!\uparrow$, and the additional requirement that a derivation in SKSf must satisfy the flow condition.

[3]SKSgf stands for "**s**ymmetric **k**lassisch (or classical) proof system in the calculus of **s**tructures, **g**lobal variant, with **f**low condition".

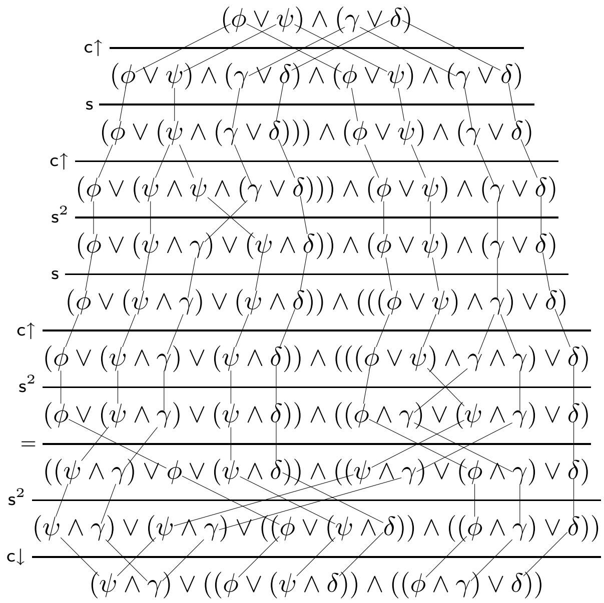

Figure 6.6: A derivation and its atomic flow for $(\phi \vee \psi) \wedge (\gamma \vee \delta) \vdash^{\mathrm{SKSgf}} (\psi \wedge \gamma) \vee \big((\phi \vee (\psi \wedge \delta)) \wedge \big((\phi \wedge \gamma) \vee \delta\big)\big)$

The example derivation in Figure 6.6 uses the contraction rule, and because the whole derivation satisfies the flow condition, this is valid use of contraction. Figure 6.7 shows another example of a derivation in SKSgf. Figure 6.8 shows a simple example of a sequence of inferences that breaks the flow condition and thus is not a derivation in SKSgf.

A decomposition result similar to the one for SKS can be obtained for SKSf.

Theorem 6.2.2. *If $\phi, \psi \in F$ and $\phi \vdash^{\mathrm{SKSf}} \psi$ then for some $\phi_1, \phi_2, \psi_1, \psi_2 \in F$,*

$$\phi \vdash^{\mathsf{ac}\uparrow} \phi_1 \vdash^{\mathsf{aw}\uparrow} \phi_2 \vdash^{\mathsf{s,m}} \psi_2 \vdash^{\mathsf{aw}\downarrow} \psi_1 \vdash^{\mathsf{ac}\downarrow} \psi$$

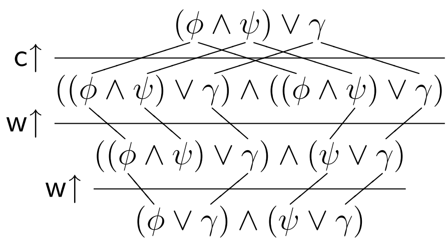

Figure 6.7: A derivation for $(\phi \wedge \psi) \vee \theta \vdash^{\mathrm{SKSgf}} (\phi \vee \theta) \wedge (\psi \vee \theta)$ and its atomic flow

$$\mathsf{c}{\downarrow}\,\frac{\theta_k \vee \theta_k}{\theta_k}$$

Figure 6.8: Not a derivation in SKSgf or SKSf; the flow condition breaks.

and, for some $\phi_3, \phi_4, \psi_3, \phi_4 \in F$*,*

$$\phi \vdash^{\mathsf{aw}\uparrow} \phi_3 \vdash^{\mathsf{ac}\uparrow} \phi_4 \vdash^{\mathsf{s,m}} \psi_4 \vdash^{\mathsf{ac}\downarrow} \psi_3 \vdash^{\mathsf{aw}\downarrow} \psi$$

such that the flow condition is satisfied in both derivations.

Proof. Kai Brünnler showed how to push all instances of $\mathsf{aw}\uparrow$ up and all instances of $\mathsf{aw}\downarrow$ down in an SKS derivation. The same technique applies to SKSf derivations because the flow condition is not violated in the transformation of the derivation. We only need to show how $\mathsf{ac}\uparrow$ can be pushed up and $\mathsf{ac}\downarrow$ can be pushed down, utilising the fact that the identity and cut rules are missing from SKSf.

Firstly, we know that $\mathsf{ac}\downarrow$ permutes under $\mathsf{aw}\downarrow$, s and m [1, Lemma 7.1.2], i.e. that for a derivation $\phi \vdash^{\mathsf{ac}\downarrow} \alpha \vdash^{\mathsf{aw}\downarrow} \psi$ there is a derivation $\phi \vdash^{\mathsf{aw}\downarrow} \beta \vdash^{\mathsf{ac}\downarrow} \psi$, etc. It remains to push an instance of $\mathsf{ac}\downarrow$ below an instance of $\mathsf{ac}\uparrow$ and below an instance of $\mathsf{aw}\uparrow$, and dually, to push an instance of $\mathsf{ac}\uparrow$ above an instance of $\mathsf{ac}\downarrow$ and above an instance of $\mathsf{aw}\downarrow$.

Let us consider an SKSf derivation with consequtive instances of $\mathsf{ac}\downarrow$ and some rule ρ that is either $\mathsf{ac}\uparrow$ or $\mathsf{aw}\uparrow$. In case the two rule instances operate in different contexts, they can be permuted with no trouble. Interaction happens only when the redex of the $\mathsf{ac}\downarrow$ instance is the contractum of the ρ instance. When ρ is $\mathsf{ac}\uparrow$, we can double the instances, push them past each other and patch the middle with an instance of m:

$$\begin{array}{rc} & \chi\{\theta_k \vee \theta_k\} \\ \mathsf{ac}\downarrow & \overline{\qquad\qquad\qquad} \\ & \chi\{\theta_k\} \\ \mathsf{ac}\uparrow & \overline{\qquad\qquad\qquad} \\ & \chi\{\theta_k \wedge \theta_k\} \end{array} \quad \leadsto \quad \begin{array}{rc} & \chi\{\theta_k \vee \theta_k\} \\ \mathsf{ac}\uparrow^2 & \overline{\qquad\qquad\qquad\qquad\qquad\qquad} \\ & \chi\{(\theta_k \wedge \theta_k) \vee (\theta_k \wedge \theta_k)\} \\ \mathsf{m} & \overline{\qquad\qquad\qquad\qquad\qquad\qquad} \\ & \chi\{(\theta_k \vee \theta_k) \wedge (\theta_k \vee \theta_k)\} \\ \mathsf{ac}\downarrow^2 & \overline{\qquad\qquad\qquad\qquad\qquad\qquad} \\ & \chi\{\theta_k \wedge \theta_k\} \end{array} \quad .$$

The atom flows that connect in the resulting derivation come from the same atom instance in the premise of the whole derivation, because they did so in the original derivation.

When ρ is $\mathsf{aw}{\uparrow}$, the $\mathsf{ac}{\downarrow}$ instance can be transformed into an $\mathsf{aw}{\uparrow}$ instance:

$$\mathsf{ac}{\downarrow}\,\dfrac{\chi\{\theta_k \vee \theta_k\}}{\mathsf{aw}{\uparrow}\,\dfrac{\chi\{\theta_k\}}{\chi\{\top\}}} \quad\rightsquigarrow\quad \mathsf{aw}{\uparrow}^2\,\dfrac{\chi\{\theta_k \vee \theta_k\}}{=\,\dfrac{\chi\{\top \vee \top\}}{\chi\{\top\}}}\ .$$

The resulting derivation has one less case of connecting atom flows, so the flow condition is still satisfied.

Considering the dual cases, pushing $\mathsf{ac}{\uparrow}$ above $\mathsf{ac}{\downarrow}$ is the same as pushing $\mathsf{ac}{\downarrow}$ below $\mathsf{ac}{\uparrow}$, which we already considered. Pushing $\mathsf{ac}{\uparrow}$ above $\mathsf{aw}{\downarrow}$ can be done as follows:

$$\mathsf{aw}{\downarrow}\,\dfrac{\chi\{\bot\}}{\mathsf{ac}{\uparrow}\,\dfrac{\chi\{\theta_k\}}{\chi\{\theta_k \wedge \theta_k\}}} \quad\rightsquigarrow\quad =\,\dfrac{\chi\{\bot\}}{\mathsf{aw}{\downarrow}^2\,\dfrac{\chi\{\bot \wedge \bot\}}{\chi\{\theta_k \wedge \theta_k\}}}\ .$$

None of the inflicted atom flows starts from the premise, so the flow condition is satisfied.

Starting from any SKSf derivation, we can now push all (co-)contractions to the outside, and then in the (co-)contraction-free part of the resulting derivation, push all (co-)weakenings to the outside. The latter pushing does not introduce new instances of (co-)contraction, so the first normal form is achieved. Similarly, we can start from any SKSf derivation, push all (co-)weakenings to the outside, and then in the (co-)weakening-free part of the resulting derivation, push all (co-)contractions to the outside. Again, the latter pushing does not introduce new instances of (co-)weakening, so the second normal form is achieved. □

6.3 Soundness of the Proof System

In order to prove the soundness of SKSgf for fragment F, we need the following definitions.

Definition 6.3.1. For a model $\mathcal{M}$, a function f is a *team collection* on $\mathcal{M}$ for $\phi \in F$ if it maps subformulas $\phi' \leq \phi$ to teams on $\mathcal{M}$, satisfying

$$f(\phi_1 \wedge \phi_2) = f(\phi_1) \cap f(\phi_2),$$
$$f(\phi_1 \vee \phi_2) = f(\phi_1) \cup f(\phi_2).$$

Definition 6.3.2 ($X^{\tau}_{\phi'}$)**.** Assume player **II** has winning strategy τ in the semantic game $\partial^{\mathrm{FOD}}(\mathcal{M}, \phi, X)$. For each possible position (ϕ', X', α) in the game, define the τ-*team* of ϕ', $X^{\tau}_{\phi'}$, to be X'. Note that this definition is unambiguous because the teams in a play of the semantic game are determined solely by player **II**'s strategy [19, Lemma 5.12].

Definition 6.3.3 ($X^{\tau,\Delta}_{\psi'}$)**.** Assume $\phi, \psi \in F$, $\mathbf{II} \uparrow \Game^{\mathrm{FOD}}(\mathcal{M}, \phi, X)$ by τ, and $\phi \vdash^{\mathrm{SKSgf}}_{\Delta} \psi$. For $\psi' \leq \psi$, define the τ-Δ-*team* of ψ', $X^{\tau,\Delta}_{\psi'}$, by induction on ψ'. If ψ' is an atom, then

$$X^{\tau,\Delta}_{\psi'} = \begin{cases} X^{\tau}_{\phi'}, & \text{where } \phi' \leq \phi \text{ flows to } \psi' \text{ in } \Delta; \\ \emptyset, & \text{if there is no such } \phi'. \end{cases}$$

Note that if there is $\phi' \leq \phi$ that flows to ψ' in Δ, it is unique by the flow condition. For compound formulas, define

$$\begin{aligned} X^{\tau,\Delta}_{\psi_1 \wedge \psi_2} &= X^{\tau,\Delta}_{\psi_1} \cap X^{\tau,\Delta}_{\psi_2}, \\ X^{\tau,\Delta}_{\psi_1 \vee \psi_2} &= X^{\tau,\Delta}_{\psi_1} \cup X^{\tau,\Delta}_{\psi_2}. \end{aligned}$$

As a special case we have that $X^{\tau}_{\phi} = X^{\tau,\Delta}_{\phi}$, where Δ is the empty derivation.

Definition 6.3.4 (Element game)**.** For a model $\mathcal{M}$, element $x \in M$, formula $\phi \in F$, and team collection f for ϕ, define a two-player game, $\Game(\phi, f, x)$, called the *element game*. The positions in a play of the game are subformulas $\phi' \leq \phi$. The starting position is ϕ itself. If the play is in position $\phi_1 \wedge \phi_2$, player **I** chooses the play to continue from ϕ_1 or ϕ_2. If the play is in position $\phi_1 \vee \phi_2$, player **II** chooses the play to continue from ϕ_1 or ϕ_2. The play ends when the position ϕ' is an atom. Player **II** wins if $x \in \mathrm{Rel}(f(\phi'))$.[4]

Lemma 6.3.5. *The following conditions are equivalent;*

1. $\mathbf{II} \uparrow \Game(\phi, f, x)$,
2. $x \in \mathrm{Rel}(f(\phi))$.

Proof. Assume player **II** has winning strategy τ for $\Game(\phi, f, x)$, and $x \in M$. We prove by induction on $\phi' \leq \phi$ that if player **II** has played by τ up to position ϕ', then $x \in \mathrm{Rel}(f(\phi'))$.

Atomic case. The play ends and **II** wins, thus $x \in \mathrm{Rel}(f(\phi'))$.

Case $\phi_1 \vee \phi_2$**.** Player **II** follows τ and chooses the play to continue from $\tau(\phi')$. Thus the induction hypothesis gives $x \in \mathrm{Rel}(f(\tau(\phi')))$, whence $x \in \mathrm{Rel}(f(\phi'))$.

[4]The technically correct expression is $(x) \in \mathrm{Rel}(f(\phi'))$ but I leave the parentheses away for simplicity.

Case $\phi_1 \wedge \phi_2$**.** Player **I** can choose the play to continue from ϕ_i for either $i \in \{1, 2\}$. In both cases player **II** has followed τ, so the induction hypothesis gives $x \in \mathrm{Rel}(f(\phi_i))$. Thus $x \in \mathrm{Rel}(f(\phi'))$.

Assume then that $x \in \mathrm{Rel}(f(\phi))$. Define strategy τ for **II** by induction on game position $\phi' \leq \phi$ while maintaining that $x \in \mathrm{Rel}(f(\phi'))$.

Case $\phi_1 \vee \phi_2$**.** From $x \in \mathrm{Rel}(f(\phi'))$ we get that $x \in \mathrm{Rel}(f(\phi_i))$ for some $i \in \{1, 2\}$. Let player **II** choose $\tau(\phi') = \phi_i$.

Case $\phi_1 \wedge \phi_2$**.** Player **I** chooses the play to continue from ϕ_i for some $i \in \{1, 2\}$. We have $x \in \mathrm{Rel}(f(\phi')) \subseteq \mathrm{Rel}(f(\phi_i))$.

Atomic case. We have maintained $x \in \mathrm{Rel}(f(\phi'))$, so τ is a winning strategy. $\square$

Theorem 6.3.6. SKSgf *is sound for* FOD *and thus* $\phi \vdash^{\mathrm{SKSgf}} \psi$ *implies* $\phi \Rightarrow \psi$ *for all* $\phi, \psi \in F$.

Proof. Assume $\mathbf{II} \uparrow \eth^{\mathrm{FOD}}(\mathcal{M}, \phi, X)$ by τ, and $\phi \vdash^{\mathrm{SKSgf}}_{\Delta} \psi$. First we prove by induction on $\psi' \leq \psi$ that $\mathcal{M}, X^{\tau,\Delta}_{\psi'} \models \psi'$.

Case θ_k**.** If $X^{\tau,\Delta}_{\psi'} = \emptyset$, then the claim holds trivially. Otherwise $X^{\tau,\Delta}_{\psi'} = X^{\tau}_{\phi'}$, where $\phi' = \theta_k$. Thus $\mathcal{M}, X^{\tau,\Delta}_{\psi'} \models \theta_k$, and the claim holds again.

Case $\psi_1 \vee \psi_2$**.** Induction hypothesis gives $\mathcal{M}, X^{\tau,\Delta}_{\psi_i} \models \psi_i$ for both $i \in \{1, 2\}$. The claim holds by splitting $X^{\tau,\Delta}_{\psi'} = X^{\tau,\Delta}_{\psi_1} \cup X^{\tau,\Delta}_{\psi_2}$.

Case $\psi_1 \wedge \psi_2$**.** Induction hypothesis gives $\mathcal{M}, X^{\tau,\Delta}_{\psi_i} \models \psi_i$ for both $i \in \{1, 2\}$. Because $X^{\tau,\Delta}_{\psi'} \subseteq X^{\tau,\Delta}_{\psi_i}$ for both $i \in \{1, 2\}$, the claim holds.

This gives $\mathcal{M}, X^{\tau,\Delta}_{\psi} \models \psi$. Next we will prove that $X \subseteq X^{\tau,\Delta}_{\psi}$, which gives us $\mathcal{M}, X \models \psi$, ending the proof.

Assume that derivation Δ is of the form

$$\rho_1 \frac{\chi_0}{\dfrac{\chi_1}{\rho_2 \dfrac{\vdots}{\rho_{n-1}\dfrac{}{\rho_n \dfrac{\chi_{n-1}}{\chi_n}}}}},$$

where $\chi_0 = \phi$ and $\chi_n = \psi$. For $k \leq n$, let f_k be the team collection for χ_k such that $f_k(\chi') = X^{\tau,\Delta_k}_{\chi'}$ for each $\chi' \leq \chi_k$, where Δ_k is Δ limited to the first

k inference steps. Note that if some $\alpha_k \leq \chi_k$ flows to some $\alpha_p \leq \chi_p$, then $X^{\tau,\Delta_k}_{\alpha_k} = X^{\tau,\Delta_p}_{\alpha_p}$. Let $x \in \mathrm{Rel}(X)$. Then $\mathbf{II} \uparrow \partial(\chi_0, f_0, x)$ by some winning strategy σ^0_x. We will form a sequence of strategies $\sigma^0_x, \sigma^1_x, \ldots, \sigma^n_x$, step by step, such that at each step we maintain the invariant

$$\mathbf{II} \uparrow \partial(\chi_k, f_k, x) \text{ by } \sigma^k_x. \tag{6.1}$$

This leads to σ^n_x being a winning strategy for $\mathbf{II}$ in $\partial(\psi, f_n, x)$, whence $x \in \mathrm{Rel}\big(f_n(\psi)\big) = \mathrm{Rel}\big(X^{\tau,\Delta}_{\psi}\big)$.

