Dependence and Independence in Logic

Juha Kontinen and Jouko Väänänen

ESSLLI 2010 CPH

Juha Kontinen and Jouko Väänänen (eds.)

Proceedings of

Dependence and Independence in Logic

ESSLLI 2010 Workshop

Copenhagen, August 16-20, 2010

Preface

Dependence and independence are common phenomena, wherever one looks: ecological systems, astronomy, human history, stock markets - but what is their role in logic and - turning the tables - what is the logic of these concepts?
The possibility of nesting quantifiers, thus expressing patterns of dependence and independence between variables, accounts for much of the expressive power of first order logic.
However, first order logic is not capable of expressing all such patterns, and as a consequence various generalizations - such as branching quantifiers, or the various variants of independence-friendly logic - have been introduced during the last fifty years.
Dependence logic is a recent formalism, which brings to the forefront the very concept of dependence, isolating it from the notion of quantifier and making it one of the primitive elements of the language. It can also be added to other logics, such as modal logic.
This has opened up an opportunity to develop logical tools for the study of complex forms of dependence, with applications to computer science, philosophy, linguistics, game theory and mathematics. Recently there has been an increasing interest in this topic, especially among young researchers.
The goal of this workshop is to provide an opportunity for researchers to further explore the very notions of dependence and independence and their role in formal logic, in particular with regard to logics of imperfect information.

Programme Committee

Wilfrid Hodges (British Academy)
Tapani Hyttinen (University of Helsinki)
Juha Kontinen (University of Helsinki)
Michał Krynicki (Cardinal Stefan Wyszyński University)
Jouko Väänänen (University of Helsinki and University of Amsterdam)

Contents

Invited lectures

Contributed talks

Relational Hidden Variables and Non-Locality

Samson Abramsky
Oxford University Computing Laboratory

Abstract

We use a simple relational framework to develop the key notions and results on *hidden variables* and *non-locality*. The extensive literature on these topics in the foundations of quantum mechanics is couched in terms of probabilistic models, and properties such as locality and no-signalling are formulated probabilistically. We show that to a remarkable extent, the main structure of the theory, through the major No-Go theorems and beyond, survives intact under the replacement of probability distributions by mere relations. In particular, probabilistic notions of independence are replaced by purely logical ones.

We also study the relationships between quantum systems, probabilistic models and relational models. Probabilistic models can be reduced to relational ones by the 'possibilistic collapse', in which non-zero probabilities are conflated to (possible) truth. We show that all the independence properties we study are preserved by the possibilistic collapse, in the sense that if the property in its probabilistic form is satisfied by the probabilistic model, then the relational version of the property will be satisfied by its possibilistic collapse. More surprisingly, we also show a *lifting property*: if a relational model satisfies one of the independence properties, then there is a probabilistic model whose possibilistic collapse gives rise to the relational model, and which satisfies the probabilistic version of the property. These probabilistic models are constructed in a canonical fashion by a form of maximal entropy or indifference principle.

Workshop on Dependence and Independence in logic
European Summer School on Logic, Language and Information ESSLLI
16 - 20 August, 2010
Copenhagen, Denmark

On imperfect information logic as [0,1]-valued logic.

Xavier Caicedo
Universidad de los Andes, Bogotá

Imperfect information logic may be given a [0,1]-valued semantics in finite models by means of the expected value of an optimal pair of mixed strategies for the associated games, so that a sentence φ which is undetermined in a model M gets an intermediate value $0 < M\ (\varphi) < 1$ (Sevenster-Sandu, Galliani). It remains the problem whether for each rational r and sentence φ the class $\{M : M$ finite and $M\ (\varphi) \geq r\}$ is definable in the ordinary sense in imperfect information logic (that is, whether it belongs to NP). In very general grounds it may be shown that there is no [0,1]-valued semantics in all models extending the partial {0,1}-valued game semantics of imperfect information logic such that $M(\neg\varphi) = 1 - M(\varphi)$, $M(\varphi \vee \psi) = \max\{M(\varphi), M(\psi)\}$ and $\{M : M\ (\varphi) \geq r\}$ is in Σ^1_1 for each rational r.

Constraint Satisfaction Problems, Partially Ordered Connectives and Dependence Logic

Lauri Hella
University of Tampere

Joint work with **Merlijn Sevenster** and **Tero Tulenheimo**

Abstract: A constraint satisfaction problem (CSP) is the problem of deciding whether there is an assignment of a set of variables in a fixed domain such that the assignment satisfies given constraints. Every CSP can be expressed alternatively as a homomorphism problem: for a given finite relational structure $\mathbf{A}$, $\mathrm{Csp}(\mathbf{A})$ is the class of all structures $\mathbf{B}$ that can be homomorphically mapped into $\mathbf{A}$. The infamous *Dichotomy Conjecture* of Feder and Vardi ([1]) states that for all $\mathbf{A}$, the problem $\mathrm{Csp}(\mathbf{A})$ is either NP-complete or it is in PTIME.

In their seminal work [1] on the descriptive complexity of CSP, Feder and Vardi introduced a natural fragment of Σ^1_1, which they call *Monotone Monadic* SNP (MMSNP). They showed that every CSP is expressible in MMSNP, and moreover, the Dichotomy Conjecture holds for CSP if and only if it holds for MMSNP.

In the article [2], we established connections between partially ordereded connectives, CSP and MMSNP. In our first main result, we characterize SNP (the fragment of Σ^1_1 consisting of formulas $\exists\vec{X}\forall\vec{x}\,\phi$, where ϕ is quantifier free) in terms of partially ordered connectives. More precisely, we prove that SNP $\equiv$ D[QF], where D[QF] is the set of all formulas with a single partially ordered connective applied to a matrix of quantifier free formulas. In our second main result we give a similar characterization for MMSNP: MMSNP $\equiv$ C_1[NEQF]. Here C is a minor variant of the set of partially ordered connectives, and the subscript 1 refers to the restriction that each row of the connective has only one universal quantifier. Furthermore, NEQF is the set of equality and quantifier free formulas such that all relation symbols of the vocabulary appear only negatively.

In the talk, I will first give a survey on CSP, MMSNP, SNP and partially ordered connectives. Then I will go through the main results in [2]. Finally, I will consider some new ideas concerning certain fragments of dependence logic, and their relationship to partially ordered connectives.

References:

1. Feder, T. and Vardi M. (1998) The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory. SIAM Journal on Computing, 28, 57–104.

2. Hella, L., Sevenster, M. and Tulenheimo, T. (2008) Partially ordered connectives and monadic monotone strict NP. Journal of Logic, Language and Information, 17, 323–344.

Epistemic Operators and Uniform Definability in Dependence Logic*

Pietro Galliani
ILLC
University of Amsterdam
The Netherlands
(pgallian@gmail.com)

Abstract

The concept of *uniform definability* of operators in the language of Dependence Logic is formally defined, and two families of operators δ^α and quantifiers $\forall^\alpha$, for α ranging over the cardinal numbers, are investigated. In particular, it is proved that for any finite n and m, all $\forall^n$, $\forall^m$, δ^n and δ^m are reciprocally uniformly definable, but neither of them is uniformly definable in Dependence Logic: this answers an open problem by Kontinen and Väänänen ([12]) about whether the $\forall^1$ quantifier is uniformly definable in Dependence Logic.

A more direct proof than that of ([12]) is also found of the fact that the $\forall^1$ quantifier (and, consequently, all $\forall^n$ and δ^n quantifiers) do not increase the expressive power of Dependence Logic, even on open formulas; furthermore, an interpretation of the δ^n operators in terms of *partial announcements* is proved to hold for the Game Theoretic Semantics of Dependence Logic, and the Ehrenfeucht-Fraïssé game for Dependence Logic is adapted to the logics $\mathcal{D}(\sqcup, \forall^\alpha)$, where $\sqcup$ is the classical disjunction.

Finally, a criterion for uniform definability in Dependence Logic is proved.

1 Dependence Logic

Logics of imperfect information ([6], [7], [15], [14]) are extensions of First Order Logic whose Game Theoretic Semantics may be derived from that of First Order Logic by restricting the amount of information available to the players at certain stages of the game - or, equivalently, by allowing the semantic dependence relation between quantified variables to differ from the syntactic scoping relation between the corresponding quantifiers.

Dependence Logic ([15]) is distinguished by other logics of imperfect information in that it separates the concept of dependence and independence from the operation of quantification: this is achieved by introducing *dependence atomic formulas* $=(t_1 \ldots t_n)$, expressing the fact that the value of the term t_n is a function of the values of the terms $t_1 \ldots t_{n-1}$, and doing away with the nonstandard forms of quantification of the logic of Branching Quantifiers ([6]), of the many variants of Independence Friendly Logic ([8], [11], [4]) and of Dependence Friendly Logic ([15], [5]).

Because of this, Dependence Logic ($\mathcal{D}$ for short) is an eminently suitable formalism for reasoning about functional dependence and independence; furthermore, Hodges' Compositional Semantics for

*This work was supported by the European Science Foundation Eurocores programme LogICCC [FP002 - Logic for Interaction (LINT)].

logics of imperfect information ([9]) can be given an especially clean formulation for it, which will now be summarized.

In brief, a *team* X is defined as a set of assignments having the same domain and representing the information state of Player II at a certain stage of a semantic game. The expression $M \models_X \phi$ may be interpreted as the statement that, if the play of a given semantic game has a reached a position in which Player II has the (true) information that the current assignment belongs to X and the current node of the game tree corresponds to the root of the formula ϕ, then Player II has a uniform winning strategy for the rest of the game.

The rules for this compositional semantics can be given as follows, where we assume for simplicity that negation occurs only in front of atomic formulas:

$M \models_X Rt_1 \ldots t_n$ if and only if for all $s \in X$, $(t_1\langle s\rangle \ldots t_n\langle s\rangle) \in R^M$;

$M \models_X \neg Rt_1 \ldots t_n$ if and only if for all $s \in X$, $(t_1\langle s\rangle \ldots t_n\langle s\rangle) \notin R^M$;

$M \models_X t = t'$ if and only if for all $s \in X$, $t\langle s\rangle = t'\langle s\rangle$;

$M \models_X \neg t = t'$ if and only if for all $s \in X$, $t\langle s\rangle \neq t'\langle s\rangle$;

$M \models_X =(t_1 \ldots t_n)$ if and only if for all $s, s' \in X$, if $t_i\langle s\rangle = t_i\langle s'\rangle$ for $i = 1 \ldots n-1$ then $t_n\langle s\rangle = t_n\langle s'\rangle$;

$M \models_X \neg =(t_1 \ldots t_n)$ if and only if $X = \emptyset$;

$M \models_X \phi \vee \psi$ if and only if there exist X_1, X_2 such that $X \subseteq X_1 \cup X_2$, $M \models_{X_1} \phi$ and $M \models_{X_2} \psi$;

$M \models_X \phi \wedge \psi$ if and only if $M \models_X \phi$ and $M \models_X \psi$;

$M \models_X \exists x\phi$ if and only if there exists a function $F : X \to Dom(M)$ such that, for $X' = X[F/x] = \{s[F(s)/x] : s \in X\}$, it holds that $M \models_{X'} \phi$;

$M \models_X \forall x\phi$ if and only if, for $X' = X[M/x] = \{s[m/x] : s \in X, m \in Dom(M)\}$, it holds that $M \models_{X'} \phi$.

A sentence ϕ is said to be *true* in a model M if and only if $M \models_{\{\emptyset\}} \phi$.

A full discussion of this definition, as well of the current body of knowledge about the properties of Dependence Logic, is far beyond the scope of this work: therefore, only a couple of simple results which will be of some relevance for the rest of this work will be mentioned.

- There is a certain asymmetry between the rules for the interpretation of the existential quantifier and of the disjunction and those for the interpretation of the universal quantifier and of the conjunction. This is the case because, in the satisfaction relation $M \models_X \phi$, the team X describes the information state of Player II at a certain point of a semantic game - and hence, when considering operators which correspond to moves of Player I (such as $\forall$ or $\wedge$) we must enlarge the current team by considering all possible choices of Player I at once, whereas when considering operators corresponding to moves of Player II we only need to select one of the choices available to the second player and update the team according to it. In particular, it must be pointed out that the rule of the universal quantification differs from

$$M \models_X \widetilde{\forall} x\phi \text{ if and only if, for all } F : X \to Dom(M),\ M \models_{X[F/x]} \phi :$$

  for example, the sentence $\exists x \widetilde{\forall} y =(x, y)$ can be seen to be true in any model M, although $\exists x \forall y =(x, y)$ is not true in any model with at least two elements. Incidentally, the $\widetilde{\forall} x$ operator thus defined is the $\forall$ operator of Team Logic, whereas the Dependence Logic $\forall$ operator is the ! ("bang") operator of the same logic ([16], [15]).

- Analogously, there is an asymmetry between the truth and the falsity conditions of the dependence atom $=(t_1 \dots t_n)$: $X \models \;=(t_1 \dots t_n)$ if and only if the relation $\{(t_1\langle s\rangle, \dots t_n\langle s\rangle) : s \in X\}$ satisfies the functional dependency condition $t_1 \dots t_{n-1} \rightarrow t_n$ (and in particular, $X \models \;=(x)$ if and only if the value of the variable x is the same in all assignments contained in X), but the negation of a dependency atomic formula does not hold in any nonempty team. This can be seen as a consequence of the fact that the relation $M \models_X \phi$ states the existence of a winning strategy for Player II in the subgame corresponding to ϕ given that the current assignment is known to be in X, and hence the following *closure condition* ([15]) holds:

  $$M \models_X \phi, X' \subseteq X \Rightarrow M \models_{X'} \phi.$$

  If X is a nonempty team, it is easy to verify that $\{s\} \models=(t_1 \dots t_n)$ for all assignments $s \in X$ and all dependence atomic formulas $=(t_1 \dots t_n)$; therefore, by the closure principle and the law of non-contradiction it cannot be the case that $M \models_X \neg =(t_1 \dots t_n)$.

2 Uniform Definability and Uniform Translatability

In the next sections, we will study a few operators over Dependence Logic formulas and discuss their uniform definability, both with respect to $\mathcal{D}$ and with respect to each other.

But what *is* uniform definability? The intuitive meaning is fairly transparent: in brief, given a logic $\mathcal{L}$ an operator O, from formulas ϕ to meanings $O(\phi)$, is uniformly definable if and only if the behaviour of O can be simulated by a suitable combination of the primitives of $\mathcal{L}$. But a formal definition of this concept requires some care.

Three very simple examples for First Order Logic may be of some use for showing how the notions of uniform definability and uniform translatability are somewhat more delicate than they might appear at first sight:

Example 2.1 (The $\exists^{\leq n}$ quantifier). *Let $FO(\exists^{\leq n} : n \in \mathbb{N})$ be First Order Logic augmented with the quantifiers $\{\exists^{\leq n} x : n \in \mathbb{N}, x \in Var\}$, whose semantics are given by the conditions*

$$M \models_s \exists^{\leq n} x\phi \Leftrightarrow |\{m \in M : M \models_{s[m/x]} \phi\}| \leq n.$$

Then $FO(\exists^{\leq n} : n \in \mathbb{N})$ not only has has precisely the same expressive power of First Order Logic, but it is uniformly translatable into FO, since $\exists^{\leq n} x$ is uniformly definable as

$$\exists^{\leq n} x\phi \equiv \exists x_1 \dots x_n \forall x(\phi \rightarrow \bigvee_{i=1}^{n} x = x_i) \tag{1}$$

for all formulas ϕ such that $x_1, \dots x_n \notin FV(\phi)$.

But what of the condition that $x_1 \dots x_n \notin FV(\phi)$? On one hand, it cannot be omitted, as for example $\exists^{\leq n} xR(x, x_1) \not\equiv \exists x_1 \dots x_n(R(x, x_1) \rightarrow \bigvee_{i=1}^{n} x = x_i)$ - but on the other it does not appear to be logically significant, as it is always possible to change the bound variables $x_1 \dots x_n$ of (1) in order to avoid naming conflicts of this kind. So what is going on here? Which properties of the "context" $\exists x_1 \dots x_n \forall x(\Xi \rightarrow \bigvee_{i=1}^{n} x = x_i)$ are essential to our notion of uniform definability, and which ones are not so?

Example 2.2. *Given two variables x and y, let $Sym_{xy}(\phi)$ be the First Order Logic operator defined as*

$$M \models_s Sym_{xy}(\phi) \Leftrightarrow \text{ for all } m, n \in M, (M \models_{s[m/x][n/y]} \phi \Leftrightarrow M \models_{s[n/x][m/y]} \phi).$$

It is obvious that such an operator does not increase the expressive power of First Order Logic, and $Sym_{xy}(\phi)$ is "uniformly" definable as

$$Sym_{xy}(\phi) := \forall xy(\phi(x, y) \leftrightarrow \phi(y, x)). \tag{2}$$

But what kind of expression is precisely the above one, and in which sense can we say that it defines uniformly Sym_{xy}? In (2) we appear to be dealing with the "argument" ϕ almost as if it were an atomic formula: is this different from what we did in (1)? And what does the expression (2) mean precisely if $FV(\phi) \neq \{x, y\}$?

Example 2.3. *Let us consider, again in First Order Logic, the* relativization *operator $(\phi)^P$ whose semantics can be given recursively as*

$$\begin{aligned}
&(Rt_1 \dots t_n)^P := Rt_1 \dots t_n;\\
&(t = t')^P := t = t';\\
&(\neg\phi)^P := \neg(\phi)^P;\\
&(\exists x\phi)^P := \exists x(Px \wedge \phi);\\
&(\forall x\phi)^P := \forall x(Px \rightarrow \phi).
\end{aligned}$$

By its own definition, this operator does not increase the expressive power of First Order Logic. But is it uniformly definable in it?

An easy argument shows that this cannot be the case: consider a model M with domain $\{a, b\}$, and consider the predicates $P^M = \{a\}$ and $Q^M = \emptyset$. Then, for all assignments s, $M \models_s \forall xPx \Leftrightarrow M \models_s \forall xQx$, as neither $\forall xPx$ and $\forall xQx$ are satisfied in M. Hence, if for some "context" $\Phi[\Xi]$ it held that $\Phi[\psi] \equiv (\psi)^P$ for all ψ, it would be the case that $M \models_s (\forall xPx)^P \Leftrightarrow M \models_s \Phi[\forall xPx] \Leftrightarrow M \models_s \Phi[\forall xQx] \Leftrightarrow M \models_s (\forall xQx)^P$. But $(\forall xPx)^P = \forall x(Px \rightarrow Px)$ is true in M, while $(\forall xQx)^P = \forall x(Px \rightarrow Qx)$ is not; and therefore, no such Φ exists.

But what sort of object is $\Phi[\Xi]$ in this argument, and for what reason exactly can we argue that if $\phi \equiv_M \psi$ then $\Phi[\phi] \equiv_M \Phi[\psi]$?

In this section, I will attempt to give a formal definition of uniform definability for Dependence Logic; definitions for other logics (e.g., for First Order Logic) can be given along entirely similar lines.

Definition 2.1 (Model classes, downwards closed model classes)**.** *Let Σ be a signature, and V a finite set of variables.*

A model class *with domain V is a set of pairs (M, X), where M is a first order model with signature Σ and X is a team over M with domain V.*

A model class K is said to be downwards closed *if*

$$(M, X) \in K, X' \subseteq X \Rightarrow (M, X') \in K$$

for all models M of signature Σ and for all teams X, X' with $Dom(X) = Dom(X') = Dom(K)$.

We write $MC_\Sigma(V)$ for the family of all model classes with domain V, and $DMC_\Sigma(V)$ for the family of all downwards closed model classes with domain V.

Given a model class K, a model M with signature $Sig(M) \supseteq Sig(K)$ and a team X with $Dom(X) \supseteq Dom(K)$, we write $M \models_X K$ for

$$(M_{\restriction Sig(K)}, X_{|Dom(K)}) \in K.$$

Definition 2.2 ($\|\phi\|_\Sigma(V)$). *Let ϕ be any Dependence Logic formula, let Σ be any signature containing the signature of ϕ, and let V be any set of variables such that $FV(\phi) \subseteq V$. Then*

$$\|\phi\|_\Sigma(V) = \{(M, X) \in M : Dom(X) = V \text{ and } M \models_X \phi\}$$

where M ranges over all first order models with signature Σ and X ranges over all teams over M.

The following lemma is just another way of stating the Downwards Closure Property of Dependence Logic ([15], Proposition 3.10):

Lemma 2.1. *For each $\phi \in \mathcal{D}$, for every Σ containing the signature of ϕ and for every $V \supseteq FV(\phi)$ it holds that $\|\phi\|_\Sigma(V) \in DMC_\Sigma(V)$.*

Definition 2.3 (Context). *Let Σ be a signature and let $n_1 \dots n_t \in \mathbb{N}$. Then a* context *over Σ of type $\langle n_1 \dots n_t\rangle$ is an expression $\psi[\Xi_1 \dots \Xi_t]$ in $\Sigma \cup \{\Xi_1 \dots \Xi_n\}$, where each Ξ_i is a new n_i-ary relation symbol which does not occur negated in ψ.*

Dependence Logic contexts induce functions from tuples of downwards closed model classes to downwards closed model classes: somewhat informally, given a context $\Phi[\Xi_1 \dots \Xi_t]$ and model classes $K_1 \dots K_t$ with domains $\{v_{1,1} \dots v_{1,n_1}\} \dots \{v_{t,1} \dots v_{t,n_t}\}$ and signatures $\Sigma_1 \dots \Sigma_t$ we assign to the expression $\Phi[K_1 \dots K_t]$ the satisfaction relation obtained by applying the recursive satisfaction definition of the Team Semantics for Dependence Logic, plus

$$M \models_X \Xi_i(t_1 \dots t_{n_i}) \Leftrightarrow M \models_{\{(v_{i,1}:s\langle t_1\rangle \dots v_{i,n_i}:s\langle t_{n_1}\rangle):s\in X\}} K$$

where the order of $v_{i,1} \dots v_{i,n_i}$ is presumed fixed.

More precisely, a Dependence Logic context operates on model classes as follows:

Definition 2.4. *For all $i \in 1 \dots t$ let $K_i \in DMC_{\Sigma_i}(V_i)$, where $|V_i| = n_i$ and $(v_{i,1} \dots v_{i,n_i})$ is a fixed ordering of the variables in V_i. Then, for all models M of the appropriate signature and for all teams X of the appropriate domains,*

1. *$M \models_X \Xi_i(t_1 \dots t_{n_i})[K_1 \dots K_t]$ if and only if $M \models_{[\overline{v_i} \mapsto \overline{t}]X} K$, where*

$$[\overline{v_i} \mapsto \overline{t}]X = \{s : Dom(s) = V_i, \exists s' \in X \text{ s.t. } s(v_{i,1}) = t_1\langle s'\rangle, \dots, s(v_{i,n_i}) = t_{n_i}\langle s'\rangle\};$$

2. *If Φ is a literal ϕ, $M \models_X \Phi[K_1 \dots K_t]$ if and only if $M \models_X \phi$;*

3. *$M \models_X (\Phi \vee \Psi)[K_1 \dots K_t]$ if and only if there exist X_1 and X_2 such that $X = X_1 \cup X_2$, $M \models_{X_1} \Phi[K_1 \dots K_t]$ and $M \models_{X_2} \Psi[K_1 \dots K_t]$;*

4. $M \models_X (\Phi \wedge \Psi)[K_1 \ldots K_t]$ *if and only if* $M \models_X \Phi[K_1 \ldots K_t]$ *and* $M \models_X \Psi[K_1 \ldots K_t]$*;*

5. $M \models_X \exists x \Phi[K_1 \ldots K_t]$ *if and only if* $M \models_{X[F/x]} \Psi[K_1 \ldots K_t]$ *for some* $F : M \to X$*;*

6. $M \models_X \forall x \Phi[K_1 \ldots K_t]$ *if and only if* $M \models_{X[M/x]} \Psi[K_1 \ldots K_t]$.

As for the case of dependence formulas, we will write $\|\Phi[K_1 \ldots K_t]\|_\Sigma(V)$ for the set $\{(M, X) : Sig(M) = \Sigma, Dom(X) = V, M \models_X \Phi[K_1 \ldots K_t]\}$.

Furthermore, if $\phi_1 \ldots \phi_t$ are formulas in $\mathcal{D}$ with $FV(\phi_i) \subseteq V_i$ for $i \in 1 \ldots t$, we write $M \models_X \Phi[\phi_1 \ldots \phi_t]$ as a shorthand for

$$M \models_X \Phi[\|\phi_1\|_{\Sigma_1}(V_1) \ldots \|\phi_t\|_{\Sigma_t}(V_t)].$$

Definition 2.5 (Uniform definability)**.** *An operator* $F : DMC_{\Sigma_1}(V_1) \times \ldots \times DMC_{\Sigma_t}(V_t) \to DMC_{\Sigma'}(V')$ *is said to be* uniformly definable *in Dependence Logic if and only if there exists a context* $\Phi[\Xi_1 \ldots \Xi_t]$*, of type* $\langle |V_1| \ldots |V_t| \rangle$*, such that*

$$\|\Phi[K_1 \ldots K_t]\|_{\Sigma'}(V') = F(K_1 \ldots K_t)$$

for all model classes $K_1 \ldots K_t$ *in* $MC(V_1) \ldots MC(V_t)$*.*

Definition 2.6 (Uniform translatability and uniform equivalence)**.** *Let* $\mathcal{L}_1, \mathcal{L}_2$ *be two extensions of Dependence Logic. We say that* $\mathcal{L}_1$ *is* uniformly translatable *in* $\mathcal{L}_2$*, and we write* $\mathcal{L}_1 \trianglelefteq \mathcal{L}_2$*, if any operator* F *as above which is uniformly definable*[1] *in* $\mathcal{L}_1$ *is also uniformly definable in* $\mathcal{L}_2$*.*[2]

If $\mathcal{L}_1 \trianglelefteq \mathcal{L}_2$ *and* $\mathcal{L}_2 \trianglelefteq \mathcal{L}_1$ *we say that* $\mathcal{L}_1$ *and* $\mathcal{L}_2$ *are* uniformly equivalent *and write* $\mathcal{L}_1 \bowtie \mathcal{L}_2$*.*

Example 2.4. *In [12], the following question is asked and immediately answered: is it possible to express, in Dependence Logic, the operator* $\exists^1 x$ *whose semantics is*

$$M \models_X \exists^1 x \phi \Leftrightarrow \exists m \in M \text{ s.t. } M \models_{X[m/x]} \phi?$$

A positive answer was given by defining $\exists^1 x \phi$ *as* $\exists x(=(x) \wedge \phi)$*: indeed, one can check that for all formulas* ϕ*, for all models* M *and for all teams* X*,* $M \models_X \exists^1 x \phi \Leftrightarrow M \models_X \exists x(=(x) \wedge \phi)$*.*

One way to represent this observation in our framework would be to ask, for any finite set of variables $\{x, x_1 \ldots x_n\}$*, whether there exists a context* $\Phi[\Xi]$ *of type* $\langle n+1 \rangle$ *such that*

$$\Phi[K] = \{(M, X) : Dom(X) \subseteq \{x_1 \ldots x_n\} \text{ and } \exists m \in Dom(M) \text{ s.t. } M \models_{X[m/x]} K\}$$

for all model classes K *with* $Dom(K) = \{x, x_1 \ldots x_n\}$*.*

[1]Uniform definability in extensions of Dependence Logic is defined exactly as it was done for Dependence Logic itself, with the obvious changes to the definition of context.

[2]This notion appears to have some relation with the concept of *compositional translation* defined in [10], and it seems likely that it will reduce to it if considered in a more abstract setting.

A positive answer can be then found by choosing $\Phi[\Xi] := \exists x(=(x) \wedge \Xi(x, x_1 \ldots x_k))$*: indeed,*

$$\begin{aligned}&M \models_X \exists x(=(x) \wedge \Xi(x, x_1 \ldots x_k))[K] \Leftrightarrow\\ &\Leftrightarrow \exists F : M \to X \text{ s.t. } M \models_{X[F/x]} =(x) \wedge M \models_{X[F/x]} K \Leftrightarrow\\ &\Leftrightarrow \exists m \in M \text{ s.t. } M \models_{X[m/x]} K\end{aligned}$$

as required.

Hence, if $\mathcal{D}(\exists^1)$ *is the logic obtained by augmenting* $\mathcal{D}$ *with the* $\exists^1$ *quantifiers we have that* $\mathcal{D}(\exists^1) \trianglelefteq \mathcal{D}$*: indeed, for any context* $\Phi[\Xi_1 \ldots \Xi_t]$ *of* $\mathcal{D}(\exists^1)$ *and sets of variables* $V_1 \ldots V_t$ *we can find a context* $\Phi'[\Xi_1 \ldots \Xi_t]$ *in* $\mathcal{D}$ *equivalent to it by "expanding" each expression* $\exists^1 x \Psi[\Xi_1 \ldots \Xi_t]$ *with* $FV(\Psi) \in \{x, x_1 \ldots x_n\}$ *as* $\exists x(=(x) \wedge \Psi[\Xi_1 \ldots \Xi_t](x, z_1 \ldots z_n))$*, where* $\Psi[\Xi_1 \ldots \Xi_n](x, z_1 \ldots z_t)$ *is simply* $\Psi[\Xi_1 \ldots \Xi_t]$*.*

As $\mathcal{D} \trianglelefteq \mathcal{D}(\exists^1)$*, we then have that* $\mathcal{D} \bowtie \mathcal{D}(\exists^1)$*.*

As this example shows, in this framework uniform definability is relative to a given domain of variables: the expressions $\exists x(=(x) \wedge \Xi(x))$ and $\exists x(=(x) \wedge \Xi(x, x_1))$ are different contexts belonging to different types, even though both of them are instances of the same informal "context" $\exists x(=(x) \wedge \Xi)$. On one hand, this limits somewhat the generality of our notion of uniform definability; but on the other, in this way we avoid entirely the possibilities of naming collisions discussed in Example 2.1 while accounting for uniform definitions such as that of Example 2.2.

Finally, some discussion about the requirement that the Ξ_is do not occur negated in a context $\Phi[\Xi_1 \ldots \Xi_t]$ may be necessary. The main reason for this condition, of course, is that we did not give a definition for the negation in Dependence Logic, although such a definition is certainly possible ([15], Definition 3.5).

But is this affecting in any way our notion of uniform definability? Is there any property concerning the truth of ϕ in some team or family of teams that could be expressed uniformly as $\Phi[\phi, \neg\phi]$ for some $\Phi[\Xi_1, \Xi_2]$, but not as $\Psi[\phi]$ for any $\Psi[\Xi]$?

By the properties of negation in Dependence Logic, this cannot be the case: indeed, in Dependence Logic there exist formulas which are true in the same models but which are not false in the same models ([15], §3.3)[3] and in particular we have that, for all formulas ϕ,

$$M \models_X \phi \Leftrightarrow M \models_X (\phi \vee \forall x =(x))$$

but

$$M \models_X \neg(\phi \vee \forall x =(x)) \Leftrightarrow X = \emptyset.$$

Hence, any condition concerning the truth of ϕ will also hold for $\phi^* = (\phi \vee \forall x =(x))$, and if this condition can be uniformly defined as $\Phi[\phi, \neg\phi]$ then it can be also uniformly defined as $\Phi[\phi^*, \neg\phi^*] \equiv \Phi[\phi, \bot]$.

[3]More in general, negation in Dependence Logic is not a semantic operation, as it was proved in a very strong sense in [3] and in [13].

3 The $\forall^n$ quantifiers and the δ^n operators

The $\forall^1 x$ quantifier is defined in the last section of [12] as

$$M \models_X \forall^1 x\phi \Leftrightarrow \text{ for all } m \in M, M \models_{X[m/x]} \phi$$

where $X[m/x] = \{s[m/x] : s \in X\}$.

Juha Kontinen and Jouko Väänänen then proved, by means of their main theorem concerning the relationship between $\mathcal{D}$ and Σ_1^1, that this quantifier does not increase the expressive power of the logic, even with respect to open formulas: if $\mathcal{D}(\forall^1)$ is Dependence Logic augmented with the $\forall^1$ quantifiers then for every $\phi \in \mathcal{D}(\forall^1)$ there exists a $\psi \in \mathcal{D}$ such that

$$M \models_X \phi \Leftrightarrow M \models_X \phi^*$$

for all models M and teams X.

Finally, they left the following problem to the future researchers ([12]):

It remains open whether the quantifier $\forall^1$ is "uniformly" definable in the logic $\mathcal{D}$.

In this section the $\forall^1$ quantifier will be studied as a member of a family of quantifiers $\forall^\alpha$, where α is a (finite or infinite) cardinal; furthermore, all of these quantifiers will be decomposed in terms of the Dependence Logic universal quantifier $\forall x$ and of the *announcement operators*[4] δ^α.

Moreover, a recursive definition of $\delta^1 x$ in Dependence Logic will be given, and it will be proved that for all $n \in \mathbb{N}_0$

$$\mathcal{D}(\forall^n), \mathcal{D}(\delta^n) \bowtie \mathcal{D}(\delta^1).$$

As a corollary, this will give us a "lower tech" proof of the statement, already shown in [12], that the $\forall^1$ quantifier does not increase the expressive power of Dependence Logic, and will prove that the same holds for all $\forall^n$ and δ^n.

Then, after two sections in which we will study the game-theoretic properties of the $\forall^\alpha$ quantifiers and of the δ^α operators, we will return in the last section to Kontinen and Väänänen's question, and we will answer it negatively by means of a characterization of uniform definability in terms of Σ_1^1 formulas.