If we have reached σ^k_x in the sequence, let σ^{k+1}_x be as follows, depending on the inference rule ρ_{k+1}.

Case s $\dfrac{\chi\{\alpha\wedge(\beta\vee\gamma)\}}{\chi\{(\alpha\wedge\beta)\vee\gamma\}}$. Define σ^{k+1}_x on the redex as

$$\sigma^{k+1}_x\big((\alpha\wedge\beta)\vee\gamma\big) = \begin{cases} \alpha\wedge\beta, & \text{if } \sigma^k_x(\beta\vee\gamma) = \beta; \\ \gamma, & \text{if } \sigma^k_x(\beta\vee\gamma) = \gamma, \end{cases}$$

and define σ^{k+1}_x to map subformulas of χ, α, β and γ like σ^k_x. To get (6.1) we only need to see that if the choice at position $(\alpha\wedge\beta)\vee\gamma$ is $\alpha\wedge\beta$, $\mathbf{II}$ is at a winning position because σ^k_x is winning both at α and at β; otherwise the choice is γ, in which case σ^k_x is winning at γ.

Case $\mathsf{w}\!\uparrow$ $\dfrac{\chi\{\alpha\}}{\chi\{\top\}}$. Define σ^{k+1}_x to map subformulas of χ like σ^k_x does. We get (6.1) because $\partial(\chi_{k+1}, f_{k+1}, x)$ is easier for $\mathbf{II}$ than $\partial(\chi_k, f_k, x)$.

Case $\mathsf{c}\!\uparrow$ $\dfrac{\chi\{\alpha\}}{\chi\{\alpha_1\wedge\alpha_2\}}$. Here $\alpha = \alpha_1 = \alpha_2$. Define σ^{k+1}_x to map subformulas of χ like σ^k_x does and to map subformulas of α_1 and α_2 like σ^k_x maps subformulas of α. To get (6.1) we only need to see that both α_1 and α_2 are winning positions for $\mathbf{II}$ because σ^k_x is winning at α.

Case $\mathsf{w}\!\downarrow$ $\dfrac{\chi\{\bot\}}{\chi\{\alpha\}}$. Define σ^{k+1}_x to map subformulas of χ like σ^k_x does and to map subformulas of α arbitrarily. We get (6.1) because the play never proceeds to α; if it did, then σ^k_x would allow reaching the position $\bot$ which is losing for $\mathbf{II}$.

Case $\mathsf{c}\!\downarrow$ $\dfrac{\chi\{\alpha_1\vee\alpha_2\}}{\chi\{\alpha\}}$. Here $\alpha = \alpha_1 = \alpha_2$. Define σ^{k+1}_x to map subformulas of χ like σ^k_x does, and to map subformulas of α like σ^k_x maps subformulas of α_1. To get (6.1), note that σ_x^{k+1} is winning at α because $X_{\alpha_1}^{\tau,\Delta_k} = X_{\alpha_2}^{\tau,\Delta_k} = X_{\alpha}^{\tau,\Delta_{k+1}}$ and $\alpha_1 = \alpha_2 = \alpha$, and thus the corresponding element games are the same. □

Corollary 6.3.7. SKSf *is sound for* FOD.

Proof. It suffices to show that derivations in SKSf are translatable to derivations in SKSgf. This is easy; first note that all atomic rules in SKSf are special cases of the corresponding rules in SKSgf. Finally, we can translate an inference by the medial rule into the derivation

$$\mathsf{w}\uparrow\frac{\mathsf{w}\uparrow\frac{\mathsf{w}\uparrow\frac{\mathsf{w}\uparrow\frac{\mathsf{c}\uparrow\frac{\chi\{(\phi\wedge\gamma)\vee(\psi\wedge\delta)\}}{\chi\{((\phi\wedge\gamma)\vee(\psi\wedge\delta))\wedge((\phi\wedge\gamma)\vee(\psi\wedge\delta))\}}}{\chi\{(\phi\vee(\psi\wedge\delta))\wedge((\phi\wedge\gamma)\vee(\psi\wedge\delta))\}}}{\chi\{(\phi\vee\psi)\wedge((\phi\wedge\gamma)\vee(\psi\wedge\delta))\}}}{\chi\{(\phi\vee\psi)\wedge(\gamma\vee(\psi\wedge\delta))\}}}{\chi\{(\phi\vee\psi)\wedge(\gamma\vee\delta)\}}\ .$$
□

6.4 Discussion on the Problem of Completeness

It is an open question if SKSgf is complete for fragment F. In this section, I present some ideas and lemmas that might help in showing that the answer is positive. Some of the claims have incomplete proofs which are marked by a square in parentheses, (□).

Note that if we can show that SKSf is complete for F, then also SKSgf is complete for F. Namely, as stated in Corollary 6.3.7, we can translate a derivation in SKSf into a derivation in SKSgf.

It seems more beneficial to try to prove completeness primarily for SKSf because it has an advantage over SKSgf, namely the property of decomposition of derivations, as shown in Theorem 6.2.2.

Conjecture 6.4.1. SKSf *is complete for fragment* F*.*

The intended proof of completeness of SKSgf for fragment F, sketched below, is based on a canonical model construction, similar to what is usually done in completeness proofs in modal logic. Because F contains dependence formulas and not sentences, satisfaction of formulas in F is defined with respect to the team in question. Thus it does not seem possible to build a canonical model where each element consists of a maximally consistent set of formulas, unlike in propositional modal logic.

The universe of the canonical model represents a set of resources and the interpretations of relations in the canonical model represent which resources are suitable for which atoms. In order to denote this suitability, the universe will also contain strategies for the element game. The resources themselves are most conveniently chosen to be the atoms themselves. Each atom is then a suitable resource for each strategy that reaches the atom or any other instance of the same formula.

Definition 6.4.2. For $\phi \in F$, define the *canonical model*, $\mathcal{M}_\phi$, to have as its universe M_ϕ the (disjoint) union of all atoms in ϕ and strategies over ϕ, that is,

$$M_\phi := \{\phi' : \phi' \leq \phi \text{ is an atom}\} \cup \{\tau : \tau \text{ is a strategy for } \mathbf{II} \text{ in } \eth(\phi)\}.$$

We say that a strategy $\tau \in M_\phi$ *reaches* a subformula $\phi' \leq \phi$ if the game $\eth(\phi)$ can reach position ϕ' when $\mathbf{II}$ plays by τ, i.e., if player $\mathbf{I}$ has a strategy σ such that when $\mathbf{I}$ plays by σ and $\mathbf{II}$ plays by τ, the play of the game eventually comes to position ϕ'. Define the relations R_k, $k < \omega$, to be interpreted in $\mathcal{M}_\phi$ as

$$R_k^{\mathcal{M}_\phi} := \{(\tau, \phi') : \tau \text{ reaches } \phi' \text{ and } \phi' = \theta_k\}.$$

For $\phi' \leq \phi$, define the *canonical team* of ϕ' in ϕ to be

$$X_{\phi'} := \{\tau \in M_\phi : \tau \text{ reaches } \phi'\}_{(x)}.$$

If $X \subseteq X_\phi$, we have $\mathcal{M}_\phi, X \models \theta_k$ if and only if $X = \emptyset$ or there is a subformula $\phi' \leq \phi$ such that $\phi' = \theta_k$ and all $\tau \in \mathrm{Rel}(X)$ reach ϕ'.

The following lemmas state basic facts about the behaviour of the canonical model and the canonical teams.

Lemma 6.4.3. *Any formula is satisfied by its canonical model and team, that is, $\mathcal{M}_\phi, X_\phi \models \phi$ for all $\phi \in F$.*

Proof. Let $\phi \in F$. We prove by induction on subformulas $\phi' \leq \phi$ that $\mathcal{M}_\phi, X_{\phi'} \models \phi'$. The atomic case is clear by definitions. The case that ϕ' is of the form $\phi_0 \wedge \phi_1$ is also clear because $X_{\phi'} = X_{\phi_0} = X_{\phi_1}$. If ϕ' is of the form $\phi_0 \vee \phi_1$, we can split $X_{\phi'} = X_{\phi_0} \cup X_{\phi_1}$ and use the induction hypothesis to complete the proof. □

Lemma 6.4.4. *Let $\phi_1, \phi_2 \leq \phi$ be two different subformulas. Then* *(i)* $X_{\phi_1} \cap X_{\phi_2} = \emptyset$ *if and only if* $\mathrm{Join}(\phi_1, \phi_2)$ *is a disjunction;*

(ii) $X_{\phi_1} \cap X_{\phi_2} \neq \emptyset$ *if and only if* $\mathrm{Join}(\phi_1, \phi_2)$ *is a conjunction.*

Proof. We can see $\mathrm{Join}(\phi_1, \phi_2)$ as the last position in $\eth(\phi)$ where the play can still reach both ϕ_1 and ϕ_2. If this position is a disjunction, it is player **II**'s decision to not be able to reach one of the two formulas anymore in the play. Every strategy of **II** has this property, and therefore $X_{\phi_1} \cap X_{\phi_2} = \emptyset$.

Assume then that the crucial decision is for player **I**, i.e. that the join is a conjunction; $\mathrm{Join}(\phi_1, \phi_2) = \phi_1' \wedge \phi_2'$, where $\phi_1 \leq \phi_1'$ and $\phi_2 \leq \phi_2'$. For $i \in \{1, 2\}$, let σ_i and τ_i be strategies for **I** and **II**, respectively, such that the play with σ_i and τ_i reaches ϕ_i. Let τ be the strategy that makes choices like τ_1 everywhere except under ϕ_2' where it makes choices like τ_2. Then the game reaches ϕ_1 when **I** and **II** play by σ_1 and τ, and the game reaches ϕ_2 when **I** and **II** play by σ_2 and τ.

Because we have excluded the case when the two subformulas are the same subformula, $\mathrm{Join}(\phi_1, \phi_2)$ cannot be an atom; it is either a conjunction or a disjunction. This proves the converse directions of (i) and (ii). $\square$

If $\mathcal{M}_\phi, X_\phi \models \psi$ for some $\psi \in F$ then player **II** has one or more winning strategies in the semantic game $\eth^{\mathrm{FOD}}(\mathcal{M}_\phi, \psi, X_\phi)$. We assume that τ is a winning strategy that chooses maximal teams and minimal covers, that is, if at position $(\mathcal{M}_\phi, \psi_1 \vee \psi_2, X)$ strategy τ chooses to split $X = X_1 \cup X_2$, then there is no winning strategy that in the same position would choose a split $X = Y_1 \cup X_2$ such that $X_1 \subsetneq Y_1$ and there is no winning strategy that in the same position would choose a split $X = X \cup \emptyset$ unless τ itself does so.

Define for each $\psi' \leq \psi$ the *game team* of ψ' (for τ), denoted $X^\tau_{\psi'}$, as the unique team that, when **II** plays by τ, is the team for ψ' in $\eth^{\mathrm{FOD}}(\mathcal{M}_\phi, \psi, X_\phi)$, i.e., the semantic game can reach position $(\psi', X^\tau_{\psi'}, \alpha)$ for some $\alpha \in \{\mathbf{I}, \mathbf{II}\}$.

Define for each $\psi' \leq \psi$ the *ϕ-canonical team* of ψ' (for τ), denoted $X^\phi_{\psi'}$, inductively as follows. Let $\psi' \leq \psi$ be an atom such that there is some strategy σ for **I** in the semantic game such that when **I** follows σ and **II** follows τ, the final position in the play of $\eth^{\mathrm{FOD}}(\mathcal{M}_\phi, \psi, X_\phi)$ is $(\psi', X', \mathbf{II})$, where $X' \neq \emptyset$. Then there is $\phi' \leq \phi$ such that $\phi' = \psi'$ and all $f \in \mathrm{Rel}(X')$ reach ϕ'. Define

$$X^\phi_{\psi'} := X_{\phi'}.$$

For all other atoms $\psi' \leq \psi$, define the ϕ-canonical team of ψ' to be the empty set. For compound subformulas in ψ define

$$X^\phi_{\psi_1 \wedge \psi_2} := X^\phi_{\psi_1} \cap X^\phi_{\psi_2},$$
$$X^\phi_{\psi_1 \vee \psi_2} := X^\phi_{\psi_1} \cup X^\phi_{\psi_2}.$$

The general idea is that game teams are the resources that **II**'s winning strategy τ assigns to subformulas of ψ, and ϕ-canonical teams describe the maximum capacity of subformulas of ψ—a ϕ-canonical team is a maximal team that can be assigned to the subformula such that **II** still wins the semantic game. The next lemma reflects this idea by stating that the resources assigned to a subformula of ψ are always a subset of the capacity of that subformula.

Lemma 6.4.5. *Let $\phi, \psi \in F$ and let τ be a winning strategy for* **II** *in the semantic game $\Game^{\mathrm{FOD}}(\mathcal{M}_\phi, X_\phi, \psi)$. If $\psi' \leq \psi$, then $X^\tau_{\psi'} \subseteq X^\phi_{\psi'}$.*

Proof. We prove the claim by induction on $\psi' \leq \psi$. If $\psi' = \theta_k$, the claim holds by definitions of $X^\tau_{\psi'}$ and $X^\phi_{\psi'}$. If $\psi' = \psi_1 \wedge \psi_2$, the induction hypothesis gives $X^\tau_{\psi'} = X^\tau_{\psi_1} \subseteq X^\phi_{\psi_1}$ and $X^\tau_{\psi'} = X^\tau_{\psi_2} \subseteq X^\phi_{\psi_2}$, whence $X^\tau_{\psi'} \subseteq X^\phi_{\psi_1} \cap X^\phi_{\psi_2} = X^\phi_{\psi'}$. If $\psi' = \psi_1 \vee \psi_2$, the induction hypothesis gives $X^\tau_{\psi'} = X^\tau_{\psi_1} \cup X^\tau_{\psi_2} \subseteq X^\phi_{\psi_1} \cup X^\phi_{\psi_2} = X^\phi_{\psi'}$. □

Lemma 6.4.6. *If $\phi, \psi \in F$ and $\mathcal{M}_\phi, X_\phi \models \psi$, then*

(i) $\mathcal{M}_\phi, X^\phi_{\psi'} \models \psi'$ for each $\psi' \leq \psi$, i.e. all subformulas in ψ are satisfied by their ϕ-canonical teams and the canonical model of ϕ;

(ii) $X^\phi_\psi = X_\phi$, i.e. the ϕ-canonical team of ψ is the canonical team of ϕ.

Proof. We prove (i) by induction on $\psi' \leq \psi$. The claim is true for ψ' if $X^\phi_{\psi'} = \emptyset$, so we need to consider only ψ' with nonempty ϕ-canonical teams. If ψ' is an atom, then $X^\phi_{\psi'} = X_{\phi'}$ for some $\phi' \leq \phi$ such that $\phi' = \psi'$, whence claim follows. If ψ' is of the form $\psi_1 \wedge \psi_2$, then the claim follows by the induction hypothesis and the fact that both $X^\phi_{\psi_1}$ and $X^\phi_{\psi_2}$ are subsets of $X^\phi_{\psi'} = X^\phi_{\psi_1} \cap X^\phi_{\psi_2}$. If ψ' is of the form $\psi_1 \vee \psi_2$, then $\mathcal{M}_\phi, X^\phi_{\psi'} \models \psi'$ by splitting $X^\phi_{\psi'} = X^\phi_{\psi_1} \cup X^\phi_{\psi_2}$ and using the induction hypothesis.

Then we prove (ii). We get $X^\phi_\psi \subseteq X_\phi$ trivially because the ϕ-canonical team of ψ is constructed as unions and intersections of some subteams of X_ϕ. To show the converse direction, let τ be the winning strategy of **II** that the ϕ-canonical teams of subformulas of ψ are based on. It suffices to prove the more general claim that for all $\psi' \leq \psi$, if the semantic game $\Game^{\mathrm{FOD}}(\mathcal{M}_\phi, \psi, X_\phi)$ can reach some position $(\psi', X', \mathbf{II})$ when **II** is following τ, then $X' \subseteq X^\phi_{\psi'}$. We prove it by induction on $\psi' \leq \psi$. If the game can reach some position $(\psi', X', \mathbf{II})$, where ψ' is an atom, **II** wins by her strategy being winning, so we know that all $f \in \mathrm{Rel}(X')$ reach some $\phi' \leq \phi$ such that $\phi' = \psi'$ and $X^\phi_{\psi'} = X_{\phi'}$. Thus $X' \subseteq X^\phi_{\psi'}$. If the game can reach some position $(\psi_1 \wedge \psi_2, X', \mathbf{II})$, the game can also reach both positions $(\psi_1, X', \mathbf{II})$ and $(\psi_2, X', \mathbf{II})$ whence by the induction hypothesis X' is contained in $X^\phi_{\psi_1} \cap X^\phi_{\psi_2} = X^\phi_{\psi_1 \wedge \psi_2}$. If the game can reach some position $(\psi_1 \vee \psi_2, X', \mathbf{II})$, then τ provides $\mathbf{II}$ with some split $X_1 \cup X_2 = X'$ such that the game can reach both positions $(\psi_1, X_1, \mathbf{II})$ and $(\psi_2, X_2, \mathbf{II})$. By the induction hypothesis $X_1 \subseteq X^\phi_{\psi_1}$ and $X_2 \subseteq X^\phi_{\psi_2}$, which gives us that $X' = X_1 \cup X_2$ is contained in $X^\phi_{\psi_1} \cup X^\phi_{\psi_2} = X^\phi_{\psi_1 \vee \psi_2}$. □

With these definitions, we can sketch how to prove the completeness of SKSgf for fragment F. The sketch is based on lemmas that are presented below.