We begin our study of the $\forall^1 x$ quantifier by introducing a new operator, which will be seen to be strictly related to it:

Definition 3.1 (δ^1)**.** *For any formula ϕ and variable x, let $\delta^1 x\phi$ be a formula with $FV(\delta^1 x\phi) = FV(\phi) \cup \{x\}$, whose satisfaction condition is*

$$M \models_X \delta^1 x\phi \Leftrightarrow \textit{for all } m \in M, M \models_{X_{x=m}} \phi$$

where $X_{x=m}$ is the team $\{s \in X : s(x) = m\}$.

We write $\mathcal{D}(\delta^1)$ for the logic obtained by adding the δ^1 operator to Dependence Logic.

[4]The name "announcement operator" has been chosen for their similarity with the public announcement operators of Dynamic Epistemic Logic ([2], [17]). It is not entirely clear at the moment how deep this similarity runs, but the author believes this to be just one of the many possible links between Dependence Logic and Dynamic Epistemic Logic.

The following result links the δ^1 and $\forall^1$ operators together:

Proposition 3.1. *For any formula ϕ of Dependence Logic, $\forall^1 x\phi \equiv \forall x \delta^1 x\phi$ and $\delta^1 x\phi \equiv \forall^1 y(x \neq y \vee \phi)$, where y is a variable which does not occur in $FV(\phi)$.*

Proof. Let M be any first-order model, let X be any team, and let ϕ be any formula with $y \notin FV(\phi)$. Then, for all X with $FV(\phi) \subseteq X \cup \{x\}$,

$$M \models_X \forall x \delta^1 x\phi \Leftrightarrow M \models_{X[M/x]} \delta^1 x\phi \Leftrightarrow \text{ for all } m \in M, M \models_{X[M/x]_{x=m}} \phi \Leftrightarrow$$
$$\Leftrightarrow \text{ for all } m \in M, M \models_{X[m/x]} \phi \Leftrightarrow M \models_X \forall^1 x\phi.$$

and for all X with $FV(\phi) \subseteq X$,

$$M \models_X \forall^1 y(x \neq y \vee \phi) \Leftrightarrow \text{ for all } m \in M, M \models_{X[m/y]} x \neq y \vee \phi \Leftrightarrow$$
$$\Leftrightarrow \text{for all } m \in M \ \exists\, Y^m, Z^m \text{ such that } X[m/y] \subseteq Y^m \cup Z^m, \text{ if } s \in Y^m \text{ then } s(x) \neq m \text{ and } M \models_{Z^m} \phi \Leftrightarrow$$
$$\Leftrightarrow \text{for all } m \in M, M \models_{X[m/y]_{x=m}} \phi \Leftrightarrow \text{for all } m \in M, M \models_{X_{x=m}} \phi \Leftrightarrow M \models_X \delta^1 x\phi$$

where we used the fact that $y \notin FV(\phi)$. □

Corollary 3.1. $\mathcal{D}(\forall^1) \bowtie \mathcal{D}(\delta^1)$.

The $\delta^1 x$ quantifier does not increase the expressive power of Dependence Logic:

Proposition 3.2. *Let ϕ be any Dependence Logic formula. Then there exists a Dependence Logic formula ϕ^* such that $\phi^* \equiv \delta^1 x\phi$.*

Proof. The proof is a simple structural induction on ϕ:

- If ϕ is a first order literal then let $\phi^* = \phi$: indeed, in this case we have that

$$M \models_X \delta^1 x\phi \Leftrightarrow \forall m \in M, M \models_{X_{x=m}} \phi \Leftrightarrow$$
$$\Leftrightarrow \forall m \in M, \forall s \in X_{x=m}, \{s\} \models \phi \Leftrightarrow \forall s \in X, \{s\} \models \phi \Leftrightarrow X \models \phi.$$

- If ϕ is an atomic dependence atom $=(t_1 \dots t_n)$, let ϕ^* be $=(x, t_1 \dots t_n)$. Indeed,

$$M \models_X \delta^1 x =(t_1 \dots t_n) \Leftrightarrow \forall m \in M, M \models_{X_{x=m}} =(t_1 \dots t_n) \Leftrightarrow$$
$$\Leftrightarrow \forall m \in M, \forall s, s' \in X_{x=m}, \text{ if } t_i\langle s\rangle = t_i\langle s'\rangle \text{ for } i = 1 \dots n-1 \text{ then } t_n\langle s\rangle = t_n\langle s'\rangle \Leftrightarrow$$
$$\Leftrightarrow \forall s, s' \in X, \text{ if } s(x) = s'(x), t_i\langle s\rangle = t_i\langle s'\rangle \text{ for } i = 1 \dots n-1 \text{ then } t_n\langle s\rangle = t_n\langle s'\rangle \Leftrightarrow$$
$$\Leftrightarrow M \models_X =(x, t_1 \dots t_n).$$

- If ϕ is a negated atomic dependence atom $\neg =(t_1 \dots t_n)$, let ϕ^* be $\neg =(x, t_1 \dots t_n)$[5]: indeed,

$$M \models_X \delta^1 x\phi \Leftrightarrow \forall m \in M, M \models_{X_{x=m}} \neg =(t_1 \dots t_n) \Leftrightarrow \forall m \in M, X_{x=m} = \emptyset \Leftrightarrow$$
$$\Leftrightarrow X = \emptyset \Leftrightarrow M \models_X \neg =(x, t_1 \dots t_n).$$

[5]Or $\neg =(t_1 \dots t_n)$, or $\bot$, or any formula which holds only in the empty assignment. Here $\neg =(x, t_1 \dots t_n)$ was chosen only because of its symmetry with the $=(x, t_1 \dots t_n)$ case.

- If $\phi = \psi \vee \theta$, let $\phi^* = \psi^* \vee \theta^*$: indeed,

$$\begin{aligned} &M \models_X \delta^1 x(\psi \vee \theta) \Leftrightarrow \forall m \in M, M \models_{X_{x=m}} \psi \vee \theta \Leftrightarrow \\ &\Leftrightarrow \forall m \in M\ \exists Y^m, Z^m \text{ such that } X_{x=m} \subseteq Y^m \cup Z^m, M \models_{Y^m} \psi \text{ and } M \models_{Z^m} \theta \Leftrightarrow \\ &\Leftrightarrow \exists Y, Z \text{ such that } X \subseteq Y \cup Z \text{ and } \forall m \in M, M \models_{Y_{x=m}} \psi \text{ and } M \models_{Z_{x=m}} \theta \Leftrightarrow \\ &\Leftrightarrow \exists Y, Z \text{ such that } X \subseteq Y \cup Z, M \models_Y \delta^1 x\psi \text{ and } M \models_Z \delta^1 x\theta \Leftrightarrow \\ &\Leftrightarrow M \models_X \psi^* \vee \theta^* \end{aligned}$$

  where for the passage from the second line to the third one we take $Y = \bigcup_{m \in M} Y^m$ and $Z = \bigcup_{m\in M} Z^m$, and for the passage from the third line to the second one we take $Y^m = Y_{x=m}$ and $Z^m = Z_{x=m}$.

- If $\phi = \psi \wedge \theta$, let $\phi^* = \psi^* \wedge \theta^*$: indeed,

$$\begin{aligned} &M \models_X \delta^1 x(\psi \wedge \theta) \Leftrightarrow \forall m \in M, M \models_{X_{x=m}} \psi \wedge \theta \Leftrightarrow \\ &\Leftrightarrow \forall m \in M, M \models_{X_{x=m}} \psi \text{ and } M \models_{X_{x=m}} \theta \Leftrightarrow M \models_X \delta^1 x\psi \text{ and } M \models_X \delta^1 x\theta \Leftrightarrow M \models_X \psi^* \wedge \theta^*. \end{aligned}$$

- If $\phi = \exists y\psi$ for some variable $y \neq x$,[6] we let $\phi^* = \exists y\psi^*$: indeed,

$$\begin{aligned} &M \models_X \delta^1 x \exists y\psi \Leftrightarrow \forall m \in M, M \models_{X_{x=m}} \exists y\psi \Leftrightarrow \\ &\Leftrightarrow \forall m \in M \exists F^m : X_{x=m} \to M \text{ s.t. } M \models_{X_{x=m}[F^m/y]} \psi \Leftrightarrow \\ &\Leftrightarrow \exists F : X \to M \text{ s.t. } \forall m \in M, M \models_{X[F/y]_{x=m}} \psi \Leftrightarrow \\ &\Leftrightarrow \exists F : X \to M \text{ s.t. } M \models_{X[F/y]} \delta^1 x\psi \Leftrightarrow M \models_X \exists y\psi^* \end{aligned}$$

  where, for the passage from the second line to the third one, we take the function F defined as

$$\forall s \in X, F(s) = F^{s(x)}(s)$$

  and, for the passage from the third line to the second one, we take for each F^m the restriction of F to X^m.

- If $\phi = \forall y\psi$ for some variable $y \neq x$, we let $\phi^* = \forall y\psi^*$. Indeed,

$$\begin{aligned} &M \models_X \delta^1 x \forall y\psi \Leftrightarrow \forall m \in M, M \models_{X_{x=m}} \forall y\psi \Leftrightarrow \forall m \in M, M \models_{X_{x=m}[M/y]} \psi \Leftrightarrow \\ &\Leftrightarrow \forall m \in M, M \models_{X[M/y]_{x=m}} \psi \Leftrightarrow M \models_{X[M/y]} \delta^1 x\psi \Leftrightarrow M \models_X \forall y\psi^*. \end{aligned}$$

□

This result implies that the logics $\mathcal{D}$, $\mathcal{D}(\forall^1)$ and $\mathcal{D}(\delta^1)$ define exactly the same classes of teams over all models and all signatures.

The δ^1 operators - and, hence, the $\forall^1$ quantifiers - are uniformly definable in Intuitionistic Dependence Logic $\mathcal{D}(\to)$ [1], that is, in Dependence Logic augmented with the *intuitionistic implication*

$$M \models_X \phi \to \psi \Leftrightarrow \forall Y \subseteq X, \text{ if } M \models_Y \phi \text{ then } M \models_Y \psi :$$

[6] If $y = x$, we define $(\exists x\psi)^* := (\exists y\psi[y/x])^*$ and $(\forall x\psi)^* := (\forall y\psi[y/x])^*$ for some new variable y.

Proposition 3.3. $\mathcal{D}(\delta^1), \mathcal{D}(\forall^1) \trianglelefteq \mathcal{D}(\to)$.

Proof. For all formulas ϕ, and all teams X,

$$M \models_X =\!(x) \to \phi \Leftrightarrow \text{ for all } Y \subseteq X, \text{ if } M \models_X =\!(x) \text{ then } M \models_X \phi \Leftrightarrow$$
$$\Leftrightarrow \text{ for all } m \in M, M \models_{X_{x=m}} \phi \Leftrightarrow M \models_X \delta^1_x \phi.$$

□

However, as Intuitionistic Dependence Logic is strictly more expressive than Dependence Logic [1], $\mathcal{D}(\to) \not\leq \mathcal{D}(\forall^1), \mathcal{D}(\delta^1)$ and in particular $\mathcal{D}(\to) \not\trianglelefteq \mathcal{D}(\forall^1), \mathcal{D}(\forall^1)$.

The above proposition suggests that, as in the case of intuitionistic implication ([1]), the public announcement operators may be used to reduce dependency atoms $=\!(t_1 \dots t_n)$ to constant atoms $= (t_i)$. This is indeed the case:

Proposition 3.4. *Let $x_1 \dots x_n$ be variables. Then*

$$=\!(x_1 \dots x_n) \equiv \delta^1 x_1 \dots \delta^1 x_{n-1} =\!(x_n).$$

Proof.

$$M \models_X \delta^1 x_1 \dots \delta^1 x_{n-1} =\!(x_n) \Leftrightarrow$$
$$\Leftrightarrow \forall m_1 \dots m_{n-1} \in M, M \models X_{x_1=m_1 \dots x_{n-1}=m_{n-1}} =\!(x_n) \Leftrightarrow$$
$$\Leftrightarrow \forall m_1 \dots m_{n-1} \in M, s, s' \in X, \text{ if } s(x_1) = s'(x_1) = m_1, \dots, s(x_{n-1}) = s'(x_{n-1}) = m_{n-1} \text{ then}$$
$$\text{then } s(x_n) = s'(x_n) \Leftrightarrow$$
$$\Leftrightarrow M \models_X =\!(x_1 \dots x_n).$$

□

In the same way, one may decompose dependency atoms of the form $=\!(t_1 \dots t_n)$ as

$$\exists x_1 \dots x_{n-1} (\bigwedge_{i=1}^{n-1} x_i = t_i) \wedge \delta^1 x_1 \dots \delta^1 x_{n-1} =\!(t_n)$$

or introduce "term announcements" $\delta^1(t)$ with the obvious semantics; hence, by removing non-constant dependency atoms from Dependence Logic and adding the δ^1 operators one may obtain a formalism with the same expressive power of Dependence Logic, in which *constancy* takes the place of *functional dependency*. This new logic $\mathcal{C}(\delta^1)$, which one may call *constancy logic with announcements*, may well be deserving of further investigation; however, this line of thought will not be pursued further here.

$\forall^1$ and δ^1 can be seen as representatives of a proper class of operators $\{\forall^\alpha, \delta^\alpha : \alpha \in Card\}$:

Definition 3.2. *($\forall^\alpha$, δ^α) For any (finite or infinite) cardinal α, for every formula ϕ and for every variable x, let $\delta^\alpha x\phi$ and $\forall^\alpha x\phi$ be formulas with $FV(\delta^\alpha x\phi) = FV(\phi) \cup \{x\}$, $FV(\forall^\alpha x\phi) = FV(\phi) \backslash \{x\}$ and truth conditions*

$$M \models_X \delta^\alpha x\phi \Leftrightarrow \forall A \subseteq^\alpha M, M \models_{X_{x \in A}} \phi;$$
$$M \models_X \forall^\alpha x\phi \Leftrightarrow \forall A \subseteq^\alpha M, M \models_{X[A/x]} \phi$$

where $A \subseteq^\alpha M$ is an abbreviation for "$A \subseteq M$ and $|A| \leq \alpha$", $X_{x \in A} = \{s \in X : s(x) \in A\}$ and $X[A/x] = \{s[m/x] : s \in X, m \in A\}$.

Again, we can define uniformly δ^α by means of $\forall^\alpha$ and vice versa:

Proposition 3.5. *For all cardinals α, formulas $\phi \in \mathcal{D}$, variables x and teams X,*

$$\forall^\alpha x \phi \equiv \forall x \delta^\alpha x \phi$$

and

$$\delta^\alpha x \phi \equiv \forall^\alpha y (y \neq x \vee \phi).$$

Proof.

$$M \models_X \forall x \delta^\alpha x \phi \Leftrightarrow M \models_{X[M/x]} \delta^\alpha x \Leftrightarrow \forall A \subseteq^\alpha M, M \models_{X[M/x]_{x \in A}} \phi \Leftrightarrow$$
$$\Leftrightarrow \forall A \subseteq^\alpha M, M \models_{X[A/x]} \Leftrightarrow M \models_X \forall^\alpha x \phi$$

and

$$M \models_X \forall^\alpha y (y \neq x \vee \phi) \Leftrightarrow \forall A \subseteq^\alpha M, M \models_{X[A/y]} (y \neq x \vee \phi) \Leftrightarrow \forall A \subseteq^\alpha M, M \models_{X[A/y]_{y=x}} \phi$$
$$\Leftrightarrow \forall A \subseteq^\alpha M, M \models_{X_{x \in A}} \phi \Leftrightarrow M \models_X \delta^\alpha x \phi,$$

where we used the fact that $y \notin FV(\phi)$. □

Corollary 3.2. *For all α, $\mathcal{D}(\forall^\alpha) \bowtie \mathcal{D}(\delta^\alpha)$.*

Furthermore, for every $n \in \mathbb{N}_0$,[7] $\mathcal{D}(\forall^n)$ is uniformly translatable into $\mathcal{D}(\forall^1)$ and $\mathcal{D}(\delta^1)$ is uniformly translatable into $\mathcal{D}(\delta^n)$:

Proposition 3.6. *For every $n \in \mathbb{N}_0$ and for all $\phi \in \mathcal{D}$ such that $FV(\phi) \cap \{x_1 \ldots x_n\} = \emptyset$,*

$$\forall^n x \phi \equiv \forall^1 x_1 \ldots \forall^1 x_n \forall x (\bigwedge_{i=1}^{n} (x \neq x_i) \vee \phi).$$

Proof.

$$M \models_X \forall^1 x_1 \ldots \forall^1 x_n \forall x (\bigwedge_{i=1}^{n} (x \neq x_i) \vee \phi) \Leftrightarrow \forall \overline{m} \in M^n, M \models_{X[M/x][\overline{m}/\overline{x}]} \bigwedge_{i=1}^{n} (x \neq x_i) \vee \phi \Leftrightarrow$$
$$\Leftrightarrow \forall \overline{m}_1 \ldots m_n \in M, M \models_{X[M/x]_{x \in \{m_1 \ldots m_n\}}} \phi \Leftrightarrow \forall A \subseteq^n B, M \models_{X[A/x]} \phi \Leftrightarrow M \models_X \forall^n x \phi.$$

□

Proposition 3.7. *For every $n \in \mathbb{N}_0$ and for every formula $\phi \in \mathcal{D}$,*

$$\delta^1 x \phi \equiv \delta^n x (\underbrace{(=(x) \wedge \phi) \vee \ldots \vee (=(x) \wedge \phi)}_{n \text{ times}}).$$

[7]For $n = 0$ we have that $\delta^0 x \phi \equiv \top$ and $\forall^0 x \phi \equiv \forall x \top$. These two operators will not be further investigated in this paper.

Proof.

$$
\begin{aligned}
&M \models_X \delta^n x(\underbrace{(=(x) \wedge \phi) \vee \ldots \vee (=(x) \wedge \phi)}_{n \text{ times}}) \Leftrightarrow \\
&\Leftrightarrow \forall A \subseteq^n M, M \models_{X_{x \in A}} (\underbrace{(=(x) \wedge \phi) \vee \ldots \vee (=(x) \wedge \phi)}_{n \text{ times}}) \Leftrightarrow \\
&\Leftrightarrow \forall A \subseteq^n M, X_{x \in A} = X_1 \cup \ldots \cup X_n \text{ s.t. } M \models_{X_i} =(x) \wedge \phi \text{ for all } i = 1 \ldots n \Leftrightarrow \\
&\Leftrightarrow \forall m_1 \ldots m_n \in M, M \models_{X_{x=m_i}} \phi \text{ for all } i = 1 \ldots n \Leftrightarrow M \models_X \delta^1 x \phi
\end{aligned}
$$

□

Corollary 3.3. *For every $n \in \mathbb{N}_0$, $\mathcal{D}(\forall^n), \mathcal{D}(\delta^n) \bowtie \mathcal{D}(\delta^1)$.*

However, this changes if we consider operators of the form $\forall^\alpha x \phi$, where α is an infinite cardinal:

Proposition 3.8. *For any infinite α, $\mathcal{D}(\forall^\alpha) \not\leq \mathcal{D}$.*

Proof. For every model M, $M \models_{\{\emptyset\}} \forall^\alpha x \exists y(=(y) \wedge x \neq y)$ if and only if $|M| > \alpha$.

But $\mathcal{D}$ and all logics semantically equivalent to it satisfy the Löwenheim-Skolem Theorem ([15], §6.2), and therefore $\mathcal{D}(\forall^\alpha) \not\leq \mathcal{D}$. □

Corollary 3.4. *For any infinite α, $\mathcal{D}(\forall^\alpha), \mathcal{D}(\delta^\alpha)$ is not uniformly translatable into $\mathcal{D}(\delta^1)$.*

4 Announcements in Game Theoretic Semantics

Apart from the above mentioned Team Semantics, Dependence Logic also has an equivalent Game Theoretic Semantics, which can be seen as a natural adaptation to the formalism of Dependence Logic of the game theoretic semantics of imperfect information developed for Henkin Quantifiers ([6]) and further developed in the context of Independence Friendly logic ([7]).

The game can be defined in the following way ([15], §5.3):

Definition 4.1 (The game $H^M_X(\phi)$)**.** *Let M be a first order model, let X be a team, and let ϕ be a formula in $\mathcal{D}$ with $FV(\phi) \subseteq Dom(X)$.*

Then the game $H^M_X(\phi)$ is defined as follows, where P is the set of all game positions of $H^M_X(\phi)$, $S : P \to \mathcal{P}(P)$ sends each position p into the set of all its possible successors, $T : P \to \{I, II\}$ indicates which one of the two players is moving from the position p, Ip is the set of the initial *positions and W is the set of the* winning *positions for Player II:*

- *The set of all positions of the game $H^M_X(\phi)$ is $P = \{(\psi, s) : \psi$ is an instance[8] of a subformula of ϕ and s is an assignment with $FV(\psi) \subseteq Dom(s)\}$;*
- *The set of all initial positions for $H^M_X(\phi)$ is $Ip = \{(\phi, s) : s \in X\}$;*

[8] This is in order to avoid confusion between different instances of the same expression, like the two dependence atomic subformulas of $=(x) \vee =(x)$. We may tacitly assume that no such confusion is possible, for example because an unique index has been associated to each atomic formula.

- *Given a position p, the active player $T(p)$ and the set of possible moves $S(p)$ are defined as follows:*
  - *If p is a literal, $T(p)$ is undefined and $S(p) = \emptyset$;*
  - *If $p = (\phi \vee \psi, s)$ then $T(p) = II$ and $S(p) = \{(\phi, s), (\psi, s)\}$;*
  - *If $p = (\phi \wedge \psi, s)$ then $T(p) = I$ and $S(p) = \{(\phi, s), (\psi, s)\}$;*
  - *If $p = (\exists x\psi, s)$ then $T(p) = II$ and $S(p) = \{(\psi, s[m/x]) : m \in M\}$;*
  - *If $p = (\forall x\psi, s)$ then $T(p) = I$ and $S(p) = \{(\psi, s[m/x]) : m \in M\}$;*
- *The set W of winning positions for Player II is defined as*

$$W = \{(\phi, s) : \phi \text{ is a first order literal and } s \models \phi\} \cup \\ \cup \{(\phi, s) : \phi \text{ is a non-negated dependence atomic formula}\}.$$

If $S(p) = \emptyset$ and $p \notin W$ we say that p is a losing *position for Player II, or equivalently a* winning *position for Player I.*

Definition 4.2 (Plays, complete plays, winning plays)**.** *Let M, X and ϕ be as above. A finite sequence $\overline{p} = p_1 \dots p_n$ of positions of $H^M_X(\phi)$ is a* play *of $H^M_X(\phi)$ if the following two conditions hold:*

1. *p_1 is in the set Ip of all initial positions of the game;*
2. *For all $i \in 1 \dots n-1$, $p_{i+1} \in S(p_i)$.*

If furthermore p_n is a terminal *position, that is, $S(p_n) = \emptyset$, we say that $\overline{p}$ is a* complete *play. Then, if $p_n \in W$ we say that $\overline{p}$ is a* winning *play; if instead p_n is terminal but not in W, we say that $\overline{p}$ is a* losing *play.*

Definition 4.3 (Strategy, plays following a strategy, winning strategy)**.** *A* strategy *for Player $\alpha \in \{I, II\}$ in $H^M_X(\phi)$ is a function σ from partial plays $\overline{p} = p_1 \dots p_i$ of $H^M_X(\phi)$ to positions $\sigma(\overline{p}) \in S(p_i)$.*

A play $\overline{p} = p_1 \dots p_n$ is said to follow *a strategy σ for Player $P \in \{I, II\}$ if, for all $i = 1 \dots n-1$, $T(p_i) = P \Rightarrow p_{i+1} = \sigma(p_1 \dots p_i)$.*

A strategy σ is winning *for Player P in $H^M_X(\phi)$ if and only if all complete plays in which P follows σ are winning for P.*

Definition 4.4 (Uniform strategy)**.** *A strategy σ for Player II in $H^M_X(\phi)$ is* uniform *if for every two complete plays $\overline{p} = p_1 \dots p_n, \overline{p'} = p'_1 \dots p'_n$ in which II followed σ, if the last positions are*

$$p_n = (=(t_1 \dots t_n), s); \qquad p'_n = (=(t_1 \dots t_n), s')$$

for the same instance of $=(t_1 \dots t_n)$ and $t_i\langle s\rangle = t_i\langle s'\rangle$ for $i = 1 \dots n-1$ then $t_n\langle s\rangle = t_n\langle s'\rangle$.

The following theorem the main result of Section 5.3 of [15]:

Theorem 4.1 ([15])**.** *Let M be a first order model, let ϕ be a Dependence Logic formula and let X be a team such that $FV(\phi) \subseteq Dom(X)$. Then $M \models_X \phi \Leftrightarrow$ Player II has a uniform winning strategy for $H^M_X(\phi)$.*

Proof. By structural induction over ϕ. $\square$

We now wish to extend this game semantics to $\mathcal{D}(\delta^1)$.

Definition 4.5 (The game ${}^1H^M_X(\phi)$)**.** *Let M be a first order model, let X be a team, and let ϕ be a formula of $\mathcal{D}(\delta^1)$ with $FV(\phi) \subseteq Dom(\phi)$. Then the game ${}^1H^M_X(\phi)$ is defined precisely as the game $H^M_X(\phi)$, with the following additional rule for the δ^1 operator:*

- *If $p = (\delta^1 x\psi, s)$, $T(p) = I$ and $S(p) = \{(\psi, s)\}$.*[9]

Starting positions, winning positions, plays, strategies and winning strategies are defined precisely as in $H^M_X(\phi)$.

However, there is a difference in the definition of uniform strategy:

Definition 4.6 (δ^1-similar plays)**.** *Let $\overline{p} = p_1 \ldots p_n$ and $\overline{p}' = p'_1 \ldots p'_{n'}$ be two plays of ${}^1H^M_X(\phi)$. Then we say that $\overline{p}$ and $\overline{p}'$ are* δ^1-similar*, and we write $\overline{p} \approx^{\delta^1} \overline{p}'$, if for all $i \in 1 \ldots n$, $i' \in 1 \ldots n'$ such that*

$$p_i = (\delta^1 x\psi, s); \qquad\qquad p'_{i'} \in (\delta^1 x\psi, s')$$

for the same instance of $\delta^1 x\psi$, we have that $s(x) = s'(x)$.

Then we limit the scope of the uniformity condition to δ^1-similar plays:

Definition 4.7 (Uniform strategies for ${}^1H^M_X(\phi)$)**.** *A strategy σ for Player II in ${}^1H^M_X(\phi)$ is* uniform *if for every two complete,* δ^1-similar *plays $\overline{p} = p_1 \ldots p_n$, $\overline{p'} = p'_1 \ldots p'_n$ in which II followed σ, if the last positions are*

$$p_n = (=(t_1 \ldots t_n), s); \qquad\qquad p'_n = (=(t_1 \ldots t_n), s')$$

for the same instance of $=(t_1 \ldots t_n)$ and $t_i\langle s\rangle = t_i\langle s'\rangle$ for $i = 1 \ldots n-1$ then $t_n\langle s\rangle = t_n\langle s'\rangle$.

The following theorem shows that the game theoretic semantics for $\mathcal{D}(\delta^1)$ induced by the games ${}^1H^M_X(\phi)$ is equivalent to the team semantics for the same logic:

Theorem 4.2. *Let M be a first order model, let ϕ be a formula of $\mathcal{D}(\delta^1)$ and let X be a team such that $FV(\phi) \subseteq Dom(X)$. Then $M \models_X \phi \Leftrightarrow$ Player II has a uniform winning strategy for ${}^1H^M_X(\phi)$.*

Proof. The proof is by structural induction over ϕ. All cases except δ^1 are treated precisely as in ([15], Theorem 5.16), and we refer to it for their proof.

Suppose that $M \models_X \delta^1 x\phi$. Then, by definition, for all $m \in M$ we have that $M \models_{X_{x=m}} \phi$. By induction hypothesis, this implies that for every $m \in M$ there exists an uniform winning strategy σ^m for Player II in ${}^1H^M_{X_{x=m}}(\phi)$. Then define the strategy σ for Player II in ${}^1H^M_X(\delta^1 x\phi)$ as follows:

- If $\overline{p} = p_1 \ldots p_k$ for $k > 1$ and $p_1 = (\delta^1 x\phi, s)$ for some $s \in X$, let $\sigma(p_1 \ldots p_k) = \sigma^{s(x)}(p_2 \ldots p_k)$.

[9] Since there is only one possible successor to a position of the form $(\delta^1 x\psi, s)$, it makes no difference at all whether the active player is I or II. However, this will not be the case for the variant ${}^\alpha H^M_X(\phi)$ for the $\mathcal{D}(\delta^\alpha)$ operator which will be presented next.

This strategy is winning, since for any complete play $\overline{p} = p_1 \ldots p_n$ of ${}^1H^M_X(\phi)$ in which $p_1 = (\delta^1 x, s)$ and II follows σ it holds that $p_2 \ldots p_n$ is a complete play of ${}^1H^M_{X_{x=s(x)}}$ in which II follows $\sigma^{s(x)}$; and it is also uniform, since for any two δ^1-similar complete plays

$$\overline{p} = p_1 \ldots p_n; \qquad \overline{p}' = p'_1 \ldots p'_{n'}$$

where $p_1 = (\delta^1 x\psi, s)$ and $p'_1 = (\delta^1 x\psi, s')$ it must hold by the definition of δ^1-similarity that $s(x) = s'(x) = m$ for some $m \in M$. Hence, by the definition of the strategy σ, both $p_2 \ldots p_n$ and $p'_2 \ldots p'_{n'}$ are complete plays in ${}^1H^M_{X_{x=m}}(\phi)$ where II followed the uniform strategy σ^m, and hence they satisfy the uniformity condition.

Conversely, suppose that II has a uniform winning strategy σ in ${}^1H^M_X(\delta^1 x\phi)$, let $m \in M$, and define the strategy σ^m for Player II in ${}^1H^M_{X_{x=m}}(\phi)$ as follows: for all partial plays $p_1 \ldots p_k$, where $p_1 = (\phi, s)$ for some $s \in X$ with $s(x) = m$, define

$$p_0 = (\delta^1 x\phi, s)$$

and

$$\sigma^m(p_1 \ldots p_k) = \sigma(p_0 \ldots p_k).$$

Then each σ^m is a winning strategy for ${}^1H^M_{X_{x=m}}(\phi)$, because σ itself is winning and each complete play of ${}^1H^M_{X_{x=m}}(\phi)$ in which II follows σ^m is included in a complete play for ${}^1H^M_X(\delta^1 x\phi)$ in which II follows σ; and furthermore, it is uniform, because any two plays $\overline{p}$ and $\overline{p}'$ of ${}^1H^M_{X_{x=m}}(\phi)$ in which II follows σ^m are included in two complete plays $p_0\overline{p}$ and $p'_0\overline{p}'$ of ${}^1H^M_X(\delta^1 x\phi)$ in which II follows σ and $p_0 = (\delta^1 x\phi, s)$, $p'_0 = (\delta^1 x\phi, s)$ for two s, s' with $s(x) = s'(x) = m$.

Hence, if $\overline{p}$ and $\overline{p}'$ are δ^1-similar over ${}^1H^M_{X_{x=m}}(\phi)$ then $p_0\overline{p}$ and $p'_0\overline{p}'$ are δ^1-similar over ${}^1H^M_X(\delta^1 x\phi)$, and in conclusion they satisfy the uniformity condition. □

The above described game theoretic semantics is the main reason why $\delta^1 x$ can be called a "announcement operator": anthropomorphizing somewhat the two agents of the game, one might think of $\delta^1 x\phi$ as the subgame in which first the value of x is announced from Player I to Player II and then the game corresponding to ϕ is played, but Player II is allowed to act accordingly to the information that has been broadcasted (that is, the value of x) even though she would not otherwise have access to it (for example, because she is selecting a y and there is a position of the form $=(y)$ which Player I can reach later in the play).

This suggests a way to find a game theoretic semantics for the logics $\mathcal{D}(\delta^\alpha)$:

Definition 4.8 (The game ${}^\alpha H^M_X(\phi)$)**.** *Let α be a (finite or infinite) cardinal, let M be a first order model, let X be a team over M, and let ϕ be a formula of $\mathcal{D}(\delta^\alpha)$ with $FV(\phi) \subseteq Dom(X)$.*

Then the game ${}^\alpha H^M_X(\phi)$ is defined precisely as $H^M_X(\phi)$, but with the following changes:

- *Positions are not pairs (ψ, s), but triples (ψ, s, A), where A is a* annotation *in the form of a set of elements of the model. The starting position is $(\psi, s, \emptyset)$, and all previously introduced rules generate the empty annotation for the next position - for example, if $p = (\exists x\psi, s, A)$ then $T(p) = II$ and $S(p) = \{(\psi, s[m/x], \emptyset) : m \in M\}$, and so on;*

- *If p is $(\delta^\alpha x\psi, s, A)$ then $T(p) = I$ and $S(p) = \{(\psi, s, B) : |B| \leq \alpha$ and $s(x) \in B\}$.*

Again, the notions of play, complete play, winning play and winning strategy are left unchanged, except of course that now the positions are triples rather than pairs (and because of this, for example, Player II might make different choices depending on which annotation Player I introduced after a δ^α position).

However, the concept of uniform strategy requires again some modifications:

Definition 4.9 (δ^α-similar plays)**.** *Let $\overline{p} = p_1 \dots p_n$ and $\overline{p'} = p'_1 \dots p'_{n'}$ be two plays of ${}^\alpha K^M_X(\phi)$. Then $\overline{p}$ and $\overline{p'}$ are* δ^α-similar *if and only if for all $i \in 1 \dots n-1$ and $j \in 1 \dots n'-1$, if*

$$p_i = (\psi, s, A); \qquad p'_j = (\psi, s', B)$$

for the same instance of ψ then the annotations A and B are the same.

In this case, we write that $p \approx^{\delta^\alpha} p'$.

Definition 4.10 (Uniform strategies for ${}^\alpha K^M_X(\phi)$.)**.** *A strategy σ for Player II in ${}^\alpha K^M_X(\phi)$. is* uniform *if for every two complete,* δ^α-similar *plays $\overline{p} = p_1 \dots p_n, \overline{p'} = p'_1 \dots p'_n$ in which II followed σ, if the last positions are*

$$p_n = (=(t_1 \dots t_n), s, A); \qquad p'_n = (=(t_1 \dots t_n), s', A)$$

for the same instance of $=(t_1 \dots t_n)$ and $t_i\langle s\rangle = t_i\langle s'\rangle$ for $i = 1 \dots n-1$ then $t_n\langle s\rangle = t_n\langle s'\rangle$.

Theorem 4.3. *Let M be a first order model, let ϕ be a formula of $\mathcal{D}(\delta^\alpha)$ and let X be a team such that $FV(\phi) \subseteq Dom(X)$. Then $M \models_X \phi \Leftrightarrow$ Player II has a uniform winning strategy for ${}^\alpha K^M_X(\phi)$.*

Proof. We proceed by structural induction on ϕ, and again all cases except the one for $\delta^\alpha x$ are dealt with precisely as in ([15], Theorem 5.16).

Suppose that $M \models_X \delta^\alpha x\psi$: then, by definition, we have that for all subsets $A \subseteq^\alpha M$ it holds that $M \models_{X_{x\in A}} \psi$. Then, by induction hypothesis, for each such A there exists a uniform winning strategy σ^A for Player II in ${}^\alpha H^M_{X_{x\in A}}(\phi)$. Let us define the strategy σ for Player II in ${}^\alpha H^M_X(\delta^\alpha x\phi)$ as the one that, whenever Player I selects a set A as annotation in the first move of the game, plays the rest of the game according to $\sigma^{A'}$ for $A' = A \cap \{s(x) : s \in X\}$. This strategy is winning, as each play contains a play of ${}^\alpha H^M_{X_{x\in A'}}(\phi)$ in which II is using $\sigma^{A'}$, and it is uniform, as any two δ^α-similar plays must have the same annotation in the second position and therefore must be played according to the same $\sigma^{A'}$, which we know by hypothesis to be uniform.

Conversely, suppose that II has a winning strategy in ${}^\alpha H^M_X(\delta^\alpha x\phi)$: then for each $A \subseteq^\alpha M$ such that $\{s(x) : s \in X\} \cap A \neq \emptyset$ Player II has a winning strategy in ${}^\alpha H^M_{X_{x\in A}}(\phi)$, and hence by induction hypothesis $M \models_{X_{x\in A}} \phi$ for all such A. If instead $|A| \leq \alpha$ and $\{s(x) : s \in X\} \cap A = \emptyset$ then $M \models_{X_{x\in A}} \phi$ trivially, and therefore $M \models_{X_{x\in A}} \phi$ for all A with $|A| \leq \alpha$.

So, in conclusion, $M \models_X \delta^\alpha x\psi$, as required. □

Again, the intuition is that of an announcement, but this time it is a *partial* one: when encountering $\delta^\alpha x\psi$, Player I does not allow Player II access to the value of x, but he chooses a set A of cardinality α and gives her the (true) information that $x \in A$.

This concept of partial announcement could be taken further, and in many different ways: here, we will only describe another case which may be of some interest.

Let us consider operators of the form

$$\delta^\theta \psi := \theta \to \psi$$

where $\to$ is the implication of Intuitionistic Dependence Logic and θ is a formula of Dependence Logic.

Then is not difficult to adapt our game theoretic semantics to the resulting logic $\mathcal{D}(\theta \to)$: the rules are precisely as in ${}^{\alpha}H_X^M(\phi)$, except that now annotations consist of teams rather than sets and the rule for δ^θ is

- If p is $(\delta^\theta\psi, s, A)$ then $T(p) = I$ and $S(p) = \{(\psi, s, X) : M \models_X \theta \text{ and } s \in X\}$.

Then, we obtain a game ${}^{\theta\to}H_X^M(\phi)$ such that the following holds:

Theorem 4.4. *Let M be a first order model, let ϕ be a formula of $\mathcal{D}(\theta \to)$ and let X be a team such that $FV(\phi) \subseteq Dom(X)$. Then $M \models_X \phi \Leftrightarrow$ Player II has a uniform winning strategy for ${}^{\theta\to}H_X^M(\phi)$, where δ^θ is interpreted as the intuitionistic implication:*[10]

$$M \models_X \delta^\theta\psi \Leftrightarrow \text{ for all } Y, \text{ if } M \models_Y \theta \text{ then } M \models_{X\cap Y} \psi.$$

Proof. Suppose that $M \models_X \delta^\theta\psi$: then, by definition, if Y satisfies θ then $X \cap Y$ satisfies ψ. Thus, for each such team Y there exists a uniform winning strategy σ^Y for II in ${}^{\theta\to}H_{X\cap Y}^M(\psi)$: by gluing them together as before, we obtain a winning strategy σ for II in ${}^{\theta\to}H_X^M(\delta^\theta\psi)$. This strategy is uniform: indeed, each σ^Y is uniform and plays corresponding to different σ^Y are never δ^θ-similar, as they have different annotations.

Conversely, suppose that II has a uniform winning strategy in ${}^{\theta\to}H_X^M(\delta^\theta\psi)$: then for each team Y such that $M \models_Y \theta$ Player II has a uniform winning strategy in ${}^{\theta\to}H_{X\cap Y}^M(\psi)$. By induction hypothesis, this means that for each such Y $M \models_{X\cap Y} \psi$, and hence by definition $M \models_X \delta^\theta\psi$. $\square$

This last result can be seen as a partial answer to the question whether Intuitionistic Dependence Logic admits a natural game-theoretic semantics. However, the fact that the satisfaction relation $\models$ is part of the game theoretic semantics for the δ^θ operator hampers severely the finitistic feel which a "good" game theoretic semantics may be required to have: in particular, in the case of nested announcement operators $\delta^{\delta^\phi\psi}\theta$ of $\mathcal{D}(\phi \to)((\delta^\phi \to \psi) \to)$,[11] or of the equivalent intuitionistic dependence logic formula $(\phi \to \psi) \to \theta$, our game theoretic semantics asks us to stack several layers of calculations.

Indeed, the game rules for ${}^{\phi\to\psi}H_X^M((\phi \to \psi) \to \theta)$ refer to the existence of winning strategies for Player II in games of the form ${}^{\phi}H_Y^M(\phi \to \psi)$, whose rules on the other hand refer to the existence of winning strategies for Player II in games of the form $H_Y^M(\phi)$.

[10] This is not the standard definition of intuitionistic implication, but it is easily seen to be equivalent to it modulo the downwards closure property of Dependence Logic and Intuitionistic Dependence Logic.

[11] With this name, we indicate the logic obtained by first adding the δ^ϕ operator to $\mathcal{D}$, and then adding the operator $\delta^{\delta^\phi\psi}$ to the resulting logic.

This may become difficultly manageable even for relatively short formulas of Intuitionistic Dependence Logic: it is not known to the author at the moment whether this is an unavoidable consequence of the higher expressive power of Intuitionistic Dependence Logic or if more straightforward games exist for its semantics.

5 An Ehrenfeucht-Fraïssé game for $\mathcal{D}(\sqcup, \forall^\alpha)$

In ([15], §6.6), the following *semiequivalence* relation between models was introduced:

Definition 5.1 ($\Rrightarrow$)**.** *Let M, N be two models, and let X, Y be teams over M and N respectively. Then $(M, X) \Rrightarrow (N, Y)$ if and only if*

$$M \models_X \phi \Rightarrow N \models_Y \phi$$

for all Dependence Logic formulas ϕ.

In this section, I will adapt the Ehrenfeucht-Fraïssé game for Dependence Logic to $\mathcal{D}(\sqcup, \forall^\alpha)$, where $\sqcup$ is the "classical disjunction" defined as

$$M \models_X \phi \sqcup \psi \Leftrightarrow M \models_X \phi \text{ or } M \models_X \psi.$$

This operation can be uniformly defined in Dependence Logic as

$$\phi \sqcup \psi := \exists x_1 \exists x_2 (=\!(x_1) \wedge =\!(x_2) \wedge ((x_1 = x_2 \wedge \phi) \vee (x_1 \neq x_2 \wedge \psi))),$$

assuming that we are working on models with at least two elements,[12] and as $\sim ((\sim \phi) \wedge (\sim \psi))$ in Team Logic ([16], [15]).

It also corresponds to the "blind disjunction" $\vee_0$ of IF-Logic ([7]), and it can be given a game semantics by adding to our uniformity condition the requirement that whenever two positions $(\phi \sqcup \psi, s)$ and $(\phi \sqcup \psi, s')$ are reached during the course of two plays for the same instance of $\phi \sqcup \psi$, Player II chooses the same disjunct for both plays.

The following definitions are the obvious modifications of those of ([15], §6.6) :

Definition 5.2 ($qr(\phi)$)**.** *Let $\phi \in \mathcal{D}(\sqcup, \forall^\alpha)$. Then its rank $qr(\phi)$ is defined inductively as follows:*

- *If ϕ is a literal, $qr(\phi) = 0$;*
- $qr(\phi \vee \psi) = \max(qr(\phi), qr(\psi)) + 1$*;*
- $qr(\phi \wedge \psi) = \max(qr(\phi), qr(\psi))$*;*
- $qr(\exists x \psi) = qr(\psi) + 1$*;*

[12] If we want to also consider one- or zero-element, we may just define

$$\phi \sqcup \psi := (\forall x_1 x_2 (x_1 = x_2) \wedge (\phi \vee \psi)) \vee \exists x_1 \exists x_2 (=\!(x_1) \wedge =\!(x_2) \wedge ((x_1 = x_2 \wedge \phi) \vee (x_1 \neq x_2 \wedge \psi))) :$$

indeed, if there are less than two elements then all teams with the same domain are the same, and hence $\phi \vee \psi$ and $\phi \sqcup \psi$ are equivalent on these models.

- $qr(\forall x\psi) = qr(\psi) + 1$;
- $qr(\forall^\alpha x\psi) = qr(\psi) + 1$;
- $qr(\phi \sqcup \psi) = \max(qr(\phi), qr(\psi))$.

Definition 5.3 ($\mathcal{D}_n(\sqcup, \forall^\alpha)$).

$$\mathcal{D}_n(\sqcup, \forall^\alpha) = \{\phi : \phi \text{ is a formula of } \mathcal{D}(\sqcup, \forall^\alpha) \text{ and } qr(\phi) \leq n\}.$$

Definition 5.4 ($\Rrightarrow^\alpha$). *Let M, N be two models, and let X, Y be teams over M and N respectively. Then $(M, X) \Rrightarrow^\alpha (N, Y)$ if and only if*

$$M \models_X \phi \Rightarrow N \models_Y \phi$$

for all formulas $\phi \in \mathcal{D}(\sqcup, \forall^\alpha)$.

Definition 5.5 ($\Rrightarrow^\alpha_n$). *Let M, N be two models, and let X, Y be teams over M and N respectively. Then $(M, X) \Rrightarrow^\alpha_n (N, Y)$ if and only if*

$$M \models_X \phi \Rightarrow N \models_Y \phi$$

for all formulas $\phi \in \mathcal{D}(\alpha)$ such that $qr(\phi) \leq n$.

Lemma 5.1. *Let M, N, X and Y be as above. Then $(M, X) \Rrightarrow^\alpha (N, Y)$ if and only if $(M, X) \Rrightarrow^\alpha_n (N, Y)$ for all $n \in \mathbb{N}$.*

The following proposition is proved analogously to the corresponding result ([15], Proposition 6.48):

Proposition 5.1. *A class of models K with assignments in a fixed domain V is definable in $\mathcal{D}_n(\sqcup, \delta^\alpha)$ if and only if it is closed under $\Rrightarrow^\alpha_n$.*

Proof. Suppose that K is $\{(M, X) : Dom(X) = V, M \models_X \phi\}$ for some formula $\phi \in \mathcal{D}_n(\sqcup, \delta^\alpha)$. Then, if $(M, X) \in K$ and $(M, X) \Rrightarrow^\alpha_n (N, Y)$ then $N \models_Y \phi$ too and hence $(N, Y) \in K$: therefore, K is closed under the $\Rrightarrow^\alpha_n$ relation.

Conversely, suppose that K is closed under the $\Rrightarrow^\alpha$ relation: then for every model $(M, X) \in K$ and for every $(N, Y) \notin K$ there exists a formula $\phi_{MX,NY}$ of rank $\leq n$ such that $M \models_X \phi_{MX,NY}$ but $N \not\models_Y \phi_{MX,NY}$. Then consider the formula

$$\phi := \bigsqcup_{(M,X)\in K} \bigwedge_{(N,Y)\notin K} \phi_{MX,NY}.$$

As there exist only finitely many logically different formulas $\psi \in \mathcal{D}_n(\sqcup, \delta^\alpha)$ with $FV(\psi) \subseteq V$, the conjunction and the classical disjunction in ϕ are finite and $\phi \in \mathcal{D}_n(\sqcup, \delta^\alpha)$.

Furthermore, $K = \{(M, X) : M \models_X \phi\}$. Indeed, if $(M, X) \in K$ then for all $(N, Y) \notin K$ it holds that $M \models_X \phi_{MX,NY}$, and if $(N, Y) \notin K$ then $N \not\models_Y \phi_{MX,NY}$ for any $(M, X) \in K$. ☐

Then, for $n \in \mathbb{N}$, we can define the $EF^\alpha_n(M, X, N, Y)$ game as follows:

Definition 5.6 ($EF_n^\alpha(M, X, N, Y)$)**.** *Let M, N be two models, let X, Y be teams with the same domain over M and N respectively, let α be any (finite or infinite) cardinal and let $n \in \mathbb{N}$. Then the game $EF_n^\alpha(M, X, N, Y)$ is defined as follows:*

- *There are two players, called $\forall$ (Abelard) and $\exists$ (Eloise);*
- *$x_1 \dots x_n$ are variables which do not occur in $Dom(X) = Dom(Y)$.*
- *The set P of all positions of the game is $\{(X^i, Y^i, i) : X^i$ is a team on M, Y^i is a team on N and $i \in 1 \dots n\}$;*
- *The starting position is $(X, Y, 0)$;*
- *For each position (X^i, Y^i, i) with $i < n$, Player $\forall$ decides which kind of move to play, among the following:*[13]

  Splitting: *$\forall$ chooses teams X' and X'' with $X' \cup X'' = X^i$. Then $\exists$ chooses teams Y' and Y'' with $Y' \cup Y'' = Y^i$, and $\forall$ decides whether the next position is $(X', Y', i + 1)$ or $(X'', Y'', i + 1)$;*

  Supplementation: *$\forall$ chooses a function $F : X^i \to M$. Then $\exists$ chooses a function $G : Y^i \to N$, and the next position is $(X^i[F/x_i], Y^i[G/x_i], i + 1)$;*

  Duplication: *The next position is $(X^i[M/x_i], Y^i[N/x_i], i + 1)$;*

  Right-α-duplication: *$\forall$ chooses a set of elements $B \subseteq^\alpha N$. Then $\exists$ chooses a set of elements $A \subseteq^\alpha M$, and the next position is $(X^i[A/x_i], Y^i[B/x_i], i + 1)$.*

- *The set of all winning positions for Player $\exists$ is $W = \{(X_n, Y_n, n) : (M, X_n) \Rrightarrow_0^\alpha (N, Y_n)\} = \{(X_n, Y_n, n) : M \models_{X_n} \phi \Rightarrow N \models_{Y_n} \phi$ for all literals $\phi\}$.*

The concepts of play, complete play, strategy and winning strategy are defined in the obvious way, and there is no uniformity condition for this game.

Theorem 5.1. *Let M, N, X and Y as above, and let $n \in \mathbb{N}$. Then $(M, X) \Rrightarrow_n^\alpha (N, Y)$ if and only if Player $\exists$ has a winning strategy for $EF_n^\alpha(M, X, N, Y)$.*

Proof. As all cases except right-α-duplication, classical disjunction and the δ^α operator are dealt with precisely as in ([15], Theorem 6.44), we will only take care of these two.

The left to right direction is proved by induction over n, and by considering all possible first moves of $\forall$.

Suppose that $(M, X) \Rrightarrow_n^\alpha (N, Y)$, and let Player $\forall$ make a right-α-duplication move and choose a set $B \subseteq^\alpha N$. Then there exists a set $A \subseteq^\alpha M$ such that $(M, X[A/x_i]) \Rrightarrow_{n-1}^\alpha (N, Y[B/x_i])$:

[13] In order to be entirely formal, we should define $T(X^i, Y^i, i) = \forall$ for all such positions, introduce next positions corresponding to the four possible choices of $\forall$, then add new positions for the subgames for splitting, supplementation and selection. As this would increase the complexity of the notation and lead to no real advantage, we will content ourselves with a somewhat informal definition here: the reader will be able to see without difficulty how this could be made more exact.

indeed, suppose instead that for each such set A there exists a formula ϕ^A of rank $\leq n-1$ such that $M \models_{X[A/x_i]} \phi^A$ but $N \not\models_{Y[B/x_i]} \phi^A$, and consider

$$\phi = \bigsqcup_{A \subseteq M, |A| \leq \alpha} \phi^A.$$

Then $qr(\forall^\alpha x_i \phi) \leq n$ and $M \models_X \forall^\alpha x_i \phi$; but since $(M, X) \Rrightarrow_n^\alpha (N, Y)$ this implies that $M' \models_Y \forall^\alpha x_i \phi$, and thus in particular $M' \models_{X[B/x_i]} \phi$ and thus $N \models_{X[B/x_i]} \phi^A$ for some A. But this is not possible, and thus there exists an A_0 such that $(M, X[A_0/x_i]) \Rrightarrow_{n-1}^\alpha (N, Y[B/x_i])$. By induction hypothesis, this implies that Player $\exists$ has a winning strategy in $EF_{n-1}^\alpha(M, X[A_0/x_i], N, Y[B/x_i])$, and thus she can win the current play by choosing A_0 and then playing according to this winning strategy.

For the right to left direction, we assume that Player $\exists$ has a winning strategy in $EF_n^\alpha(M, X, N, Y)$ and we prove, by structural induction on ϕ, that if $qr(\phi) \leq n$ and $M \models_X \phi$ then $N \models_Y \phi$ too.

Suppose that ϕ is of the form $\psi \sqcup \theta$, where $qr(\phi) = \max(qr(\psi), qr(\theta)) \leq n$ and $M \models_X \phi$. Then by the definition of the classical disjunction, $M \models_X \psi$ or $M \models_X \theta$: let us assume, without loss of generality, that $M \models_X \psi$. Then, by our induction hypothesis[14], $N \models_Y \psi$, and hence $N \models_Y \psi \sqcup \theta$ too.

If instead ϕ is of the form $\forall^\alpha x_i \psi$ and $M \models_X \phi$ then, by definition, for all subsets $A \subseteq M$ such that $|A| \leq \alpha$ we have that $M \models_{X[A/x_i]} \psi$. Suppose now that for some subset $B_0 \subseteq N$ such that $|B_0| \leq \alpha$, $N \not\models_{X[B_0/x_i]} \psi$: then, as $qr(\psi) \leq n-1$ and by induction hypothesis, Player $\forall$ has a winning strategy in $EF_{n-1}^\alpha(M, X[A/x_i], N, Y[B_0/x_i])$ for all sets A as above.[15] But then Player $\forall$ can win $EF_n^\alpha(M, X[A/x_i], N, Y[B_0/x_i])$ by selecting this B_0 and playing the strategy corresponding to the A picked in answer by Player $\exists$. This contradicts our assumption: therefore, there is no such B_0 and for all $B \subseteq^\alpha N$ it holds that $N \models_{Y[B/x_i]} \psi$, so in conclusion $B \models \forall^\alpha \psi$, as required. □

One may wonder if there exists an Ehrenfeucht-Fraïssé game for $\mathcal{D}(\sqcup, \delta^\alpha)$. It turns out that such a game exists, and it is obtained simply changing the right-α-duplication of $EF_n^\alpha(M, X, N, Y)$ into the following *right-α-selection* rule:

Right-α-selection: $\forall$ chooses a variable $x \in Dom(X) = Dom(Y)$ and a set of elements $B \subseteq N$ such that $|B| \leq \alpha$. Then $\exists$ chooses a set of elements $A \subseteq M$ with $|A| \leq \alpha$, and the next position is $(X^i_{x_i \in A}, Y^i_{x_i \in B}, i+1)$.

The proof that this rule captures correctly the δ^α connective is mirrors exactly the one for $\forall^\alpha$.

One may also wonder which connectives correspond to the *left-α-duplications* and *left-α-selection* rules:

Left-α-duplication: $\forall$ chooses a set of elements $A \subseteq M$ such that $|A| \leq \alpha$. Then $\exists$ chooses a set of elements $B \subseteq N$ with $|B| \leq \alpha$, and the next position is $(X^i[A/x_i], Y^i[B/x_i], i+1)$.

[14] As here we are working by *structural induction* on ϕ rather than by induction on $qr(\phi)$, the fact that $qr(\psi)$ is not necessarily smaller than $qr(\phi)$ is not an issue.

[15] As the EF games are finite games of perfect information which do not allow for draws, by Zermelo's Theorem ([18]) one of the two players has a winning strategy in $EF_{n-1}(M, X[A/x_i], N, Y[B/x_i])$. As $(M, X[A/x_i]) \not\Rrightarrow_{n-1}^U (N, Y[B_0/x_i])$, by induction hypothesis Player $\exists$ does not have a winning strategy. Hence, Player $\forall$ does.

Right-α-selection: $\forall$ chooses a variable $x \in Dom(X) = Dom(Y)$ and a set of elements $A \subseteq M$ such that $|A| \leq \alpha$. Then $\exists$ chooses a set of elements $B \subseteq N$ with $|B| \leq \alpha$, and the next position is $(X^i_{x_i \in A}, Y^i_{x_i \in B}, i+1)$.

However, these rules do not correspond to anything interesting - indeed, if Player $\forall$ uses them then, since $(M, X) \Rrightarrow^\alpha_n (N, \emptyset)$ for all n, M and X, Player $\exists$ can always win the play choosing $B = \emptyset$,

6 Uniform Definability and Σ^1_1

We immediately state and prove the main result of this section:[16]

Theorem 6.1. *Let $\Phi[\Xi_1 \ldots \Xi_t]$ be a context in $\mathcal{D}$ of type $\langle n_1 \ldots n_t \rangle$. Then there exists a formula $\Phi^*(P, \overline{Y_1} \ldots \overline{Y_t})$ in Σ^1_1, where each $\overline{Y_i}$ is a tuple of second order variables $Y_{i,1} \ldots Y_{i,l_i}$ of arity n_i, such that, for all models M, teams X with domain $\{x_1 \ldots x_n\}$ and downwards closed model classes $K_1 \ldots K_t$ with domains $\{y_{i,1} \ldots y_{i,n_i}\}$ $(i \in 1 \ldots t)$,*

$$M \models_X \Phi[K_1 \ldots K_t] \Leftrightarrow (M, Rel(X), \mathcal{K}^1 \ldots \mathcal{K}^t) \models \exists \overline{Y_1} \ldots \overline{Y_t}(\bigwedge_{i=1}^{t} \bigwedge_{j=1}^{l_t} \mathcal{K}^i(Y_{i,j}) \wedge \Phi^*(P, \overline{Y_1} \ldots \overline{Y_t}))$$

where $Rel(X) = \{(s(x_1) \ldots s(x_n)) : Dom(X) = \{x_1 \ldots x_n\}$ and $s \in X\}$ and the $\mathcal{K}^i$ are the second order predicates $\{Rel(Z) : (M_{\upharpoonright Sig(K_i)}, Z^i) \in K_i\}$.

Furthermore, P occurs only negatively and the $Y_{i,j}$ occur only positively in Φ^*.

Proof. The proof is a straightforward induction over $\Phi[\Xi_1 \ldots \Xi_t]$, where each case coincides transparently with one of the conditions of the recursive satisfaction relation over contexts of Definition 2.4:

- If $\Phi[\overline{\Xi}] = \Xi_i(t_1 \ldots t_n)$ for some $i \in 1 \ldots n$, let $\Phi^*(R, Y_{i,1})$ be
$$\forall x_1 \ldots x_n(P(x_1 \ldots x_n) \to Y_{i,1}(t_1 \ldots t_n));$$
- If $\Phi[\overline{\Xi}]$ is a first order atomic literal ϕ, let $\Phi^*(R)$ be
$$\forall x_1 \ldots x_n(P(x_1 \ldots x_n) \to \phi);$$
- If $\Phi[\overline{\Xi}]$ is a atomic dependence formula $=(t_1 \ldots t_n)$, let $\Phi^*(P)$ be
$$\forall x_1 \ldots x_n x'_1 \ldots x'_n((P(x_1 \ldots x_n) \wedge P(x'_1 \ldots x'_n) \wedge t_1 = t_1[\overline{x'}/\overline{x}] \wedge \ldots \wedge t_{n-1} = t_{n-1}[\overline{x'}/\overline{x}]) \to \\ \to t_n = t_n[\overline{x'}/\overline{x}]),$$
where $t[\overline{x'}/\overline{x}]$ is the term obtained substituting each instance of each variable x_i with an instance of the corresponding x'_i;
- If $\Phi[\overline{\Xi}]$is a negated atomic dependence formula $\neg =(t_1 \ldots t_n)$, let $\Phi^*(P)$ be $\forall x_1 \ldots x_n \neg P(x_1 \ldots x_n)$;

[16]An old version of the results of this section, including Proposition 6.1 and a weaker version of Theorem 6.1, was presented at the University of Tampere. The author wishes to thank Juha Kontinen for the suggestions he gave him afterwards, and that have been partially incorporated in the current version of Theorem 6.1.

- If $\Phi[\Xi]$ is $\Psi[\Xi] \vee \Theta[\Xi]$ and $\Psi^*(P', \overline{Y'}_1 \ldots \overline{Y'_t})$, $\Theta^*(P'', \overline{Y''}_1 \ldots \overline{Y''}_t)$ are the Σ_1^1 formulas corresponding to Ψ and Θ, let $\Phi^*(P, \overline{Y'}_1\overline{Y''}_1 \ldots \overline{Y'}_t\overline{Y''}_t)$ be

$$\exists P'P''((\forall x_1 \ldots x_n P(x_1 \ldots x_n) \to P'(x_1 \ldots x_n) \vee P''(x_1 \ldots x_n)) \wedge \Psi^*(P', \overline{Y'}_1 \ldots \overline{Y'_t}) \wedge \\ \wedge \Theta^*(P'', \overline{Y''}_1 \ldots \overline{Y''}_t));$$

- If $\Phi[\Xi]$ is $\Psi[\Xi] \wedge \Theta[\Xi]$ and $\Psi^*(P', \overline{Y'}_1 \ldots \overline{Y'_t})$, $\Theta^*(P'', \overline{Y''}_1 \ldots \overline{Y''}_t)$ are the Σ_1^1 formulas corresponding to Ψ and Θ, let $\Phi^*(P, \overline{Y'}_1\overline{Y''}_1 \ldots \overline{Y'}_t\overline{Y''}_t)$ be

$$\Psi^*(P, \overline{Y'}_1 \ldots \overline{Y'_t}) \wedge \Theta^*(P, \overline{Y''}_1 \ldots \overline{Y''}_t);$$

- If $\Phi[\Xi]$ is $\exists x_{n+1}\Psi[\Xi]$ and $\Psi^*(P', \overline{Y'}_1 \ldots \overline{Y'_t})$ is the Σ_1^1 formula corresponding to Ψ,[17] let $\Phi^*(P, \overline{Y'}_1 \ldots \overline{Y'}_t)$ be

$$\exists P'((\forall x_1 \ldots x_n P(x_1 \ldots x_n) \to \exists x_{n+1} P'(x_1 \ldots x_n x_{n+1})) \wedge \Psi^*(P', \overline{Y'}_1 \ldots \overline{Y'_t}));$$

- If $\Phi[\Xi]$ is $\forall x_{n+1}\Psi[\Xi]$ and $\Psi^*(P', \overline{Y'}_1 \ldots \overline{Y'_t})$ is the Σ_1^1 formula corresponding to Ψ, let $\Phi^*(P, \overline{Y'}_1 \ldots \overline{Y'}_t)$ be

$$\exists P'((\forall x_1 \ldots x_n P(x_1 \ldots x_n) \to \forall x_{n+1} P'(x_1 \ldots x_n x_{n+1})) \wedge \Psi^*(P', \overline{Y'}_1 \ldots \overline{Y'_t})).$$

□

As a consequence of this theorem, we can prove that the $\forall^1$ quantifier - and, hence, all $\forall^i$ and δ^i operators for $i \in \mathbb{N}_0$ - is not uniformly definable:

Proposition 6.1. *The $\forall^1 x$ quantifier is not uniformly definable in Dependence Logic.*

Proof. Suppose that $\forall^1 x$ were uniformly definable: then there would exist a context $\Psi[K]$ of type $\langle 1 \rangle$ such that

$$M \models_{\{\emptyset\}} \Psi[\phi] \Leftrightarrow \forall m \in M, M \models_{\{(x:m)\}} \phi$$

for all formulas $\phi \in \mathcal{D}$. In particular, consider the model $\mathbb{N}$ of all natural numbers, with the signature Σ associating a constant c_n to each number n: since $\mathbb{N} \models_{\{(x:n)\}} =\!(x)$ for all $n \in \mathbb{N}$, we must have that $\mathbb{N} \models_{\{\emptyset\}} \Phi[=\!(x)]$, and since by the above theorem

$$M \models_X \Phi[K] \Leftrightarrow (M, Rel(X), \mathcal{K}) \models \exists Y_{1,1} \ldots Y_{1,l_1}(\bigwedge_{j=1}^{l_1} \mathcal{K}(Y_{1,j}) \wedge \Phi^*(P, Y_{1,1} \ldots Y_{1,l_1}))$$

where $\mathcal{K}$ is the representation of the model class K as a second order predicate and for some formula $\Phi^* \in \Sigma_1^1$ where P occurs only negatively and the $Y_{1,j}$ occur only positively, it must hold that

$$(\mathbb{N}, Rel(\{\emptyset\}), \mathcal{K}) \models \exists Y_{1,1} \ldots Y_{1,l_1}(\bigwedge_{j=1}^{l_1} \mathcal{K}(Y_{1,j}) \wedge \Phi^*(P, Y_{1,1} \ldots Y_{1,l_1}))$$

[17] Quantification over already used variables can be dealt with in an entirely analogous way.

where $\mathcal{K} = \{Rel(Z) : (\mathbb{N}_{\restriction\emptyset}, Z) \in \| =(x)\|_\emptyset(\{x\})\} = \{Q \subseteq \mathbb{N} : |Q| = 1\}$.

Hence there exist singletons $A_1 = \{a_{1,1}\} \ldots A_{l_1} = \{a_{1,1}\}$ such that $\mathbb{N}, Rel(\{\emptyset\}) \models \Phi^*(P, A_1 \ldots A_{l_1})$.

But now, take $b \in \mathbb{N}\backslash\{a_{1,1} \ldots a_{1,l_1}\}$, and define the model class $K' = \|x \neq c_b\|_\Sigma(\{x\}) = \{(M, X) : Sig(M) = \Sigma, Dom(X) = \{x\} \text{ and } x \neq c_b^M\}$: then the corresponding second order predicate is

$$\mathcal{K}' = \{Rel(Z) : (\mathbb{N}_{\restriction\Sigma}, Z) \in K'\} = \{Q \subseteq \mathbb{N} : b \notin Q\}.$$

Therefore, $A_1 \ldots A_{l_1} \in \mathcal{K}'$; hence,

$$(\mathbb{N}, Rel(\{\emptyset\}), \mathcal{K}') \models \exists Y_{1,1} \ldots Y_{1,l_1}(\bigwedge_{j=1}^{l_1} \mathcal{K}'(Y_{1,j}) \wedge \Phi^*(P, Y_{1,1} \ldots Y_{1,l_1}))$$

and thus $\mathbb{N} \models_{\{\emptyset\}} \Phi[K']$, or in other words $\mathbb{N} \models_{\{\emptyset\}} \Phi[x \neq c_b]$.

But $\mathbb{N} \not\models_{\{\emptyset\}} \forall^1 x(x \neq c_b)$, and hence $\Phi[\Xi]$ does not define $\forall^1 x$. □

Corollary 6.1. *For every $n \in \mathbb{N}_0$, the quantifier $\forall^n$ and the operator δ^n are not uniformly definable in $\mathcal{D}$.*

References

[1] S. Abramsky and J. Väänänen. From IF to BI, a tale of dependence and separation. ILLC Publications, PP–2008–27, 2008.

[2] A. Batlag, L. S. Moss, and S. Solecki. Logic of public announcements and common knowledge and private suspicions. *TARK: Theoretical Aspects of Reasoning about Knowledge*, 7, 1998.