Sketch of proof of Conjecture 6.4.1. The plan is to establish a finite sequence $\chi_0, \chi_1, \ldots$ of formulas while maintaining the invariant that $\mathcal{M}_\phi, X_\phi \models \chi_n$ and $\chi_n \vdash^{\mathrm{SKSf}} \psi$. When we end up with $\chi_n = \phi$, the proof is finished. Forming the sequence consists of phases. The rules in one phase are repeated until the ending condition of the phase holds. Then we proceed with the rules of the next phase.

We start by choosing $\chi_0 = \psi$. Then the invariant clearly holds. In general, assume that we have built the sequence $\chi_0, \ldots, \chi_n$ and the invariant holds.

Phase 1. If $\chi_n = \chi_n\{\theta_k\}$ such that $X^\tau_{\theta_k} = \emptyset$, then choose $\chi_{n+1} = \chi_n\{\bot\}$. We retain the invariant $\mathcal{M}_\phi \models \chi_{n+1} \vdash \psi$ by Lemma 6.4.7 and the inference

$$\mathsf{aw}{\downarrow}\,\frac{\chi_n\{\bot\}}{\chi_n\{\theta_k\}}\ .$$

Phase 1 ends when for all atomic $\theta \le \chi_n$ the game team X^τ_θ is nonempty. Then, in fact, the same condition holds also for all non-atomic subformulas of χ_n. Namely, if some $\psi' \le \chi_n$ had $X^\tau_{\psi'} = \emptyset$, all its subformulas—including at least one atom—would have an empty game team.

Phase 2. If $\chi_n = \chi_n\{(\psi_1 \wedge \psi_2) \vee \psi_3\}$ such that $X^\tau_{\psi_3} \subseteq X^\phi_{\psi_1}$, then choose $\chi_{n+1} := \chi_n\{\psi_1 \wedge (\psi_2 \vee \psi_3)\}$. We retain the invariant $\mathcal{M}_\phi \models \chi_{n+1} \vdash \psi$ by Lemma 6.4.9 and the inference

$$\mathsf{s}\,\frac{\chi_n\{\psi_1 \wedge (\psi_2 \vee \psi_3)\}}{\chi_n\{(\psi_1 \wedge \psi_2) \vee \psi_3\}}\ .$$

If $\chi_n = \chi_n\{\psi'\}$, where $\psi' = (\psi_1 \vee \psi_2) \wedge (\psi_3 \vee \psi_4)$, and there is a split $X_1 \cup X_2 = X^\tau_{\psi'}$ such that $X_1 \subseteq X^\phi_{\psi_1} \cap X^\phi_{\psi_3}$ and $X_2 \subseteq X^\phi_{\psi_2} \cap X^\phi_{\psi_4}$, then choose $\chi_{n+1} := \chi_n\{(\psi_1 \wedge \psi_3) \vee (\psi_2 \wedge \psi_4)\}$. We retain the invariant $\mathcal{M}_\phi \models \chi_{n+1} \vdash \psi$ by Lemma 6.4.10 and the inference

$$\mathsf{m}\,\frac{\chi_n\{(\psi_1 \wedge \psi_3) \vee (\psi_2 \wedge \psi_4)\}}{\chi_n\{(\psi_1 \vee \psi_2) \wedge (\psi_3 \vee \psi_4)\}}\ .$$

Phase 2 ends when the above rules cannot be applied anymore. Then we get from Lemma 6.4.11 that, for all $\psi' := \bigwedge_{i\in I} \psi_i \leq \psi$ and all $i \in I$, $X^{\phi}_{\psi_i} = X^{\phi}_{\psi'}$. Further, from Lemma 6.4.12 we get that for all $\psi' \leq \psi$ there is $\phi' \leq \phi$ such that $X^{\phi}_{\psi'} = X_{\phi'}$. Finally, Lemma 6.4.13 gives $\phi \vdash^{\mathsf{c}\uparrow,\mathsf{w}\uparrow} \psi$. ($\square$)

The next lemma states that under a certain condition, if the canonical model and team of ϕ satisfy some formula ψ, they also satisfy a formula from which we can infer ψ by atomic weakening.

Lemma 6.4.7. *Let $\phi, \psi\{\theta_k\} \in F$. If $\mathcal{M}_\phi, X_\phi \models \psi\{\theta_k\}$ and $X^{\tau}_{\theta_k} = \emptyset$, then $\mathcal{M}_\phi, X_\phi \models \psi\{\bot\}$.*

Proof. We need to provide a winning strategy τ' for $\mathbf{II}$ in the semantic game $\partial^{\mathrm{FOD}}(\mathcal{M}_\phi, \psi\{\bot\}, X_\phi)$. We can, in fact, choose τ' to be τ. $\square$

A similar lemma is formulated below for atomic contraction. Alternatively, the below lemma could be replaced by a nondeterministic approach in the proof of Conjecture 6.4.1; duplicate each atom instance in ψ some number of times, where this number is whatever suits us later in the construction of the derivation in Conjecture 6.4.1.

Lemma 6.4.8. *Let $\phi, \psi\{\psi'\} \in F$, where $\psi' := \theta_k \vee (\gamma \wedge \delta)$. If $\mathcal{M}_\phi, X_\phi \models \psi\{\psi'\}$ and $X^{\phi}_{\psi'} = X^{\phi'}$ for some $\phi' \leq \phi$, then $\mathcal{M}_\phi, X_\phi \models \psi\{\theta_k \vee \psi'\}$.* ($\square$)

The following lemma states that under a certain condition, if the canonical model and team of ϕ satisfy some formula ψ, they also satisfy a formula from which we can infer ψ by switch. The idea is that when the condition holds, the switch is a step closer to ϕ from ψ.

Lemma 6.4.9. *Let $\phi, \psi\{\psi'\} \in F$, where $\psi' := (\psi_1 \wedge \psi_2) \vee \psi_3$. If $\mathcal{M}_\phi, X_\phi \models \psi\{\psi'\}$ and $X^{\tau}_{\psi_3} \subseteq X^{\phi}_{\psi_1}$, then $\mathcal{M}_\phi, X_\phi \models \psi\{\psi''\}$, where $\psi'' := \psi_1 \wedge (\psi_2 \vee \psi_3)$.*

Proof. We must give player $\mathbf{II}$ a winning strategy τ' in $\partial^{\mathrm{FOD}}(\mathcal{M}_\phi, \psi\{\psi''\}, X_\phi)$. Let τ' be otherwise like τ but with slight changes when the play reaches position $(\mathcal{M}_\phi, \psi'', X^{\tau}_{\psi'})$. At the conjunction, player $\mathbf{I}$ can choose the subformula ψ_1, so we need to ensure that τ' is still winning at position $(\mathcal{M}_\phi, \psi_1, X^{\tau}_{\psi'})$. We know that $X^{\tau}_{\psi'} = X^{\tau}_{\psi_1} \cup X^{\tau}_{\psi_3} \subseteq X^{\phi}_{\psi_1}$, so $\mathcal{M}_\phi, X^{\tau}_{\psi'} \models \psi_1$ as we wanted. If player $\mathbf{I}$ chooses $\psi_2 \vee \psi_3$ instead, let τ' split $X^{\tau}_{\psi'} = X^{\tau}_{\psi_2} \cup X^{\tau}_{\psi_3}$. The play proceeds to position $(\mathcal{M}_\phi, \psi_k, X^{\tau}_{\psi_k})$ for either $k \in \{2, 3\}$. Again, τ' is winning at this position because τ is. $\square$

The following lemma states that under a certain condition, if the canonical model and team of ϕ satisfy some formula ψ, they also satisfy a formula from which we can infer ψ by medial.

Lemma 6.4.10. *Let $\phi, \psi\{\psi'\} \in F$, where $\psi' := (\psi_1 \vee \psi_2) \wedge (\psi_3 \vee \psi_4)$. If there is a split $X_1 \cup X_2 = X^\tau_{\psi'}$ such that $X_1 \subseteq X^\phi_{\psi_1} \cap X^\phi_{\psi_3}$ and $X_2 \subseteq X^\phi_{\psi_2} \cap X^\phi_{\psi_4}$, then $\mathcal{M}_\phi, X_\phi \models \psi\{\psi''\}$, where $\psi'' := (\psi_1 \wedge \psi_3) \vee (\psi_2 \wedge \psi_4)$.*

Proof. We need to give player **II** a winning strategy τ' in the semantic game $\eth^{\mathrm{FOD}}(\mathcal{M}_\phi, \psi\{\psi''\}, X_\phi)$. Let τ' be like τ except in subformulas of ψ''. When the play reaches position $(\mathcal{M}_\phi, \psi'', X^\tau_{\psi'})$, let τ' split $X^\tau_{\psi'} = X_1 \cup X_2$. The play proceeds to position $(\mathcal{M}, \psi_k \wedge \psi_{k+2}, X_k)$ for some $k \in \{1, 2\}$. Choose τ' to play the remaining game according to a winning strategy given by the fact that $\mathcal{M}_\phi, X_k \models \psi_k \wedge \psi_{k+2}$. $\square$

The following conjectures tell us when all the necessary inferences by switch and medial have been done, and what then remains of ψ can be derived from ϕ by atomic co-weakening and atomic co-contraction.

Lemma 6.4.11. *Let $\phi, \psi \in F$. If there is no $(\psi_1 \wedge \psi_2) \vee \psi_3 \leq \psi$ such that $X^\tau_{\psi_3} \subseteq X^\phi_{\psi_1}$ and there is no $\psi' := (\psi_1 \vee \psi_2) \wedge (\psi_3 \vee \psi_4)$ and $X_1 \cup X_2 =: X^\tau_{\psi'}$ such that $X_1 \subseteq X^\phi_{\psi_1} \cap X^\phi_{\psi_3}$ and $X_2 \subseteq X^\phi_{\psi_2} \cap X^\phi_{\psi_4}$, then for all $\bigwedge_{i\in I} \psi_i \leq \psi$ and $i, j \in I$ we have $X^\phi_{\psi_i} = X^\phi_{\psi_j}$.* ($\square$)

Lemma 6.4.12. *Let $\phi, \psi \in F$. If for all $\psi' := \bigwedge_{i\in I} \psi_i \leq \psi$ and $i \in I$, $X^\phi_{\psi_i} = X^\phi_{\psi'}$, then for all $\psi' \leq \psi$ there is $\phi' \leq \phi$ such that $X^\phi_{\psi'} = X_{\phi'}$.* ($\square$)

Lemma 6.4.13. *Let $\phi, \psi \in F$. If for all $\psi' := \bigwedge_{i\in I} \psi_i \leq \psi$ and all $i \in I$, $X^\phi_{\psi_i} = X^\phi_{\psi'}$, and for all $\psi' \leq \psi$ there is $\phi' \leq \phi$ such that $X^\phi_{\psi'} = X_{\phi'}$, then $\phi \vdash^{\mathsf{c}\uparrow,\mathsf{w}\uparrow} \psi$.*

Proof. We prove by induction on $\psi' \leq \psi$ the claim that if $\phi' \leq \phi$ is maximal such that $X^\phi_{\psi'} = X'_\phi$, then $\begin{array}{c}\phi'\\ \Big\Vert{\scriptstyle \mathsf{c}\uparrow,\mathsf{w}\uparrow}\\ \psi'\end{array}$.

Atomic case If $\psi' = \theta_k$, then $\phi' = \bigwedge_{i\in I} \phi_i$ and $\phi_i = \theta_k$ for some $i \in I$. We can infer

$$\mathsf{w}\uparrow \frac{\phi'}{\psi'} \ .$$

Case $\wedge$ If $\psi' = \psi_1 \wedge \psi_2$, then by $X^\phi_{\psi_i} = X^\phi_{\psi'}$, ϕ' is maximal also with respect to $X^\phi_{\psi_i} = X_{\phi'}$ for both $i \in \{1, 2\}$. Induction hypothesis gives derivations

Δ_1 and Δ_2 such that we can derive

$$\dfrac{\dfrac{\phi'}{\phi' \wedge \phi'}\,\mathsf{c}\!\uparrow}{\psi_1 \wedge \phi'}\,\{\Delta_1\}\wedge\phi' \,\Big\|\, \mathsf{c}\!\uparrow,\mathsf{w}\!\uparrow \quad \dfrac{\psi_1 \wedge \phi'}{\psi_1 \wedge \psi_2}\,\psi_1\wedge\{\Delta_2\} \,\Big\|\, \mathsf{c}\!\uparrow,\mathsf{w}\!\uparrow .$$

Case $\vee$ Open question. $(\Box)$

Chapter 7

1-Semantics

In this chapter I develop a new semantics for the syntax of dependence logic and compare it with the previous semantics. The new semantics is called *1-semantics*. I prove several properties of 1-semantics; it strictly contains the old semantics, in the sense that the old semantics can be computed from 1-semantics but the contrary is not possible. I describe a translation of formulas of existential second order logic into 1-semantics, proving that 1-semantics has expressive power equal to existential second order logic. 1-semantics contains a semantic definition for negation, and thus 1-semantics is defined for all $\phi \in \mathrm{FOD}$ without resorting to any negation normal form. The law of excluded middle, formulated suitably, holds for 1-semantics. Finally, the definition of 1-semantics comes in a natural way as a type shift from the Tarskian semantics of first order logic. The type shift can be applied to any connective and quantifier of first order logic. I also consider 1-semantics in light of game theoretic semantics and present a novel game that I conjecture to yield 1-semantics.

7.1 Definitions

In order to be clear and precise, we must start by defining 1-semantics for teams whose domain is the set of all element variables. We can then refine the definition into teams with finite domains.

7.1.1 1-Semantics in ω-Teams

Define an *ω-team* X for a model $\mathcal{M}$ to be any relation on M of arity ω, i.e. $X \subseteq M^\omega$. We call the elements of an ω-team tuples and denote them $\overline{a} = (a_1, a_2, \dots)$ or similarly. For $\overline{a} \in X$, let $s_{\overline{a}}$ denote the assignment

$$\begin{array}{lcl}
\mathcal{M}, X \models^1 \bot & \iff & X = \emptyset \\
\mathcal{M}, X \models^1 \top & \iff & X = M^\omega \\
\mathcal{M}, X \models^1 Rt_1 \ldots t_n & \iff & X = \big\{\overline{a} : \mathcal{M}, s_{\overline{a}} \models Rt_1 \ldots t_n\big\} \\
\mathcal{M}, X \models^1 (t_1 \ldots t_n) \rightsquigarrow u & \iff & \text{there is } f \text{ s.t. } X = \big\{\overline{a} : \\
 & & s_{\overline{a}}(u) = f\big(s_{\overline{a}}(t_1) \ldots s_{\overline{a}}(t_n)\big)\big\} \\
\mathcal{M}, X \models^1 \neg\phi & \iff & \mathcal{M}, \complement X \models^1 \phi \\
\mathcal{M}, X \models^1 \phi \vee \psi & \iff & \text{there is } Y, Z \text{ s.t. } Y \cup Z = X \text{ and} \\
 & & \mathcal{M}, Y \models^1 \phi \text{ and } \mathcal{M}, Z \models^1 \psi \\
\mathcal{M}, X \models^1 \phi \wedge \psi & \iff & \text{there is } Y, Z \text{ s.t. } Y \cap Z = X \text{ and} \\
 & & \mathcal{M}, Y \models^1 \phi \text{ and } \mathcal{M}, Z \models^1 \psi \\
\mathcal{M}, X \models^1 \exists x_n \phi & \iff & \text{there is } Y \text{ s.t. } X = \mathsf{C}_n Y \text{ and } \mathcal{M}, Y \models^1 \phi \\
\mathcal{M}, X \models^1 \forall x_n \phi & \iff & \text{there is } Y \text{ s.t. } \complement X = \mathsf{C}_n \complement Y \text{ and } \mathcal{M}, Y \models^1 \phi
\end{array}$$

Figure 7.1: 1-semantics in ω-teams

that maps $s(x_i) = a_i$ for all i. For $\overline{a} \in X$, $a \in M$ and $n < \omega$, denote $\overline{a}(n \mapsto a) := (a_1, \ldots, a_{n-1}, a, a_{n+1}, \ldots)$. Given an ω-team X for a model $\mathcal{M}$, denote *cylindrification* of X along the n'th column by $\mathsf{C}_n X := \{\overline{a}(n \mapsto a) : \overline{a} \in X \text{ and } a \in M\}$.[1]

Definition 7.1.1. Satisfaction of a formula $\phi \in \mathrm{FOD}$ in a model $\mathcal{M}$ by an ω-team X in 1-semantics, denoted $\mathcal{M}, X \models^1 \phi$, or simply $X \models^1 \phi$ when the model is clear from context, is defined as in Figure 7.1.[2]

Theorem 7.1.2 (Only Free Variables Matter)**.** *For an ω-team X for some model $\mathcal{M}$ and formula $\phi \in \mathrm{FOD}$, if $X \models^1 \phi$ then for all $\overline{a} \in X$ and $\overline{c} \in M^\omega$ we have $\overline{a}(\phi, \overline{c}) \in X$, where $\overline{a}(\phi, \overline{c}) := \overline{a}(n \mapsto c_n)_{x_n \notin \mathrm{FV}(\phi)}$ denotes a tuple where the values of irrelevant (with respect to the syntax of ϕ) variables have been replaced.*

Proof. Induction on $\phi \in \mathrm{FOD}$.

Case $\bot$. If $X \models^1 \bot$ then $X = \emptyset$, whence the claim is holds vacuously.

[1] The cylindrification operation was introduced with cylindric algebras, see [11].

[2] With 1-semantics, we need not consider negation as shorthand notation. Thus, Definition 2.4.2 is to be applied only to FOD formulas that are interpreted with the semantics in Definition 2.4.1.

Case $\top$**.** If $X \models^1 \top$ then $X = M^\omega$, whence $\overline{a}(\top, \overline{c}) = \overline{c} \in X$ for all $\overline{a} \in X$ and $\overline{c} \in M^\omega$.

Case $Rt_1 \ldots t_n$**.** If $X \models^1 \phi$, $\overline{a} \in X$ and $\overline{c} \in M^\omega$, then $\mathcal{M}, s_{\overline{a}(\phi,\overline{c})} \models \phi$ because $s_{\overline{a}(\phi,\overline{c})}$ equals $s_{\overline{a}}$ in all free variables of ϕ and in FO semantics only the interpretation of free variables affects satisfaction. Thus $\overline{a}(\phi, \overline{c}) \in X$.

Case $(t_1 \ldots t_n) \rightsquigarrow u$**.** If $X \models^1 \phi$, $\overline{a} \in X$ and $\overline{c} \in M^\omega$, then $s_{\overline{a}(\phi,\overline{c})}$ and $s_{\overline{a}}$ agree on the evaluation of the terms t_i and u because $s_{\overline{a}(\phi,\overline{c})}$ equals $s_{\overline{a}}$ in all free variables of ϕ. Thus $\overline{a}(\phi, \overline{c}) \in X$.

Case $\neg\psi$**.** If $X \models^1 \phi$ and $\overline{a} \in X$, then $\complement X \models^1 \psi$ and $\overline{a} \notin \complement X$. If $\overline{a}(\psi, \overline{c}) \in \complement X$ for any $\overline{c} \in M^\omega$, then $\overline{a} \in \complement X$ by the induction hypothesis, a contradiction. Thus $\overline{a}(\phi, \overline{c}) = \overline{a}(\psi, \overline{c}) \in X$ for all $\overline{c} \in M^\omega$.

Case $\psi_1 \vee \psi_2$**.** If $X \models^1 \phi$ and $\overline{a} \in X$, then there are ω-teams Y_1 and Y_2 such that $X = Y_1 \cup Y_2$, $Y_i \models^1 \psi_i$ for both $i \in \{1, 2\}$, and $\overline{a} \in Y_{i_0}$ for some i_0. Let $\overline{c} \in M^\omega$. Choose $\overline{e} \in M^\omega$ such that $\overline{e}$ agrees with $\overline{a}$ in $\mathrm{FV}(\phi)$ and with $\overline{c}$ in all other variables. By the induction hypothesis, $\overline{a}(\psi_{i_0}, \overline{e}) \in Y_{i_0}$. Then $\overline{a}(\phi, \overline{c}) = \overline{a}(\phi, \overline{e}) = \overline{a}(\psi_{i_0}, \overline{e}) \in X$.

Case $\psi_1 \wedge \psi_2$**.** If $X \models^1 \phi$ and $\overline{a} \in X$, then there are ω-teams Y_1 and Y_2 such that $X = Y_1 \cap Y_2$, $Y_i \models^1 \psi_i$ for both $i \in \{1, 2\}$, and $\overline{a} \in Y_i$ for both $i \in \{1, 2\}$. Let $\overline{c} \in M^\omega$. Choose $\overline{e} \in M^\omega$ such that $\overline{e}$ agrees with $\overline{a}$ in $\mathrm{FV}(\phi)$ and with $\overline{c}$ in all other variables. By the induction hypothesis, $\overline{a}(\psi_i, \overline{e}) \in Y_i$ for both $i \in \{1, 2\}$. Then $\overline{a}(\phi, \overline{c}) = \overline{a}(\phi, \overline{e}) = \overline{a}(\psi_i, \overline{e}) \in Y_i$ for both $i \in \{1, 2\}$, thus $\overline{a}(\phi, \overline{c}) \in X$.

Case $\exists x_n \psi$**.** If $X \models^1 \phi$ and $\overline{a} \in X$, then there is an ω-team Y such that $X = \mathsf{C}_n Y$ and $Y \models^1 \psi$. Let $\overline{c} \in M^\omega$. From $\overline{a} \in \mathsf{C}_n Y$ we get $\overline{a}(n \mapsto e) \in Y$ for some $e \in M$. By the induction hypothesis, $\overline{a}(n \mapsto e)(\psi, \overline{c}) \in Y$, whence $\overline{a}(\phi, \overline{c}) \in \mathsf{C}_n Y = X$.

Case $\forall x_n \psi$**.** If $X \models^1 \phi$ and $\overline{a} \in X$, then there is an ω-team Y such that $\complement X = \mathsf{C}_n \complement Y$ and $Y \models^1 \psi$. Let $\overline{c} \in M^\omega$. From $\overline{a} \in \complement \mathsf{C}_n \complement Y$ we get $\overline{a}(n \mapsto e) \in Y$ for all $e \in M$. Combining this with the induction hypothesis, we get $\overline{a}(\psi, \overline{c})(n \mapsto e) \in Y$ for all $e \in M$, whence $\overline{a}(\phi, \overline{c}) \in \complement \mathsf{C}_n \complement Y = X$. □

Note that we do not need to assume anything about the axiom of choice in the proof of Theorem 7.1.2. This is in contrast to the similar theorem of dependence logic, Theorem 2.4.7, where we need to assume the axiom of choice.

Theorem 7.1.2 suggests viewing ω-teams as having two parts, divided by the domain; the *trivial part* of an ω-team X is the subset $V \subseteq \omega$ such that $s(x_n \mapsto a_n)_{n \in V} \in X$ for all $s \in X$ and $\overline{a} \in M^\omega$. The *nontrivial part* of an ω-team is the complement of its trivial part. Theorem 7.1.2 shows that the concept of trivial part is coherent with the set of variables in a formula whose interpretation may affect the satisfaction of the formula; free variables are always in the nontrivial part of an ω-team that satisfies the formula.

7.1.2 Type Shifting

In linguistics, there is a concept called *type shifting* [18]. A type shift is a mapping of objects of one type to objects of another type. A type can be an element, a set of elements, a set of sets of elements, etc. One of the main differences between first order logic and dependence logic (both P- and 1-semantics) is the type of semantic objects. In first order logic, semantic objects are assignments, whereas in dependence logic, semantic objects are sets of assignments. This brings forth the question, is there a type shift that leads from first order logic to dependence logic. It turns out that there indeed is one for 1-semantics.

To give a Tarskian definition of formula satisfaction is essentially to give an operation for each atomic formula and each kind of compound formula that computes the interpretation of the formula. For example, there must be an operation such that, given the interpretations of two subformulas, the operation computes the interpretation of the conjunction of the subformulas. In the case of an atomic formula, the operation is constant, not taking arguments. Thus each atomic formula has its own operation.[3]

In order to shift first order semantics from assignments to ω-teams, it therefore suffices to shift the operations on formula interpretations. The operations that correspond to connectives and quantifiers take one or two formula interpretations as arguments and map them to another formula interpretation. A nullary operation represents a named formula interpretation.

Let us call *1-shift* the type shift that maps an operation

$$(X_1, \ldots, X_n) \mapsto f(X_1, \ldots, X_n)$$

[3]It is conventional that satisfaction of an atomic formula is defined in one step where parts of the atomic formula are arguments to the operation. For example, the operation computing the interpretation of atomic formulas of the form $Rt_1 \ldots t_n$, the relation symbol R and terms $t_1, \ldots, t_n$ are the arguments. This is practical in truth definitions but not in the present context, as the arguments are not formulas but finer-grained elements that do not have an interpretation with respect to a model in the form of a set of semantic objects. Therefore I skip this further level of detail.

to the operation

$$(\mathcal{X}_1,\dots,\mathcal{X}_n) \mapsto \{f(X_1,\dots,X_n) : X_1 \in \mathcal{X}_1,\ \dots,\ X_n \in \mathcal{X}_n\}.$$

Here, the sets X_i intend to be interpretations of first order formulas, i.e. sets of assignments, and the sets $\mathcal{X}_i$ stand for interpretations of dependence formulas, i.e. sets of teams.

Theorem 7.1.3. *1-semantics for* FO *formulas is the 1-shift of* FO *semantics.*

Proof. We prove by induction on formulas $\phi \in \mathrm{FO}$ that applying 1-shift to FO semantics on assignments leads to the definition of 1-semantics on ω-teams, Definition 7.1.1. In this proof, I consider assignments as infinite tuples.

The atomic formulas $\bot$ and $\top$ have the constant interpretations $f_\bot() = \emptyset$ and $f_\top() = M^\omega$, where M is the universe of the model in question. 1-shifting them gives the 1-interpretations $g_\bot() = \{\emptyset\}$ and $g_\top() = \{M^\omega\}$. Thus, $X \in [\![\bot]\!]^1$ if and only if $X \in g_\bot()$ if and only if $X = \emptyset$. Similarly, $X \in [\![\top]\!]^1$ if and only if $X \in g_\top()$ if and only if $X = M^\omega$.

Each atomic formula of the form $Rt_1 \dots t_n$ has a constant interpretation $f() = \{\overline{a} : (s_{\overline{a}}(t_1),\dots,s_{\overline{a}}(t_n)) \in R^{\mathcal{M}}\}$. 1-shifting it gives the constant 1-interpretation $g() = \{f()\}$. Thus, $X \in [\![Rt_1 \dots t_n]\!]^1$ if and only if $X \in g()$ if and only if $X = \{\overline{a} : (s_{\overline{a}}(t_1),\dots,s_{\overline{a}}(t_n)) \in R^{\mathcal{M}}\}$.

Negation $\neg\phi$ in FO has the interpretation $f(X) = \{\overline{a} : \overline{a} \notin X\} = \complement X$, where X is the interpretation of ϕ. 1-shifting it gives the 1-interpretation $g(\mathcal{X}) = \{f(X) : X \in \mathcal{X}\} = \{\complement X : X \in \mathcal{X}\}$, where $\mathcal{X}$ is the 1-interpretation of ϕ. Thus, $X \in [\![\neg\phi]\!]^1$ if and only if $X \in g([\![\phi]\!]^1)$ if and only if $X = \complement Y$ for some $Y \in [\![\phi]\!]^1$.

Disjunction $\phi \vee \psi$ in FO has the interpretation $f(X,Y) = \{\overline{a} : \overline{a} \in X \text{ or } \overline{a} \in Y\} = X \cup Y$, where X and Y are the interpretations of ϕ and ψ, respectively. 1-shifting it gives the 1-interpretation $g(\mathcal{X},\mathcal{Y}) = \{X \cup Y : X \in \mathcal{X} \text{ and } Y \in \mathcal{Y}\}$, where $\mathcal{X}$ and $\mathcal{Y}$ are the 1-interpretations of ϕ and ψ, respectively. Thus, $X \in [\![\phi \vee \psi]\!]^1$ if and only if $X \in g([\![\phi]\!]^1, [\![\psi]\!]^1)$ if and only if $X = Y \cup Z$ for some $Y \in [\![\phi]\!]^1$ and $Z \in [\![\psi]\!]^1$.

Conjunction $\phi \wedge \psi$ in FO has the interpretation $f(X,Y) = \{\overline{a} : \overline{a} \in X \text{ and } \overline{a} \in Y\} = X \cap Y$, where X and Y are the interpretations of ϕ and ψ, respectively. 1-shifting it gives the 1-interpretation $g(\mathcal{X},\mathcal{Y}) = \{X \cap Y : X \in \mathcal{X} \text{ and } Y \in \mathcal{Y}\}$, where $\mathcal{X}$ and $\mathcal{Y}$ are the 1-interpretations of ϕ and ψ, respectively. Thus, $X \in [\![\phi \wedge \psi]\!]^1$ if and only if $X \in g([\![\phi]\!]^1, [\![\psi]\!]^1)$ if and only if $X = Y \cap Z$ for some $Y \in [\![\phi]\!]^1$ and $Z \in [\![\psi]\!]^1$.

Existential quantification $\exists x_n \phi$ in FO has the interpretation $f(X) = \mathsf{C}_n X$, where X is the interpretation of ϕ. 1-shifting it gives the 1-interpretation $g(\mathcal{X}) = \{\mathsf{C}_n X : X \in \mathcal{X}\}$, where $\mathcal{X}$ is the 1-interpretation of ϕ.

Thus, $X \in [\![\exists x_n\phi]\!]^1$ if and only if $X \in g([\![\phi]\!]^1)$ if and only if $X = \mathsf{C}_n Y$ for some $Y \in [\![\phi]\!]^1$.

Lastly, universal quantification $\forall x_n \phi$ in FO has the interpretation $f(X) = \complement\mathsf{C}_n\complement X$, where X is the interpretation of ϕ. 1-shifting it gives the 1-interpretation $g(\mathcal{X}) = \{\complement\mathsf{C}_n\complement X : X \in \mathcal{X}\}$, where $\mathcal{X}$ is the 1-interpretation of ϕ. Thus, $X \in [\![\forall x_n\phi]\!]^1$ if and only if $X \in g([\![\phi]\!]^1)$ if and only if $X = \complement\mathsf{C}_n\complement Y$ for some $Y \in [\![\phi]\!]^1$. □

Thus 1-shift defines 1-semantics for all formulas except D-formulas. The semantics of D-formulas is just a suitable adaptation from the corresponding formulas in dependence logic. A D-formula states that the ω-team is (the graph of) a function, modulo transformations done by interpretation of terms in the D-formula.

Thinking of formula interpretations as nullary operations, we see that there is a translation of FO into 1-semantics with the identity function being the syntactic translation and the mapping $X \mapsto \{X\}$ being the semantic translation, where X is an interpretation of a first order formula. That is, $[\![\phi]\!]^1 = \{[\![\phi]\!]^{\mathrm{FO}}\}$ for all $\phi \in \mathrm{FO}$. The name "1-semantics" refers to this singleton correspondence between 1-interpretations and FO-interpretations. Following this idea, I use the nickname *P-semantics* for the semantics that Hodges gave for formulas of independence friendly logic and for the semantics that Väänänen uses for dependence logic, defined in Chapter 2. P refers to the fact that $[\![\phi]\!]^{\mathrm{P}} = \mathcal{P}[\![\phi]\!]^{\mathrm{FO}}$ for all $\phi \in \mathrm{FO}$, i.e. the powerset correspondence between P-interpretations and FO-interpretations. We may denote satisfaction in P-semantics by $\mathcal{M}, X \models^{\mathrm{P}} \phi$, to make the necessary distinction.[4]

7.1.3 1-Semantics in Teams

I will now give an equivalent definition of 1-semantics where in place of ω-teams we use teams without any restriction on their domains, as is done in P-semantics. The complement of a team we take in the corresponding full team, $\complement X = X^{\mathcal{M}}_{\mathrm{Dom}(X)} \setminus X$, where X is a team for model $\mathcal{M}$. Given a team X for a model $\mathcal{M}$, denote *cylindrification* of X along variable x_n by $\mathsf{C}_{x_n} X := \{s(x_n \mapsto a) : s \in X \text{ and } a \in M\}$. Define the *cylindric restriction* of X to V as

$$X \mathbin{\phi} V := \{s \restriction V : s(x_n \mapsto a_n)_{x_n \in \mathrm{Dom}(X) \setminus V} \in X \text{ for all } \overline{a} \in M^\omega\}.$$

[4]Note that $\models^{\mathrm{P}}$ is the same semantics that Chapter 2 denotes by $\models^{\mathrm{FOD}}$. Because in this chapter we have two different semantics for the same set of formulas, FOD, it no longer makes sense to denote either semantics by the symbol of the set of formulas, hence the change in notation.

Finally, define the *cylindric co-restriction* of X from V as

$$X\mathbin{\mathring{\downharpoonright}}V := X\mathbin{\mathring{\upharpoonright}}(\mathrm{Dom}(X)\setminus V).$$

If $V = \{x\}$, we denote $X\upharpoonright V = X\upharpoonright x$, $X\downharpoonright V = X\downharpoonright x$, $X\mathbin{\mathring{\upharpoonright}}V = X\mathbin{\mathring{\upharpoonright}}x$, and $X\mathbin{\mathring{\downharpoonright}}V = X\mathbin{\mathring{\downharpoonright}}x$.

Teams can have any set of variables as their domain. In order to express 1-semantics, as it is defined on ω-teams, using teams that in general are missing some of the information that ω-teams carry, we must decide on which ω-team each team corresponds to. Theorem 7.1.2 suggests a many-to-one correspondence between teams and ω-teams; a team always contains the nontrivial part of an ω-team (and optionally some of the trivial part, as well). More precisely, a nonempty team X corresponds to its *ω-closure* which is the ω-team

$$\overline{X} := \{\overline{a} : \text{ there is } s \in X \text{ such that } s(x_n) = a_n \text{ for all } x_n \in \mathrm{Dom}(X)\}.$$

Turning the definition around, a nonempty ω-team corresponds to each of its cylindric restrictions to a set of variables that contains the nontrivial part of the ω-team.[5] The empty team corresponds to the empty ω-team and vice versa.

Theorem 7.1.4. *ω-closure commutes with the complement, union, intersection and cylindrification operations on teams.*

Proof. **Complement.** $\overline{a} \in \complement\overline{X}$ iff there is no $s \in X$ such that $s = \overline{a}\upharpoonright\mathrm{Dom}(X)$ iff $\overline{a}\upharpoonright\mathrm{Dom}(X) \in \complement X$ iff there is $s \in \complement X$ such that $s = \overline{a}\upharpoonright\mathrm{Dom}(X)$ iff $\overline{a} \in \overline{\complement X}$.

Union. Assuming $\mathrm{Dom}(X_1) = \mathrm{Dom}(X_2) =: V$, $\overline{a} \in \overline{X_1} \cup \overline{X_2}$ iff $\overline{a} \in \overline{X_1}$ or $\overline{a} \in \overline{X_2}$ iff there is $s \in X_1$ or $s \in X_2$ such that $s = \overline{a}\upharpoonright V$ iff there is $s \in X_1 \cup X_2$ such that $s = \overline{a}\upharpoonright V$ iff $\overline{a} \in \overline{X_1 \cup X_2}$.