[3] J.P. Burgess. A remark on Henkin sentences and their contraries. *Notre Dame Journal of Formal Logic*, 3(44):185–188, 2003.

[4] F. Caicedo, X. Dechesne and T. Jannsen. Equivalence and Quantifier Rules for Logic with Imperfect Information. *Logic Journal of the IGPL*, 17(1):91–129, 2009.

[5] P. Galliani. Game Values and Equilibria for Undetermined Sentences of Dependence Logic. MSc Thesis. ILLC Publications, MoL–2008–08, 2008.

[6] L. Henkin. Some Remarks on Infinitely Long Formulas. In *Infinitistic Methods. Proc. Symposium on Foundations of Mathematics*, pages 167–183. Pergamon Press, 1961.

[7] J. Hintikka. *The Principles of Mathematics Revisited.* Cambridge University Press, 1996.

[8] J. Hintikka and G. Sandu. Informational independence as a semantic phenomenon. In J.E Fenstad, I.T Frolov, and R. Hilpinen, editors, *Logic, methodology and philosophy of science*, pages 571–589. Elsevier, 1989.

[9] W. Hodges. Compositional Semantics for a Language of Imperfect Information. *Journal of the Interest Group in Pure and Applied Logics*, 5 (4):539–563, 1997.

[10] T.M.V. Janssen. Algebraic translations, correctness and algebraic compiler construction. *Theoretical Computer Science*, (199):25–56, 1998.

[11] T.M.V. Janssen. Independent Choices and the Interpretation of IF Logic. *Journal of Logic, Language and Information*, 11:367–387, 2002.

[12] J. Kontinen and J. Väänänen. On definability in dependence logic. *Journal of Logic, Language and Information*, 3(18):317–332, 2009.

[13] J. Kontinen and J. Väänänen. A Remark on Negation of Dependence Logic. Forthcoming, 2010.

[14] T. Tulenheimo. Independence Friendly Logic. http://plato.stanford.edu/entries/logic-if/, 2009.

[15] J. Väänänen. *Dependence Logic.* Cambridge University Press, 2007.

[16] J. Väänänen. Team Logic. In J. van Benthem, D. Gabbay, and B. Löwe, editors, *Interactive Logic. Selected Papers from the 7th Augustus de Morgan Workshop*, pages 281–302. Amsterdam University Press, 2007.

[17] J. van Benthem, J. van Eijck, and B. Kooi. Logics of communication and change. *INFCTRL: Information and Computation (formerly Information and Control)*, 204, 2006.

[18] E. Zermelo. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels. In *Proceedings of the Fifth Congress of Mathematicians*, volume 2, pages 501–504. Cambridge University Press, 1913.

Lottery Semantics*

Pietro Galliani and Allen L. Mann

Abstract

We present a compositional semantics for a logic of imperfect information and prove its equivalence to *equilibrium semantics* ([10]), thus extending to mixed (rather than just behavioural) strategies part of the work of ([2], [3]).

1 Dependence-Friendly Logic

Logics of imperfect information ([4], [5], [12], [11]) are extensions of First Order Logic which allow for more general patterns of dependence and independence between connectives. In this work, for ease of notation we will consider Dependence Friendly Logic ([12], [2]); however, all of this can be easily adapted to the cases of Dependence Logic ([12]) or of Independence-Friendly Logic ([5]).

Definition 1.1 (Dependence-Friendly Logic formulas)**.** *The* formulas *of Dependence-Friendly Logic $\mathcal{DF}$ in the signature Σ are those generated by the grammar:*

$$\phi ::= Rt_1 \dots t_n \mid t = t' \mid \neg\phi \mid \phi \vee \phi \mid \exists x_{\backslash V}\phi$$

where n ranges over $\mathbb{N}$, R ranges over all n-ary relation symbols in Σ, $t_1 \dots t_n, t$ and t' are terms over the signature Σ, x is any variable in the infinite set $Var = \{x_1, x_2 \dots\}$, and $V \subseteq Var$.

We will freely use the standard abbreviations of $\phi \wedge \psi$ for $\neg(\neg\phi \vee \neg\psi)$, $\forall x_{\backslash V}\phi$ for $\neg\exists x_{\backslash V}\neg(\psi)$, $\exists x\phi$ for $\exists x_{\backslash Var}\phi$ and $\forall x\phi$ for $\forall x_{\backslash Var}\phi$.

Definition 1.2 (Free variables in $\mathcal{DF}$)**.** *Let ϕ be any DF-Logic formula. Then its set of* free variables *$FV(\phi)$ is defined inductively as follows:*

- *$FV(Rt_1 \dots t_n) = Var(t_1) \cup \dots \cup Var(t_n)$, where $Var(t_i)$ is the set of all variables occurring in t_i;*
- *$FV(t = t') = Var(t) \cup Var(t')$;*
- *$FV(\neg\phi) = FV(\phi)$;*
- *$FV(\phi \vee \psi) = FV(\phi) \cup FV(\psi)$;*

*This work was supported by the European Science Foundation Eurocores programme LogICCC [FP002 - Logic for Interaction (LINT)]. The second author wishes to gratefully acknowledge the generous financial support provided by the Academy of Finland (grant 129208) in the context of the European Science Foundation EUROCORES project Logic for Interaction (LINT).

- $FV(\exists x_{\backslash V}\phi) = (FV(\phi)\backslash\{x\}) \cup V$.

Just as most[1] logics of imperfect information, the semantics of Dependence Friendly Logic may be defined by means of *semantic games*:

Definition 1.3 (The game $G_s^M(\phi)$)**.** *Let M be a first order model of signature Σ, let s be a variable assignment, and let ϕ be any DF-Logic formula of signature Σ with $FV(\phi) \subseteq Dom(s)$. Then the game $G_s^M(\phi)$ is defined as follows:*

- *Positions are triples (ψ, s', α), where ψ is an instance of a subformula of ϕ, s' is an assignment such that $FV(\psi) \subseteq Dom(s')$ and $\alpha \in \{I, II\}$. We write P for the set of all positions of the game; furthermore, for $\alpha \in \{I, II\}$ we write P_α for the set of all positions of the form (ψ, s', α).*
- *The starting position is (ϕ, s, II);*
- *Given any position p, the set of its* successors *$S(\phi)$ is defined as follows:*
  1. *If p is of the form $(Rt_1 \dots t_n, s', \alpha)$ or $(t = t', s', \alpha)$ then $S(p) = \emptyset$;*
  2. *If p is of the form $(\neg\psi, s', \alpha)$ then $S(p) = \{(\psi, s', \alpha^*)\}$, where $II^* = I$ and $I^* = II$;*
  3. *If p is of the form $(\psi \vee \theta, s', \alpha)$ then $S(p) = \{(\psi, s', \alpha), (\theta, s', \alpha)\}$;*
  4. *If p is of the form $(\exists x_{\backslash V}\psi, s', \alpha)$ then $S(p) = \{(\psi, s'[m/x], \alpha) : m \in M\}$, where m ranges over all elements of M and $s'[m/x]$ is the assignment defined as*
  $$s'[m/x](y) = \begin{cases} m & \text{if } x = y; \\ s'(y) & \text{otherwise} \end{cases}$$
  for all $y \in Dom(s') \cup \{x\}$;
- *The set W_α of* winning positions for player *$\alpha \in \{I, II\}$ is defined as*
$$\begin{aligned} W_\alpha = &\{(Rt_1 \dots t_n, s', \alpha) : (t_1\langle s'\rangle \dots t_n\langle s'\rangle) \in R^M)\} \cup \{(t = t', s, \alpha) : t\langle s\rangle = t'\langle s'\rangle\} \cup \\ &\cup \{(Rt_1 \dots t_n, s', \alpha^*) : (t_1\langle s'\rangle \dots t_n\langle s'\rangle) \notin R^M)\} \cup \{(t = t', s, \alpha^*) : t\langle s\rangle \neq t'\langle s'\rangle\}. \end{aligned}$$

Definition 1.4 (Play, complete play, winning play)**.** *Let M, s and ϕ be as above.*

A play *$\overline{p}$ of $G_s^M(\phi)$ is a finite sequence of positions $p_1 \dots p_n$ such that*

1. *p_1 is the initial position of $G_s^M(\phi)$;*
2. *For all $i < n$, $p_{i+1} \in S(p_i)$.*

If furthermore p_n is a terminal *position (that is, $S(p_n) = \emptyset$) then we say that $\overline{p}$ is a* complete play*; in this case, we say that $\overline{p}$ is* winning *for Player α if and only if $p_n \in W_\alpha$.*

It follows immediately from the definition of W_α that every complete play of every game $G_s^M(\phi)$ is winning for Player I or Player II, and that no play is winning for both of them; in other words, the games $G_s^M(\phi)$ thus defined are *zero-sum* games.

[1]But not necessarily all: for example, at the present time no natural semantics in terms of games of imperfect information is known for Team Logic ([13]).

Definition 1.5 (Strategy, play following a strategy)**.** *Let M, s, and ϕ be as above, and let $\alpha \in \{I, II\}$. Then a* strategy *σ for α in $G_s^M(\phi)$ is a function from P_α to P such that*

$$\sigma(p) \in S(p)$$

for all $p \in P_\alpha$.

A play $\overline{p} = p_1 \dots p_n$ is said to follow *such a strategy σ if and only if*

$$p_i \in P_\alpha \Rightarrow p_{i+1} = \sigma(p_i)$$

for all $i \in 1 \dots n-1$.

These strategies will be sometimes called *pure strategies*, in order to distinguish them from the *mixed strategies* considered in the next sections.

Lemma 1.1. *Let σ and τ be two strategies for Player I and Player II in $G_s^M(\phi)$. Then there exists one and only one complete play $\overline{p} = (\sigma; \tau)$ of $G_s^M(\phi)$ which follows σ and τ.*

Definition 1.6 (Winning strategy)**.** *A strategy σ for Player α in $G_s^M(\phi)$ is said to be* winning *if and only if all complete plays of $G_s^M(\phi)$ which follow σ are winning for Player α.*

So far, the set of backslashed variables V of our quantifiers $\exists x_{\backslash V}$ played no role. From an informal point of view, V specifies the set of variables on which the choice of the value for x may depend, as the following definition makes clear:

Definition 1.7 (Uniform strategy)**.** *A strategy σ for Player α in $G_s^M(\phi)$ is* uniform *if and only if for all instances of subformulas of ϕ of the form $\exists x_{\backslash V} \psi$ and for all assignments s', s'' such that*

- *$Dom(s') = Dom(s'') \supseteq FV(\exists x_{\backslash V} \psi)$ and*
- *$s'_{|V} = s''_{|V}$*

it holds that

$$\sigma(\exists x_{\backslash V} \psi, s', \alpha) = \sigma(\exists x_{\backslash V} \psi, s'', \alpha).$$

Then we may define truth of a DF-Logic formula in terms of existence of uniform winning strategies in the corresponding game.

Definition 1.8 (Truth of DF-Logic formulas)**.** *Let M be a first order model of signature Σ, let ϕ be a DF-Logic formula of the same signature and let s be an assignment such that $FV(\phi) \subseteq Dom(s)$. Then we say that ϕ is* true *in M according to the assignment s, and we write $M \models_s \phi$, if and only if Player II has a uniform winning strategy in $G_s^M(\phi)$.*

For some time, it was an open problem whether logics of imperfect information admitted a natural compositional semantics ([5]). This was answered positively by Wilfrid Hodges ([6]), whose argument we now show for the case of DF-Logic.

Definition 1.9 (Team)**.** *Let M be a model, and V a set of variables. Then a* team *X over M with domain V is a set of assignments with the same domain V.*

From an informal point of view, a team[2] represents the knowledge a player has about the current assignment s.

It is now possible to adapt the games G_s^M to teams:

Definition 1.10 ($G_X^M(\phi)$)**.** *Let M be a first order model of signature Σ, let ϕ be a DF-Logic formula of the same signature and let X be a team with $FV(\phi) \subseteq Dom(X)$. Then the game $G_X^M(\phi)$ is defined precisely as $G_s^M(\phi)$ except for the following change to the definition of starting position:*

- *The starting positions are $\{(\phi, s, II) : s \in X\}$.*

The definitions of play, complete play, winning play, strategy, winning strategy and uniform winning strategy carry over from $G_s^M(\phi)$ to $G_X^M(\phi)$ without any trouble[3]: for example, a *play* $\overline{p}$ of $G_X^M(\phi)$ is a sequence of positions $p_1 \ldots p_n$ such that p_1 is an initial position of $G_X^M(\phi)$ and $p_{i+1} \in S(p_i)$ for all $i = 1 \ldots n-1$, and so on.

Hence, it is possible to define the following *satisfaction relation* over teams:

Definition 1.11 ($M \models_X^\pm \phi$)**.** *Let M be a first order model of signature Σ, let ϕ be a DF-formula for the same signature and let X be a team with $FV(\phi) \subseteq Dom(X)$. Then we say that X* satisfies *ϕ in M, and we write $M \models_X^+ \phi$, if and only if Player II has a uniform winning strategy in $G_X^M(\phi)$.*

Analogously, we write $M \models^- \phi$ if and only if Player I has a uniform winning strategy in $G_X^M(\phi)$.

Lemma 1.2. *For all suitable models M, assignments s and formulas ϕ,*

$$M \models_s \phi \Leftrightarrow M \models_{\{s\}}^+ \phi.$$

The following theorem then shows how to define compositionally the relations $\models^\pm$:

Theorem 1.1 (Hodges)**.** *Let M be any first order model and let X be any suitable team. Then*

1. *$M \models_X^+ Rt_1 \ldots t_n$ if and only if, for all $s \in X$, $(t_1\langle s\rangle \ldots t_n\langle s\rangle) \in R^M$;*
2. *$M \models_X^+ t = t'$ if and only if, for all $s \in X$, $t\langle s\rangle = t'\langle s\rangle$;*
3. *$M \models_X^+ \neg\phi$ if and only if $M \models_X^- \phi$;*
4. *$M \models_X^+ \phi \vee \psi$ if and only if there exist teams X_1, X_2 such that*
    - *$X \subseteq X_1 \cup X_2$,*
    - *$M \models_{X_1}^+ \phi$, and*
    - *$M \models_{X_2}^- \psi$;*
5. *$M \models_X^+ \exists x_{\backslash V}\phi$ if and only if there exists a function $F : X \to M$ such that $M \models_{X[F/x]} \phi$, where*

    $$X[F/x] = \{s[F(s)/x] : s \in X\},$$

    and furthermore F depends only on V, in the sense that

    $$\forall s, s' \in X, \text{ if } s_{|V} = s'_{|V} \text{ then } F(s) = F(s').$$

[2]Hodges ([6]) calls a winning team a *trump*, and a losing team a *cotrump*; hence the name *trump semantics*. Väänänen ([12]) introduces the term *team*.

[3]Although, of course, Lemma 1.1 does not hold anymore, as there are many different possible starting positions.

Furthermore,

1. $M \models^-_X Rt_1 \dots t_n$ *if and only if, for all* $s \in X$*,* $(t_1\langle s\rangle \dots t_n\langle s\rangle) \notin R^M$*;*

2. $M \models^-_X t = t'$ *if and only if, for all* $s \in X$*,* $t\langle s\rangle \neq t'\langle s\rangle$*;*

3. $M \models^-_X \neg\phi$ *if and only if* $M \models^+_X \phi$*;*

4. $M \models^-_X \phi \vee \psi$ *if and only if* $M \models^-_X \phi$ *and* $M \models^-_X \psi$*;*

5. $M \models^-_X \exists x_{\backslash V}\phi$ *if and only if* $M \models^-_{X[M/x]} \phi$*, where*

$$X[M/x] = \{s[m/x] : s \in X \text{ and } m \in M\}.$$

We finish this section with a few general comments about logics of imperfect information and Hodges' construction that will be of some use in the rest of the work.

- The principal insight that can be found in Hodges' construction consists in the passage from assignments to teams, and from the games $G^M_s(\phi)$ to the games $G^M_X(\phi)$. By increasing the complexity of the semantical objects upon which the satisfaction relation is predicated, he managed to let the truth condition of any formula be a function of the truth conditions of the components; or, and by the above theorem this is the same, he managed to let the existence of uniform winning strategies for $G^M_X(\phi)$ be a function of the existence of uniform winning strategies for games corresponding to the subformulas of ϕ.

- The fact that "positive satisfiability" and "negative satisfiability" had to be considered separately is a consequence of the well-known fact that the law of the excluded middle does not hold in logics of imperfect information ([5]).

  A typical example of this can be seen by considering the formula

  $$\phi := \forall x(\exists y_{\backslash\emptyset})(x = y),$$

  corresponding to the game

  1. Player I (Abélard) picks an element $m \in M$;
  2. Player II (Eloïse) picks, *independently from the value of* m, an element $n \in M$;
  3. Player II wins if $m = n$, otherwise Player I wins.

  It can be easily seen that, if the model M has at least two elements, neither Abélard nor Eloïse[4] have a winning strategy for the game $G^M_{\{\emptyset\}}(\phi)$; and indeed, by applying Hodges' compositional semantics one can verify that

  $$|M| > 1 \Rightarrow M \not\models^+_{\{\emptyset\}} \phi,\ M \not\models^-_{\{\emptyset\}} \phi.$$

[4]In the rest of this work, we will use "Abélard and Eloïse" or "Player I and Player II" interchangeably, with a preference for the latter.

- There is an asymmetry between the rules for positive satisfiability and those for negative satisfiability. Not only the conditions for positive satisfiability of disjunction and quantification are formally different from those for the negative satisfiability of the same connectives, but the rule for $M \models^-_X \exists x_{\backslash V}\phi$ does not even mention the set of variables V!

  The reason for this is that restricting what information is accessible to Abélard does not influence in any way the existence of uniform winning strategies for Eloïse: if she can guarantee a victory against all "blind" strategies of Abélard, she can also guarantee a victory if Abélard has access to all possible information about her moves. As the converse is obvious, this implies that - as long as we are only concerned with the existence of winning strategies of Eloïse, that is, with positive satisfaction - slashed universal quantification behaves precisely as unslashed universal quantification: if ϕ is any formula, and ϕ^* is obtained from ϕ by substituting each negatively occurring subformula $\exists x_{\backslash V}\psi$ with $\exists x\psi^*$, we have that

  $$M \models^+_X \phi \leftrightarrow M \models^+_X \phi^*.$$

  The same phenomenon would occur if we added to our language "backslashed disjunctions" $\phi \vee_V \psi$: one could verify that $M \models^+_X \phi \vee_V \psi$ if and only if there exist teams X_1 and X_2 such that

  - $X \subseteq X_1 \cup X_2$;
  - $M \models^+_{X_1} \phi$;
  - $M \models^+_{X_2} \psi$;
  - For all $s, s' \in X$, if $s_{|V} = s'_{|V}$ and $s \in X_1$ then $s' \in X_1$ too

  but $M \models^-_X \phi \vee_V \psi$ if and only if $M \models^-_X \phi$ and $M \models^-_X \psi$.

  For simplicity reasons, the backslashed disjunction connectives will not be considered in the rest of this work; they could however be added without much difficulty, if one were so inclined, by generalizing the notion of *splitting function* (Definition 4.4) in order to be able to require a splitting function Sp to be determined by a set V of variables.

2 Equilibrium Semantics

As we saw, it is possible for a formula ϕ of DF-Logic to be neither true nor false in a model M and with respect to a team X, and it is hence possible that neither player has a uniform winning strategy in the corresponding game $G^M_X(\phi)$.

Hence, as Miklos Ajtai first noticed ([1]), it may be worthwhile to ask what is the value of the game when we allow both player to randomize their strategies. This intuition was made more precise and turned into a formal definition in ([9]); later, ([2]) developed - independently from Sevenster's result - a compositional semantics for Probabilistic Dependence Logic and proved the equivalence between the values it computes and those of Behavioural Nash Equilibria for the corresponding semantic games.[5]

[5] In a behavioural strategy, at each stage of the game the active player chooses a probability distribution for the

Independently from Galliani, and working with mixed strategies rather than with behavioural ones, Sevenster and Sandu then defined *Equilibrium Semantics* for IF-Logic in ([10]) and proved a number of results about its expressive power and the complexity of computing these equilibria; however, no compositional semantics equivalent to Equilibrium Semantics was presented.

In the rest of this section, we will adapt Sevenster and Sandu's approach to DF-Logic; then, in the next two, we will extend Hodges' approach from the usual "Winning Strategy Semantics" to Equilibrium Semantics, thus deriving a compositional semantics which is equivalent to it.

From now on, when we talk about strategies we will always implicitly require them to be *uniform*, unless otherwise specified.

Definition 2.1 (Value of a pair of pure strategies)**.** *Let M be a finite model of signature Σ, let s be an assignment, and let ϕ be any DF-Logic formula with $FV(\phi) \subseteq Dom(s)$. Then, for any two strategies σ and τ for I and II in $G_s^M(\phi)$, we define the* value *of this pair (σ, τ) as*

$$V_s^M(\phi; \sigma, \tau) = \begin{cases} 1 & \text{if } (\sigma;\tau) \text{ is winning for } II; \\ 0 & \text{if } (\sigma;\tau) \text{ is winning for } I \end{cases}$$

where $(\sigma; \tau)$ is the unique play of $G_s^M(\phi)$ in which Player I follows σ and Player II follows τ.

Definition 2.2 (Mixed Strategy)**.** *Let M be a finite model of signature Σ, let s be an assignment, and let ϕ be any DF-Logic formula with $FV(\phi) \subseteq Dom(s)$. Then a* mixed strategy *for Player $\alpha \in \{I, II\}$ in the game $G_s^M(\phi)$ is a probability distribution over all pure strategies for α in the same game, that is, a function from pure strategies for α to $[0, 1]$ such that*

$$\sum \{\mu(\sigma) : \sigma \text{ is a pure strategy for } \alpha \text{ in } G_s^M(\phi)\} = 1.$$

Definition 2.3 (Value of a pair of strategies)**.** *Let μ, ν be two mixed strategies for Players I and II in $G_s^M(\phi)$, where M, s and ϕ are as above. Then the* value *of the pair of strategies for Player II is*

$$V_s^M(\phi; \mu, \nu) = \sum_\sigma \sum_\tau \mu(\sigma)\nu(\tau)V_s^M(\phi; \sigma, \tau)$$

---

next position, whereas in a mixed strategy the player picks a pure (deterministic) strategy at the beginning of the game and then plays according to it.

It is not difficult to see that all behavioural strategies correspond to mixed strategies, in the sense that for every behavioural strategy γ there exists a mixed strategy γ' such that the payoff of the strategy profile (γ, ρ) is equal to that of the strategy profile (γ', ρ) for all pure strategies (and, hence, also for all mixed and for all behavioral strategies) ρ of the other player. Moreover, Kuhn' Theorem ([7]) states that in all games of perfect recall (that is, those in which a player has always access to the values of the choices previously made by him/her) all mixed strategies correspond to behavioural strategies.

However, our semantic games are not necessarily games of perfect recall, and hence Kuhn's Theorem does not hold: for example, if the domain of the model M contains two elements a and b and ϕ is $\exists x \exists y_{\backslash \emptyset}(x = y)$ then in the game $G_\emptyset^M(\phi)$ the mixed strategy $1/2(x := a, y := a) + 1/2(x := b, y := b)$ does not correspond to any behavioural strategy.

Choosing between considering mixed and behavioural strategies corresponds to choosing a specific concept of dependence and independence between variables: for example, one may ask whether, according to the desired notion of dependence and to the probability distribution over assignments induced by the above mixed strategy, the values of x and y are independent on each other.

However, it must be stressed that the Minimax theorem only guarantees the existence of *mixed* strategy equilibria for semantic games $G_s^M(\phi)$, and only for *finite* models: if we are working with behavioural strategies or infinite models then, in general, there is no guarantee that an equilibrium exists.

where σ ranges over all pure strategies of Player I in $G_s^M(\phi)$ and τ ranges over all pure strategies of Player II in $G_s^M(\phi)$.

Definition 2.4 (Nash Equilibrium)**.** *Let $G_s^M(\phi)$, μ and ν be as above. Then we say that the pair (μ, ν) is a* Nash Equilibrium *if*

- *For all mixed strategies μ' for Player I in $G_s^M(\phi)$, $V_s^M(\phi; \mu, \nu) \leq V_s^M(\phi; \mu', \nu)$;*
- *For all mixed strategies ν' for Player II in $G_s^M(\phi)$, $V_s^M(\phi; \mu, \nu) \geq V_s^M(\phi; \mu, \nu')$;*

Lemma 2.1. *If (μ, ν) and (μ', ν') are equilibria in $G_s^M(\phi)$ then $V_s^M(\phi; \mu, \nu) = V_s^M(\phi; \mu', \nu') = V_s^M(\phi; \mu', \nu) = V_s^M(\phi; \mu, \nu')$.*

Proof. By definition of equilibrium,

$$V_s^M(\phi; \mu, \nu) \leq V_s^M(\phi; \mu', \nu) \leq V_s^M(\phi; \mu', \nu')$$

where the first inequality comes from (μ, ν) being a Nash equilibrium, and the second one comes from (μ', ν') being a Nash equilibrium.

Analogously,

$$V_s^M(\phi; \mu', \nu') \leq V_s^M(\phi; \mu, \nu') \leq V_s^M(\phi; \mu, \nu)$$

and the result follows. □

The following is a restatement of the Minimax Theorem ([8]) for our semantic games:

Theorem 2.1. *Let M be a finite model of signature Σ, let ϕ be a DF-Logic formula and let s be an assignment with $FV(\phi) \subseteq Dom(s)$. Then there exist two mixed strategies μ, ν for Players I and II in $G_s^M(\phi)$ such that (μ, ν) is a Nash equilibrium.*

These results justify the following definition (which is simply that of [10], adapted to the language of Dependence Friendly Logic):

Definition 2.5 (Equilibrium Semantics)**.** *Let M, s and ϕ be as above. Then the* value $V_s^M(\phi)$ *of ϕ in M according to s is $V_s^M(\phi; \mu, \nu)$ where (μ, ν) is a mixed strategy Nash equilibrium over $G_s^M(\phi)$.*

Example 2.1. *Let us consider again the formula*

$$\phi := \forall x \exists y_{\backslash \emptyset}(x = y)$$

and let M be a model with n elements.

Then $V_\emptyset^M(\phi) = 1/n$: indeed, for any $m \in M$ let τ^m be the strategy for Player II defined as

$$\tau^m(\exists y_{\backslash \emptyset}(x = y), s, II) = m, \text{ for all assignments } s$$

and let ν be such that $\nu(\tau^m) = 1/n$ for all $m \in M$.

Then for all mixed strategies μ' of Player I it holds that $V_s^M(\phi; \mu', \nu) = 1/n$: indeed, if $x\langle\sigma\rangle$ is the value that Player I assigns to x according to the pure strategy σ then

$$V_s^M(\phi; \mu', \nu) = \sum_\sigma \mu'(\sigma) \sum_\tau \nu(\tau) V_s^M(\phi; \sigma, \tau) = \sum_\sigma \mu'(\sigma) \sum_{m \in M} V_s^M(\phi; \sigma, \tau^m)/n = \sum_\sigma \mu'(\sigma) \cdot 1/n = 1/n$$

where we used the fact that

$$V_s^M(\phi;\sigma,\tau^m) = \begin{cases} 1 & \textit{if } x\langle\sigma\rangle = m; \\ 0 & \textit{otherwise.} \end{cases}$$

By the exact same argument one can show that if $\mu(\sigma^m) = 1/n$ *for all* $m \in M$*, where* σ^m *is the pure strategy for Player I which assigns the value* m *to* x*,* $V_s^M(\phi;\mu,\nu') = 1/n$ *for all mixed strategies* ν' *of Player II; hence,* (μ,ν) *is a Nash equilibrium, and*

$$V_s^M(\phi) = V_s^M(\phi;\mu,\nu) = 1/n.$$

3 Lottery Augmented Games

In this section and in the next one we will adapt Hodges' construction to Equilibrium Semantics for DF-Logic, thus obtaining an equivalent compositional semantics.

The main obstacle to doing this is the fact that mixed strategies are complex objects, much more so than pure strategies or even behavioral strategies: in particular, the behaviours that a mixed strategy μ induces on the subgames cannot, in general, be represented as mixed strategies over the information partitions of the subgames.

Example 3.1. *Let the domain of* M *be* $\{a,b\}$*, let* $\phi = \exists x \exists y_{\backslash\emptyset}(x = y)$*, and consider the mixed strategy*

$$\nu = 1/2(x := a, y := a) + 1/2(x := b, y := b)$$

in the game $G_\emptyset^M(\phi)$*.*[6]

Over the two subgames corresponding to $\exists y_{\backslash\emptyset}(x = y)$*, which are indistinguishable from the point of view of Player II,* ν *induces the behaviour "if* $x = a$*, let* $y = a$*; if instead* $x = b$*, let* $y = b$*". But this is not expressible as a mixed strategy, since the choice of* y *is supposed to be independent on the choice of* x*!*

Taking a hint from Hodges' construction, we sidestep this issue by increasing the complexity of our games, thus turning the mixed strategies of the original game into pure strategies for the modified game, which we will then be able to carry over to the subgames in a more direct way.

The following observation can be used both as the starting point and as justification of the change that we will make to the definition of the games $G_s^M(\phi)$:

Definition 3.1 (Graphical representation of mixed strategies)**.** *Let* $\sigma^1, \sigma^2, \ldots \sigma^n$ *be an enumeration of all pure strategies of Player* $\alpha \in \{I, II\}$ *in* $G_s^M(\phi)$*, where* M *is finite, and let* μ *be a mixed strategy for* α *in the same game.*

Then we can represent the strategy μ *as a* partition $\mathcal{A}$ *of the unit segment in* n *intervals* $A_1 \ldots A_n$*, where the length of each* A_i *is exactly* $\mu(\sigma_i)$*.*

---

[6]Here, as in Footnote 5, $(x := a, y := a)$ is the pure strategy τ_1 which assigns the value a to both x and y, $(x := b, y := b)$ is the pure strategy τ_2 which assigns the value b to both x and y, and $1/2(x := a, y := a) + 1/2(x := b, y := b)$ is the mixed strategy which selects τ_1 and τ_2 with equal probability $1/2$.

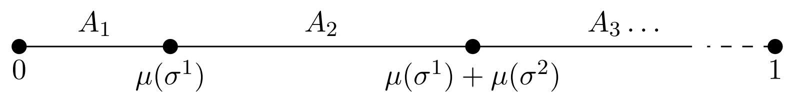

Figure 3.1: Graphical representation of a mixed strategy

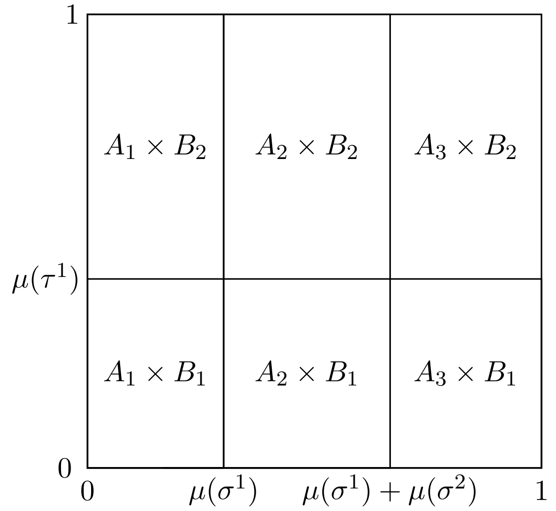

Figure 3.2: The graphical representation of a strategy profile

Given this representation, we can then recover our probability distribution by considering, for each $i = 1 \ldots n$ and according to the uniform distribution over $[0, 1]$, the probability that $x \in A_i$: indeed,

$$Prob(x \in A_i) = \int_{A_i} dx = \mu(\sigma_i).$$

Definition 3.2 (Graphical representation of strategy profiles)**.** *Let μ and ν be two mixed strategies for Player I and Player II in $G_s^M(\phi)$, and let $\mathcal{A} = (A_i)_{i \in I}$, $\mathcal{B} = (B_j)_{j \in J}$ be the representations of μ and ν as partitions of the unit intervals.*

Then consider the partition $(A_i \times B_j)_{i \in I, j \in J}$ of the unit square $[0, 1]^2$: we say that this partition is a representation *of the strategy profile (μ, ν).*

For any $i \in I$ and $j \in J$, the area of $A_i \times B_j$ (and, hence, the probability that a point in the unit square, picked according to the uniform probability distribution, belongs to it) is given by $\int_{A_i \times B_j} dxdy = \int_{A_i} dx \cdot \int_{B_j} dy = \mu(\sigma^i)\nu(\tau^j)$. Hence the following result holds:

Proposition 3.1 (Values of strategy profiles as measures)**.** *Let M be a finite model of signature Σ, let ϕ be a DF-Logic formula in the same signature, and let s be an assignment with $FV(\phi) \subseteq$*

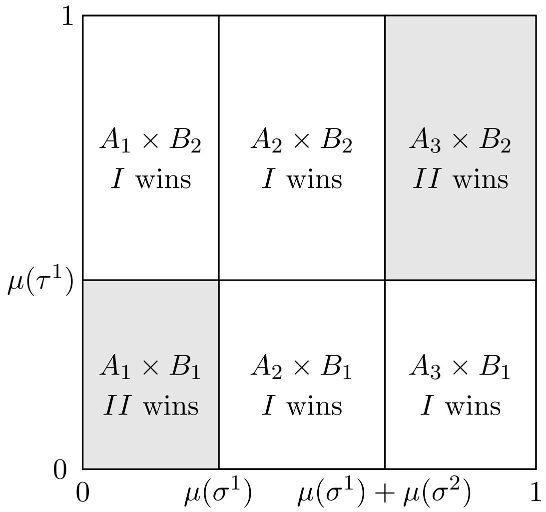

Figure 3.3: The same profile represented in Figure 3.2, but with the winning region for Player II grayed out

$Dom(s)$. Then, if $\mathcal{A} = (A_i)_{i \in I}$ and $\mathcal{B} = (B_j)_{j \in J}$ are the representations of μ and ν and define the subset $W_{II} \subseteq [0,1]^2$ as

$$W_{II}(\mathcal{A}, \mathcal{B}) = \{(a,b) \in [0,1]^2 : \exists i \in I, j \in J \text{ s.t. } a \in A_i, b \in B_j \text{ and } II \text{ wins } (\sigma^i; \tau^j)\} = \\ = \bigcup \{A_i \times B_j : V_s^M(\phi; \sigma^i, \tau^j) = 1\}$$

we have that

$$V_s^M(\phi; \mu, \nu) = \int_{W_{II}(\mathcal{A},\mathcal{B})} dxdy.$$

Proof.

$$V_s^M(\mu, \nu) = \sum_{i \in I} \sum_{j \in J} \mu(\sigma^i) \nu(\tau^j) V_s^M(\phi; \sigma^i, \tau^j) = \sum_{i \in I} \sum_{j \in J} V_s^M(\phi; \sigma^i, \tau^j) \int_{A_i \times B_j} dxdy = \int_{W_{II}(\mathcal{A},\mathcal{B})} dxdy.$$

□

At this point, it may help the reader's intuition to think of Players I and II not as single players, but as coalitions of players whose interests are aligned. Before a given play of the game, each coalition is allowed to come to an agreement about which pure strategy they will use (possibly with the aid of a coin-flip or the roll of a die). Once play begins no further coordination between the players is allowed; however, during play some players' actions may be observed by other players. The players are allowed to use their observations when deciding how to act.