Intersection. Assuming $\mathrm{Dom}(X_1) = \mathrm{Dom}(X_2) =: V$, $\overline{a} \in \overline{X_1} \cap \overline{X_2}$ iff $\overline{a} \in \overline{X_1}$ and $\overline{a} \in \overline{X_2}$ iff there is $s_1 \in X_1$ and $s_2 \in X_2$ such that $s_1 = \overline{a}\upharpoonright V$ and $s_2 = \overline{a}\upharpoonright V$ iff there is $s \in X_1 \cap X_2$ such that $s = \overline{a}\upharpoonright V$ iff $\overline{a} \in \overline{X_1 \cup X_2}$.

Cylindrification. Letting $n < \omega$, $\overline{a} \in \overline{\mathsf{C}_{x_n} X}$ iff there is $s \in \mathsf{C}_{x_n} X$ such that $s = \overline{a}\upharpoonright\mathrm{Dom}(\mathsf{C}_{x_n} X)$ iff there is $s \in X$ and $a \in M$ such that $s(x_n \mapsto a) = \overline{a}\upharpoonright\mathrm{Dom}(\mathsf{C}_{x_n} X)$ iff there is $\overline{c} \in \overline{X}$ and $a \in M$ such that $\overline{a} = \overline{c}(n \mapsto a)$ iff $\overline{a} \in \mathsf{C}_{x_n}\overline{X}$. □

[5] Here we implicitly convert from tuples to assignments, and from indices to variables, in the obvious way.

Lemma 7.1.5. *For teams X and Y, if $\overline{X} = \overline{Y}$ and $\mathrm{Dom}(X) = \mathrm{Dom}(Y)$, then $X = Y$.*

The concept of ω-closure acts as a link between ω-teams and teams of arbitrary domains. It leads us to the following definition of 1-semantics for all teams X.

Definition 7.1.6. If $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(X)$ then define

$$\mathcal{M}, X \models^1 \phi \iff \mathcal{M}, \overline{X} \models^1 \phi.$$

Theorem 7.1.2 enables us to give an alternative, equivalent definition for 1-semantics which may suit better those who wish to work with teams with finite domains. Finite domains are desirable in certain translations. For example, when expressing 1-semantics or P-semantics in second order logic by encoding a team as the value relation variable, one has only relation variables of finite arity.

Theorem 7.1.7. *The definition of 1-semantics for all teams X can be formulated equivalently in the form of Figure 7.2, where we assume that $\mathrm{Dom}(X)$ contains all the free variables of the formula whose satisfaction we are defining.*

Proof. We prove by induction on $\phi \in \mathrm{FOD}$ that satisfaction as in Definition 7.1.6 is equivalent with the conditions listed above. The proof is based on Theorem 7.1.2, Theorem 7.1.4 and the fact that $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(X)$.

Case $\bot$. $X \models^1 \bot$ iff $\overline{X} \models^1 \bot$ iff $\overline{X} = \emptyset$ iff $X = \emptyset$.

Case $\top$. $X \models^1 \top$ iff $\overline{X} \models^1 \top$ iff $\overline{X} = M^\omega$ iff $X = X^{\mathcal{M}}_{\mathrm{Dom}(X)}$.

Case $Rt_1 \ldots t_n$. $X \models^1 \phi$ iff $\overline{X} \models^1 \phi$ iff $\overline{X} = \{\overline{a} : \mathcal{M}, s_{\overline{a}} \models \phi\}$ iff $X = \{s \in X^{\mathcal{M}}_{\mathrm{Dom}(X)} : \mathcal{M}, s \models \phi\}$.

Case $(t_1 \ldots t_n) \rightsquigarrow u$. $X \models^1 \phi$ iff $\overline{X} \models^1 \phi$ iff there is f such that $\overline{X} = \{\overline{a} : s_{\overline{a}}(u) = f(s_{\overline{a}}(t_1), \ldots, s_{\overline{a}}(t_n))\}$ iff $X = \{s \in X^{\mathcal{M}}_{\mathrm{Dom}(X)} : s(u) = f(s(t_1), \ldots, s(t_n))\}$.

Case $\neg\psi$. $X \models^1 \phi$ iff $\overline{X} \models^1 \phi$ iff $\complement\overline{X} \models^1 \psi$ iff $\overline{\complement X} \models^1 \psi$ iff $\complement X \models^1 \psi$.

$$
\begin{array}{lcl}
\mathcal{M}, X \models^1 \bot & \iff & X = \emptyset \\
\mathcal{M}, X \models^1 \top & \iff & X = X^{\mathcal{M}}_{\mathrm{Dom}(X)} \\
\mathcal{M}, X \models^1 Rt_1 \ldots t_n & \iff & X = \big\{ s \in X^{\mathcal{M}}_{\mathrm{Dom}(X)} : \mathcal{M}, s \models^1 Rt_1 \ldots t_n \big\} \\
\mathcal{M}, X \models^1 (t_1 \ldots t_n) \rightsquigarrow u & \iff & \text{there is } f \text{ s.t. } X = \big\{ s \in X^{\mathcal{M}}_{\mathrm{Dom}(X)} : \\
 & & s(u) = f\big(s(t_1), \ldots, s(t_n)\big) \big\} \\
\mathcal{M}, X \models^1 \neg\phi & \iff & \mathcal{M}, \complement X \models^1 \phi \\
\mathcal{M}, X \models^1 \phi \vee \psi & \iff & \text{there is } Y, Z \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)} \text{ s.t. } Y \cup Z = X \text{ and} \\
 & & \mathcal{M}, Y \models^1 \phi \text{ and } \mathcal{M}, Z \models^1 \psi \\
\mathcal{M}, X \models^1 \phi \wedge \psi & \iff & \text{there is } Y, Z \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)} \text{ s.t. } Y \cap Z = X \text{ and} \\
 & & \mathcal{M}, Y \models^1 \phi \text{ and } \mathcal{M}, Z \models^1 \psi \\
\mathcal{M}, X \models^1 \exists x_n \phi & \iff & \text{there is } Y \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X) \cup \{x_n\}} \text{ s.t.} \\
 & & \mathsf{C}_{x_n} X = \mathsf{C}_{x_n} Y \text{ and } \mathcal{M}, Y \models^1 \phi \\
\mathcal{M}, X \models^1 \forall x_n \phi & \iff & \text{there is } Y \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X) \cup \{x_n\}} \text{ s.t.} \\
 & & \mathsf{C}_{x_n} \complement X = \mathsf{C}_{x_n} \complement Y \text{ and } \mathcal{M}, Y \models^1 \phi
\end{array}
$$

Figure 7.2: 1-semantics in teams

Case $\psi_1 \vee \psi_2$**.** $X \models^1 \phi$ iff $\overline{X} \models^1 \phi$ iff there are ω-teams Y_1 and Y_2 such that $\overline{X} = Y_1 \cup Y_2$ and $Y_i \models^1 \psi_1$ for both $i \in \{1,2\}$. By Theorem 7.1.2 and the fact that $\mathrm{FV}(\psi_i) \subseteq \mathrm{FV}(\phi)$ for both $i \in \{1,2\}$, given such Y_i, there are teams Z_1 and Z_2 such that $\mathrm{Dom}(Z_i) = \mathrm{Dom}(X)$ and $\overline{Z_i} = Y_i$ for both $i \in \{1,2\}$. Now $X \models^1 \phi$ iff there is $Z_1, Z_2 \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)}$ such that $\overline{X} = \overline{Z_1} \cup \overline{Z_2}$ and $\overline{Z_i} \models^1 \psi_i$ for both $i \in \{1,2\}$. From $\mathrm{Dom}(Z_i) = \mathrm{Dom}(X)$ and $\overline{X} = \overline{Z_1} \cup \overline{Z_2} = \overline{Z_1 \cup Z_2}$, we get $X = Z_1 \cup Z_2$. Thus $X \models^1 \phi$ iff there is $Z_1, Z_2 \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)}$ such that $X = Z_1 \cup Z_2$ and $Z_i \models^1 \psi_i$ for both $i \in \{1,2\}$.

Case $\psi_1 \wedge \psi_2$**.** $X \models^1 \phi$ iff $\overline{X} \models^1 \phi$ iff there are ω-teams Y_1 and Y_2 such that $\overline{X} = Y_1 \cap Y_2$ and $Y_i \models^1 \psi_1$ for both $i \in \{1,2\}$. By Theorem 7.1.2 and the fact that $\mathrm{FV}(\psi_i) \subseteq \mathrm{FV}(\phi)$ for both $i \in \{1,2\}$, given such Y_i, there are teams Z_1 and Z_2 such that $\mathrm{Dom}(Z_i) = \mathrm{Dom}(X)$ and $\overline{Z_i} = Y_i$ for both $i \in \{1,2\}$. Now $X \models^1 \phi$ iff there is $Z_1, Z_2 \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)}$ such that $\overline{X} = \overline{Z_1} \cap \overline{Z_2}$ and $\overline{Z_i} \models^1 \psi_i$ for both $i \in \{1,2\}$. From $\mathrm{Dom}(Z_i) = \mathrm{Dom}(X)$ and $\overline{X} = \overline{Z_1} \cap \overline{Z_2} = \overline{Z_1 \cap Z_2}$, we get $X = Z_1 \cap Z_2$. Thus $X \models^1 \phi$ iff there is $Z_1, Z_2 \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)}$ such that $X = Z_1 \cap Z_2$ and $Z_i \models^1 \psi_i$ for both $i \in \{1,2\}$.

Case $\exists x_n \psi$**.** $X \models^1 \phi$ iff $\overline{X} \models^1 \phi$ iff there is ω-team Y such that $\overline{X} = \mathsf{C}_{x_n} Y$ and $Y \models^1 \psi$. By Theorem 7.1.2 and the fact that $\mathrm{FV}(\psi) \subseteq \mathrm{FV}(\phi) \cup \{x_n\}$, given such Y there is team Z such that $\mathrm{Dom}(Z) = \mathrm{Dom}(X) \cup \{x_n\}$ and $\overline{Z} = Y$. Now $X \models^1 \phi$ iff there is $Z \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)\cup\{x_n\}}$ such that $\overline{X} = \mathsf{C}_{x_n}\overline{Z}$ and $\overline{Z} \models^1 \psi$. From $\overline{X} = \mathsf{C}_{x_n}\overline{Z}$, we get $\overline{\mathsf{C}_{x_n} X} = \overline{X} = \mathsf{C}_{x_n}\overline{Z} = \overline{\mathsf{C}_{x_n} Z}$, and together with $\mathrm{Dom}(\mathsf{C}_{x_n} Z) = \mathrm{Dom}(Z) = \mathrm{Dom}(X) \cup \{x_n\} = \mathrm{Dom}(\mathsf{C}_{x_n} X)$ it gives $\mathsf{C}_{x_n} X = \mathsf{C}_{x_n} Z$. Thus $X \models^1 \phi$ iff there is $Z \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)\cup\{x_n\}}$ such that $\mathsf{C}_{x_n} X = \mathsf{C}_{x_n} Z$ and $Z \models^1 \psi$.

Case $\forall x_n \psi$**.** $X \models^1 \phi$ iff $\overline{X} \models^1 \phi$ iff there is ω-team Y such that $\complement\overline{X} = \mathsf{C}_{x_n}\complement Y$ and $Y \models^1 \psi$. By Theorem 7.1.2 and the fact that $\mathrm{FV}(\psi) \subseteq \mathrm{FV}(\phi) \cup \{x_n\}$, given such Y there is team Z such that $\mathrm{Dom}(Z) = \mathrm{Dom}(X) \cup \{x_n\}$ and $\overline{Z} = Y$. Now $X \models^1 \phi$ iff there is $Z \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)\cup\{x_n\}}$ such that $\complement\overline{X} = \mathsf{C}_{x_n}\complement\overline{Z}$ and $\overline{Z} \models^1 \psi$. From $x_n \notin \mathrm{FV}(\neg\phi)$ and Theorem 7.1.2 we get $\mathsf{C}_{x_n}\complement X = \complement X$, which together with $\complement\overline{X} = \mathsf{C}_{x_n}\complement\overline{Z}$ gives $\overline{\mathsf{C}_{x_n}\complement X} = \overline{\complement X} = \complement\overline{X} = \mathsf{C}_{x_n}\complement\overline{Z} = \overline{\mathsf{C}_{x_n}\complement Z}$, and together with $\mathrm{Dom}(\mathsf{C}_{x_n}\complement Z) = \mathrm{Dom}(Z) \cup \{x_n\} = \mathrm{Dom}(X) \cup \{x_n\} = \mathrm{Dom}(\mathsf{C}_{x_n}\complement X)$ we get $\mathsf{C}_{x_n}\complement X = \mathsf{C}_{x_n}\complement Z$. Thus $X \models^1 \phi$ iff there is $Z \subseteq X^{\mathcal{M}}_{\mathrm{Dom}(X)\cup\{x_n\}}$ such that $\mathsf{C}_{x_n}\complement X = \mathsf{C}_{x_n}\complement Z$ and $Z \models^1 \psi$. $\square$

Theorem 7.1.8 (Extra variables do not matter)**.** *For all teams X and Y and formulas $\phi \in \mathrm{FOD}$, if $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(X)$ and $\mathrm{FV}(\phi) \subseteq \mathrm{Dom}(Y)$ and $X \oint \mathrm{FV}(\phi) = Y \oint \mathrm{FV}(\phi)$, then $X \models^1 \phi \iff Y \models^1 \phi$.*

Proof. From $X \oint \mathrm{FV}(\phi) = Y \oint \mathrm{FV}(\phi)$ we get $\overline{X} = \overline{Y}$. Thus $X \models^1 \phi$ iff $\overline{X} \models^1 \phi$ iff $\overline{Y} \models^1 \phi$ iff $Y \models^1 \phi$. □

As a corollary of the previous theorem, if $X \models^1 \phi$, then also $\overline{X} \models^1 \phi$. Combined with Theorem 7.1.2, this makes finite-domain teams and ω-teams interchangeable in satisfaction of formulas in 1-semantics. The operations that link the interchangeable teams and ω-teams are ω-closure and cylindric restriction (to a set of variables that contains the free variables of the formula in question).

7.2 Basic Properties

The two important properties of P-semantics are that formula interpretations are closed downwards and the empty team satisfies all formulas (Theorem 2.4.4). Both properties fail in 1-semantics as can be seen by inspecting any satisfiable first order formula in 1-semantics. However, every formula is still satisfied by some team, although it may not be the empty team.

Theorem 7.2.1. *For all $\phi \in \mathrm{FOD}$ and all models $\mathcal{M}$, there is a team X such that $\mathcal{M}, X \models^1 \phi$.*

Proof. Induction on ϕ. For first order atomic formulas, choose $X = [\![\phi]\!]^{\mathrm{FO}}$. For atomic D-formulas $(t_1 \ldots t_n) \rightsquigarrow u$, choose any n-ary function f and $X = \{s : s(u) = f(s(t_1), \ldots, s(t_n))\}$. For the remaining cases, let Y_1 and Y_2 be teams that satisfy subformulas ψ_1 and ψ_2, respectively, obtained by the induction hypothesis. For $\neg\psi_1$, choose $X = \complement Y_1$. For $\psi_1 \vee \psi_2$, choose $X = Y_1 \cup Y_2$. For $\psi_1 \wedge \psi_2$, choose $X = Y_1 \cap Y_2$. For $\exists x_n \psi_1$, choose $X = \mathsf{C}_{x_n} Y_1$. For $\forall x_n \psi_1$, choose $X = \complement \mathsf{C}_{x_n} \complement Y_1$. □

7.2.1 Similarity of P-Semantics and 1-Semantics

1-semantics and P-semantics relate with respect to formula interpretations. The following theorem shows that 1-semantics holds enough information to determine P-semantics. We will later show a simple example where the converse is not true. Hence, 1-semantics gives more information of a FOD formula than P-semantics.

Theorem 7.2.2. *For all* $\phi \in \mathrm{FOD}$ *in strict negation normal form and all teams* X *it holds that* $X \models^{\mathrm{P}} \phi$ *if and only if there is* $Y \supseteq X$ *such that* $Y \models^{1} \phi$.

Proof. The claim is clear if ϕ is first order (i.e. it does not contain occurrences of D-formulas) because then $[\![\phi]\!]^{\mathrm{P}} = \mathcal{P}[\![\phi]\!]^{\mathrm{FO}}$ and $[\![\phi]\!]^{1} = \{[\![\phi]\!]^{\mathrm{FO}}\}$. The general case we prove by induction on ϕ for an arbitrary team X.

Cases $\bot$**,** $\top$**,** $Rt_1 \dots t_n$ **and** $\neg Rt_1 \dots t_n$**.** The claim is clear, see above.

Case $(t_1 \dots t_n) \rightsquigarrow u$**.** By definition, $X \models^{\mathrm{P}} \phi$ is equivalent to there being a function f such that for all $s \in X$: $s(u) = f\big(s(t_1), \dots, s(t_n)\big)$, and $X \models^{1} \phi$ is equivalent to there being a function f such that $X = \big\{s \in X_{\mathrm{FV}(\phi)} : s(u) = f\big(s(t_1), \dots, s(t_n)\big)\big\}$. We see that the claim holds.

Case $\phi_1 \wedge \phi_2$**.** If $X \models^{\mathrm{P}} \phi$ then $X \models^{\mathrm{P}} \phi_1$ and $X \models^{\mathrm{P}} \phi_2$. The induction hypothesis gives some $Y_1 \supseteq X$ and $Y_2 \supseteq X$ such that $Y_1 \models^{1} \phi_1$ and $Y_2 \models^{1} \phi_2$. Now $X \subseteq Y_1 \cap Y_2$ and $Y_1 \cap Y_2 \models^{1} \phi$.

If $Y \models^{1} \phi$ for some $Y \supseteq X$, then there are Y_1 and Y_2 such that $Y = Y_1 \cap Y_2$, $Y_1 \models^{1} \phi_1$ and $Y_2 \models^{1} \phi_2$. Note that $X \subseteq Y_1$ and $X \subseteq Y_2$. Thus the induction hypothesis gives $X \models^{\mathrm{P}} \phi_1$ and $X \models^{\mathrm{P}} \phi_2$, whence $X \models^{\mathrm{P}} \phi$.

Case $\phi_1 \vee \phi_2$**.** If $X \models^{\mathrm{P}} \phi$ then there are teams X_1 and X_2 such that $X = X_1 \cup X_2$, $X_1 \models^{\mathrm{P}} \phi_1$ and $X_2 \models^{\mathrm{P}} \phi_2$. The induction hypothesis gives some $Y_1 \supseteq X_1$ and $Y_2 \supseteq X_2$ such that $Y_1 \models^{1} \phi_1$ and $Y_2 \models^{1} \phi_2$. Now $X \subseteq Y_1 \cup Y_2$ and $Y_1 \cup Y_2 \models^{1} \phi$.

If $Y \models^{1} \phi$ for some $Y \supseteq X$, then there are Y_1 and Y_2 such that $Y = Y_1 \cup Y_2$, $Y_1 \models^{1} \phi_1$ and $Y_2 \models^{1} \phi_2$. By choosing $X_1 = X \cap Y_1$ and $X_2 = X \cap Y_2$ we get $X_1 \subseteq Y_1$, $X_2 \subseteq Y_2$ and $X = X_1 \cup X_2$. By the induction hypothesis, $X_1 \models^{\mathrm{P}} \phi_1$ and $X_2 \models^{\mathrm{P}} \phi_2$, whence $X \models^{\mathrm{P}} \phi$.

Case $\exists x_n \psi$**.** If $X \models^{\mathrm{P}} \phi$ then there is a function F such that $X(x_n \mapsto F) \models^{\mathrm{P}} \psi$. The induction hypothesis gives some $Y \supseteq X(x_n \mapsto F)$ such that $Y \models^{1} \psi$. Now $Y' \models^{1} \phi$, where $Y' := (\mathsf{C}_{x_n} Y)\upharpoonright\mathrm{Dom}(X)$. Because $Y' = Y\upharpoonright\mathrm{Dom}(X)$, also $Y' \supseteq X$ holds.

If $Y \models^{1} \phi$ for some $Y \supseteq X$, then there is $Z \subseteq X_{\mathrm{Dom}(X) \cup \{x_n\}}$ such that $\mathsf{C}_{x_n} Y = \mathsf{C}_{x_n} Z$ and $Z \models^{1} \psi$. Assuming the axiom of choice, there is a function $F \colon X \to M$ such that $F(s) \in \{a \in M : s(x_n \mapsto a) \in Z\}$ for all $s \in X$. Because $X(x_n \mapsto F) \subseteq Z$, the induction hypothesis gives $X(x \mapsto F) \models^{\mathrm{P}} \psi$. Thus $X \models^{\mathrm{P}} \phi$.

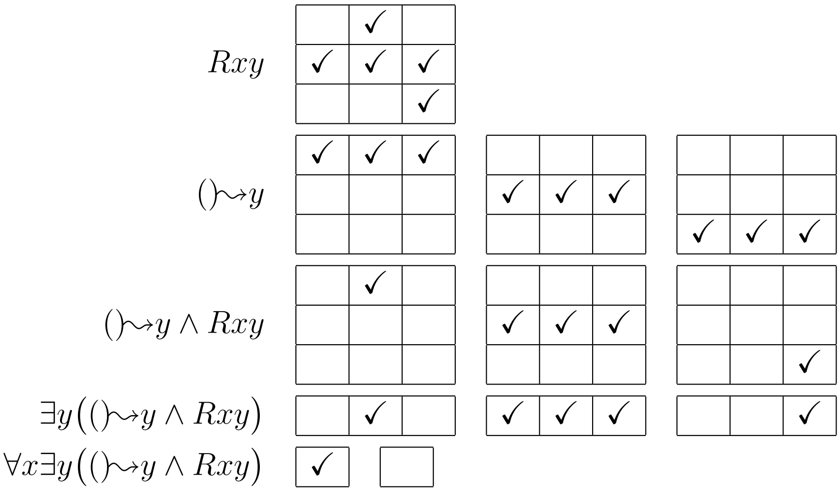

Figure 7.3: 1-interpretations of certain formulas

Case $\forall x_n \psi$**.** If $X \models^{\mathrm{P}} \phi$, then $X(x \mapsto M) \models^{\mathrm{P}} \psi$. By the induction hypothesis, there is some $Y \supseteq X(x \mapsto M)$ such that $Y \models^1 \psi$. Now $Y' \supseteq X$ and $Y' \models^1 \phi$, where $Y' := (\complement \mathsf{C}_{x_n} \complement Y) \restriction \mathrm{Dom}(X)$.

If $Y \models^1 \phi$ for some $Y \supseteq X$, then there is $Z \subseteq X_{\mathrm{Dom}(X) \cup \{x_n\}}$ such that $\mathsf{C}_{x_n} \complement Z = \mathsf{C}_{x_n} \complement Y$ and $Z \models^1 \psi$. Because $X(x_n \mapsto M) \subseteq Z$, the induction hypothesis gives $X(x \mapsto M) \models^{\mathrm{P}} \psi$, whence $X \models^{\mathrm{P}} \phi$. □

The above theorem shows that if we know $[\![\phi]\!]^1$ for any $\phi \in \mathrm{FOD}$, we can compute also $[\![\phi]\!]^{\mathrm{P}}$. The converse does not hold as the following example shows.

Example 7.2.3. Consider the sentence $\phi := \forall x \exists y \big(() \rightsquigarrow y \wedge Rxy\big)$ in the three-element model $\mathcal{M} = (\{1, 2, 3\}, R^{\mathcal{M}})$, with the relation symbol interpreted as $R^{\mathcal{M}} = \{(2, 1), (1, 2), (2, 2), (3, 2), (3, 3)\}$. Despite the facts that this simple formula holds little interest in what it defines and that it has an equivalent first order formula,[6] it is useful in illustrating the mechanics of 1-semantics. Figure 7.3 shows the 1-interpretations of all subformulas in ϕ, where each team in the interpretation is represented as a table where a checkmark at the b'th row and a'th column means that the assignment $\{x \mapsto a, y \mapsto b\}$ is in the team.

[6]Sentence ϕ is equivalent to the first order sentence $\exists y \forall x Rxy$, regardless of whether we interpret ϕ in 1-semantics or P-semantics.

The P-interpretations of the subformulas of ϕ are the downward closures of the corresponding 1-interpretations. Thus, $[\![Rxy]\!]^{\mathrm{P}}_{\mathcal{M}}$ consists of 2^5 teams, $[\![()\rightsquigarrow y]\!]^{\mathrm{P}}_{\mathcal{M}}$ consists of $3 \cdot 2^3 - 2$ teams, and so on. 1-interpretation and P-interpretation agree on ϕ but not on any of its proper subformulas.

Consider the subformula $\phi' := \exists y\big(()\rightsquigarrow y \wedge Rxy\big)$. Its P-interpretation is $\mathcal{P}X^{\mathcal{M}}_{\{x,y\}}$. Now, if we change the model $\mathcal{M}$ by interpreting R as the full relation, the 1-interpretation of ϕ' changes while its P-interpretation remains the same. Thus, although P-interpretation is straightforwardly computable as the downward closure of 1-interpretation, it is not possible to compute the 1-interpretation of a formula in a model if we are given only the P-interpretation of the formula in the model. $\square$

Another interesting thing to note about the previous example is that it is also an example of a sentence ϕ and model $\mathcal{M}$ such that $\mathcal{M} \models^1 \phi$ and $\mathcal{M} \models^1 \neg\phi$. This kind of phenomenon has been studied by the name of *paraconsistent logic*. In paraconsistent logic, one abandons some classical inference rules in order to be able to deduce formulas from a contradiction of the form $\phi \wedge \neg\phi$ while avoiding the principle of explosion, i.e. the ability to deduce every formula. Indeed, 1-semantics rejects the rule of *modus tollendo ponens*, i.e. that from $\mathcal{M}, X \models^1 \phi \vee \psi$ and $\mathcal{M}, X \models^1 \neg\phi$ one can deduce $\mathcal{M}, X \models^1 \psi$.

An explanation to the paraconsistency of 1-semantics can be seen later in Theorem 7.2.6. The theorem shows that 1-semantics of FOD reflects existential second order logic. A formula $\phi \in$ FOD corresponds to some formula $\psi \in$ ESO with free element and function variables. If $X \models^1 \phi$, it corresponds to saying that there are some values for the free function variables in ψ such that X contains all the values for the element variables that satisfy ψ together with the chosen functions. In this case, X also contains the values of those functions, evaluated on the chosen values for the element variables.

This is to say that negation in 1-semantics works on the "first order level". Negation in 1-semantics is not the classical negation of second order logic; it is the classical negation of first order logic. This fact is tied to the way we translate formulas between these logics, in particular, the semantic translation and how it encodes first order and second order objects into teams and back.

7.2.2 The Law of Excluded Middle

In first order logic, the law of excluded middle holds. The law of excluded middle is the property of the first order truth definition $\models$ that $\mathcal{M} \models \forall\overline{x}(\phi(\overline{x}) \vee \neg\phi(\overline{x}))$ for all first order formulas ϕ and models $\mathcal{M}$. It is generally known that the law of excluded middle fails for sentences of independence friendly logic. Therefore it also fails for FOD sentences interpreted in P-semantics. Because the law fails already for sentences it has not been studied how it could be adapted more generally to all FOD formulas.

There are two ways to adapt the law of excluded middle to FOD formulas. One is the obvious analogy of the first order statement, obtained by the same formula construction for a given FOD formula ϕ.

Definition 7.2.4 (LEM1)**.** A truth definition $\models$ of FOD formulas has the property LEM1 if for all $\phi \in$ FOD and models $\mathcal{M}$ it holds that $\mathcal{M} \models \forall\overline{x}\big(\phi(\overline{x}) \vee \neg\phi(\overline{x})\big)$.

Another formulation is obtained by rephrasing the first order statement into the form "for all assignments s it holds that $\mathcal{M}, s \models \phi \vee \neg\phi$".

Definition 7.2.5 (LEM2)**.** A truth definition $\models$ of FOD formulas has the property LEM2 if for all $\phi \in$ FOD, all teams X and all models $\mathcal{M}$ it holds that $\mathcal{M}, X \models \phi(\overline{x}) \vee \neg\phi(\overline{x})$.

Clearly LEM2 implies LEM1. Both of them are natural generalisations of the law of excluded middle for first order logic.

Not surprisingly, both LEM1 and LEM2 fail for $\models^{\mathrm{P}}$. This follows simply from the fact that for $\phi := \forall x \exists y(() \rightsquigarrow y \wedge x = y)$ we have both $\mathcal{M} \not\models^{\mathrm{P}} \phi$ and $\mathcal{M} \not\models^{\mathrm{P}} \neg\phi$ for all models $\mathcal{M}$ with at least two elements.[7] Also note that with Theorem 2.4.6 one can construct a plethora of sentences that manifest the failure of LEM1.

LEM1 holds for $\models^1$. LEM1 applied to a FOD formula ϕ is equivalent to $\mathcal{M}, X_{\mathrm{FV}(\phi)} \models^1 \phi \vee \neg\phi$. This holds if and only if we can split $X_{\mathrm{FV}(\phi)} = Y \cup Z$ such that $\mathcal{M}, Y \models^1 \phi$ and $\mathcal{M}, \complement Z \models^1 \phi$. This can be done if and only if there is some team Y such that $\mathcal{M}, Y \models^1 \phi$; just choose $Z = \complement Y$. By Theorem 7.2.1, this is always the case.

LEM2 fails also for $\models^1$. Consider any first order formula ϕ. It is satisfied in 1-semantics only by one team, $[\![\phi]\!]^{\mathrm{FO}}$, and thus $\phi \vee \neg\phi$ is satisfied in 1-semantics only by the full team. In a way, this makes sense. The full team represents "all first order semantic objects" whereas the empty team represents "no first order semantic objects". It is unreasonable to demand that a first order formula would hold for all and no semantic objects at the same time.

[7] Note that $\neg\phi$ in the context of P-semantics denotes the result of a syntactic manipulation of ϕ, as stated in Definition 2.4.2.

7.2.3 Expressive Power

It is easy to see that 1-semantics of FOD can be expressed in existential second order logic. This gives a translation of 1-semantics to ESO. I also give a translation in the converse direction. Note that the translation works for all formulas in ESO as opposed to Theorem 5.1.2 for P-semantics where we must restrict to downward closed formulas in ESO.

Theorem 7.2.6. *Let $\phi \in$ ESO with free variables $x_1, \dots, x_m$ and free function variables $f_1, \dots, f_n$ such that each f_i appears only in occurrences of the term $t_i := f_i u^i_1, \dots u^i_{k(i)}$, where each u^i_j is a variable. Then there is $\psi \in$ FOD with free variables $x_1, \dots, x_m$ such that for all models $\mathcal{M}$ and teams X: $\mathcal{M}, X \models^1 \psi$ if and only if there are functions $g_1, \dots, g_n$ such that*

$$X = \{s : \mathcal{M}, s(f_i \mapsto g_i)_{i \le n} \models \phi\}. \tag{7.1}$$

Proof. Without loss of generality, we may assume that $\phi \in$ ESO is in Skolem normal form,

$$\phi := \exists f_{n+1} \dots \exists f_{n'} \forall x_{m+1} \dots \forall x_{m'} \theta,$$

where θ is a quantifier-free formula in variables $f_1, \dots, f_{n'}$ and $x_1, \dots, x_{m'}$ for some $n' \ge n$ and $m' \ge m$. Furthermore, we can assume that each f_i appears only in occurrences of the term $t_i := f_i u^i_1 \dots u^i_{k(i)}$, where each u^i_j is a variable.

Let $\psi \in$ FOD be

$$\psi := \forall x_{m+1} \dots \forall x_{m'} \exists y_1 \dots \exists y_{n'} \bigwedge_{i \le n} \alpha_i \wedge \bigwedge_{n < i \le n'} \beta_i \wedge \theta(t_i \mapsto y_i)_{i \le n'}$$

$$\alpha_i := (u^i_1 \dots u^i_{k(i)}) \rightsquigarrow y_i$$

$$\beta_i := (u^i_1 \dots u^i_{k(i)} x_1 \dots x_m) \rightsquigarrow y_i.$$

Assume $X \models^1 \psi$ and show that there are functions $g_1, \dots, g_n$ such that (7.1) holds. We may assume without loss of generality that $x_i, y_j \in \mathrm{Dom}(X)$ for all $i \le m'$ and $j \le n'$. For $m < i \le m'$, there are $X_i \subseteq X_{\mathrm{Dom}(X)}$ such that $\mathsf{C}_{x_i} \complement X_i = \mathsf{C}_{x_i} \complement X_{i-1}$, where we denote $X_m := X$, and for $i \le n'$ there are $Y_i \subseteq X_{\mathrm{Dom}(X)}$ such that $\mathsf{C}_{y_i} Y_i = \mathsf{C}_{y_i} Y_{i-1}$, where we denote $Y_0 := X_{m'}$, and there are functions g_i for $i \le n'$ such that

$$\begin{aligned} Y_n = Z &\cap \{s : s(y_i) = g_i(s(u^i_1), \dots, s(u^i_{k(i)})) \text{ for all } i \le n\} \\ &\cap \{s : s(y_i) = g_i(s(u^i_1), \dots, s(u^i_{k(i)}), s(x_1), \dots, s(x_m)) \text{ for all } n < i \le n'\}, \end{aligned} \tag{7.2}$$

where $Z \models^1 \theta(t_i \mapsto y_i)_{i \le n'}$, i.e. $Z = [\![\theta(t_i \mapsto y_i)_{i \le n'}]\!]^{\mathrm{FO}}$. In particular, each g_i for $i \le n$ is $k(i)$-ary, and each g_i for $n < i \le n'$ is $(k(i) + m)$-ary. We also have $X = \complement \mathsf{C}_{x_{m+1}} \dots \mathsf{C}_{x_{m'}} \complement \mathsf{C}_{y_1} \dots \mathsf{C}_{y_{n'}} Y_{n'}$.

Now we have $g_1, \dots, g_n$ and it is left to show that they satisfy (7.1). Let $s \in X$. Note that by varying s we can get arbitrary values for variables $x_{m+1}, \dots, x_{m'}$ while keeping the values of other variables fixed. We have $s' \in Y_{n'}$, where

$$s' := s\big(y_i \mapsto g_i\big(s(u_1^i), \dots, s(u_{k(i)}^i)\big)\big)_{i \le n} \\ \big(y_i \mapsto g_i\big(s(u_1^i), \dots, s(u_{k(i)}^i), s(x_1), \dots, s(x_m)\big)\big)_{n < i \le n'}.$$

Denote

$$s'' := s(f_i \mapsto g_i)_{i \le n}(f_i \mapsto h_i^s)_{n < i \le n'},$$

where h_i^s for $n < i \le n'$ are functions that map

$$h_i^s(b_1, \dots, b_{k(i)}) = g_i\big(b_1, \dots, b_{k(i)}, s(x_1), \dots, s(x_m)\big) \tag{7.3}$$

for all $b_1, \dots, b_{k(i)} \in M$. Then we get $s'(y_i) = s''(t_i)$ for all $i \le n'$. Now from $s' \models \theta(t_i \mapsto y_i)_{i \le n'}$ we get $s'' \models \theta$. From this we get that

$$s(f_i \mapsto g_i)_{i \le n} \models \phi \tag{7.4}$$

for all $s \in X$, concluding the proof of the inclusion to the right in (7.1).