For example, imagine two American football teams facing each other on the field. When the clock is stopped, the coach of the team on offense calls the play and relays it to the quarterback.

At the same time, the defensive coach chooses a defense and informs the middle linebacker. Both sides form a huddle. The quarterback tells the offense which play was called, taking care not to let the defense overhear. On the other side of the ball, the middle linebacker tells the defense how to line up. Once the ball is snapped, the players execute their assignments more or less independently, reacting as best they can to what they see happening on the field. For instance, if the middle linebacker sees the quarterback hand the ball off to the running back, he will try to tackle him, whereas if the he sees the quarterback drop back to pass, he will fall back into pass coverage.

Now imagine the members of each coalition are dispersed around the globe so that they are unable to gather together to coordinate their strategies. For example, suppose that Player I is a team of CIA agents, and Player II is a group of KGB agents. Before leaving headquarters, each group distributes a code book to its members. Once in the field, the agents await a signal indicating which strategy they should execute. For example, if, at a certain time, BBC Radio 4 broadcasts an advertisement for a non-existent brand of laundry detergent, the CIA agents will execute the strategy on page 19 of their code book. At approximately the same time, Soviet state television broadcasts an homage to Vladimir Lenin containing a certain agreed-upon word, telling the KGB agents to execute the strategy on page 42 of their code book. During the course of the operation a given CIA agent may be able to observe the actions taken by some of her fellow agents, as well as the opposing agents, before executing her assignment. Her actions may be observed in turn and affect subsequent choices made by friend and foe alike.

In terms of our mixed-strategy diagrams, we can think of the real number a as encoding a signal to Player I, and b as encoding a signal to Player II; and these signals allow each coalition to coordinate its pure strategy without leaking any information to their opponents. In order to formalize such signals, we define a new version of the extensive game that adds two random moves the beginning of each play in order to extract the “lottery numbers” a and b, and we adapt Hodges’ definition of team in order to represent the partial plays associated to each pair of lottery numbers.

Definition 3.3 (Grids, functions respecting grids). *A* grid *is a pair $\langle \mathcal{A}, \mathcal{B} \rangle$ of finite, measurable partitions of* $[0, 1]$. *A partial function H on* $[0, 1]^2$ *is said to* respect *such a grid if it is constant over the rectangles $A \times B$ for all $A \in \mathcal{A}$ and $B \in \mathcal{B}$, that is,*

$$a, a' \in A \in \mathcal{A} \text{ and } b, b' \in B \in \mathcal{B} \quad \text{imply} \quad H(a, b) = H(a', b').$$

Here we allow ourselves to write $H(a, b) = H(a', b')$ in the case when both $H(a, b)$ and $H(a', b')$ are undefined.

The following definition is for Equilibrium Semantics what the concept of Team is for Winning Strategy Semantics (that is, for the usual game semantics in terms of winning strategies for logics of imperfect information):

Definition 3.4 (Strategy guide). *Let V be a finite set of variables, let M be a suitable finite structure, and let $\langle \mathcal{A}, \mathcal{B} \rangle$ be a grid. A* strategy guide *with assignments over V in the model M is a partial function $H : [0, 1]^2 \to M^V$ that sends points in the unit square to assignments with domain V and respects $\langle \mathcal{A}, \mathcal{B} \rangle$.*

Definition 3.5 (Lottery-augmented games). *Let M be a finite model of signature Σ, let ϕ be a DF-Logic formula, and let H be a strategy guide with assignments over some $V \supseteq FV(\phi)$.*

Then we define the game $\underline{G}^M_H(\phi)$ as follows:

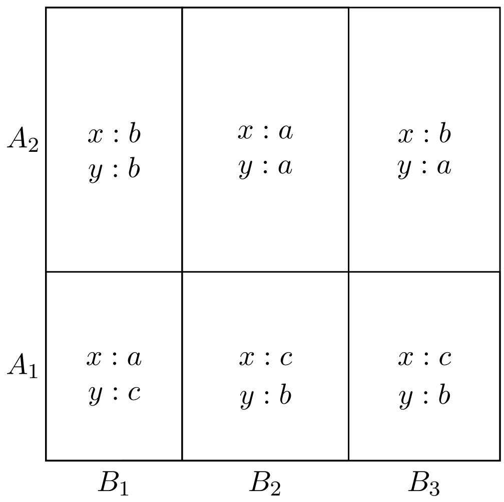

Figure 3.4: Graphical representation of a strategy guide

1. *The set of all positions of the game is given by* $P = \{(\psi, s, \alpha) : \psi$ *is an instance of a subformula of* ϕ*,* s *is an assignment with* $Dom(s) \supseteq FV(\psi)$ *and* $\alpha \in \{I, II\}\}$*.*

   As before, we write P_I *for the positions in the form* (ψ, s, I) *and* P_{II} *for the positions in the form* (ψ, s, II)*.*

2. *At the beginning of the play, a pair* $(a, b) \in [0, 1]^2$ *is extracted according to the uniform distribution. If* $H(a, b) = s$*, the initial position is* (ϕ, s, II)*; if instead* $H(a, b)$ *is undefined, the play ends with a* draw *(neither player receives any payoff).*

3. *The set* $S(p)$ *of the successors of a given position and the set* W_α *of the winning positions for player* $\alpha \in \{I, II\}$ *is exactly as for the game* $G_s^M(\phi)$ *(Definition 1.3).*

Definition 3.6 ((a, b)-play, Complete play, Winning play)**.** *Let* M*,* H *and* ϕ *be as above, and let* $(a, b) \in Dom(H)$*. Then an* (a, b)-play *is a sequence* $p_1 \ldots p_n$ *for some* $n \geq 0$*, where*

- $p_1 = (\phi, H(a, b), II)$*;*
- *For all* $i < n$*,* $p_{i+1} \in S(p_i)$*.*

Such a play is complete *if* p_n *is a terminal position, and is* winning *for Player* $\alpha \in \{I, II\}$ *if and only if the last position is winning for* α*, in the sense of Definition 1.3.*

Definition 3.7 (Lottery-augmented strategy, Play following a strategy)**.** *A* strategy $\underline{\sigma}$ *for Player* I *in* $\underline{G}_H^M(\phi)$ *is a family* $\{\sigma_a : a \in [0, 1]\}$*, where each* σ_a *is a uniform strategy for* $G_{H(a,*)}^M(\phi)$*, and*

$$H(a, *) = \{H(a, b) : b \in [0, 1], H(a, b) \textit{ is defined}\}.$$

Player I is said to follow *a strategy $\underline{\sigma}$ in an (a,b)-play $p_1 \dots p_n$ if and only if, for all $i \in 1 \dots n-1$,*

$$p_i \in P_I \Rightarrow p_{i+1} = \sigma_a(p_i).$$

Analogously, a strategy $\underline{\tau}$ for Player II in $\underline{G}_H^M(\phi)$ is a family $\{\tau_b : b \in [0,1]\}$, where each τ_b is a uniform strategy for $G^M_{H(*,b)}(\phi)$ and

$$H(*,b) = \{H(a,b) : a \in [0,1], H(a,b) \text{ is defined}\}.$$

and Player II is said to follow *$\underline{\tau}$ in an (a,b)-play $p_1 \dots p_n$ if and only if*

$$p_i \in P_{II} \Rightarrow p_{i+1} = \tau_b(p_i).$$

Given two strategies $\underline{\sigma}$, $\underline{\tau}$ and given $(a,b) \in Dom(H)$, we let

$$(\underline{\sigma}; \underline{\tau})_{a,b} := (\sigma_a; \tau_b)$$

where $(\sigma_a; \tau_b)$ is the only play of $G^M_{H(a,b)}(\phi)$ in which Player I follows σ_a and Player II follows τ_b.

Definition 3.8 (Measurable strategy). *A strategy $\underline{\sigma}$ (respectively $\underline{\tau}$) for Player I (respectively II) in $\underline{G}_H^M(\phi)$ is said to be* measurable *if and only if, for the parameter a (respectively b) ranging over $[0,1]$, the equivalence classes*

$$\|a\|_\sigma = \{a' \in [0,1] : \sigma_a = \sigma_{a'}\}$$

(respectively, $\|b\|_\tau = \{b' \in [0,1] : \tau_b = \tau_{b'}\}$) partition $[0,1]$ in finitely many Lebesgue-measurable intervals.

Definition 3.9 (Winning regions). *Let M be a finite first order model, let H be a strategy guide over a set V of variables, let ϕ be a DF-Logic formula with $FV(\phi) \subseteq V$, and let $\underline{\sigma}$, $\underline{\tau}$ be strategies for Players I and II over $\underline{G}_H^M(\phi)$.*

Then the winning regions *for Player I and Player II are given by*

$${W_I}_H^M(\phi; \underline{\sigma}, \underline{\tau}) = \{(a,b) : \text{Player } I \text{ wins } (\underline{\sigma}; \underline{\tau})_{a,b}\}$$

and

$${W_{II}}_H^M(\phi; \underline{\sigma}, \underline{\tau}) = \{(a,b) : \text{Player } II \text{ wins } (\underline{\sigma}; \underline{\tau})_{a,b}\}.$$

For ease of notation, we will write $W_H^M(\phi; \underline{\sigma}, \underline{\tau})$ for ${W_{II}}_H^M(\phi; \underline{\sigma}, \underline{\tau})$ and $\widehat{W}_H^M(\phi; \underline{\sigma}, \underline{\tau})$ for ${W_I}_H^M(\phi; \underline{\sigma}, \underline{\tau})$.

Lemma 3.1. *$W_H^M(\phi; \underline{\sigma}, \underline{\tau}) \cup \widehat{W}_H^M(\phi; \underline{\sigma}, \underline{\tau}) = Dom(H)$ and $W_H^M(\phi; \underline{\sigma}, \underline{\tau}) \cap \widehat{W}_H^M(\phi; \underline{\sigma}, \underline{\tau}) = \emptyset$.*

Proof. Let $a, b \in \mathbb{R}$ be such that $H(a,b)$ is defined. Then the play $(\underline{\sigma}; \underline{\tau})_{a,b}$ is winning for either Player I or Player II, and hence

$$(a,b) \in W_H^M(\phi; \underline{\sigma}, \underline{\tau}) \cup \widehat{W}_H^M(\phi; \underline{\sigma}, \underline{\tau}).$$

Furthermore, no play $(\underline{\sigma}, \underline{\tau})_{a,b}$ is winning for both players, and therefore

$$W_H^M(\phi; \underline{\sigma}, \underline{\tau}) \cap \widehat{W}_H^M(\phi; \underline{\sigma}, \underline{\tau}) = \emptyset.$$

□

Definition 3.10 (Value of a strategy profile). *Let M, H, ϕ, $\underline{\sigma}$ and $\underline{\tau}$ be as above. Then the value of the strategy profile $(\underline{\sigma}, \underline{\tau})$ is the probability that, once the point $(a, b) \in [0, 1]^2$ has been extracted according to the uniform distribution, the play $(\underline{\sigma}, \underline{\tau})_{p,q}$ is winning for Player II, that is,*

$$\underline{V}_H^M(\phi; \underline{\sigma}, \underline{\tau}) = \int_{W_H^M(\phi;\underline{\sigma},\underline{\tau})} dxdy.$$

Analogously,

$$\widehat{\underline{V}}_H^M(\phi; \underline{\sigma}, \underline{\tau}) = \int_{\widehat{W}_H^M(\phi;\underline{\sigma},\underline{\tau})} dxdy.$$

Proposition 3.2. *Let M, H, ϕ, $\underline{\sigma}$ and $\underline{\tau}$ be as above. Then*

$$\underline{V}_H^M(\phi; \underline{\sigma}, \underline{\tau}) + \widehat{\underline{V}}_H^M(\phi; \underline{\sigma}, \underline{\tau}) = \int_{Dom(H)} dxdy.$$

Proof. By Lemma 3.1,

$$\int_{Dom(H)} dxdy = \int_{W_H^M(\phi;\underline{\sigma},\underline{\tau}) \cup \widehat{W}_H^M(\phi;\underline{\sigma},\underline{\tau})} dxdy = \int_{W_H^M(\phi;\underline{\sigma},\underline{\tau})} dxdy + \int_{\widehat{W}_H^M(\phi;\underline{\sigma},\underline{\tau})} dxdy =$$
$$= \underline{V}_H^M(\phi; \underline{\sigma}, \underline{\tau}) + \widehat{\underline{V}}_H^M(\phi; \underline{\sigma}, \underline{\tau})$$

as required. □

Definition 3.11 (Equilibria in $\underline{G}_H^M(\phi)$). *Let M, H and ϕ be as usual. Then a couple of strategies $\underline{\sigma}$, $\underline{\tau}$ for $\underline{G}_H^M(\phi)$ are said to be in* equilibrium *if and only if*

$$\underline{V}_H^M(\phi; \underline{\sigma}, \underline{\tau}') \leq \underline{V}_H^M(\phi; \underline{\sigma}, \underline{\tau}) \leq \underline{V}_H^M(\phi; \underline{\sigma}', \underline{\tau})$$

for all pairs of strategies $\underline{\sigma}'$ and $\underline{\tau}'$.

Lemma 3.2. *Let $(\underline{\sigma}, \underline{\tau})$ and $(\underline{\sigma}', \underline{\tau}')$ be equilibria for $\underline{G}_H^M(\phi)$. Then*

$$\underline{V}_H^M(\phi; \underline{\sigma}, \underline{\tau}) = \underline{V}_H^M(\phi; \underline{\sigma}', \underline{\tau}').$$

Definition 3.12 ($\underline{V}_H^M(\phi)$). *Suppose that $(\underline{\sigma}, \underline{\tau})$ is an equilibrium in $\underline{G}_H^M(\phi)$: then we say that*

$$\underline{V}_H^M(\phi) = \underline{V}_H^M(\phi; \underline{\sigma}, \underline{\tau}).$$

Definition 3.13 (H_s). *Let s be any variable assignment over a finite model M. Then we define H_s as the unique strategy guide such that*

$$H_s(a, b) = s \text{ for all } (a, b) \in [0, 1]^2.$$

As the graphical representation of mixed strategies suggests, mixed strategies in a game $G_s^M(\phi)$ correspond precisely to (pure, uniform) strategies in $\underline{G}_{H_s}^M(\phi)$, and vice versa:

Definition 3.14 ($Lot(\mu)$, $Lot(\nu)$)**.** *Let ϕ be a DF-Logic formula, let M be a suitable finite structure, let s be a variable assignment with domain M, let $\sigma^1 \dots \sigma^n$ be an enumeration of all (pure, uniform) strategies for Player I in $G_s^M(\phi)$ and let μ be a mixed strategy for Player I in the same game.*

Then let $A_1 \dots A_n$ be the segments of the unit interval defined as

$$\begin{aligned} &A_1 = [0, \mu(\sigma^1));\\ &A_2 = [\mu(\sigma^1), \mu(\sigma^1) + \mu(\sigma^2));\\ &\dots\\ &A_n = \left[\sum_{k=1}^{n-1} \mu(\sigma^k), 1\right] \end{aligned}$$

and let $Lot(\mu)$ be the strategy for Player I in $\underline{G}_{H_s}^M(\phi)$ such that, for all $i \in 1 \dots n$ and for all $a \in A_i$, $Lot(\mu)_a = \sigma^i$.

Analogously, if $\tau^1 \dots \tau^m$ are the pure strategies for Player II in the same game and ν is a mixed strategy we let $B_1 \dots B_m$ be the segments of the unit interval defined as

$$\begin{aligned} &B_1 = [0, \nu(\tau^1));\\ &B_2 = [\nu(\tau^1), \nu(\tau^1) + \nu(\tau^2));\\ &\dots\\ &B_m = \left[\sum_{k=1}^{m-1} \nu(\tau^k), 1\right] \end{aligned}$$

and we let the strategy $Lot(\nu)$ for Player II in $\underline{G}_{H_s}^M(\phi)$ be such that, for all $j \in 1 \dots m$ and for all $b \in B_j$, $Lot(\nu)_b = \tau^j$.

Proposition 3.3. *Let ϕ be a DF-Logic formula, let M be a suitable finite structure, let s be an assignment with $FV(\phi) \subseteq Dom(s)$, and let μ, ν be two mixed strategies for Players I and II in $G_s^M(\phi)$.*

Then

$$V_s^M(\phi; \mu, \nu) = \underline{V}_{H_s}^M(\phi; Lot(\mu), Lot(\nu)).$$

Proof. By definition,

$$\underline{V}_{H_s}^M(\phi; Lot(\mu), Lot(\nu)) = \int_{W_{H_s}^M(\phi; Lot(\mu), Lot(\tau))} dxdy = \sum_{i \in I} \sum_{j \in J} V_s^M(\phi; \sigma^i, \tau^j) \int_{A_i \times B_j} dxdy =$$

$$= \sum_{i \in I} \sum_{j \in J} \mu(\sigma^i) \nu(\tau^j) V_s^M(\phi; \sigma^i, \tau^j) = V_s^M(\phi; \mu, \nu).$$

□

Definition 3.15. *$Mix(\underline{\sigma})$, $Mix(\underline{\tau})$ Let $\underline{\sigma}$ and $\underline{\tau}$ be strategies for Players I and II in $\underline{G}_{H_s}^M(\phi)$. Then $Mix(\underline{\sigma})$ and $Mix(\underline{\tau})$ are the mixed strategies for Players I and II in $G_s^M(\phi)$ defined as*

$$\begin{aligned} Mix(\underline{\sigma})(\sigma^i) &= \int_{\{a \in [0,1] : \sigma_a = \sigma^i\}} dx;\\ Mix(\underline{\tau})(\tau^j) &= \int_{\{b \in [0,1] : \tau_b = \tau^j\}} dy \end{aligned}$$

for all pure strategies σ^i and τ^j.

Proposition 3.4. *Let M, s, and ϕ be as usual, and let $\underline{\sigma}$, $\underline{\tau}$ be strategies for Player I and Player II in $\underline{G}^M_{H_s}(\phi)$. Then*

$$\underline{V}^M_{H_s}(\phi; \underline{\sigma}, \underline{\tau}) = V^M_s(\phi; Mix(\underline{\sigma}), Mix(\underline{\tau})).$$

Proof. For each suitable i and j, let

$$A^{\underline{\sigma}}_i = \{a \in [0,1] : \sigma_a = \sigma^i\}$$

and

$$B^{\underline{\tau}}_j = \{b \in [0,1] : \tau_b = \tau^j\}.$$

Then

$$\underline{V}^M_{H_s}(\phi; \underline{\sigma}, \underline{\tau}) = \int_{W^M_{H_s}(\phi; \underline{\sigma}, \underline{\tau})} dxdy = \sum_i \sum_j V^M_s(\phi; \sigma^i, \tau^j) \int_{A^{\underline{\sigma}}_i \times B^{\underline{\tau}}_j} dxdy =$$
$$= \sum_i \sum_j Mix(\underline{\sigma})(\sigma^i), Mix(\underline{\tau})(\tau^j) V^M_s(\phi; \sigma^i, \tau^j) = V^M_s(\phi; Mix(\underline{\sigma}), Mix(\underline{\tau}))$$

as required. □

Lemma 3.3. *Let μ and ν be mixed strategies for Players I and II in $G^M_s(\phi)$. Then $Mix(Lot(\mu)) = \mu$ and $Mix(Lot(\nu)) = \nu$.*

Proof. By definition, for all pure strategies σ^i of Player I

$$Mix(Lot(\mu))(\sigma^i) = \int_{\{a : Lot(\mu)_a = \sigma^i\}} dx = \int_{A^i} dx = \mu(\sigma^i)$$

and

$$Mix(Lot(\nu))(\tau^j) = \int_{\{b : Lot(\mu)_b = \sigma^j\}} dy = \int_{B^j} dy = \nu(\tau^j)$$

where A^i and B^j are the sets used in Definition 3.14. □

In general, we do not know if there exists an equilibrium in a game $\underline{G}^M_H(\phi)$. However, the following corollary follows from Theorem 2.1, Propositions 3.3 and 3.4 and Lemma 3.3:

Corollary 3.1. *For all formulas ϕ, finite models M and assignments s with $FV(\phi) \subseteq Dom(s)$, the game $\underline{G}^M_{H_s}(\phi)$ has an equilibrium and*

$$V^M_s(\phi) = \underline{V}^M_{H_s}(\phi).$$

Proof. By Theorem 2.1, there exists a Nash equilibrium (μ, ν) for the game $G^M_s(\phi)$. Consider now the strategy profile $(Lot(\mu), Lot(\nu))$ for $\underline{G}^M_{H_s}(\phi)$: by Proposition 3.3,

$$\underline{V}^M_{H_s}(\phi; Lot(\mu), Lot(\nu)) = V^M_s(\phi; \mu, \nu) = V^M_s(\phi).$$

Moreover, $(\underline{Lot}(\mu), Lot(\nu))$ is an equilibrium for $\underline{G}^M_{H_s}(\phi)$: indeed, suppose that

$$\underline{V}^M_{H_s}(\phi; \underline{\sigma}', Lot(\nu)) < \underline{V}^M_{H_s}(\phi; Lot(\mu), Lot(\nu)).$$

Then

$$V^M_s(\phi; Mix(\underline{\sigma}'), \nu) = V^M_s(\phi; Mix(\underline{\sigma}'), Mix(Lot(\nu))) = \underline{V}^M_{H_s}(\phi; \underline{\sigma}', Lot(\nu)) < \\ < \underline{V}^M_{H_s}(\phi; Lot(\mu), Lot(\nu)) = V^M_s(\phi; \mu, \nu),$$

where the first equality follows from Lemma 3.3, the second follows from Proposition 3.4 and the third follows from Proposition 3.3: and therefore, (μ, ν) is not a Nash equilibrium for the game $G^M_s(\phi)$, contradicting our assumption.

Analogously, if

$$\underline{V}^M_{H_s}(\phi; Lot(\mu), \underline{\tau}') > \underline{V}^M_{H_s}(\phi; Lot(\mu), Lot(\nu))$$

then

$$V^M_s(\phi; \mu, Mix(\underline{\tau}')) = V^M_s(\phi; Mix(Lot(\mu)), Mix(\underline{\tau}')) = \underline{V}^M_{H_s}(\phi; Lot(\mu), \underline{\tau}') > \\ > \underline{V}^M_{H_s}(\phi; Lot(\mu), Lot(\nu)) = V^M_s(\phi; \mu, \nu)$$

and again (μ, ν) is not a Nash equilibrium for $G^M_s(\phi)$.

So in conclusion $(Lot(\mu), Lot(\nu))$ is an equilibrium and

$$\underline{V}^M_{H_s}(\phi) = \underline{V}^M_{H_s}(\phi; Lot(\mu), Lot(\nu)),$$

as required. □

4 A compositional semantics for Lottery games

In this section, we will develop a similar result to Theorem 1.1 for lottery-augmented games, thus finding a compositional semantics for the computation of $V^M_H(\phi)$ (if it exists). By the last corollary of the previous section, this will also give us a compositional semantics for the computation of the Nash equilibria $V^M_s(\phi)$.

We will need to consider three operations over strategy guides:

Definition 4.1 (Transposition)**.** *Let M be a finite model, let $V \subseteq Var$, and let H be a strategy guide over the set V of variables. Then the* transpose *H^T of H is given by*

$$H^T(b, a) = H(a, b)$$

for all $(a, b) \in Dom(H)$.

Lemma 4.1. *If H is a strategy guide, H^T is a strategy guide too. Furthermore,*

$$\int_{Dom(H)} dxdy = \int_{Dom(H^T)} dxdy.$$

Proof. Let $\langle \mathcal{A}, \mathcal{B} \rangle$ be a grid for H. Then $\langle \mathcal{B}, \mathcal{A} \rangle$ is a grid for H^T: indeed, for all $b, b' \in B \in \mathcal{B}$ and $a, a' \in A \in \mathcal{A}$ we have that $H(a, b) = H(a', b')$ and hence that $H^T(b, a) = H^T(b', a')$, as required.

The proof of the equivalence $\int_{Dom(H)} dxdy = \int_{Dom(H^T)} dxdy$ is trivial. □

Definition 4.2 (Supplementation function, measurable supplementation function)**.** *Let M be a finite model and let $V \subseteq Var$. Then a family of functions $F = \{F_b : M^V \to M \mid b \in [0,1]\}$ is a* supplementation function *over the model M and the domain X, and it is* measurable *if and only if, for every $b \in [0, 1]$, the equivalence classes*

$$\|b\|_F = \{b' \in [0, 1] : F_{b'} = F_b\}$$

partition $[0, 1]$ in finitely many measurable sets.

Given a set of variables $W \subseteq V$, we say that F is determined *by W if and only if, for all assignments s, s' with domain V and all $b \in [0, 1]$,*

$$s_{|W} = s'_{|W} \Rightarrow F_b(s) = F_b(s').$$

Definition 4.3 (Supplementation)**.** *Let M be a finite model, let $V \subseteq Var$, let H be a strategy guide over the set V of variables, let $x \in Var$ and let F be a supplementation function over M with domain V. Then $H[F/x]$ is defined as*

$$H[F/x](a, b) = H(a, b)[F_b(H(a, b))/x]$$

for all $a, b \in [0, 1]$.

Lemma 4.2. *If H is a strategy guide and F is a measurable supplementation function, then $H[F/x]$ is a strategy guide.*

Proof. Let $\langle \mathcal{A}, \mathcal{B} \rangle$ be a grid that H respects, and define

$$\mathcal{B}' = \{B \cap \|b\|_F : b \in [0, 1]\}$$

where, as before,

$$\|b\|_F = \{b' \in [0, 1] : F(b') = F(b)\}.$$

Since each $B \cap \|b\|_F$ is measurable, $\langle \mathcal{A}, \mathcal{B}' \rangle$ is a grid; furthermore, if $a, a' \in A \in \mathcal{A}$ and $b, b' \in B \in \mathcal{B} \cap \|b\|_F \in \mathcal{B}'$, then

$$H[F/x](a, b) = H(a, b)[F_b(H(a, b))/x] = H(a', b')[F_{b'}(H(a', b'))/x] = H[F/x](a', b')$$

and therefore $H[F/x]$ respects $\langle \mathcal{A}, \mathcal{B}' \rangle$. □

Definition 4.4 (Splitting function)**.** *Let M be a finite model and let V be a set of variables. Then a* splitting function *Sp over M with domain V is a family of functions $\{Sp_b : M^V \to \{L, R\}\}$. Such a splitting function is* measurable *if and only if, for every $b \in [0, 1]$, the equivalence classes*

$$\|b\|_{Sp} = \{b' \in [0, 1] : Sp_b = Sp_{b'}\}$$

partition $[0, 1]$ into finitely many measurable sets.

Definition 4.5 (Splitting). *Let M be a finite model, let V be a set of variables, let H be a strategy guide over M and V, and let Sp be a splitting function over M with domain V. Then we say that* Sp splits H *into* H_1 *and* H_2*, where*

$$Dom(H_1) = \{(a, b) \in Dom(H) : Sp_b(H(a, b)) = L\},$$
$$H_1(a, b) = H(a, b) \text{ for all } (a, b) \in Dom(H_1)$$

and

$$Dom(H_2) = \{(a, b) \in Dom(H) : Sp_b(H(a, b)) = R\},$$
$$H_2(a, b) = H(a, b) \text{ for all } (a, b) \in Dom(H_2).$$

Lemma 4.3. *If H is a strategy guide, Sp is a measurable splitting function and Sp splits H into H_1 and H_2 then H_1 and H_2 are strategy guides.*

Proof. Let $\langle \mathcal{A}, \mathcal{B} \rangle$ be a grid that H respects, and define

$$\mathcal{B}' = \{B \cap \|b\|_{Sp} : B \in \mathcal{B}\}$$

where $\|b\|_{Sp}$ is defined as in Definition 4.4.

Then each $B \cap \|b\|_{Sp}$ is measurable, and hence $\langle \mathcal{A}, \mathcal{B}' \rangle$ is a grid.

Let us verify that H_1 and H_2 respect $\langle \mathcal{A}, \mathcal{B}' \rangle$. Suppose that $a, a' \in A \in \mathcal{A}$ and $b, b' \in B \in \mathcal{B} \cap \|b\|_{Sp} \in \mathcal{B}'$, and furthermore that $(a, b), (a', b') \in Dom(H)$: then, since H respects $\langle \mathcal{A}, \mathcal{B} \rangle$ and $\mathcal{B}'$ is a refinement of $\mathcal{B}$, $H(a, b) = H(a', b')$. Moreover, as $b' \in \|b\|_{Sp}$ we have that $Sp_b(H(a, b)) = Sp_{b'}(H(a', b'))$.

By the definition of splitting, this implies that $(a, b) \in H_i \Leftrightarrow (a', b') \in H_i$ for $i \in \{1, 2\}$; furthermore, if $(a, b), (a', b') \in H_i$ we have that $H_i(a, b) = H(a, b) = H(a', b') = H_i(a', b')$, as required. ☐

We may now write and prove the main result of this work:

Theorem 4.1. *Let M be a finite model, let ϕ be a formula of the same signature, and let H be a strategy guide over the model M and the variables $V \supseteq FV(\phi)$. Furthermore, suppose that the game $\underline{G}_H^M(\phi)$ has an equilibrium.*

Then

1. *If ϕ is atomic,*
$$V_H^M(\phi) = \int_{\phi \circ H} dxdy$$
where
$$\phi \circ H = \{(a, b) \in Dom(H) : M \models_{H(a,b)} \phi\};$$

2. *If ϕ is $\neg\psi$,*
$$V_H^M(\phi) = \int_{Dom(H)} dxdy - V_{H^T}^M(\psi);$$

3. *If* ϕ *is* $\psi_1 \vee \psi_2$,

$$V_H^M(\phi) = \max_{Sp}(V_{H_1}^M(\psi_1) + V_{H_2}^M(\psi_2))$$

where the maximum is taken over all measurable splitting functions Sp*, and* H_1, H_2 *are the strategy guides that* H *is split into according to* Sp*;*

4. *If* ϕ *is* $\exists x_{\backslash W}\psi$,

$$V_H^M(\phi) = \max_F V_{H[F/x]}^M(\phi)$$

where the maximum is taken over all supplementing functions F *which are determined by* W*.*

Proof. 1. If ϕ is atomic, the game ends as soon as the initial assignment $s = H(a,b)$ is calculated, and therefore the only strategies available to the two players are the trivial ones $\underline{\sigma}_0$ and $\underline{\tau}_0$. If $H(a,b)$ is undefined, neither player receives any payoff; otherwise, Player II wins if and only if $M \models_s \phi$. Thus the set of pairs (a,b) such that Player II wins the play $(\underline{\sigma}_0, \underline{\tau}_0)_{a,b}$ is

$$W_H^M(\underline{\sigma}_0, \underline{\tau}_0) = \{(a,b) : II \text{ wins } (\underline{\sigma}_0; \underline{\tau}_0)_{a,b}\} = \{(a,b) \in Dom(H) : M \models_{H(a,b)} \phi\} = \phi \circ H.$$

Then, by definition,

$$V_H^M(\phi) = V_H^M(\phi; \underline{\sigma}_0, \underline{\tau}_0) = \int_{W_H^M(\phi;\underline{\sigma}_0,\underline{\tau}_0)} dxdy = \int_{\phi\circ H} dxdy,$$

as required.