Assume that assignment s with $\mathrm{Dom}(s) = \mathrm{Dom}(X)$ satisfies (7.4). Then there are functions $h_{n+1}^s, \dots, h_{n'}^s$ such that for all $a_{m+1}, \dots, a_{m'} \in M$ we have $s'' \models \theta$, where

$$s'' := s(x_i \mapsto a_i)_{m < i \le m'}(f_i \mapsto g_i)_{i \le n}(f_i \mapsto h_i^s)_{n < i \le n'}.$$

Note that at this point we still remember from (7.2) the functions $g_1, \dots, g_n$. Instead, the functions $g_{n+1}, \dots, g_{n'}$ that (7.2) gave are useless and therefore, for $n < i \le n'$, we use h_i^s to redefine g_i as the $(k(i)+m)$-ary function that maps elements as in (7.3) for all s that satisfy (7.4) and is defined arbitrarily elsewhere. Define, for all $i \le n'$,

$$s' := s_1\big(y_i \mapsto g_i\big(s_1(u_1^i), \dots, s_1(u_{k(i)}^i)\big)\big)_{i \le n} \\ \big(y_i \mapsto g_i\big(s_1(u_1^i), \dots, s_1(u_{k(i)}^i), s_1(x_1), \dots, s_1(x_m)\big)\big)_{n < i \le n'},$$

where $s_1 := s(x_i \mapsto a_i)_{m < i \le m'}$. Then $s'(y_i) = s''(t_i)$ for all $i \le n'$, whence $s' \models \theta(t_i \mapsto y_i)_{i \le n'}$. Now $s' \in Y_{n'}$, where $Y_{n'}$ is as in (7.2), whence $s \in X$. This concludes the proof that the functions g_i satisfy (7.1).

For the other direction, let $g_1, \dots, g_n$ be given and show that $X \models^1 \psi$, where X is as in (7.1). We may assume without loss of generality that

$x_i, y_j \in \mathrm{Dom}(X)$ for all $i \leq m'$ and $j \leq n'$. For each $s \in X$ there are functions $h^s_{n+1}, \dots, h^s_{n'}$ such that $s'' \models \forall x_1 \dots \forall x_m \theta$, where

$$s'' := s(f_i \mapsto g_i)_{i \leq n}(f_i \mapsto h^s_i)_{n<i\leq n'}.$$

For each $n < i \leq n'$, let g_i be the $(k(i)+m)$-ary function that maps as in (7.3) for all $s \in X$ and is defined arbitrarily on other values. Define

$$\begin{aligned} s' := s\big(y_i \mapsto g_i\big(s(u^i_1), \dots, s(u^i_{k(i)})\big)\big)_{i\leq n} \\ \big(y_i \mapsto g_i\big(s(u^i_1), \dots, s(u^i_{k(i)}), s(x_1), \dots, s(x_m)\big)\big)_{n<i\leq n'}. \end{aligned}$$

To show that $X \models^1 \psi$, it suffices to show that $Y = Y_{n'}$, where $Y := \{s' : s \in X\}$ and $Y_{n'}$ is defined as in (7.2).

Let $s' \in Y$. Clearly $s'(y_i) = g_i\big(s(u^i_1), \dots, s(u^i_{k(i)})\big)$ for all $i \leq n$, and $s'(y_i) = g_i\big(s(u^i_1), \dots, s(u^i_{k(i)}), s(x_1), \dots, s(x_m)\big)$ for all $n < i \leq n'$. Note that $s''(x_i \mapsto s'(x_i))_{m<i\leq m'} = s''$ because $s'(x_i) = s(x_i)$ for all $m < i \leq m'$. Thus from the fact that $s'' \models \theta$ and $s''(t_i) = s'(y_i)$ for all $i \leq n'$, we have $s' \models \theta(t_i \mapsto y_i)_{i\leq n'}$. Therefore $s' \in Y_{n'}$.

Let then $s_0 \in Y_{n'}$. Looking at the definition of $Y_{n'}$, we see that s_0 defines the values of y_i based on the functions g_i; therefore we can present $s_0 =: s'$ for some $s \in X$. Furthermore, because $s' \in Y_{n'}$, we have $s' \models \theta(t_i \mapsto y_i)_{i\leq n'}$. Again, it holds that $s'(y_i) = s''(t_i)$ for all $i \leq n'$, and therefore $s'' \models \theta$, from which we get $s(f_i \mapsto g_i)_{i\leq n} \models \phi$, and further, $s_0 \in Y$. This completes the proof. □

7.3 Game Theoretic Semantics

In this section I sketch some ideas about what impact 1-semantics has on semantic games. The goal is to give a semantic game that characterises the truth definition of 1-semantics as presented in Definition 7.1.6.

A *strategy* (for a semantic test) for a model $\mathcal{M}$ and formula $\phi \in \mathrm{FOD}$ is a partial function σ that is defined on some pairs (ψ, s) where ψ is a subformula of ϕ and s is an assignment for $\mathcal{M}$ defined on $\mathrm{FV}(\psi)$. A strategy maps these pairs to the union of the universe M and the set of subformulas of ϕ such that $\sigma(\psi_1 \vee \psi_2, s) \in \{\psi_1, \psi_2\}$ and $\sigma(\exists x \psi, s) \in M$.

Definition 7.3.1. I define the *semantic test procedure*, denoted T, as follows. There is a test supervisor called ∃loise and her job is to guide the test so that it passes, denoted $\top$. If a test does not pass, it fails, denoted $\bot$. The result of a semantic test we denote by $\mathrm{T}(\mathcal{M}, \phi, s, \sigma)$ for a model $\mathcal{M}$, formula $\phi \in \mathrm{FOD}$, assignment s that is defined on the free variables of ϕ, and strategy σ for $\mathcal{M}$

$$
\begin{aligned}
\mathrm{T}(\mathcal{M}, \bot, s, \sigma) &\;=\!=\; \bot \\
\mathrm{T}(\mathcal{M}, \top, s, \sigma) &\;=\!=\; \top \\
\mathrm{T}(\mathcal{M}, Rt_1 \dots t_n, s, \sigma) &\iff \big(s(t_1), \dots, s(t_n)\big) \in R^{\mathcal{M}} \\
\mathrm{T}(\mathcal{M}, (t_1 \dots t_n)\rightsquigarrow u, s, \sigma) &\;=\!=\; \top \text{ (see below for passing tests uniformly)} \\
\mathrm{T}(\mathcal{M}, \neg\psi, s, \sigma) &\iff \text{not } \mathrm{T}(\mathcal{M}, \psi, s, \sigma) \\
\mathrm{T}(\mathcal{M}, \psi_1 \vee \psi_2, s, \sigma) &\iff \mathrm{T}(\mathcal{M}, \psi_n, s, \sigma), \text{ where } \psi_n = \sigma(\phi, s) \\
\mathrm{T}(\mathcal{M}, \psi_1 \wedge \psi_2, s, \sigma) &\iff \mathrm{T}(\mathcal{M}, \psi_n, s, \sigma) \text{ for both } n \in \{1, 2\} \\
\mathrm{T}(\mathcal{M}, \exists x\psi, s, \sigma) &\iff \mathrm{T}(\mathcal{M}, \psi, s(x \mapsto a), \sigma), \text{ where } a = \sigma(\phi, s) \\
\mathrm{T}(\mathcal{M}, \forall x\psi, s, \sigma) &\iff \mathrm{T}(\mathcal{M}, \psi, s(x \mapsto a), \sigma) \text{ for all } a \in M
\end{aligned}
$$

Figure 7.4: Result of a semantic test, $\mathrm{T}(\mathcal{M}, \phi, s, \sigma)$

and ϕ. We may think of $\mathrm{T}(\mathcal{M}, \phi, s, \sigma)$ as a predicate that yields true if and only if the test passes. The test result is defined as in Figure 7.4, based on the form of ϕ.

It is an open question if there is a game such that $\mathrm{T}(\mathcal{M}, \phi, s, \sigma)$ is equivalent to σ being a winning strategy in the game on $\mathcal{M}$, ϕ and s.
The semantic test procedure characterises first order semantics.

Theorem 7.3.2. *For any model $\mathcal{M}$, formula $\phi \in \mathrm{FO}$ and assignment s there is σ such that $\mathcal{M}, s \models \phi$ if and only if $\mathrm{T}(\mathcal{M}, \phi, s, \sigma) = \top$.*

Proof. Induction on $\phi \in \mathrm{FOD}$.

Case $\bot$. Both $\mathcal{M}, s \models \bot$ and $\mathrm{T}(\mathcal{M}, \bot, s, \sigma) = \top$ never hold.

Case $\top$. Both $\mathcal{M}, s \models \top$ and $\mathrm{T}(\mathcal{M}, \top, s, \sigma) = \top$ always hold.

Case $Rt_1 \dots t_n$. The claim is clear as we can choose σ to be the empty function.

Case $\neg\psi$. Let σ be the same as for ψ. Then $\mathcal{M}, s \models \phi$ iff $\mathcal{M}, s \not\models \psi$ iff $\mathrm{T}(\mathcal{M}, \psi, s, \sigma) = \bot$ iff $\mathrm{T}(\mathcal{M}, \phi, s, \sigma) = \top$.

Case $\psi_1 \vee \psi_2$. Let σ map ϕ to ψ_n such that $\mathcal{M}, s \models \psi_n$, if such n exists, or to ψ_1 otherwise, and elsewhere let σ map like σ_1 and σ_2 which we get for ψ_1 and ψ_2 from the induction hypothesis. Then $\mathcal{M}, s \models \phi$ iff $\mathcal{M}, s \models \psi_n$ for some n iff $\mathrm{T}(\mathcal{M}, \psi_n, s, \sigma_n) = \top$ for some n. We can now see that $\mathcal{M}, s \models \phi$ iff $\mathrm{T}(\mathcal{M}, \phi, s, \sigma) = \top$.

Case $\psi_1 \wedge \psi_2$. Let σ map like σ_1 and σ_2 which we get for ψ_1 and ψ_2 from the induction hypothesis. Then $\mathcal{M}, s \models \phi$ iff $\mathcal{M}, s \models \psi_n$ for both $n \in \{1,2\}$ iff $\mathrm{T}(\mathcal{M}, \psi_n, s, \sigma_n) = \top$ for both $n \in \{1,2\}$ iff $\mathrm{T}(\mathcal{M}, \phi, s, \sigma) = \top$.

Case $\exists x\psi$. Let σ map ϕ to $a \in M$ such that $\mathcal{M}, s(x \mapsto a) \models \psi$, if such a exists, or to an arbitrary element otherwise, and elsewhere let σ map like σ' which we get for ψ from the induction hypothesis. Then $\mathcal{M}, s \models \phi$ iff $\mathcal{M}, s(x \mapsto a) \models \psi$ for some $a \in M$ iff $\mathrm{T}(\mathcal{M}, \psi, s(x \mapsto a), \sigma') = \top$ for some $a \in M$. We can now see that $\mathcal{M}, s \models \phi$ iff $\mathrm{T}(\mathcal{M}, \phi, s, \sigma) = \top$.

Case $\forall x\psi$. Let σ map like σ' which we get for ψ from the induction hypothesis. Then $\mathcal{M}, s \models \phi$ iff $\mathcal{M}, s(x \mapsto a) \models \psi$ for all $a \in M$ iff $\mathrm{T}(\mathcal{M}, \psi, s(x \mapsto a), \sigma') = \top$ for all $a \in M$ iff $\mathrm{T}(\mathcal{M}, \phi, s, \sigma) = \top$. □

For a model $\mathcal{M}$, formula $\phi \in \mathrm{FOD}$, team X and strategy σ, we say that $\mathrm{T}(\mathcal{M}, \phi, s, \sigma)$ *passes uniformly for* $s \in X$, if for all instances of D-formulas in ϕ, enumerated as $\psi_1, \ldots, \psi_k$, there are functions $f_1, \ldots, f_k$ such that, for all $s \in X$, $\mathrm{T}(\mathcal{M}, \phi, s, \sigma) = \top$ and if the test $\mathrm{T}(\mathcal{M}, \phi, s, \sigma)$ ends in the final test $\mathrm{T}(\mathcal{M}, \psi_i, s', \sigma)$ then $s'(u) = f_i\big(s'(t_1), \ldots, s'(t_n)\big)$, where ψ_i is $(t_1 \ldots t_n) \rightsquigarrow u$.

There is also a semantic game in the traditional sense in which we process several semantic tests at once. A similar game for P-semantics was presented already by Väänänen [19, Definition 5.5].

Definition 7.3.3. The *1-semantic game on teams*, $\partial^1(\mathcal{M}, \phi, X)$ for a model $\mathcal{M}$, formula $\phi \in \mathrm{FOD}$ and team X that is defined on the free variables of ϕ is defined as follows. There are two players, $\forall$belard and $\exists$loise, whose actions are limited by the game according to the form of ϕ as follows.

Case $\bot$. If $X = \emptyset$, $\exists$loise wins. Otherwise she loses.

Case $\top$. $\exists$loise wins unconditionally.

Case $Rt_1 \ldots t_n$. If $X = \big\{s : \big(s(t_1), \ldots, s(t_n)\big) \in R^{\mathcal{M}}\big\}$, $\exists$loise wins. Otherwise she loses.

Case $(t_1 \ldots t_n) \rightsquigarrow u$. If there is an n-ary function f such that $X = \big\{s : s(u) = f\big(s(t_1), \ldots, s(t_n)\big)\big\}$, $\exists$loise wins. Otherwise she loses.

Case $\neg\phi$. The game continues according to $\partial^1(\mathcal{M}, \phi, \complement X)$.

Case $\phi \vee \psi$. $\exists$loise chooses teams Y and Z such that $X = Y \cup Z$, and $\forall$belard chooses whether the game continues according to $\partial^1(\mathcal{M}, \phi, Y)$ or $\partial^1(\mathcal{M}, \psi, Z)$.

Case $\phi \wedge \psi$. $\exists$loise chooses teams Y and Z such that $X = Y \cap Z$, and $\forall$belard chooses whether the game continues according to $\eth^1(\mathcal{M}, \phi, Y)$ or $\eth^1(\mathcal{M}, \psi, Z)$.

Case $\exists x\phi$. $\exists$loise chooses a team Y such that $Y{\upharpoonright}\mathrm{FV}(\exists x\phi) = X{\upharpoonright}\mathrm{FV}(\exists x\phi)$, and the game continues according to $\eth^1(\mathcal{M}, \phi, Y)$.

Case $\forall x\phi$. $\exists$loise chooses a team Y such that $\complement Y{\upharpoonright}\mathrm{FV}(\exists x\phi) = \complement X{\upharpoonright}\mathrm{FV}(\exists x\phi)$, and the game continues according to $\eth^1(\mathcal{M}, \phi, Y)$.

A *game strategy* (for $\exists$loise) for a model $\mathcal{M}$ and formula $\phi \in \mathrm{FOD}$ is a partial function σ that is defined on some pairs (ψ, X) where ψ is a subformula of ϕ and X is a team for $\mathcal{M}$ defined on $\mathrm{FV}(\psi)$. A game strategy maps these pairs to teams or pairs of teams on $\mathcal{M}$ such that pairs of the forms $(\psi_1 \vee \psi_2, X)$ and $(\psi_1 \wedge \psi_2, X)$ are mapped to pairs of teams, and pairs of the forms $(\exists x\psi, X)$ and $(\forall x\psi, X)$ are mapped to teams.

The semantic test procedure and the 1-semantic game seem to be closely related.

Conjecture 7.3.4. *For any formula $\phi \in \mathrm{FOD}$, model $\mathcal{M}$, team X and game strategy σ there is a strategy σ^* such that σ is winning in $\eth^1(\mathcal{M}, \phi, X)$ if and only if X is maximal with respect to $\mathrm{T}(\mathcal{M}, \phi, s, \sigma^*)$ passing uniformly for $s \in X$.*

Theorem 7.3.5. *For all formulas ϕ, models $\mathcal{M}$ and teams X, $\exists$loise has a winning strategy in $\eth^1(\mathcal{M}, \phi, X)$ if and only if $\mathcal{M}, X \models^1 \phi$.*

Proof. We prove the claim for arbitrary $\mathcal{M}$ and X by induction on $\phi \in \mathrm{FOD}$.

Atomic cases. For $\bot$, $\top$, $Rt_1 \dots t_n$ and $(t_1 \dots t_n){\rightsquigarrow}u$ the claim clearly holds.

Case $\neg\psi$. Claim clear.

Case $\psi_1 \vee \psi_2$. If σ is winning in $\eth^1(\mathcal{M}, \phi, X)$ then σ gives Y_1 and Y_2 such that $X = Y_1 \cup Y_2$ and σ is winning in $\eth^1(\mathcal{M}, \psi_n, Y_n)$ for both $n \in \{1, 2\}$. By the induction hypothesis, $\mathcal{M}, Y_n \models^1, \psi_n$ for both $n \in \{1, 2\}$, thus $\mathcal{M}, X \models^1 \phi$.

If $\mathcal{M}, X \models^1 \phi$, then there is Y_1 and Y_2 such that $X = Y_1 \cup Y_2$ and $\mathcal{M}, Y_n \models \psi_n$ for both $n \in \{1, 2\}$. By the induction hypothesis, $\exists$loise has a winning strategy σ_n for $\eth^1(\mathcal{M}, \psi_n, Y_n)$ for both $n \in \{1, 2\}$. Let σ be the strategy that maps (ϕ, X) to (Y_1, Y_2) and that maps otherwise like σ_1 and σ_2. Then σ is winning in $\eth^1(\mathcal{M}, \phi, X)$.

Case $\psi_1 \wedge \psi_2$. Similarly to above.

Case $\exists x\psi$**.** If σ is winning in $\eth^1(\mathcal{M}, \phi, X)$ then σ gives some team Y such that $X\upharpoonright\mathrm{FV}(\phi) = Y\upharpoonright\mathrm{FV}(\phi)$ and σ is winning in $\eth^1(\mathcal{M}, \psi, Y)$. By the induction hypothesis, $\mathcal{M}, Y \models^1 \psi$, thus $\mathcal{M}, X \models \phi$.

If $\mathcal{M}, X \models \phi$, then there is Y such that $X\upharpoonright\mathrm{FV}(\phi) = Y\upharpoonright\mathrm{FV}(\phi)$ and $\mathcal{M}, Y \models^1 \psi$. By the induction hypothesis, $\exists$loise has a winning strategy σ' for $\eth^1(\mathcal{M}, \psi, Y)$. Let σ be the strategy that maps (ϕ, X) to Y and that maps otherwise like σ'. Then σ is winning in $\eth^1(\mathcal{M}, \phi, X)$.

Case $\forall x\psi$**.** Similarly to above. □

The point of presenting the semantic test procedure is to place a critical view on the concept of game theoretic semantics. The semantic test procedure is an alternative "game theoretic" truth definition for first order logic (and also for 1-semantics, given that Conjecture 7.3.4 holds) that handles negation in a different way than is done in the traditional approach, fronted by Hintikka, where the semantic game is played by two players and negation corresponds to the players swapping roles. Theorem 7.3.2 shows that the semantic test procedure is as adequate a definition for game theoretic semantics for first order logic as Hintikka's two-player game. However, as I suggest in Conjecture 7.3.4, when shifting from first order logic to dependence logic (or independence friendly logic), the semantic test procedure is the more practical choice of the two. Assuming that the conjecture holds, the semantic test procedure yields 1-semantics, whereas Hintikka's semantic game yields P-semantics.

7.4 Further Ideas

Let us consider an analogy to Theorem 2.3.2 and Theorem 2.3.3 which state that in second order logic one can equivalently restrict to function quantifiers or relation quantifiers. Whereas a D-formula expresses the existence of a function, we could have another kind of formula that expressed the existence of a relation. In a sense, we already have this kind of quantification built in 1-semantics of first order formulas; a block of n quantifiers existential state the existence of a new team that extends the old team in n variables, thus in practice expressing the existence of an n-ary relation. What 1-semantics on FOD lacks, however, is an explicit way of expressing atomic second order formulas, that is, containment of a tuple in a quantified relation. We *are* able to express containment via the detour through translation to function quantifiers and then to FOD but this is not enough for practical use of the feature.

Define *C-formula* as a formula of the form $(t_1 \dots t_n) \subset (u_1 \dots u_n)$, where all t_i and u_i are terms. We obtain *containment logic* by equipping the syntax of FO with C-formulas and we denote it FOC. C stands for containment as in "tuples defined by $(u_i)_{i \leq n}$ contain tuples defined by $(t_i)_{i \leq n}$".

First we need a helpful definition. Let X be a team and let $t_1, \dots, t_n$ be terms. Extend the previously defined operation Rel that turns a team into a relation in the following way, allowing us to specify with terms how the relation is decoded from the team:

$$\begin{aligned}&\mathrm{Rel}(X, t_1, \dots, t_n) := \\ &\qquad \{(a_1, \dots, a_n) : \text{there is } s \in X \text{ such that } s(t_i) = a_i \text{ for all } i \leq n\}.\end{aligned}$$

We can now define the 1-semantics of C-formulas on ω-teams as follows.

$$\begin{aligned}&\mathcal{M}, X \models^1 (t_1 \dots t_n) \subset (u_1 \dots u_n) \iff \\ &\qquad\qquad \mathrm{Rel}(X, t_1, \dots, t_n) \subseteq \mathrm{Rel}(X, u_1, \dots, u_n)\end{aligned}$$

C-formulas express the containment of one relation in another, where both relations are encoded into the team by sequences of terms. This is a generalisation of our previously stated goal of expressing the containment of just one tuple in a relation. In teams containing several assignments, it would be difficult to pick just one tuple. Therefore this more general expression is easier to define. In fact, there should be no need to express the containment of just one tuple in a relation. The reason is that semantics defined on teams is in any case supposed to speak about several first order assignments at once, that is, we have several simultaneous values for "first order variables" and these values are grouped into a relation of their own.

I conjecture that FOC has the same expressive power as FOD. At least it is easy to see that 1-semantics for FOC formulas is expressible in ESO. The converse is formulated as the following conjecture, mimicking Theorem 7.2.6.

Conjecture 7.4.1. *Let $\phi \in$ ESO with free variables $x_1, \dots, x_m$ and free relation variables $R_1, \dots, R_n$ such that each R_i appears only in occurrences of the atomic formula $\theta_i := R_i u_1^i, \dots u_{k(i)}^i$, where each u_j^i is a variable. Then there is $\psi \in$ FOC with free variables $x_1, \dots, x_m$ and y_j^i for all $i \leq n$ and $j \leq k(i)$ such that for all models $\mathcal{M}$ and teams X, $\mathcal{M}, X \models^1 \psi$ if and only if there are relations $S_1, \dots, S_n$ such that*

$$\mathrm{Rel}\big(X, y_1^i, \dots, y_{k(i)}^i\big) = S_i$$

for all $i \leq n$, and

$$\mathrm{Rel}(X, x_1, \dots, x_m) = \\ \big\{ \big(s(x_1), \dots, s(x_m)\big) : \mathcal{M}, s \models \phi \textit{ and } s(R_i) = S_i \textit{ for all } i \leq n \big\}.$$

Furthermore, FOC might have an advantage over FOD by being more suitable for finding proof systems that capture entailment in fragments of FOC. Recall from Chapter 6 the construction of a fragment of FOD by taking a fragment of classical propositional logic and replacing propositions P_k by fixed formulas $\theta_k \in \mathrm{FOD}$. Fragment F from Chapter 6 is constructed in this way. It turned out that proof systems of classical logic are unsound for the fragment F because of $\theta_k \vee \theta_k \not\Rightarrow \theta_k$. In its simplest form, this negative entailment is of the form $(t_1 \dots t_n)\rightsquigarrow y \vee (t_1 \dots t_n)\rightsquigarrow y \not\Rightarrow (t_1 \dots t_n)\rightsquigarrow y$. Intuitively, it states that even if we know that a team can be split in two parts such that in each part the value of y is the value of a function on terms $t_1, \dots, t_n$, there might not be a function that computes the value of y in the whole team from the values of the terms $t_1, \dots, t_n$.

Proposition 7.4.2. $(x)\sqsubset(y) \vee (x)\sqsubset(y) \Rightarrow (x)\sqsubset(y)$

Proof. If $X \models^1 (x)\sqsubset(y) \vee (x)\sqsubset(y)$ then we can split $X = Y \cup Z$ such that $Y \models^1 (x)\sqsubset(y)$ and $Z \models^1 (x)\sqsubset(y)$. Therefore for all $s \in Y$ there is $s' \in Y$ such that $s(x) = s'(y)$ and for all $s \in Z$ there is $s' \in Z$ such that $s(x) = s'(y)$. Let $s \in X$. Then $s \in Y$ or $s \in Z$, whereby there is $s' \in Y$ or $s' \in Z$ such that $s(x) = s'(y)$. Obviously $s' \in X$. This shows $X \models^1 (x)\sqsubset(y)$. □

This little proposition suggests a notable difference between FOD and FOC when it comes to finding proof systems. FOC might, in some ways, be better suited for adapting proof systems of classical propositional logic for fragments constructed from propositional sentences by replacing propositional symbols by FOC formulas.

Chapter 8

Conclusions

In this thesis, I have investigated dependence logic from several aspects. On the practical side, I have presented some rules for quantifier swapping in dependence logic and team logic. Such rules are among the basic tools one must be familiar with in order to gain the required intuition for using the logic for practical purposes. For comparison, similar rules for the old and well established logics such as first order logic are so central that they are taught in elementary logic courses.

I have also looked into Ehrenfeucht-Fraïssé (EF) games. I have compared the EF games of first order logic and dependence logic and I have defined a third EF game that characterises a mixed case where first order formulas are measured in dependence rank. I have also provided an effective conversion between winning strategies.

These two areas of research provide basic facts about dependence logic. The facts in themselves may not be of particular interest but they form part of the basic understanding of dependence logic. More such research is needed before we can claim to understand dependence logic and before the logic can gain ground in more practical applications.

I have provided detailed proofs of several translations between dependence logic, team logic, second order logic and its existential fragment. Translations can be used in showing a relationship between the expressive powers of two logics. Translations are also useful on a more detailed level—by inspecting the form of the translated formulas, one can see how an aspect of one logic can be expressed in the other logic. For example, in the translation of second order logic to team logic, one can see how function quantifiers in second order logic are expressed in team logic by a similar "first order" quantifier and a suitable D-formula.

In this thesis I have also investigated proof theory in dependence logic, an area that is mostly untouched in literature. My work in this field is still very much in the beginning. My attempts focused on finding a complete proof system for some modest yet nontrivial fragment of dependence logic. In particular, I investigated a fragment that one could describe as being a tiny step from classical propositional first order logic towards dependence logic. Even this fragment is a "tough nut". I addressed a key problem in adapting a known proof system of classical propositional logic to become a proof system for the fragment, namely that the rule of contraction is needed but is unsound in its unrestricted form. I provided a proof system for the fragment but its completeness is yet only a conjecture.

Finally, I have investigated the foundation of dependence logic. I provided an alternative semantics for the syntax of dependence logic. I call the new semantics 1-semantics and the old semantics P-semantics because of the way they relate to first order semantics. Whereas it is always easy to come up with new semantics, 1-semantics stands out because it is derived from first order semantics by a natural type shift. This means that 1-semantics reflects an established semantics in a coherent manner. As a positive side effect, one can shift any quantifier or connective from first order logic to 1-semantics by the same shift.

In contrast, in P-semantics the meaning of each connective and quantifier is defined separately without a unifying principle. This is illustrated by the fact that one can define disjunction in P-semantics equivalently by referring to a disjoint union,

$$\begin{aligned} X \models^{\mathrm{P}} \phi \vee \psi \iff & \text{ there are disjoint } Y, Z \text{ s.t. } X = Y \cup Z \text{ and} \\ & Y \models^{\mathrm{P}} \phi \text{ and } Z \models^{\mathrm{P}} \phi, \end{aligned} \tag{8.1}$$

as well as by referring to union of potentially overlapping teams,

$$\begin{aligned} X \models^{\mathrm{P}} \phi \vee \psi \iff & \text{ there are } Y, Z \text{ s.t. } X = Y \cup Z \text{ and} \\ & Y \models^{\mathrm{P}} \phi \text{ and } Z \models^{\mathrm{P}} \phi. \end{aligned} \tag{8.2}$$

Both (8.1) and (8.2) produce the same semantics. A similar degree of freedom is in the existential quantifier; one can define it equivalently by extending a team by a multi-valued function or a single-valued function.

1-semantics is closely related to P-semantics. In terms of formula interpretations, P-semantics is the downward closure of 1-semantics. By this relationship, one may be able to transfer some results between the two semantics. Because of the downward closure, P-semantics hides some of the information that 1-semantics carries. In other words, a simple operation (namely downward closure) can turn the 1-interpretation of a formula into its P-interpretation, but there is no operation that can turn P-interpretations into corresponding 1-interpretations.

Despite the fact that the definition of 1-semantics differs only slightly from P-semantics, 1-semantics has several additional properties. Most importantly, negation is a semantic operation in 1-semantics, just like any other connective. Negation also behaves differently; the law of excluded middle, translated suitably from first order logic, holds for 1-semantics. This is related to the definition of negation as complementing the team in question. In contrast, the syntactic negation of P-semantics moves to some subset of the complement of the team in question, depending on the syntax of the formula.

I have provided a detailed translation of existential second order logic into 1-semantics. Interestingly, it seems natural to translate formulas where second order quantifiers are function quantifiers. The corresponding translation into P-semantics works naturally when second order quantifiers are relation quantifiers. We also know that only downward closed formulas of existential second order logic can be translated to P-semantics. This comes as no surprise, of course, knowing that P-semantics is the downward closure of 1-semantics. 1-semantics is free of this restriction.

Game theoretic semantics is the origin of P-semantics. First there was game theoretic semantics for first order logic as a two-player game. Hintikka made a twist in the game, providing a means to hide information from the players, resulting in independence friendly logic. Hodges then came up with P-semantics as a representation of game positions of several semantic games that are played simultaneously. 1-semantics emerges the other way; we start with the Tarskian semantics for first order logic, 1-shift it to teams, and finally add D-formulas. In this thesis I have briefly explored how to provide also a game theoretic form of 1-semantics. I conjecture that the semantic test procedure will fulfill this task.

The key difference between Hintikka's semantic game and the semantic test procedure is the handling of negation. Hintikka's negation switches the roles of the two players. This is in effect the same syntactic operation on compound formulas that I define for P-semantics. Hintikka's negation is a dramatic operation from ∃loise's perspective—her winning strategy for some formula ϕ has in general nothing to do with the game that is played on $\neg\phi$ because the negation makes ∃loise play in totally different places in the formula. In the semantic test procedure, negation works differently. It lets ∃loise keep her strategy. Negation negates the result of one test. This is of course different from negating the satisfaction of the formula which is characterised by there existing a strategy for ∃loise with what the test that the semantic test procedure specifies for the formula, model and assignment, passes.

Dependence logic and independence friendly logic are like twins; the former is the Tarskian twin and the latter is the game theoretic twin. It seems that studying one can benefit both via the strong bond they have. After all, dependence logic emerged from a reformulation of independence friendly logic. On the other hand, 1-semantics may shed light on game theoretic semantics. Therefore the study of these logics should not be left for philosophers only as seems to have been the case so far. Logicians from the fields of mathematics and philosophy have the best chance of making the most out of these new logics by working together.

Bibliography

[1] Kai Brünnler, *Deep inference and symmetry in classical proofs*, Logos Verlag, Berlin, 2004, http://www.iam.unibe.ch/~kai/Papers/phd.pdf.

[2] Kai Brünnler and Alwen Fernanto Tiu, *A local system for classical logic*, LPAR 2001 (R. Nieuwenhuis and A. Voronkov, eds.), Lecture Notes in Artificial Intelligence, vol. 2250, Springer-Verlag, 2001, http://www.iam.unibe.ch/~kai/Papers/lcl-lpar.pdf, pp. 347–361.

[3] John P. Burgess, *A remark on Henkin sentences and their contraries*, Notre Dame J. Formal Logic **44** (2003), no. 3, 185–188 (electronic) (2004).

[4] Xavier Caicedo, Francien Dechesne, and Theo M.V. Janssen, *Equivalence and quantifier rules for logic with imperfect information*, Logic Jnl IGPL **17** (2009), no. 1, 91–129, http://dx.doi.org/10.1093/jigpal/jzn030.

[5] Peter Cameron and Wilfrid Hodges, *Some combinatorics of imperfect information*, J. Symbolic Logic **66** (2001), no. 2, 673–684. MR MR1833470 (2002e:03046)

[6] Herbert B. Enderton, *Finite partially-ordered quantifiers*, Z. Math. Logik Grundlagen Math. **16** (1970), 393–397.

[7] Alessio Guglielmi, *Deep inference and the calculus of structures*, Web site at http://alessio.guglielmi.name/res/cos.

[8] Alessio Guglielmi and Tom Gundersen, *Normalisation control in deep inference via atomic flows*, Submitted. http://cs.bath.ac.uk/ag/p/NormContrDIAtFl.pdf, 2007.

[9] David Harel, *Characterizing second-order logic with first-order quantifiers*, Z. Math. Logik Grundlag. Math. **25** (1979), no. 5, 419–422.

[10] L. Henkin, *Some remarks on infinitely long formulas*, Infinitistic Methods (Proc. Sympos. Foundations of Math., Warsaw, 1959), Pergamon, Oxford, 1961, pp. 167–183.

[11] Leon Henkin, J. Donald Monk, and Alfred Tarski, *Cylindric algebras, Part I*, North-Holland Publishing Co., Amsterdam, 1971, Studies in Logic and the Foundations of Mathematics, Vol. 64.

[12] Jaakko Hintikka, *The principles of mathematics revisited*, Cambridge University Press, 1996.

[13] Jaakko Hintikka and Gabriel Sandu, *Informational independence as a semantical phenomenon*, Logic, methodology and philosophy of science, VIII (Moscow, 1987), Stud. Logic Found. Math., vol. 126, North-Holland, Amsterdam, 1989, pp. 571–589.

[14] Wilfrid Hodges, *Compositional semantics for a language of imperfect information*, Log. J. IGPL **5** (1997), no. 4, 539–563 (electronic).

[15] ———, *Some strange quantifiers*, Structures in logic and computer science, Lecture Notes in Comput. Sci., vol. 1261, Springer, Berlin, 1997, pp. 51–65.

[16] Juha Kontinen and Ville Nurmi, *Team logic and second-order logic*, 16th International Workshop, WoLLIC 2009, Tokyo, Japan, June 21-24, 2009. Proceedings, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, June 2009, http://dx.doi.org/10.1007/978-3-642-02261-6_19.

[17] Juha Kontinen and Jouko Väänänen, *On definability in dependence logic*, Journal of Logic, Language and Information (2009), http://dx.doi.org/10.1007/s10849-009-9082-0.

[18] Barbara H. Partee, *Noun phrase interpretation and type-shifting principles*, Studies in Discourse Representation Theory and the Theory of Generalized Quantifiers (Jeroen Groenendijk, Dick de Jongh, and Martin Stokhof, eds.), Groningen-Amsterdam Studies in Semantics, vol. 8, Foris Publications, Dordrecht, 1986, pp. 115–144.

[19] Jouko Väänänen, *Dependence logic: A new approach to independence friendly logic*, London Mathematical Society Student Texts, vol. 70, Cambridge University Press, 2007.

[20] Wilbur John Walkoe, Jr., *Finite partially-ordered quantification*, J. Symbolic Logic **35** (1970), 535–555.

Index