2. Suppose that ϕ is $\neg\psi$ and let $(\underline{\sigma}, \underline{\tau})$ be an equilibrium for $\underline{G}_H^M(\neg\psi)$. Then consider the strategies $\underline{\tau}^T, \underline{\sigma}^T$ for $\underline{G}_{H^T}^M(\psi)$ defined as

$$\tau_b^T(\theta, s, I) = \tau_b(\theta, s, II);$$
$$\sigma_a^T(\theta, s, II) = \sigma_a(\theta, s, I)$$

for all subformulas θ of ψ: then every (a,b)-play $(\underline{\sigma}, \underline{\tau})_{a,b}$ of $\underline{G}_H^M(\phi)$ contains a (b,a)-play $(\underline{\tau}^T, \underline{\sigma}^T)_{b,a}$ of $\underline{G}_{H^T}^M$ with the roles of the players swapped, and hence

$$W_H^M(\neg\psi; \underline{\sigma}; \underline{\tau}) = \widehat{W}_H^M(\neg\psi; \underline{\tau}^T; \underline{\sigma}^T).$$

Therefore,

$$\underline{V}_H^M(\neg\psi) = \int_{W_H^M(\neg\psi;\underline{\sigma},\underline{\tau})} dxdy = \int_{\widehat{W}_{H^T}^M(\psi;\underline{\tau}^T,\underline{\sigma}^T)} dxdy =$$
$$= \int_{Dom(H^T)} dxdy - \int_{W_{H^T}^M(\psi;\underline{\tau}^T,\underline{\sigma}^T)} dxdy = \int_{Dom(H)} dxdy - \underline{V}_{H^T}^M(\psi; \underline{\tau}^T, \underline{\sigma}^T).$$

Furthermore, $(\underline{\tau}^T, \underline{\sigma}^T)$ is an equilibrium for $\underline{G}_{H^T}^M(\psi)$: indeed, if it were the case that

$$\underline{V}_{H^T}^M(\psi; \underline{\rho}, \underline{\sigma}^T) < \underline{V}_{H^T}^M(\psi; \underline{\tau}^T, \underline{\sigma}^T)$$

for some strategy $\underline{\rho}$ of Player I then, for the strategy $\underline{\rho}'$ of Player II in $\underline{G}^M_H(\neg\psi)$ defined as

$$\rho'_b(\neg\psi, s, II) = (\psi, s, I);$$
$$\rho'_b(\theta, s, II) = \rho_b(\theta, s, I) \text{ for all subformulas } \theta \text{ of } \psi$$

it would hold that $W^M_H(\neg\psi; \underline{\sigma}, \underline{\rho}') = \widehat{W}^M_{H^T}(\psi; \underline{\rho}, \underline{\sigma}^T)$ and hence that

$$\underline{V}^M_H(\neg\psi; \underline{\sigma}, \underline{\rho}') = \int_{Dom(H)} dxdy - \underline{V}^M_{H^T}(\psi; \rho, \underline{\sigma}^T) > \int_{Dom(H)} dxdy - \underline{V}^M_{H^T}(\psi; \underline{\tau}^T, \underline{\sigma}^T) = \underline{V}^M_H(\neg\psi; \underline{\sigma}, \underline{\tau})$$

and thus $(\underline{\sigma}, \underline{\tau})$ would not an equilibrium for $\underline{G}^M_H(\neg\psi)$. A similar argument takes care of the strategies for the other player, so in conclusion $(\underline{\tau}^T, \underline{\sigma}^T)$ is an equilibrium for $\underline{G}^M_{H^T}(\psi)$ and

$$\underline{V}^M_H(\neg\psi) = \underline{V}^M_H(\neg\psi; \underline{\sigma}, \underline{\tau}) = \int_{Dom(H)} dxdy - \underline{V}^M_{H^T}(\psi; \underline{\tau}^T, \underline{\sigma}^T) = \int_{Dom(H)} dxdy - \underline{V}^M_{H^T}(\psi)$$

as required.

3. Suppose that ϕ is $\psi_1 \vee \psi_2$, and let $(\underline{\sigma}; \underline{\tau})$ be a Nash equilibrium for $\underline{G}^M_H(\phi)$. Then consider the splitting function Sp defined as follows:

$$Sp_b(s) = \begin{cases} L & \text{if } \tau_b(\psi_1 \vee \psi_2, s, II) = (\psi_1, s, II); \\ R & \text{if } \tau_b(\psi_1 \vee \psi_2, s, II) = (\psi_2, s, II). \end{cases}$$

Sp is also measurable: indeed, $\underline{\tau}$ is a measurable strategy, and hence for every $b \in [0, 1]$

$$\begin{aligned} \|b\|_{Sp} &= \{b' : Sp_b = Sp_{b'}\} = \{b' : \forall s, \tau_{b'}(\psi_1 \vee \psi_2, s, II) = (\psi_1, s, II) \Leftrightarrow Sp_b(s) = L\} = \\ &= \bigcup\{\|b'\|_{\underline{\tau}} : \forall s, \tau_{b'}(\psi_1 \vee \psi_2, s, II) = (\psi_1, s, II) \Leftrightarrow Sp_b(s) = L\}, \end{aligned}$$

where each $\|b'\|_{\underline{\tau}} = \{b'' : \tau_{b''} = \tau_{b'}\}$ is measurable by hypothesis, and there exist only finitely many distinct $\|b'\|_{\underline{\tau}}$. But the union of finitely many measurable sets is measurable, and hence Sp is measurable, as required.

Thus, Sp splits H into the two strategy guides

$$H_1 = \{(a, b) \in Dom(H) : \tau_b(\psi_1 \vee \psi_2, H(a, b), II) = (\psi_1, H(a, b), II)\}$$

and

$$H_2 = \{(a, b) \in Dom(H) : \tau_b(\psi_1 \vee \psi_2, H(a, b), II) = (\psi_2, H(a, b), II)\}.$$

Now, let us consider the strategies $\underline{\sigma}'$, $\underline{\tau}'$, $\underline{\sigma}''$ and $\underline{\tau}''$ for the two players, in $\underline{G}^M_{H_1}(\psi_1)$ and $\underline{G}^M_{H_2}(\psi_2)$ respectively, defined as

$$\left.\begin{aligned} \sigma'_a(\theta, s', I) &= \sigma_a(\theta, s', I), \\ \tau'_b(\theta, s', II) &= \tau_b(\theta, s', II) \end{aligned}\right\} \text{ for all subformulas } \theta \text{ of } \psi_1;$$
$$\left.\begin{aligned} \sigma''_a(\chi, s', I) &= \sigma_a(\chi, s', I), \\ \tau''_b(\chi, s', II) &= \tau_b(\chi, s', II) \end{aligned}\right\} \text{ for all subformulas } \chi \text{ of } \psi_2.$$

Now, $(\underline{\sigma}', \underline{\tau}')$ is an equilibrium for $\underline{G}^M_{H_1}(\psi_1)$ and $(\underline{\sigma}'', \underline{\tau}'')$ is an equilibrium for $\underline{G}^M_{H_2}(\psi_2)$, otherwise $(\underline{\sigma}, \underline{\tau})$ would not be an equilibrium for $\underline{G}^M_H(\phi)$; and moreover, $W^M_H(\phi; \underline{\sigma}, \underline{\tau})$ is the disjoint union of $W^M_H(\psi_1; \underline{\sigma}', \underline{\tau}')$ and $W^M_H(\psi_2; \underline{\sigma}'', \underline{\tau}'')$. Therefore,

$$\underline{V}^M_H(\phi) = \underline{V}^M_H(\phi; \underline{\sigma}, \underline{\tau}) = \int_{W^M_H(\phi; \underline{\sigma}, \underline{\tau})} dxdy = \int_{W^M_{H_1}(\psi_1; \underline{\sigma}', \underline{\tau}')} dxdy + \int_{W^M_{H_2}(\psi_2; \underline{\sigma}'', \underline{\tau}'')} dxdy =$$

$$= \underline{V}^M_{H_1}(\psi_1; \underline{\sigma}', \underline{\tau}') + \underline{V}^M_{H_2}(\psi_2; \underline{\sigma}'', \underline{\tau}'') = \underline{V}^M_{H_1}(\psi_1) + \underline{V}^M_{H_2}(\psi_2).$$

Now, suppose that Sp' is another measurable function splitting H into H'_1 and H'_2, and let $(\underline{\sigma}^{(1)}, \underline{\tau}^{(1)})$ and $(\underline{\sigma}^{(2)}, \underline{\tau}^{(2)})$ be equilibria for $\underline{G}^M_{H'_1}(\psi_1)$ and $\underline{G}^M_{H'_2}(\psi_2)$ respectively. Then define the strategies $\underline{\sigma}'$ and $\underline{\tau}'$ for Player I and II in $\underline{G}^M_H(\psi_1 \vee \psi_2)$ as

$$\sigma'_a(\theta, s', I) = \begin{cases} \sigma^{(1)}_a(\theta, s', I) & \text{if } \theta \text{ is a subformula of } \psi_1; \\ \sigma^{(2)}_a(\theta, s', I) & \text{if } \theta \text{ is a subformula of } \psi_2 \end{cases}$$

and as

$$\tau'_b(\psi_1 \vee \psi_2, s, II) = \begin{cases} (\psi_1, s, II) & \text{if } Sp'_b(s) = L; \\ (\psi_2, s, II) & \text{if } Sp'_b(s) = R; \end{cases}$$

$$\tau'_b(\theta, s', II) = \begin{cases} \tau^{(1)}_b(\theta, s', II) & \text{if } \theta \text{ is a subformula of } \psi_1; \\ \tau^{(2)}_b(\theta, s', II) & \text{if } \theta \text{ is a subformula of } \psi_2. \end{cases}$$

Then $W^M_H(\psi_1 \vee \psi_2; \underline{\sigma}', \underline{\tau}')$ is the disjoint union of $W^M_{H'_1}(\psi_1; \underline{\sigma}^{(1)}, \underline{\tau}^{(1)})$ and $W^M_{H'_2}(\psi_2; \underline{\sigma}^{(2)}, \underline{\tau}^{(2)})$, and therefore

$$\underline{V}^M_H(\psi_1 \vee \psi_2; \underline{\sigma}', \underline{\tau}') = \int_{W^M_H} (\psi_1 \vee \psi_2; \underline{\sigma}', \underline{\tau}') dxdy = \int_{W^M_{H'_1}(\psi_1; \underline{\sigma}^{(1)}, \underline{\tau}^{(1)})} dxdy + \int_{W^M_{H'_2}(\psi_2; \underline{\sigma}^{(2)}, \underline{\tau}^{(2)})} dxdy =$$

$$= \underline{V}^M_{H'_1}(\psi_1; \underline{\sigma}^{(1)}, \underline{\tau}^{(1)}) + \underline{V}^M_{H'_2}(\psi_2; \underline{\sigma}^{(2)}, \underline{\tau}^{(2)}) = V^M_{H'_1}(\psi_1) + V^M_{H'_2}(\psi_2).$$

The pair of strategies $(\widehat{\underline{\sigma}}, \widehat{\underline{\tau}})$ may not be an equilibrium for $\underline{G}^M_H(\phi)$, but we know that Player I cannot decrease the expected utility by modifying $\underline{\sigma}'$ because otherwise either $(\underline{\sigma}^{(1)}, \underline{\tau}^{(1)})$ or $(\underline{\sigma}^{(2)}, \underline{\tau}^{(2)})$ would not be an equilibrium for the corresponding game. Therefore,

$$\underline{V}^M_{H_1}(\psi_1) + \underline{V}^M_{H_2}(\psi_2) = \underline{V}^M_H(\psi_1 \vee \psi_2; \underline{\sigma}', \underline{\tau}') \leq \underline{V}^M_H(\psi_1 \vee \psi_2; \underline{\sigma}, \underline{\tau}) = V^M_H(\psi_1 \vee \psi_2),$$

which proves that the splitting function Sp is optimal.

4. Suppose that ϕ is $\exists x_{\setminus W}\psi$, and let $(\underline{\sigma}, \underline{\tau})$ be an equilibrium for $\underline{G}^M_H(\phi)$. Then, for each $b \in [0,1]$, let $F_b : M^V \to M$ be such that

$$\tau_b(\exists x_{\setminus W}\psi, s, II) = (\psi, s[F_b(s)/x], II).$$

By definition, $F = \{F_b : b \in [0,1]\}$ is determined by W, and it is measurable since $\underline{\tau}$ is measurable; let us then consider the strategies $\underline{\sigma}'$, $\underline{\tau}'$ for $\underline{G}^M_{H[F/x]}(\psi)$ defined by

$$\sigma'_a(\theta, s', I) = \sigma_a(\theta, s', I);$$
$$\tau'_b(\theta, s', II) = \tau_b(\theta, s', II)$$

for all subformulas θ of ψ.

Then $W_H^M(\exists x_{\backslash W}\psi; \underline{\sigma}, \underline{\tau}) = W_{H[F/x]}^M(\psi; \underline{\sigma}', \underline{\tau}')$, and hence

$$\underline{V}_H^M(\exists x_{\backslash W}\psi; \underline{\sigma}, \underline{\tau}) = V_{H[F/x]}^M(\psi; \underline{\sigma}', \underline{\tau}').$$

Furthermore, $(\underline{\sigma}', \underline{\tau}')$ is an equilibrium in $\underline{G}_{H[F/x]}^M(\psi)$, since if either player could improve their payoff by changing strategies in this game they could also improve their payoff in $\underline{G}_H^M(\exists x_{\backslash W}\psi; \underline{\sigma}, \underline{\tau})$, and thus

$$\underline{V}_H^M(\exists x_{\backslash W}\psi) = \underline{V}_{H[F/x]}^M(\psi).$$

Now suppose that $F' = \{F_b' : M^V \to M \mid b \in [0,1]\}$ is another measurable supplementation function which is determined by W, and let $(\underline{\sigma}', \underline{\tau}')$ be an equilibrium for $\underline{G}_{H[F/x]}^M(\psi)$.

Then, extend $\underline{\sigma}'$ and $\underline{\tau}'$ to strategies $\underline{\sigma}''$ and $\underline{\tau}''$ for $\underline{G}_H^M(\exists x_{\backslash W}\psi)$ by setting

$$\sigma_a''(\theta, s, I) = \sigma_a'(\theta, s', I)$$

and

$$\tau_b''(\exists x_{\backslash W}\psi, s, II) = (\psi, s[F_b'(s)/x], II);$$
$$\tau_b''(\theta, s', II) = \tau_b'(\theta, s', II).$$

Then $W_{H[F'/x]}^M(\psi; \underline{\sigma}', \underline{\tau}') = W_H^M(\exists x_{\backslash W}\psi; \underline{\sigma}'', \underline{\tau}'')$, and hence

$$\underline{V}_{H[F'/x]}^M(\psi; \underline{\sigma}', \underline{\tau}') = \underline{V}_H^M(\exists x_{\backslash W}\psi; \underline{\sigma}'', \underline{\tau}'').$$

The pair of strategies $(\underline{\sigma}', \underline{\tau}')$ may not be an equilibrium for $\underline{G}_H^M(\exists x_{\backslash W}\psi)$, but we know that Player I cannot decrease the expected utility by modifying $\underline{\sigma}'$ because otherwise $(\underline{\sigma}'', \underline{\tau}'')$ would not be an equilibrium for $\underline{G}_{H[F'/x]}^M(\psi)$. Therefore,

$$V_{H[F'/x]}^M(\psi) = V_{H[F'/x]}^M(\psi; \underline{\sigma}', \underline{\tau}') = V_H^M(\exists x\psi; \underline{\sigma}'', \underline{\tau}'') \leq V_H^M(\exists x_{\backslash W}\psi) = V_{H[F/x]}^M(\psi)$$

which proves that F is optimal.

□

This result, combined with Corollary 3.1, yields a compositional semantics equivalent to Equilibrium Semantics, and this concludes this work.

References

[1] A. Blass and Y. Gurevich. Henkin Quantifiers and Complete Problems. *Annals of Pure and Applied Logic*, 32:1–16, 1986.

[2] P. Galliani. Game Values and Equilibria for Undetermined Sentences of Dependence Logic. MSc Thesis. ILLC Publications, MoL–2008–08, 2008.

[3] P. Galliani. Probabilistic Dependence Logic. Forthcoming, 2008.

[4] L. Henkin. Some Remarks on Infinitely Long Formulas. In *Infinitistic Methods. Proc. Symposium on Foundations of Mathematics*, pages 167–183. Pergamon Press, 1961.

[5] J. Hintikka. *The Principles of Mathematics Revisited.* Cambridge University Press, 1996.

[6] W. Hodges. Compositional Semantics for a Language of Imperfect Information. *Journal of the Interest Group in Pure and Applied Logics*, 5 (4):539–563, 1997.

[7] H.W. Kuhn. Extensive Games and the Problem of Information. In *Contributions to the Theory of Games III*, pages 193–216. Princeton University Press, 1953.

[8] J. Von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische Annalen*, 100(1):295–320, 1928.

[9] M. Sevenster. Branches of imperfect information: logic, language, and computation. *PhD Thesis, Institute for Logic, Language and Computation, DS–2006–06*, 2006.

[10] M. Sevenster and G. Sandu. Equilibrium semantics of languages of imperfect information. *Annals of Pure and Applied Logic*, 161:618–631, 2010.

[11] T. Tulenheimo. Independence Friendly Logic. http://plato.stanford.edu/entries/logic-if/, 2009.

[12] J. Väänänen. *Dependence Logic.* Cambridge University Press, 2007.

[13] J. Väänänen. Team Logic. In J. van Benthem, D. Gabbay, and B. Löwe, editors, *Interactive Logic. Selected Papers from the 7th Augustus de Morgan Workshop*, pages 281–302. Amsterdam University Press, 2007.

A Comparison of Independence Friendly logic and Dependence logic

Theo M.V. Janssen

Introduction

Independence Friendly logic, henceforth IF logic, is introduced by Hintikka (see e.g. [1]). It is a logic in which it can be indicated that some quantifier Q_1 which occurs within the scope of a quantifier Q_2, nevertheless is independent of Q_2. The logic is claimed to be useful for many phenomena: in natural language semantics examples are the *de dicto-de re* ambiguity and branching quantifier sentences, and in mathematics an example is the distinction between *continuity* and *uniform continuity.*

Dependence logic, introduced by Väänänen, see [2], is a logic with the complementary approach. For a quantifier Q_3 it can be indicated on which subset of quantifiers it depends (from the quantifiers that have scope over Q_3). So instead of marking independence as a special case, dependency is marked. This logic should be applicable to the above mentioned phenomena as well.

In a certain sense the two logics seem to have the same expressive power: for a sentence (i.e. a closed formula) in the one logic there is one in the other logic that is true in the same models (but that is misleading, because the logics have richer notions of meaning). For open formulas there is not such an relation: for an open formula in the one logic there is no formula at all in the other logic that it true with respect to the same sets of assigments. As a consequence there cannot be compositional translation from the one logic into the other.

In this contribution we will compare these two logics for their possibilities in (two) applications; which logic is most suitable? We will argue that in IF logic a compositional semantic for interesting phenomena in natural logic is possible, whereas that is not possible for Dependence Logic. Since in both approaches claims are made about applications, but not really worked out, we had to do so for an example; and we took the classical distinction between a *de dicto* reading and a *de re* reading.

De dicto-de re

The sentence *John believes that a stranger crippled his cow* is ambiguous. One reading concerns the situation that John has no particular person in mind, but that he believes that whatever the situation is, it will be such that some stranger crippled his cow (the *de dicto* reading). The other reading is that there is particular person in reality of whom John believes that he crippled his cow (the *de re* reading).

Let $Bel(w, J)$ denote the belief-alternatives of John in possible world w, $C(y, J)$ that y is the cow of John, $Cr(w, x, y)$ hat x crippled y in world w, and let a be the actual world. Then the *de dicto* reading is represented in IF logic by (1), and the *de re* reading by (2). In (2) the subscript $_{/w}$ indicates that x is independent of w.

(1) $\exists y\,[C(y, J) \wedge \forall w_{\in Bel(a,J)}\,\exists x\,[Str(x) \wedge Cr(w, x, y)]]$.

(2) $\exists y[\,C(y) \wedge \forall w_{\in Bel(a,J)}\,\exists x_{/w}[Str(x) \wedge Cr(w, x, y)]]$.

In Dependence Logic we have to indicate what the quantifier depends on. Then *de dicto* reading is represented in (3), where $=\!(y, w, x)$ indicates that variable x depends only on y and w. The *de re* reading is given in (4), where the w is omitted from the dependency information.

(3) $\exists y\,[C(y, J) \wedge \forall w_{\in Bel(a,J)}\,\exists x\,[=\!(y, w, x) \wedge Str(x) \wedge Cr(w, x, y)]]$.

(4) $\exists y[\,C(y, J) \wedge \forall w_{\in Bel(a,J)}\,\exists x[=\!(y, x) \wedge Str(x) \wedge Cr(w, x, y)]]$.

Now consider the variant *Some man believes that a stranger crippled his cow.* In IF logic the *de dicto* and the *de re* reading are respectively (5) and (6):

(5) $\exists z\exists y\,[M(z) \wedge C(y, z) \wedge \forall w_{\in Bel(a,z)}\exists x\,[Str(x) \wedge Cr(w, x, y)]]$.

(6) $\exists z\exists y\,[M(z) \wedge C(y, z) \wedge \forall w_{\in Bel(a,z)}\exists x_{/w}[Str(x) \wedge Cr(w, x, y)]]$.

In Dependence logic the *de dicto* reading in (7), and the *de re* reading in (8).

(7) $\exists z\exists y\,[M(z) \wedge C(y, z) \wedge \forall w_{\in Bel(a,z)}\exists x\,[=\!(z, y, w, x) \wedge Str(x) \wedge Cr(w, x, y)]]$.

(8) $\exists z\exists y\,[M(z) \wedge C(y, z) \wedge \forall w_{\in Bel(a,z)}\exists x\,[=\!(z, y, x) \wedge Str(x) \wedge Cr(w, x, y)]]$.

This example exhibits the difference between the two logics. In IF it is possible to give a representation for the de re reading of *believes that a stranger cripples y's cow* (viz. (9)), but that is not possibe in Dependence Logic because there the representation depends on the context in which the formula is used. For the de dicto reading the same holds, in IF we have (10), but in dependence logic the representation depends on the context.

(9) $\exists x[Str(x) \wedge Cr(w, x, y)]$

(10) $\exists x_{/w}[Str(x) \wedge Cr(w, x, y)]$.

The same situation arises when one considers further embeddings, such as in *A woman thinks that some man believes that a stranger crippled his cow*: in IF logic the same representations for the embedded clause can be used for the *de dicto* and for the *de re* reading respectively. For Dependence Logic this is not the case: again new the representations are needed in order to account for the additional quantifier introduced by *a woman.*

This shows that using IF logic a compositional semantics for compound sentences can be given. For Dependence Logic this is not the case. Hence with that logic a compositional semantics for natural language seems not possible.

(Uniform) continuity

In both logics the standard versions of *continuous* and *uniformly continuous* can be represented. If we consider the variants *continuous on an interval* and *uniformly continuous on an interval*, a quantifier over intervals is introduced, and the other quantifiers depend on this. In IF logic the difference between the two properties remains the same: independence of value of the argument (i.e. the place in the the interval). So the representations for *continuous* and *uniformly continuous* can be used in the new situation as well. In dependence logic a new representation must be used, thus suggesting that a new property plays a role. Again, IF logic is possible to formalize the continuity notions in a compositional way, whereas Dependence Logic this is not, because it considers the phenomenon from the angle of dependency.

Discussion

An explanation might be that the phenomenona under discussion are a phenomenon of independence, and Independence Friendly Logic is suitable to describe this. Dependence Logic describes a complementary phenomenon, viz. dependence, and it essential of that phenomenon that the dependency relation may change under embeddings. Dependence Logic is suitable for situation where dependence is basic, maybe data bases are an example, but in many other applications independence seems to be the fundamental notion, and Independence Friendly Logic is the logic to be used.

References

[1] J. Hintikka and G. Sandu (1977), 'Game-theoretical semantics', *in* J. van Benthem and A. ter Meulen, *Handbook of Logic and Language* Elsevier, Amsterdam, pp. 361-410.

[2] J. Väänänen (2007), *Dependence logic; a new approach to Independence Friendly logic*, vol. 70 in London mathematical Society students texts, Cambridge university Press, Cambridge.

Coherence and computational complexity of quantifier-free dependence logic formulas

Jarmo Kontinen

June 21, 2010

Abstract

We study the computational complexity of the model checking for quantifier-free dependence logic ($\mathcal{D}$) formulas. We point out three thresholds in the computational complexity: logarithmic space, non-deterministic logarithmic space and non-deterministic polynomial time.

1 Introduction

Dependence logic $\mathcal{D}$ [5] incorporates explicit dependence relations between terms into first-order logic (FO). The dependence relations between terms are expressed in terms of dependence atoms

$$=\!(t_1, \ldots, t_n) \tag{1}$$

which are taken as atomic formulas. The intuitive meaning of (1) is that the values of the terms $t_1, \ldots, t_{n-1}$ determine the value of the term t_n. The expressive power of $\mathcal{D}$ equals to that of existential second order logic (Σ_1^1) [5].

We are interested in characterizing the computational complexity of model checking of $\mathcal{D}$-formulas. The problem of charting fragments of logics which fall under specific computational classes is widely studied question. Fagin's classical result [1] establishes a perfect match between Σ_1^1-formulas and languages in NP. It is known that $\mathcal{D}$-formulas have a definition in Σ_1^1 and vice versa [5]. When we combine this with Fagin's result that $\Sigma_1^1 \equiv NP$, we get that the properties of finite structures definable in $\mathcal{D}$ are exactly the ones recognized in NP.

We characterize the computational complexity of the model checking for quantifier-free formulas of dependence logic. An essential notion we will use in the characterization is the k-coherence of a formula. We say that a formula ϕ is k-coherent, $k \in \mathbb{N}$, if for all teams $\mathcal{X}$ it holds that ϕ is satisfied by $\mathcal{X}$ if and only if all k-element subsets of the team $\mathcal{X}$ satisfy ϕ. Coherence allows us to evaluate the satisfiability of the formula in finite fixed size sub-teams, which is very useful as the teams are potentially very large.

We will first give characterization for the k-coherence of a formula. We will show that all atomic formulas are coherent and that conjunction preserves coherence whereas disjunction does not. We will show that disjunction of two distinct dependence atoms is not coherent for any $k \in \mathbb{N}$.

We use coherence to characterize the model checking for quantifier-free $\mathcal{D}$-formulas. We will show that for all k-coherent formulas the model checking can be done in logarithmic space (L). When there can be at most one disjunction in the formula, the model checking can be done in non-deterministic logarithmic space (NL). Furthermore we will show that the model checking of two distinct dependence atoms is complete for NL. Last we will show that the model checking becomes already non-deterministic polynomial time (NP)-complete when one allows more than one disjunction in the formulas.

2 Preliminaries

Definition 2.1. The syntax of $\mathcal{D}$ extends FO, defined in terms of $\vee$, $\wedge$, $\neg$, $\forall$, $\exists$, by new atomic formulas of the form

$$=\!(t_1, \ldots, t_n) \tag{2}$$

where $t_1, \ldots, t_n$ are terms. We will denote the set of free variables of ϕ by $Fr(\phi)$.

The semantics of $\mathcal{D}$ is given in terms of sets of assignments, teams. We define two operations on teams before giving the semantics.

Definition 2.2. Suppose $\mathcal{X}$ is a team of domain V and range M, and $F : \mathcal{X} \to M$ is a function. Let $\mathcal{X}(F, x_n)$ denote the *supplement team*

$$\{s(F(s)/x_n) \mid s \in \mathcal{X}\},$$

where $s(F(s)/x_n)$ is the tuple obtained by replacing $(x_n, s(x_n))$ in s with $(x_n, F(s))$.

Definition 2.3. Suppose $\mathcal{X}$ is a team of domain V and range M. Let $\mathcal{X}(M, x_n)$ denote the *duplicated team*

$$\{s(a/x_n) \mid a \in M,\ s \in \mathcal{X}\},$$

where $s(a/x_n)$ is the tuple obtained by replacing $(x_n, s(x_n))$ in s with (x_n, a).

Definition 2.4. (*Semantics*) Suppose τ is a vocabulary, $\mathcal{X}$ is a team of domain V and range M, $\mathcal{M}$ a τ-structure and ϕ and θ formulas of $\mathcal{D}(\tau)$. The semantics of $\mathcal{D}$-formulas is defined in the following way:

1. $\mathcal{M} \models_{\mathcal{X}} =\!(t)$, iff for all $s, s' \in \mathcal{X}$ it holds $s(t) = s'(t)$.
2. $\mathcal{M} \models_{\mathcal{X}} =\!(t_1, \ldots, t_n)$, $n > 1$, iff for all $s, s' \in \mathcal{X}$ it holds that, if $s(t_i) = s'(t_i)$ for $i \leq n-1$, then $s(t_n) = s'(t_n)$.
3. $\mathcal{M} \models_{\mathcal{X}} \neg =\!(t_1, \ldots, t_n)$ iff $\mathcal{X} = \emptyset$.
4. $\mathcal{M} \models_{\mathcal{X}} \approx t_1 t_2$, iff for every $s \in \mathcal{X}$, $s(t_1) = s(t_2)$.
5. $\mathcal{M} \models_{\mathcal{X}} \neg \approx t_1 t_2$, iff for every $s \in \mathcal{X}$, $s(t_1) \neq s(t_2)$.
6. $\mathcal{M} \models_{\mathcal{X}} R(t_1, \ldots, t_n)$, iff for every $s \in \mathcal{X}$, $(s(t_1), \ldots, s(t_n)) \in R^{\mathcal{M}}$.
7. $\mathcal{M} \models_{\mathcal{X}} \neg R(t_1, \ldots, t_n)$, iff for every $s \in \mathcal{X}$, $(s(t_1), \ldots, s(t_n)) \notin R^{\mathcal{M}}$.
8. $\mathcal{M} \models_{\mathcal{X}} \phi \wedge \theta$, iff $\mathcal{M} \models_{\mathcal{X}} \phi$ and $\mathcal{M} \models_{\mathcal{X}} \theta$.
9. $\mathcal{M} \models_{\mathcal{X}} \phi \vee \theta$, iff there exists $\mathcal{Y}$ and $\mathcal{Z}$, such that $\mathcal{Y} \cup \mathcal{Z} = \mathcal{X}$, $\mathcal{M} \models_{\mathcal{Y}} \phi$ and $\mathcal{M} \models_{\mathcal{Z}} \theta$.
10. $\mathcal{M} \models_{\mathcal{X}} \exists x \phi(x)$, iff there is $F : V \to M$, such that $\mathcal{M} \models_{\mathcal{X}(F,x)} \phi(x)$.
11. $\mathcal{M} \models_{\mathcal{X}} \forall x \phi(x)$, iff $\mathcal{M} \models_{\mathcal{X}(M,x)} \phi(x)$.

Finally, a sentence ϕ is true in a structure $\mathcal{M}$ if $\mathcal{M} \models_{\{\emptyset\}} \phi$.

Theorem 2.5. *[5](*Downwards closure*) Suppose $\phi \in \mathcal{D}$ and $\mathcal{X}$ and $\mathcal{Y}$ are teams, such that $\mathcal{Y} \subseteq \mathcal{X}$. Then the following holds:*

$$\mathcal{M} \models_{\mathcal{X}} \phi \Rightarrow \mathcal{M} \models_{\mathcal{Y}} \phi.$$

3 Coherence

We will characterize quantifier-free formulas in terms of coherence. We will show that conjunction preserves coherence whereas disjunction does not.

Definition 3.1. Suppose $\phi(x_1 \ldots, x_n) \in \mathcal{D}$ is a quantifier-free formula. Then ϕ is *k-coherent* if and only if for all structures $\mathcal{M}$ and teams $\mathcal{X}$ of range $dom(\mathcal{M})$, such that $Fr(\phi) \subseteq Dom(\mathcal{X})$ the following are equivalent:

1. $\mathcal{M} \models_{\mathcal{X}} \phi$.

2. For all k-element sub-teams $\mathcal{Y} \subseteq \mathcal{X}$ it holds that $\mathcal{M} \models_{\mathcal{Y}} \phi$.

We will start by going through the atomic formulas and connectives of $\mathcal{D}$ and study the effect they have on coherence. We will observe that the satisfiability of first-order formulas can be done in restricting to singleton sub-teams whereas dependence atoms can be checked in restricting to two-element sub-teams. Furthermore, we will show that conjunction preserve coherence, whereas disjunction does not.

The following two propositions follow directly from the definition of semantics for the atomic formulas.

Proposition 3.2. *First-order atomic formulas and negated atomic first-order formulas are* 1*-coherent.*

Proposition 3.3. *All dependence atoms are* 2*-coherent.*

Proposition 3.4. *Suppose ϕ and ψ are quantifier-free formulas such that ϕ is k-coherent and ψ is l-coherent for $l, k \in \mathbb{N}$, $l \leq k$. Then $\phi \wedge \psi$ is k-coherent.*

Proof. Suppose $\mathcal{X} \models \phi \wedge \psi$. Then by Theorem 2.5 all subsets of $\mathcal{X}$ satisfy $\phi \wedge \psi$, especially the k-element subsets.

The other direction: Suppose all k-element subsets $Y \subseteq \mathcal{X}$ satisfy $\phi \wedge \psi$. Then it holds that $\mathcal{Y}$ satisfy ϕ and ψ for all $\mathcal{Y}$. Then, by the coherence of ϕ it follows that $\mathcal{X}$ satisfies ϕ. By downward closure and the fact that all l-element subsets of $\mathcal{X}$ are contained in some k-element subset, we conclude that all l-element subsets of $\mathcal{X}$ satisfy ψ. Then by coherence of ψ, also $\mathcal{X}$ satisfies ψ. Thus $\mathcal{X}$ satisfies $\phi \wedge \psi$. □

Disjunction does not preserve coherence in general. In some cases, however, we can show that disjunction does preserve it.

Proposition 3.5. *Suppose ϕ and ψ are quantifier-free $\mathcal{D}$-formulas, such that ϕ is* 1*-coherent and ψ is k-coherent for some $k \in \mathbb{N}$. Then $\phi \vee \psi$ is k-coherent.*

Proof. Suppose it holds that $\mathcal{X} \models \phi \vee \psi$, then by Theorem 2.5 $\mathcal{X}_k \models \phi \vee \psi$ holds for all k-element subsets $\mathcal{X}_k \subseteq \mathcal{X}$.

The other direction: Suppose that $\mathcal{X}_k \models \phi \vee \psi$ holds for all k-element subsets $\mathcal{X}_k \subseteq \mathcal{X}$. Now the division of $\mathcal{X}$ into $\mathcal{Y}$ and $\mathcal{Z}$, such that $\mathcal{Y} \models \phi$ and $\mathcal{Z} \models \psi$ is obtained in the following way:

- $s \in \mathcal{Y}$ iff $\{s\} \models \phi$ and $s \in \mathcal{Z}$ otherwise.

Clearly, it holds that $\mathcal{Y} \models \phi$. Let us show that $\mathcal{Z} \models \psi$. By k-coherence of ψ we have to check that for all k-element subsets $\mathcal{Z}_k \subseteq \mathcal{Z}$, it holds that

$\mathcal{Z}_k \models \psi$. Suppose $\mathcal{Z}_k \subseteq \mathcal{Z}$, such that $\mathcal{Z}_k$ fails ψ. Since all the singletons $s \in \mathcal{Z}_k$ fail ϕ it holds that $\mathcal{Z}_k \not\models \phi \vee \psi$, which is a contradiction with the assumption. Thus all the k-element subsets of $\mathcal{Z}$ satisfy ψ. Which means by k-coherence of ψ that $\mathcal{Z} \models \psi$. Thus $\mathcal{Y} \cup \mathcal{Z} \models \phi \vee \psi$ holds. $\square$

As established in the previous proposition, combining a k-coherent formula with a 1-coherent formula does not increase the coherence level. Thus we have to have both disjuncts at least 2-coherent. Namely we have to consider disjunctions over dependence atoms.

We denote the disjunction of size k over a single dependence atom $=(x_1, \ldots, x_n)$, by $\bigvee_k =(x_1, \ldots, x_n)$. We will next show that disjunctions over the same dependence atom increases the coherence-level, i.e. the coherence-level is increased by 1 for each disjunct.

Proposition 3.6. *Suppose $k \in \mathbb{N}$ and ϕ is a dependence atom. Then $\bigvee_k \phi$ is $(k+1)$-coherent.*

Proof. Suppose ϕ is the dependence atom $=(x_1, \ldots, x_n)$ and $\mathcal{X}$ is a team of type $\bigvee_k =(x_1, \ldots, x_n)$. Then, by downwards closure property all $(k+1)$-element subsets of $\mathcal{X}$ satisfy $\bigvee_k =(x_1, \ldots, x_n)$.

Other direction: Suppose all $k+1$-element subsets of $\mathcal{X}$ satisfy $\bigvee_k = (x_1, \ldots, x_n)$. Let $S(a_1, \ldots, a_{n-1})$ be defined in the following way for each $(a_1, \ldots, a_{n-1}) \in M^{n-1}$:

$$S(a_1, \ldots, a_{n-1}) = \{s \in \mathcal{X} \mid (s(x_1), \ldots, s(x_{n-1})) = (a_1, \ldots, a_{n-1})\}.$$

Let $|S(a_1, \ldots, a_{n-1})|^*$ be the number of different values of x_n under the assignments in $S(a_1, \ldots, a_{n-1})$. We will show that the following are equivalent:

1. $\mathcal{X} \models \bigvee_k =(x_1, \ldots, x_n)$.
2. $|S(\bar{a})|^* \leq k+1$ for each $\bar{a} \in M^{n-1}$.

Suppose (2) holds. Then each $S(a_1, \ldots, a_{n-1})$ can be divided into $(k+1)$-element sets $S(a_1, \ldots, a_{n-1})^i$, $1 \leq i \leq k+1$, such that x_n is constant in each $S(a_1, \ldots, a_{n-1})^i$. Now the following partition of $\mathcal{X}$ into sets $\mathcal{X}_i$, $1 \leq i \leq k+1$, is what we are looking for:

$$\mathcal{X}_i = \bigcup_{\bar{a} \in M^{n-1}} S(a_1, \ldots, a_{n-1})^i.$$

Next we will show that $\mathcal{X}_i \models =(x_1, \ldots, x_n)$ for each $\mathcal{X}_i$, $1 \leq i \leq k+1$.

Suppose $s, s' \in \mathcal{X}_i$, such that s and s' are from the same $S(a_1, \ldots, a_{n-1})^i$ for some $(a_1, \ldots, a_{n-1}) \in M^{n-1}$. Now s and s' will agree on x_n since x_n

is constant in each $S(a_1,\ldots,a_{n-1})^i$. Thus $\{s,s'\} \models =(x_1,\ldots,x_n)$ holds. Suppose s and s' are from different sets, say s from $S(a_1,\ldots,a_{n-1})$ and s' from $S(a'_1,\ldots,a'_{n-1})$. Then s and s' will disagree on the sequence $(x_1,\ldots,x_{n-1})$. Thus $\{s,s'\} \models =(x_1,\ldots,x_n)$ holds. Now $\mathcal{X}_i \models =(x_1,\ldots,x_n)$ holds for each $\mathcal{X}_i$, $1 \le i \le k+1$ by 2-coherence of dependence atoms.

Other direction: Suppose (2) does not hold. Then there is $(a_1,\ldots,a_{n-1}) \in M^n$, such that $|S(a_1,\ldots,a_{n-1})| > k+1$. By *pigeon hole principle*[1] it is not possible to divide $S(a_1,\ldots,a_n)$ into $k+1$ subsets $S(a_1,\ldots,a_{n-1})^i$, $1 \le i \le k+1$, so that in each set $S(a_1,\ldots,a_{n-1})^i$ the value of x_n would be constant. Since all the tuples in $S(a_1,\ldots,a_n)$ agree on sequence $x_1,\ldots,x_{n-1}$ it follows that $=(x_1,\ldots,x_n)$ will be failed in some subset independent of the division of $S(a_1,\ldots,a_n)$. Thus $\mathcal{X}$ does not satisfy $\bigvee_k =(x_1,\ldots,x_n)$.

The original assumption was that each $(k+1)$-element sub-team of $\mathcal{X}$ satisfies ϕ. Thus it holds that there are no such $(k+1)$-element subsets in $\mathcal{X}$ where the assignments agree on the first $n-1$ terms and all disagree on the last term. Thus for each tuple $(a_1,\ldots,a_{n-1}) \in M^{n-1}$ it holds that $|S(a_1,\ldots,a_{n-1})|^* \le k+1$. Thus the claim follows with the above established equivalence. □

3.1 Incoherence

We will show that disjunction does not preserve coherence. Given a team $\mathcal{X}$ and a disjunction of dependence atoms $=(X_i,y_i)$, $i \in I$, denoted by $\bigvee_{i\in I} =(X_i,y_i)$, we interpret the team as a multigraph in such a way that the $|I|$-colorability of the multigraph corresponds to $\mathcal{X}$ satisfying $\bigvee_{i\in I} =(X_i,y_i)$. Each assignment translates into a vertex in the graph. Each dependence atom $=(X_i,y_i)$, induces edges between the vertices in such a way that if two assignments fail the dependence atom, then the corresponding vertices share the corresponding edge E_i.

Definition 3.7. Suppose $\mathcal{X} = \{s_1,\ldots,s_n\}$ is a team of domain V and range M and $\phi \in \mathcal{D}$ is of the form $\bigvee_{i\in I} =(X_i,y_i)$. For each $\mathcal{X}$ and ϕ we construct a multi graph $\mathcal{G}^{\phi}_{X} = (V,\{E_i|\ i \in I\})$ in the following way:

1. $V = \{v_j \mid s_j \in \mathcal{X}\}$.

2. For each $i \in I$, if $\{s_j,s_l\} \not\models =(X_i,y_i)$, then $(v_j,v_l) \in E_i$.

The k-colorability of a multigraph is defined as an existence of a coloring function $\sigma : V \to |I|$, such that if two nodes share an edge E_i then they

[1] Formally it states that there does not exist an injective function on finite sets whose codomain is smaller than its domain.

cannot be colored both with the same color i. The existence of a coloring function matches exactly with the semantical condition of the disjunction in Team-semantics under the interpretation 3.7.

Proposition 3.8. *Suppose $\mathcal{G}_X^\phi$ is a multigraph defined as in 3.7 for a team $\mathcal{X}$ and formula $\phi =: \bigvee_{i\in I} =\!(X_i, y_i)$. Then the following two conditions are equivalent:*

1. *There exists a function $\sigma : V \to I$, such that if $\sigma(v_i) = \sigma(v_j) = m$, then $(v_i, v_j) \notin E_m$.*
2. *$\mathcal{X} \models \bigvee_{i\in I} =\!(X_i, y_i)$.*

Proof. Suppose $\sigma : V \to I$ is a function, such that if $\sigma(v_i) = \sigma(v_j) = m$, $m \in I$, then $(v_i, v_j) \notin E_m$. Let X_i, $1 \leq i \leq m$, be defined the following way:

$$X_i = \{s_n \mid s_n \in X \wedge \sigma(v_n) = i\}.$$

Since σ is defined on the domain $\mathcal{G}_X^\phi$, it holds that $\mathcal{X} = \bigcup_{i\in m} \mathcal{X}_i$. We will show next that $\mathcal{X}_i \models =\!(X_i, y_i)$ holds for each $\mathcal{X}_i$ $1 \leq i \leq m$:

Suppose $s_l, s_k \in \mathcal{X}_i$. Then, the corresponding vertices v_l and v_k are assigned the value i under σ. Then, by assumption on σ, it follows that $(v_l, v_k) \notin E_i$. Thus by 3.7, it follows that $\{s_l, s_k\} \models =\!(X_i, y_i)$. Further, it follows from the 2-coherence of the dependence atoms that $X_i \models =\!(X_i, y_i)$. Thus $\mathcal{X} \models \bigvee_{i\in I} =\!(X_i, y_i)$ holds.

The other direction: Suppose $\mathcal{X} \models \bigvee_{i\in I} =\!(X_i, y_i)$ holds. Then, there is a partition of $\mathcal{X}$ into sets $\mathcal{X}_i$, such that $\mathcal{X}_i \models =\!(X_i, y_i)$ for each $i \in I$, and $\mathcal{X} = \bigcup_{i\in I} \mathcal{X}_i$. Then, let σ be defined the following way:

- $\sigma(v_n) = i$, if $s_n \in \mathcal{X}_i$.

Clearly, σ is well defined and it holds that, if $\sigma(v_i) = \sigma(v_j) = m$, then $(v_i, v_j) \notin E_m$. □

Next lemma will show that disjunction does not preserve coherence. An important detail to notice is whether the disjuncts share some variables or not. As we will show later, disjunctions of the same dependence formula stay coherent.

Theorem 3.9. $=\!(x, y) \vee =\!(z, v)$ *is not k-coherent for any $k \in \mathbb{N}$.*

Proof. We will actually show that a stronger claim holds, namely that $=\!(x, y) \vee =\!(z, v)$ is not $f(n)$-coherent for any function $f : \mathbb{N} \to \mathbb{N}$, such that $f(n) < n$, for all n. Here the meaning of $f(n)$-coherence is, that a formula

ϕ is $f(n)$-coherent, if for all teams $\mathcal{X}$, such that $|\mathcal{X}| = n$, it holds that $\mathcal{X} \models \phi \Leftrightarrow \mathcal{Y} \models \phi$ for every $\mathcal{Y} \subseteq \mathcal{X}$, such that $|\mathcal{Y}| = f(n)$.

We will construct a team $\mathcal{X}$ for every $k \in \mathbb{N}$ so that every proper subset of the team satisfies $=(x, y) \vee =(z, v)$, but the whole team fails to satisfy $=(x, y) \vee =(z, v)$. We represent the team as a multigraph as in 3.7. Each of the vertices correspond to an assignment of the team. Suppose $s_v, s_w \in \mathcal{X}$. There are two type of edges we assign between vertices in the following way.

- If $\{s_v, s_w\} \not\models =(x, y)$, then we assign a smooth edge between the vertices v and w.
- If $\{s_v, s_w\} \not\models =(z, v)$, then we assign a wavy edge between the corresponding vertices v and w.

We will use "black" color to denote the vertices that do not allow wavy edge and "white" color to denote the vertices that do not allow smooth edges. A coloring of the multigraph will be a partition of the universe into two sets, black and white vertices, such that the black vertices do not share any smooth edges and the white vertices do not share any wavy edges. The graph in figure 1 is such that every proper subgraph is 2-colorable, but the whole multigraph is not.

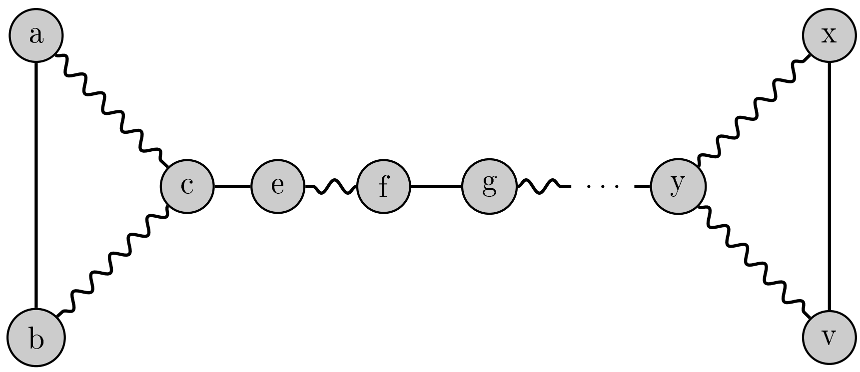

Figure 1: Graph $\mathcal{G}_{\mathcal{X}}$

$\mathcal{G}_{\mathcal{X}}$ **is not** 2**-colorable**: Suppose both the nodes c and y are colored black. Then vertices a, b, x, v should be colored white as they all share a wavy edge either with c or y. But since there is a smooth edge between a and b as well as between v and x and the fact that white color did not allow smooth edges, this cannot be a proper coloring. Thus the only way to properly color the triangles is to color both c and y white. The colors of a, b, x, v can be chosen black or white as long as v, x and a, b are not both white. The two triangles

$\{a, b, c\}$ and $\{x, y, z\}$ are connected with a path (of even length). The path is such that the edge alternates between smooth and wavy, which forces the proper coloring also to alternate between black and white for the nodes on the path. Since the length of the path is even, there cannot be a coloring for the whole graph as the color of c totally determines the coloring of the whole path, in the same way as the color of y. They both force different colors on the path, thus making the proper coloring impossible. Thus the whole graph is not 2-colorable.

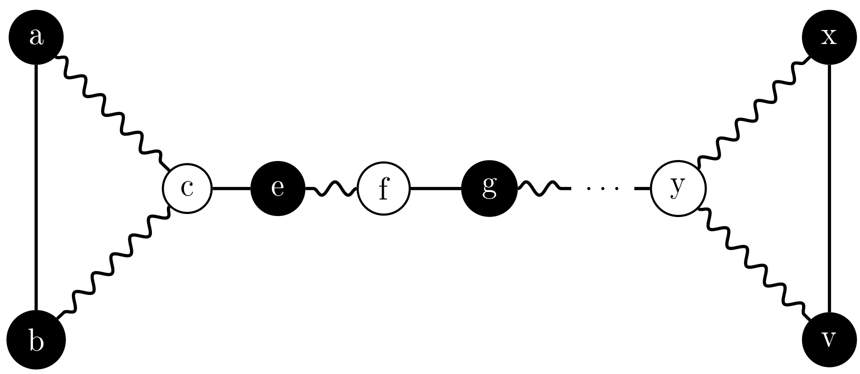

Figure 2: Coloring of the graph $\mathcal{G}_{\mathcal{X}}$

Every proper subgraph of $\mathcal{G}_{\mathcal{X}}$ is 2-colorable: We will show that if we remove a vertex from either of the triangles, then the coloring of the vertex c (or y), which is connected to the path, can be chosen either black or white. Suppose a is removed. Then we can choose so that c is colored black and b is with white. The vertex y has to be still colored with white. Now, since c and y are colored with different colors and the path connecting them is even, it holds that the whole graph can be colored. The cases where we remove any other vertex from the two triangles are analogous to this one.

On the other hand, suppose one of the vertices from the path connecting the two triangles is removed. Then we have two components of the graph that are not connected by edges. The coloring of the whole graph reduces to the coloring of the two subgraphs for which there is a trivial coloring induced by the coloring of two nodes c and y. The team, which corresponds to the graph $\mathcal{G}_{\mathcal{X}}$ is the following table 1.

As one can observe the values of the whole path are not explicitly given in the picture. If two vertices share a smooth edge, the corresponding assignments in the team in table 1 are assign the same value for x and different one for y. Similarly, if two vertices share an wavy edge the corresponding assignments assign the same value for z and different one for v. When we choose

assignment	x	y	z	v
s_a	0	0	0	0
s_b	0	2	0	0
s_c	1	2	0	1
s_e	1	1	1	2
s_f	2	3	1	3
s_g	2	4	2	4
.				
.				
.				
s_v	4	7	3	5
s_x	4	6	3	5
s_y	3	5	3	4

Table 1:

the values for the assignments that correspond to a vertex in the the path, we use always new values for the variables if possible. This way we ensure that there will be no unintended edges between the vertices in the triangle and the vertices in the path, just the ones that appear in the picture.

For example, for the assignment s_e the value of x is assigned the same as $s_c(x)$ and $s_e(y)$ is assigned different to $s_c(y)$, but $s_e(z)$ and $s_e(v)$ can be chosen new values. With the next vertex on the path, which is f, we can already assign the new values for x and y. We have to take care that the values of $s_f(z)$ and $s_f(v)$ are assigned so that the dependence $=(z, v)$ is failed. At this point of the path the values, which the assignment s_a, s_b and s_c assign to variables x, y, z, v are no more assigned when we go left in the path. Thus, the ranges of the variables under the assignments corresponding to the vertices of the triangles are disjoint with ranges of variables under the assignment that correspond to the vertices on the path (excluding the endpoints of the path).

Let us show that the team in table 1 indeed translates into the graph in figure 1. Recall that smooth edges are generated when two assignments fail the dependence $=(x, y)$ and wavy edges when the $=(z, v)$ is failed;

Smooth edges from vertex a: $s_a(x) = 0$. It holds that $s_b(x) = 0$ and since $s_b(y) \neq s_a(y)$ there is a smooth edge (a, b). All the other assignments assign other value than 0 for x, thus there cannot be smooth edges from a to other vertices.

Wavy edges from vertex a: $s_a(z) = s_c(z) = 0$ and $s_a(v) = 0 \neq 2 = s_z(v)$, thus there is a wavy edge (a,c). Indeed $s_b(z) = 0$ but $s_b(v) \neq s_a(v)$, thus there is no wavy edge (a,b). All the other assignments assign a value different to 0 for z, thus there are no wavy edges between a and other vertices.

Smooth edges from vertex b: The only smooth edge from b is the one that is shared with a. All other assignments assign a value different to 0 for x, thus there are no other smooth edges from b.

Wavy edges from vertex b: $s_b(z) = s_c(z) = 0$ and $s_b(v) = 0 \neq 2 = s_z(v)$, thus there is a wavy edge (b,c). Again, s_b and s_a agree on z but disagree on v, thus there is no wavy edge between them. All the other assignments assign a value different to 0 for z. Thus there are no other wavy edges from b.

Smooth edges from vertex c: Only assignment that assigns x as 1 is s_e. Also they disagree on y. Thus there is smooth edge (c,e). As we earlier noted, after the next vertex on the path, the values that are assigned by the assignments that correspond to nodes that appear in the triangle do not appear in the ranges of the assignments that correspond to the vertices that come later in the path. Thus all the other assignments assign a value different to 1 for x. Thus there are no other smooth edges from the node c.

Wavy edges from vertex c: The edges (c,a) and (c,b) have been already established. Again, the value that s_c assigns for z does not appear as a value for z under assignment corresponding the nodes which appear later in the path. Thus there are no other wavy edges from z.

The other triangle $\{x,y,z\}$ is isomorphic to that of $\{a,b,c,\}$. It can be checked analogous that exactly the edges that appear in the graph will be generated under the translation 3.7.

Now we have given a construction of collection of graphs like in 1,which are not 2-colorable, but for which hold that every proper sub-graph is. Now by 3.8 the following are equivalent:

1. $\mathcal{G}_{\mathcal{X}}$ is 2-colorable.

2. $\mathcal{X} \models =\!(x,y) \vee =\!(z,v)$.

Thus the whole team $\mathcal{X}$ as in table 1 does not satisfy $=\!(x,y) \vee =\!(z,v)$, but every proper sub-team of $\mathcal{X}$ satisfies $=\!(x,y) \vee =\!(z,v)$. By increasing the length of the path which connects the two triangles, we get the same counter example for different cardinalities. $\square$

4 Computational complexity of quantifier-free $\mathcal{D}$-formulas

Model checking is one of the central problems considered in the finite model theory. Given a structure $\mathcal{M}$ and a formula ϕ it is to decide wether $\mathcal{M}$ is a model of ϕ. When we fix the formula ϕ and let the structure vary, we talk about the model checking problem for a formula ϕ. Each $\phi \in \mathcal{D}(\tau)$ defines a collection of pairs $(\mathcal{M}, \mathcal{X})$, where $\mathcal{M}$ is a τ-structure and $\mathcal{X}$ is a team of range M, such that $Fr(\phi) = dom(\mathcal{X})$. Given a team $\mathcal{X}$ we define a relation $Rel(\mathcal{X})$ in the following way:

$$Rel(\mathcal{X}) = \{(s(x_0), \ldots, s(x_{n-1})) \mid s \in \mathcal{X}\}.$$

Definition 4.1. Suppose $\phi \in \mathcal{D}(\tau)$. *Model checking problem for a formula ϕ, MC(ϕ)*, is to decide wether it holds that

$$\mathcal{M} \models_{\mathcal{X}} \phi,$$

where $\mathcal{M}$ is a τ-structure and $\mathcal{X}$ is a team, such that $Fr(\phi) = dom(\mathcal{X})$

We look for a connection between the computational complexity of the model checking problem for a quantifier-free $\mathcal{D}$-formula and the syntactic form of ϕ. We will use the following versions of Boolean satisfiability problem in to show the NL- and NP-completeness of $MC(\phi)$ for certain $\phi \in \mathcal{D}$.

Definition 4.2. *Boolean satisfiability problem* (SAT) is a decision problem to determine wether a given propositional first order formula is satisfiable. The variables are boolean and may occur positively or negatively in the formula. The formulas are assumed to be in the conjunctive normal form. The problem is to determine, whether there is an assignment, which evaluates the given formula true. There are several variations of SAT from which we consider the following two:

- 2-SAT: At most 2 disjuncts in each clause.
- 3-SAT: At most 3 disjuncts in each clause.

4.1 Logarithmic space

We will start by showing that the model checking for k-coherent formulas can be done in LOGSPACE. We will establish this by showing that for every k-coherent τ-formula there is an equivalent FO-sentence over vocabulary $\tau \cup \{R\}$, where R is an $|dom(\mathcal{X})|$-ary relation symbol interpreting the team.

Theorem 4.3. *Suppose ϕ is a quantifier-free k-coherent $\mathcal{D}(\tau)$-formula. Then there is a sentence $\phi^* \in FO(\tau \cup \{R\}$, such that the following are equivalent:*

1. $\mathcal{M} \models_{\mathcal{X}} \phi$
2. $(\mathcal{M}, Rel(\mathcal{X}) \models \phi^*(R)$

for all τ-structures $\mathcal{M}$ and for all teams $\mathcal{X}$, such that $Fr(\phi) \subseteq dom(\mathcal{X})$.

Proof. Suppose ϕ is a k-coherent $\mathcal{D}(\tau)$-formula. Then for all teams $\mathcal{X}$, such that $FR(\phi) \subseteq dom(\mathcal{X})$ the following are equivalent:

1. $\mathcal{M} \models_{\mathcal{X}} \phi$
2. For all k assignment sub-teams $\mathcal{Y} \subseteq \mathcal{X}$ holds $\mathcal{M} \models_{\mathcal{Y}} \phi$.

Each k-assignment $\mathcal{Y} \subseteq \mathcal{X}$ defines a finite relation $Rel(\mathcal{Y})$, which can be characterize in FO up to isomorphism. Since the vocabulary is finite, there are only finite number of different τ-isomorphism types of $Rel(\mathcal{Y})$. Given k distinct assignments $\mathcal{Y} = \{s_1, \ldots, s_k\}$ let $\psi(\bar{x}_1, \ldots \bar{x}_k) \in$ FO be isomorphism type of the relation $Rel(\mathcal{Y})$. Let I^ϕ be the set of all different isomorphism types of k-assignment teams $\mathcal{Y}$, such that $\mathcal{M} \models_{\mathcal{Y}} \phi$. Now $\phi^*(R)$ can be written in the following way:

$$\phi^* =: \forall \bar{x}_1 \ldots \forall \bar{x}_k ((\bigwedge_{i \in k} \bar{x}_i \in R \bigwedge_{i \neq j} \bar{x}_i \neq \bar{x}_j) \to \bigvee_{\psi \in I^\phi} \psi(\bar{x}_1, \ldots, \bar{x}_k))$$

Suppose $\mathcal{M} \models_{\mathcal{X}} \phi$. Then by k-coherence of ϕ it holds that each k-assignment sub-team $\mathcal{Y} \subseteq \mathcal{X}$ satisfies ϕ. Thus for each k-assignment sub-team $\mathcal{Y} \subseteq \mathcal{X}$ holds that Rel($\mathcal{Y}$) is of the proper isomorphism type. Thus, ϕ^* holds.

Suppose $(\mathcal{M}, Rel(\mathcal{X})) \models \phi^*(R)$. Then, each k-element sub-team $\mathcal{Y} \subseteq \mathcal{X}$ is of proper isomorphism type. Thus it holds that $\mathcal{M} \models_{\mathcal{Y}} \phi$ for each k-element sub-team. $\mathcal{M} \models_{\mathcal{X}} \phi$ follows by k-coherence of ϕ.

□

It is known that the data complexity of the model checking of first order formulas can be done in LOGSPACE (see [2] for details). This gives us the following corollary:

Corollary 4.4. *Suppose $\phi \in \mathcal{D}$ is a k-coherent formula. Then $MC(\phi) \in$ LOGSPACE.*

4.2 Non-deterministic logarithmic space

We already established that all quantifier-free formulas without disjunction are coherent. Furthermore we showed that with the use of one disjunction one obtains already formulas which are incoherent. In this section we will show that the model checking for all quantifier-free formulas with at most one disjunction can be done in non-deterministic logarithmic space. We will also show that the model checking of the formula $=(x, y) \vee =(z, u)$ is complete for NL.

Notice that in the following theorem we do not restrict the number of disjunctions in the formula, but rather the coherence level of the disjuncts. All formulas without disjunctions are at most 2-coherent.

Theorem 4.5. *Suppose ϕ and ψ are 2-coherent quantifier-free $\mathcal{D}$-formulas. Then*

$$MC(\phi \vee \psi) \leq_{LOGSPACE} 2 - SAT.$$

Proof. Suppose we are given a team $\mathcal{X} = \{s_1, \ldots, x_k\}$. We will go through all the two-element subsets $\{s_i, s_j\} \subseteq \mathcal{X}$, and construct an instance of 2-SAT in the following way:

- If $\{s_i, s_j\} \not\models \phi$, then $(x_i \vee x_j) \in C$.
- If $\{s_i, s_j\} \not\models \psi$, then $(\neg x_i \vee \neg x_j) \in C$.

We let $\Theta_X = \bigwedge_{\phi \in C} \phi$. Clearly, Θ_X is a proper instance of 2-SAT. We will next show that there is an assignment S, which satisfies Θ_X if and only if $\mathcal{X} \models \phi \vee \psi$ holds: Suppose there is an assignment $S : Var(\Theta_X) \to \{0, 1\}$, which evaluates Θ_X true. Let us define the partition of $\mathcal{X}$ in the following way:

- $\mathcal{Z} = \{s_i \in \mathcal{X} \mid S(x_i) = 1\}$.
- $\mathcal{Y} = \mathcal{X} \setminus \mathcal{Z}$.

Clearly it holds that $\mathcal{X} = \mathcal{Z} \cup \mathcal{Y}$. Let us show that $\mathcal{Z} \models \psi$ and $\mathcal{Y} \models \phi$ hold:

Suppose $s_i, s_j \in \mathcal{Z}$. Since S satisfies Θ_X, $(\neg x_i \vee \neg x_j)$ cannot be a clause in $C_{\mathcal{X}}$. By the construction above, it follows that $\{s_i, s_j\} \models \psi$ holds. Now, by 2-coherence of ψ it follows that $\mathcal{Z} \models \psi$.

Suppose $s_i, s_j \in \mathcal{Y}$. Since S was assumed to satisfy $C_{\mathcal{X}}$, $(x_i \vee x_j)$ cannot be a clause in $C_{\mathcal{X}}$. It follows by the construction above that $\{s_i, s_j\} \models \phi$ holds. Again, from 2-coherence of ϕ it follows that $\mathcal{Y} \models \phi$ holds.

The other direction: Suppose $\mathcal{X} \models \phi \vee \psi$ holds. Then, by Definition 2.4 it holds that there is a division of $\mathcal{X}$ into two sets $\mathcal{Z}$ and $\mathcal{Y}$, such that $\mathcal{X} = \mathcal{Z} \cup \mathcal{Y}$, $\mathcal{Z} \cap \mathcal{Y} = \emptyset$, $\mathcal{Y} \models \phi$ and $\mathcal{Z} \models \psi$. Let S be defined the following way:

- $S(x_i) = 1$, if $s_i \in \mathcal{Z}$.
- $S(x_i) = 0$, if $s_i \in \mathcal{Y}$.

Clearly it holds that $S : Var(\Theta_X) \to \{0, 1\}$ is a function. Let us show that S satisfies Θ_X: Suppose $\theta \in C$ of form $(x_i \vee x_j)$. Then $\{s_i, s_j\}$ fails ϕ by the construction of $\Theta_{\mathcal{X}}$. Then s_i and s_j cannot be both in $\mathcal{Y}$, since $\mathcal{Y}$ was supposed to satisfy ϕ. Thus, either s_i or s_j must be in $\mathcal{Z}$. Then, it holds that $S(x_i) = 1$ or $S(x_j) = 1$, which implies that $S(x_i \vee x_j) = 1$.

Suppose θ is $(\neg x_i \vee \neg x_j)$. Then, by the construction of $\Theta_{\mathcal{X}}$, it holds that $\{s_i, s_j\}$ fails ψ. Then, s_i and s_j cannot be both in $\mathcal{Z}$, since $\mathcal{Z}$ was supposed to satisfy ψ. Thus either s_i or s_j must be in $\mathcal{Y}$. Then, it holds that $S(x_i) = 0$ or $S(x_j) = 0$, which implies that $S(\neg x_i \vee \neg x_j) = 1$.

Last, the complexity of this reduction is in LOGSPACE: We need to go through the 2-element subsets of the team $\mathcal{X}$ and check if they fail ϕ or ψ. All the 2 assignment sub-teams of $\mathcal{X}$ can be generated in LOGSPACE when $\mathcal{X}$ is given. Since ϕ and ψ were coherent, the model checking for both of these formulas can be done in LOGSPACE. □

Now we will obtain the following corollary.

Corollary 4.6. *Suppose ϕ and ψ are 2-coherent $\mathcal{D}$-formulas. Then*

$$MC(\phi \vee \psi) \in NL.$$

Next we will show that the model checking of the formula $=(x, y) \vee =(z, u)$ is complete for NL. We will reduce 2-SAT to the model checking problem of the formula $=(x, y) \vee =(z, v)$.

Theorem 4.7. $2-SAT \leq_{LOGSPACE} MC(=(x, y) \vee =(z, v))$.

Proof. Suppose $\theta(p_0, \ldots, p_{m-1})$ is an instance of 2-SAT of the form $\bigwedge_{i \in I} E_i$, where each conjunct $E_i = (A_{i_1} \vee A_{i_2})$, $i \in I$, where A_{i_j}, $j \leq 1$, are positive or negative boolean variables.

We will construct a team $\mathcal{X}$, such that the following are equivalent:

1. $\mathcal{X} \models =(x, y) \vee =(z, v)$.
2. $\theta(p_0, \ldots, p_{m-1})$ is satisfiable.

For each conjunct E_i, $i \in I$, we create a team $\mathcal{X}_{E_i}$ where we code the information required to satisfy E_i. Now, E_i will be satisfied if one of the disjuncts will be true. Thus it has two conditions for being satisfied. We will code these conditions into the team we construct in the following way:

We will have a variable z denote the clause E_i, x denote the variables of the clause, y the truth value of the corresponding variable and v that makes sure we choose at least one of the assignments form each $\mathcal{X}_{E_i}$ into the sub-set of $\mathcal{X}$ which eventually codes the assignment which evaluates θ true. Each disjunct A_{ij} gives a rise to one assignment. Now $\mathcal{X}$ is the union $\bigcup_{i \in I} \mathcal{X}_{Ei}$.

For example, the team $\mathcal{X}_{E_i}$ for a clause $(p_k \vee p_j)$ is the one in Table 2. The team for the whole instance of 2−SAT:

$$(A_{01} \vee A_{02}) \wedge (A_{11} \vee A_{12}) \wedge \ldots \wedge (A_{I_1} \vee A_{I_2})$$

is the one in Table 3, where $t(A_i) = 1$ if A_i is a positive variable and $t(A_i) = 0$, if A_i is a negated variable.

z	x	y	v
i	p_k	1	1
i	p_j	1	2

Table 2: Team for $(p_k \vee p_j)$.

z	x	y	v
0	A_{01}	$t(A_{01})$	1
0	A_{02}	$t(A_{02})$	2
1	A_{11}	$t(A_{11})$	1
1	A_{12}	$t(A_{02})$	2
2	A_{21}	$t(A_{21})$	1
2	A_{22}	$t(A_{22})$	2
.	.	.	.
.	.	.	.
.	.	.	.
n	A_{I1}	$t(A_{I1})$	1
n	A_{I2}	$t(A_{I2})$	2

Table 3: Team $\bigcup_{i \in I} \mathcal{X}_{Ei}$.

Suppose $\theta(p_0, \ldots, p_{m-1})$ is satisfiable. Then there exists an assignment $F : \{p_0, \ldots, p_{m-1}\} \to \{0, 1\}$, such that F evaluates $\theta(p_0, \ldots, p_{m-1})$ true. We define the partition of the team $\mathcal{X}$ into two sets in the following way for each $s \in \mathcal{X}$: $s \in \mathcal{X}_1$ if the following condition holds:

$$(s(x) = p_i) \to F(p_i) = s(y). \tag{3}$$

Otherwise $s \in \mathcal{X}_2$.

Condition (3) guarantees that the tuples that agree with the assignment F are chosen to $\mathcal{X}_1$. Since F evaluates $\bigwedge_{i \in I} E_i$ to true, it evaluates every conjunct E_i true. As the satisfying conditions of each E_i are coded into $\mathcal{X}_{E_i}$, the condition (3) is satisfied by at least one of the assignments in each $\mathcal{X}_{E_i}$. Thus there will be at most one tuple from each $\mathcal{X}_{E_i}$ in $\mathcal{X}_2$. Thus $\mathcal{X}_2$ trivially satisfies $=(z, v)$ since all tuples in $\mathcal{X}_2$ disagree on z. Next we will show that $\mathcal{X}_1$ satisfies $=(x, y)$: Let $s, s' \in \mathcal{X}_1$, such that $s(x) = s'(x) = p_i$. Then by (3) it follows that $s(y) = F(p_i) = s'(y)$ holds. Thus $\mathcal{X}_1 \models\ =(x, y)$.

The other direction: Suppose $\mathcal{X} \models\ =(x, y) \vee\ =(z, v)$. Then there is a partition of $\mathcal{X}$ into $\mathcal{X}_1$ and $\mathcal{X}_2$, such that $\mathcal{X}_1 \models =(x, y)$ and $\mathcal{X}_2 \models =(z, v)$. We will define the assignment $F : \{p_0, \ldots, p_m\} \to \{0, 1\}$ in the following way:

- If $\exists s \in \mathcal{X}_1$, such that $s(x) = p_i$, then $F(p_i) = s(y)$.
- If $\forall s \in \mathcal{X}_1$ it holds $s(x) \neq p_i$, then $F(p_i) = 1$.[2]

Let us show that $F : \{p_0, \ldots, p_{m-1}\} \to \{0, 1\}$ is a function, which evaluates $\theta(p_0, \ldots, p_{m-1})$ true:

1. Clearly, $Dom(F) = \{p_0, \ldots, p_{m-1}\}$ and $Range(F) = \{0, 1\}$.

2. F is a function: Let $p_i \in \{p_0, \ldots, p_m\}$. Suppose there exists $s, s' \in \mathcal{X}_1$, such that $s(x) = s'(x) = p_i$ holds. Since $\mathcal{X}_1 \models\ =(x, y)$ holds, it follows that $s(y) = s'(y)$ holds. Suppose there are no $s \in \mathcal{X}_1$, such that $s(x) = p_i$. Then by definition of F it holds that $F(p_i) = 1$.

3. F evaluates Θ_X true: Note that z is constant and v is assigned different value by each tuple in each $\mathcal{X}_{E_i}$. Thus $\mathcal{X}_1$ contains at least one of the tuples from each $\mathcal{X}_{E_i}$. Let $s_0 \in \mathcal{X}_{E_i}$, such that $s_0 \in \mathcal{X}_1$. Since each tuple codes a satisfying condition of E_i it follows that F evaluates one of the disjuncts in E_i true. Thus $S(E_i) = 1$.

Each conjunct of θ gives rise to a constant size team of two assignments with domain $\{x, y, z, v\}$. Thus the team $\mathcal{X}$ can be constructed in LOGSPACE when given θ.

□

The problem 2−SAT is known to be complete for NL [4]. Now we have the following corollary:

Corollary 4.8. *$MC(=(x, y) \vee =(z, v)$ is complete for NL.*

[2] If for all the assignments $s \in X_1$ holds $s(x) \neq p_i$, then the value of p_i is not relevant to the satisfiability of Θ. Thus the value of p_i can be chosen 0 or 1.

Next we will show that when we consider formulas with two disjunctions, the model checking becomes NP-complete for certain formulas.

4.3 Non-deterministic polynomial time

We will reduce 3-SAT to the model checking problem of the formula $=(x, y) \vee =(z, u) \vee =(z, v)$.

Recall that an instance $\theta \in$ 3-SAT is a first-order formula in conjunctive normal form, where each conjunct has at most three variables: $\bigwedge_{i \in I} E_i$, where I is finite. Each E_i is of form $(A_{i_0} \vee A_{i_2} \vee A_{i_3})$, where A_i is either a positive or a negated boolean variable. θ is accepted if there is an assignment, which evaluates θ true. The reduction is analogous with the reduction given in Theorem 4.7.

Theorem 4.9. $3 - SAT \leq_{LOGSPACE} MC(=(x, y) \vee =(z, v) \vee =(z, v))$.

Proof. Suppose $\theta(p_0, \ldots, p_{m-1})$ is an instance of 3-SAT with conjuncts E_i, $i \in I$. We will construct a team $\mathcal{X}$, such that the following are equivalent:

- $\mathcal{X} \models =(x, y) \vee =(z, v) \vee =(z, v)$.
- $\theta(p_0, \ldots, p_{m-1})$ is satisfiable.

For each conjunct E_i, $i \in I$, we create a team $\mathcal{X}_{E_i}$ where we code all the satisfying conditions of the clause E_i. Let $\mathcal{X} = \bigcup_{i \in I} \mathcal{X}_{E_i}$. For example, a clause $E_i = (p_l \vee \neg p_j \vee \neg p_k)$ will be satisfied if $p_l = 1$ or $p_j = 0$ or $p_k = 0$. The team for $(p_l \vee \neg p_j \vee \neg p_k)$ is the one in Table 4.

z	x	y	v
1	p_l	1	0
1	p_j	0	1
1	p_k	0	2

Table 4: A team for $(p_l \vee \neg p_j \vee \neg p_k)$.

Suppose $\theta(p_0, \ldots, p_{m-1})$ is satisfiable. Then there exists an assignment $F : \{p_0, \ldots, p_{m-1}\} \to \{0, 1\}$, such that F evaluates $\theta(p_0, \ldots, p_{m-1})$ true. We define $\mathcal{X}_1$ in the following way for all $s \in \mathcal{X}$: $s \in \mathcal{X}_1$ if

$$s(x) = p_) \to F(p_i) = s(y) \tag{4}$$

Since F evaluates $\bigwedge_{i \in n} E_i$ true, it evaluates every conjunct E_i true. Furthermore, since we coded all the satisfying conditions of E_i into $\mathcal{X}_{E_i}$, it holds that at least one assignment from each $\mathcal{X}_{E_i}$ satisfies the condition (4). Thus $\mathcal{X}_1$ contains at least one assignment from each $\mathcal{X}_{E_i}$. Thus the two "leftover"-assignment form each $\mathcal{X}_{E_i}$ can be easily divided into $\mathcal{X}_2$ and $\mathcal{X}_3$ in such a way that $=(z, v)$ holds in both of them. We just place one of the assignments into $\mathcal{X}_2$ and one into $\mathcal{X}_3$.

Let us show that $\mathcal{X}_1 \models =(x, y)$: Suppose $s, s' \in \mathcal{X}_1$, such that $s(x) = s'(x) = p_i$. Then, by (4), it follows that $s(y) = s'(y) = F(p_i)$. Thus $\mathcal{X}_1 \models = (x, y)$.

The other direction: Suppose $\mathcal{X} \models =(x, y) \vee =(z, v) \vee =(z, v)$ holds. Then by the truth definition of the disjunction, it follows that $\mathcal{X}$ can be partitioned into three sets $\mathcal{X}_1$, $\mathcal{X}_2$ and $\mathcal{X}_3$, such that $\mathcal{X}_1 \models =(x, y)$, $\mathcal{X}_2 \models =(z, v)$ and $\mathcal{X}_3 \models =(z, v)$ hold. Let F be defined in the following way for each variable p_i.

- If $\exists s \in \mathcal{X}_1$, such that $s(x) = p_i$, then $F(p_i) = s(y)$.
- If $\forall s \in \mathcal{X}_1$ it holds $s(x) \neq p_i$, then $F(p_i) = 1$.

Let us show that $F : \{p_0, \ldots, p_m\} \to \{0, 1\}$ is a function, which evaluates $\theta(p_0, \ldots, p_{m-1})$ true:

1. Clearly, F is well defined and the domain of F is $\{p_0, \ldots, p_{m-1}\}$ and the range is $\{0, 1\}$.
2. F is a function: Let $p_i \in \{p_0, \ldots, p_m\}$. Suppose there exists $s, s' \in \mathcal{X}_1$, such that $s(x) = s'(x) = p_i$ holds. Since $\mathcal{X}_1 \models =(x, y)$ holds, it follows that $s(y) = s'(y) = F(p_i)$ holds. If there exists no $s \in \mathcal{X}_1$, such that $s(x) = p_i$, then it holds by the definition of F, that $F(p_i) = 1$.
3. We will show that S evaluates each E_i, $i \in I$, true: Note that z is constant and v gets different value by each tuple in each $\mathcal{X}_{E_i}$. Thus $\mathcal{X}_1$ must contain at least one of the tuples from each X_{E_i}. Since each tuple in $\mathcal{X}_{E_i}$ codes a satisfying condition for E_i it means that F agrees with one of the satisfying conditions for E_i. Thus F satisfies E_i.

Each conjunct of theta gives rise to a constant size team of three assignments with domain $\{x, y, z, v\}$. Thus given θ, $\mathcal{X}$ can be constructed in LOGSPACE.
□

3-SAT is complete for NP [3]. We have the following corollary:

Corollary 4.10. *$MC(=(x, y) \vee =(z, v) \vee =(z, v))$ is complete for NP.*

References

[1] Ronald Fagin, *Generalized First-Order Spectra and Polynomial-Time Recognizable Sets*, Complexity of Computation, ed. R. Karp, SIAM-AMS Proceedings 7, 27-41, 1974.

[2] E. Grädel, P. G. Kolaitis, L. Libkin, *Finite Model Theory And Its Applications* Springer, 2007.

[3] R. M. Karp, *Reducibility Among Combinatorial Problems*, in R. E. Miller and J. W. Thatcher (editors). Complexity of Computer Computations. New York: Plenum. pp. 85-103, 1972.

[4] C.H. Papadimitriou, *Computational Complexity*, Addison Wesley, 1993.

[5] Jouko Väänänen, *Dependence Logic: A New Approach to Independence Friendly Logic*, Cambridge University Press, 2007.

Logics of Imperfect Information without Identity

Antti Kuusisto*†
Department of Mathematics and Statistics
University of Tampere
Finland

Abstract

We investigate the expressive power of sentences of the family of independence-friendly (IF) logics in the equality-free setting. Various natural equality-free fragments of logics in this family translate into the version of existential second-order logic with prenex quantification of function symbols only and with the first-order parts of formulae equality-free. We study this version of existential second-order logic. Our principal result is that over finite models with a vocabulary consisting of unary relation symbols only, this fragment of second-order logic is weaker in expressive power than first-order logic. Such results could turn out useful in the study of independence-friendly modal logics. In addition to proving results of a technical nature, we consider issues related to a perspective where IF logic is regarded as a specification framework for games, and also discuss the significance of understanding fragments of second-order logic in investigations related to non-classical logics.

1 Introduction

We investigate the family of *independence-friendly* (IF) *logics* introduced by Hintikka and Sandu in [8]. See also [7] for an early exposition of the main ingredients leading to the idea of IF logic, and of course [5] for an even earlier discussion of ideas closely related to IF logic. Variants of IF logic have received a lot of attention recently; see [1, 4, 9, 10, 11, 13, 16] for example. Therefore we believe that the time is beginning to be mature for investigations not directly related to technical aspects concerning semantical issues. The focus of our work is the expressive power of the equality-free fragment of IF logic without slashed *connectives.* To be exact, we study the

*email: antti.j.kuusisto@uta.fi

†Participates in the European Science Foundation project Logic for Interaction (LINT). The research leading to the current article was supported by MALJA Graduate School in Mathematical Logic (Academy of Finland grant 118815) and Tampere Graduate School in Information Science and Engineering TISE (Academy of Finland grant 129892).

fragment of the system IF* (see [1]) without equality and without slashed connectives. We denote this fragment by IF$_{wo=}$.

Even though motivated by questions related to the expressive power of IF$_{wo=}$, our study concerns a wider range of logics. In fact, our study focuses on the system fESO$_{wo=}$ which is the version of existential second-order logic where the second-order quantifiers quantify function symbols only and where equality is not used. Here we allow for the function symbols to be nullary, i.e., to be interpreted as constants. With a careful inside-out Skolemization procedure preceeded by some preprocessing, any sentence of IF$_{wo=}$ can be turned into a sentence of fESO$_{wo=}$ that defines exactly the same class of models as the original IF$_{wo=}$ sentence. However, results about fESO$_{wo=}$ automatically apply to a wider range of logics. For example, the delightfully exotic looking expressions of the form

$$\begin{pmatrix} \forall x_1 & \exists x_2 \\ \forall x_3 & \exists x_4 \end{pmatrix} \varphi(x_1, x_2, x_3, x_4),$$

where a finite partially ordered quantifier preceeds an equality-free FO formula $\varphi(x_1, x_2, x_3, x_4)$ (with the free variables x_1, x_2, x_3, x_4), are equivalent to sentences of fESO$_{wo=}$ by the definition of Henkin [5]. Hence, whatever is inexpressible in fESO$_{wo=}$, is automatically inexpressible with expressions of the above type. Thus by studying fESO$_{wo=}$ we can kill multiple birds with one stone. This is part of a more general phenomenon. Results about fragments of second-order logic are very useful in the study of non-classical logics with devices giving them the capacity to express genuinely second-order properties. A typical such non-classical logic often immediately translates into a fragment of second-order logic. Then, armed with theorems about fragments of second-order logic, one may immediately obtain a range of metatheoretic results concerning the non-classical logic in question. Such results could be, for example, related to decidability issues. By directing attention to *fragments* of second-order logic rather than the full system of second-order logic, one can often easily identify, for example, truth preserving model transformations etc. The very high expressive power of second-order logic seems to often make it very difficult to obtain results like truth preserving model transformation theorems applying to *all* sentences of the system. These considerations provide part of the motivation for our study of the system fESO$_{wo=}$.

In addition to contributing to the general program of studying fragments of second-order logic, we believe that insights about sentences of the equality-free systems IF$_{wo=}$ and fESO$_{wo=}$ can be more or less directly useful in the study of the independence-friendly modal logics of Tulenheimo [15] and Tulenheimo and Sevenster [14] and others. This is due to the fact that formulae of such systems tend to translate to formulae of IF$_{wo=}$. This realization provides an example that demonstrates the significance of the claim made about the study of fragments of second-order logic above.

In this paper we study the expressivity of *sentences* of $\mathrm{IF}_{wo=}$ only. A sentence of $\mathrm{IF}_{wo=}$ *defines* the class of models on which *Eloise* has a winning strategy in the related semantic game. We begin the paper by observing that $\mathrm{fESO}_{wo=}$ can define properties not definable in first-order logic FO (with equality), when the vocabulary under consideration contains at least one binary relation symbol. We then define a simple model-transformation that preserves the truth of $\mathrm{fESO}_{wo=}$ sentences, but not FO sentences. Therefore we observe that $\mathrm{fESO}_{wo=}$ and FO are incomparable with regard to expressive power. The same transformation of course also preserves the truth of $\mathrm{IF}_{wo=}$ sentences. We discuss the significance of this observation in relation to the use of IF logic as a specification language for games.

Finally, we ask whether $\mathrm{fESO}_{wo=}$ and FO are also incomparable with regard to expressive power when attention is limited to vocabularies containing only unary relation symbols. Our principal result is that over finite models with such a vocabulary,

$$\mathrm{FO}_{wo=} < \mathrm{fESO}_{wo=} < \mathrm{FO},$$

where $\mathrm{FO}_{wo=}$ denotes first-order logic without equality. So far we have not succeeded in establishing these results without the use of somewhat involved combinatorial arguments.

2 Preliminary Considerations

We assume the reader is familiar with first-order logic and independence-friendly logic. For a tour of properties of IF logic, see [1]. The version of IF logic studied in this paper is the version where *slashed quantifiers* $\exists x/\{y_0, ...y_i\}$, $\forall x/\{y_0, ...y_i\}$ are allowed, but disjunctions and conjunctions do not have slash sets associated with them. To be exact, we study the fragment of the system IF^* (see [1]) without equality and without slashed connectives. We call this logic $\mathrm{IF}_{wo=}$. For the semantics of $\mathrm{IF}_{wo=}$, see Definition 4.2 in [1].

Our main tool in investigating $\mathrm{IF}_{wo=}$ is the logic $\mathrm{fESO}_{wo=}$, whose formulae are exactly the formulae of the type $\exists \overline{f}\varphi$, where $\overline{f}$ is a finite vector of function symbols and φ is an FO formula without equality. The function symbols are allowed to be nullary, i.e., to be interpreted as constants. The formulae of $\mathrm{fESO}_{wo=}$ are interpreted according to the natural semantics.

A sentence φ of IF^* is called equivalent to a sentence ψ of $\mathrm{fESO}_{wo=}$ (or, equivalently, a sentence of FO) if and only if *Eloise* has a winning strategy in the semantic game defined by φ exactly on those models where ψ is true. Any equality-free sentence of IF^* without slashed *connectives*, i.e., a sentence of $\mathrm{IF}_{wo=}$, can be transformed to an equivalent sentence of $\mathrm{fESO}_{wo=}$. We base this claim on Theorem 10.2 of [1] which implies that any sentence of $\mathrm{IF}_{wo=}$ can be put to an equivalent prenex normal form with exactly the original propositional skeleton, and the transformation can be done so that connectives and quantifiers without slash sets associated with them remain unslashed. As the propositional skeleton of the new prenex sentence is the same as that of the original sentence, the transformation process does not introduce equality symbols. Furthermore, we obtain a sentence that is *regular*, implying that no quantifier for a variable occurs within the scope of another quantifier for the same varible. See [1] for details. A sentence in this normal form can then be Skolemized in a careful inside-out fashion. The procedure eliminates existential quantifiers and introduces fresh function symbols. The related functions encode the way *Eloise* can play the semantic game. The procedure does not introduce equality or slashed connectives. The slash sets associated with universal quantifiers get eliminated. Finally, the fresh function symbols are prenex quantified existentially, resulting in a sentence of $\mathrm{fESO}_{wo=}$ equivalent to the original $\mathrm{IF}_{wo=}$ sentence.

The reader uneasy about this translation should note that the results in the current paper are mainly about for $\mathrm{fESO}_{wo=}$, and the statements about $\mathrm{IF}_{wo=}$ are mostly nothing more than direct corollaries to results concerning $\mathrm{fESO}_{wo=}$.

3 Expressivity of $\mathrm{IF}_{wo=}$ and $\mathrm{fESO}_{wo=}$ over Models with a Relational Vocabulary

We begin the section by making the simple observation that $\mathrm{IF}_{wo=}$ and FO are incomparable with regard to the expressive power of sentences over vocabularies containing at least one binary relation symbol. Here we do not limit our attention to finite models only.

Proposition 3.1. *Let V be a vocabulary containing at least one binary relation symbol R. Then there is a class of V-models definable by a sentence of* $\mathrm{IF}_{wo=}$ *and also a sentence of* $\mathrm{fESO}_{wo=}$ *that is not definable by a sentence of* FO.

Proof. It is well known that there is an IF sentence φ (*with* equality and without slashed connectives) that is true in exactly those models whose domain has an even or an infinite cardinality. Let φ' be the sentence obtained from φ by replacing each atom of the type $x = y$ by the atom $R(x, y)$. Let C be the class of finite V-models $\mathfrak{A}$ such that

$$R^{\mathfrak{A}} = \{\ (a, a) \mid a \in Dom(\mathfrak{A})\ \}.$$

It is clear that with respect to C, the sentence φ' defines the class C_{even} of models whose domain is even. A straightforward Ehrenfeucht-Fraïssé argument shows that the class C_{even} is not definable with respect to C by any FO sentence. Since there is no FO sentence that is equivalent over C to φ', there is no FO sentence equivalent to φ'.

Since φ' can be transformed to an equivalent fESO$_{wo=}$ sentence, it follows that fESO$_{wo=} \not\leq$ FO with regard to expressive power of sentences over the class of V-models. □

3.1 Bloating Models

We now define a model-transformation under which the truth of fESO$_{wo=}$ sentences is preserved.

Definition 3.2. Let V be a relational vocabulary containing only unary and binary relation symbols. (We restrict our attention to at most binary relation symbols for the sake of simplicity.) Let $\mathfrak{A}$ be a V-model with the domain A, and let $a \in A$. Let S be some set such that $S \cap A = \emptyset$. Define the V-model $\mathfrak{B}$ as follows.

1. The domain of $\mathfrak{B}$ is the set $A \cup S$.
2. Let $P \in V$ be a unary relation symbol. We define $P^{\mathfrak{B}}$ as follows.
   (a) For all $v \in A$, $v \in P^{\mathfrak{B}}$ iff $v \in P^{\mathfrak{A}}$.
   (b) For all $s \in S$, $s \in P^{\mathfrak{B}}$ iff $a \in P^{\mathfrak{A}}$.
3. Let $R \in V$ be a binary relation symbol. We define $R^{\mathfrak{B}}$ as follows.
   (a) For all $\overline{v} \in A \times A$, $\overline{v} \in R^{\mathfrak{B}}$ iff $\overline{v} \in R^{\mathfrak{A}}$.
   (b) For all $s \in S$ and all $v \in A$, $(v, s) \in R^{\mathfrak{B}}$ iff $(v, a) \in R^{\mathfrak{A}}$.
   (c) For all $s \in S$ and all $v \in A$, $(s, v) \in R^{\mathfrak{B}}$ iff $(a, v) \in R^{\mathfrak{A}}$.
   (d) For all $s, s' \in S$, $(s, s') \in R^{\mathfrak{B}}$ iff $(a, a) \in R^{\mathfrak{A}}$.

We call the model $\mathfrak{B}$ a *bloating* of $\mathfrak{A}$. Figure 1 illustrates how this model transformation affects models.

Theorem 3.3. *Let V be a vocabulary containing unary and binary relation symbols only. The truth of* fESO$_{wo=}$ *sentences is preserved from V-models to their bloatings.*

Proof. Let $\mathfrak{A}$ be a V-model and φ a sentence of fESO$_{wo=}$. We assume that φ is of the form $\exists \overline{f} \psi$, where the symbols f are function symbols (some of them perhaps nullary) and ψ is a first-order sentence without existential quantifiers and with negations pushed to the atomic level. This normal form is obtained by first transferring the first-order part of φ into negation normal form and then Skolemizing the resulting sentence. The freshly introduced Skolem functions are prenex quantified existentially, so the vocabulary of

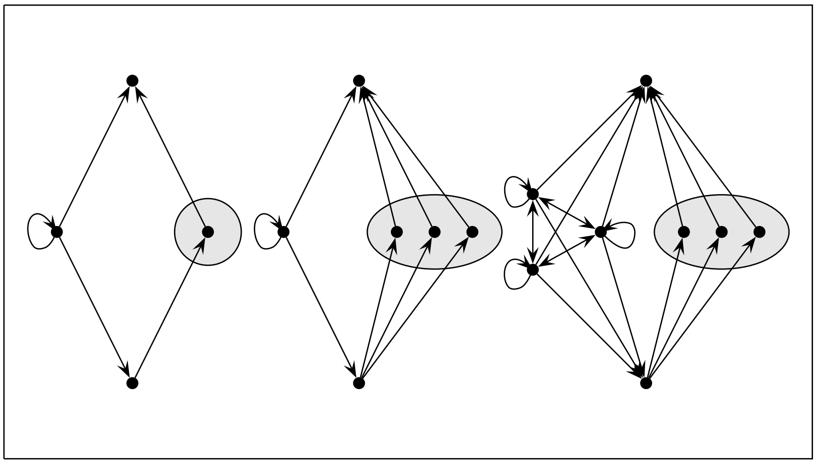

Figure 1: The figure shows three structures of a vocabulary consisting of one binary and one unary relation symbol. The shaded areas correspond to the extensions of the unary relation symbol. The structure in the middle is a bloating of the structure on the left. The structure in the middle is obtained from the one on the left by adding two new copies of the middle right element. The structure on the right is a bloating of the structure in the middle obtained by adding two copies of the middle left element.

$\exists\overline{f}\psi$ is the same as that of φ. The process of transferring φ into the described normal form does not introduce equality.

Let $\mathfrak{A}$ and $\mathfrak{B}$ be as in Definition 3.2. The models there had the domains A and $A\cup S$, respectively, and the element $a \in A$ was used in order to define $\mathfrak{B}$. We assume that $\mathfrak{A} \models \exists\overline{f}\psi$ and expand $\mathfrak{A}$ to a model $\mathfrak{A}' = (\mathfrak{A}, \overline{f^{\mathfrak{A}'}})$ such that $\mathfrak{A}' \models \psi$. We then expand $\mathfrak{B}$ to a model $\mathfrak{B}' = (\mathfrak{B}, \overline{f^{\mathfrak{B}'}})$ as follows.

1. For each k-ary symbol f, we let $f^{\mathfrak{B}'} \upharpoonright A^k = f^{\mathfrak{A}'} \upharpoonright A^k$. Note that when $k = 0$, i.e., when f is a constant symbol, then $f^{\mathfrak{B}'} = f^{\mathfrak{A}'}$.

2. For each k-tuple $\overline{w} \in (A \cup S)^k$ containing points from the set S, we define the k-tuple $\overline{w}'$, where each co-ordinate value $s \in S$ of $\overline{w}$ is replaced by the element a. We then set $f^{\mathfrak{B}'}(\overline{w}) = f^{\mathfrak{A}'}(\overline{w}')$.

We then establish that $\mathfrak{B}' \models \psi$. The proof is a simple induction on the structure of ψ. For each variable assignment h with codomain A, let $g(h)$ denote the set of all variable assignments with codomain $A \cup S$ that can be obtained from h by allowing some subset of the variables mapping to the element a to map to elements in S. We prove that for every variable assignment h with codomain A and every subformula χ of ψ,

$$\mathfrak{A}', h \models \chi \;\Rightarrow\; \forall h' \in g(h)(\mathfrak{B}', h' \models \chi).$$

The cases for atomic and negated atomic formulae form the basis of the induction. The claim for these formulae follows immediately with the help of the observation that $h(t) = h'(t)$ for all h and $h' \in g(h)$ and terms t that *contain function symbols*, i.e., terms that are not variable symbols. We will next establish this claim by induction on the nesting depth of function symbols.

The basis of the induction deals with the terms of nesting depth one, i.e., terms of the type $f(x_1, ..., x_k)$ and c, where the symbols $x_1, ..., x_k$ are variable symbols and the symbol c is a constant symbol. It is immediate that $h(t) = h'(t)$ for all h and $h' \in g(h)$ and all such terms t of nesting depth one.

Now let $f(t_1, ..., t_k)$ be a term of nesting depth $n + 1$. By the induction hypothesis, for each one of the terms t_i that is not a variable symbol, we have $h(t_i) = h'(t_i)$. For the terms t_i that are variable symbols and for which $h(t_i) \neq a$, we have $h(t_i) = h'(t_i)$. For the terms t_i that are variable symbols and for which $h(t_i) = a$, we have either $h'(t_i) = a$ or $h'(t_i) \in S$. We therefore notice that we obtain the tuple $(h(t_1), ..., h(t_k))$ from the tuple $(h'(t_1), ..., h'(t_k))$ by replacing the elements $u \in S$ of the tuple $((h'(t_1), ..., h'(t_k))$ by the element a. Therefore we conclude, by the definition of the function $f^{\mathfrak{B}'}$, that

$$f^{\mathfrak{B}'}(h'(t_1), ..., h'(t_k)) = f^{\mathfrak{A}'}(h(t_1), ..., h(t_k)).$$

This concludes the induction on terms and therefore the basis of the original induction on the structure of ψ has now been established. We return to the original induction.

The connective cases are trivial and the quantifier case relatively straightforward. We discuss the details of the quantifier case here.

Assume $\mathfrak{A}', h \models \forall x \alpha(x)$. We need to show that for all $h' \in g(h)$, $\mathfrak{B}', h' \models \forall x \alpha(x)$. Assume, for contradiction, that for some $h'' \in g(h)$ we have $\mathfrak{B}', h'' \not\models \forall x \alpha(x)$. Therefore, for some $u \in A \cup S$, we have $\mathfrak{B}', h''\frac{u}{x} \not\models \alpha(x)$. It suffices to show that $h''\frac{u}{x} \in g(h\frac{v}{x})$ for some $v \in A$. This suffices, as the assumption $\mathfrak{A}', h \models \forall x \alpha(x)$ first implies that $\mathfrak{A}', h\frac{v}{x} \models \alpha(x)$, which in turn then implies, by the induction hypothesis, that $\mathfrak{B}', h''\frac{u}{x} \models \alpha(x)$.

If $u \in A$, let $v = u$. Then, as $h'' \in g(h)$, we have $h''\frac{u}{x} = h''\frac{v}{x} \in g(h\frac{v}{x})$. If $u \in S$, we let $v = a$. Then, as $h'' \in g(h)$, we have $h''\frac{u}{x} \in g(h\frac{a}{x}) = g(h\frac{v}{x})$. □

An immediate consequence of Theorem 3.3 is that FO $\not\leq$ fESO$_{wo=}$ because there exist first-order sentences whose truth is not preserved under bloating.

Theorem 3.3 is interesting when regarding IF logic as a kind of a specification language for games. Let V be a vocabulary of the type defined in Theorem 3.3. Let the equality-free and slash connective-free IF sentence φ of the vocabulary V specify some class of games and assume we know some board (i.e., a V-model) on which *Eloise* wins the game (i.e., φ is true on that model). The theorem then gives us a whole range of new, larger boards where she wins the game specified by φ. On the other hand, *non-winning* and in fact even *indeterminacy* are clearly preserved in reverse bloatings. This follows directly by a dualization argument.

4 Expressivity of $\mathrm{fESO}_{wo=}$ and $\mathrm{IF}_{wo=}$ over Finite Models with a Unary Relational Vocabulary

We now turn our attention to finite models whose vocabulary contains only unary relation symbols. Over such finite models, the picture is quite different from the case where there is a binary relation symbol in the vocabulary. We will show that over the class of finite models whose vocabulary contains only unary relation symbols,

$$\mathrm{FO}_{wo=} < \mathrm{fESO}_{wo=} < \mathrm{FO}.$$

We first discuss the latter inequality and then the former one.

4.1 $\mathrm{fESO}_{wo=} < \mathrm{FO}$ over the Class of Finite Models with a Unary Vocabulary

In this subsection, we establish that $\mathrm{fESO}_{wo=} < \mathrm{FO}$ over the class of finite models with a unary relational vocabulary. Therefore also $\mathrm{IF}_{wo=} < \mathrm{FO}$ over that class. We begin by making a number of auxiliary definitions.

Let U be a finite vocabulary containing unary relation symbols only. A *unary U-type* (with the free variable x) is a conjunction τ with $|U|$ conjuncts such that for each $P \in U$, exactly one of the formulae $P(x)$ and $\neg P(x)$ is a conjunct of τ. Let $T = \{\tau_1, ..., \tau_{|T|}\}$ be the set of unary U-types. The domain of each (finite) U-model $\mathfrak{A}$ is partitioned into some number $n \leq |T|$ of sets S_i such that the elements of S_i *realize*, i.e., satisfy, the type $\tau_i \in T$. (Here an element $a \in Dom(\mathfrak{A})$ realizes (satisfies) the type τ_i if and only if $\mathfrak{A} \models \tau_i(a)$ in the usual sense of first-order logic.)

Let $n \in \mathbb{N}_{\geq 1}$, and let $k = 2^n$. Any relation

$$R \subseteq \mathbb{N}^k \setminus \{0\}^k$$

is called a *spectrum*. We associate sentences of FO and $\mathrm{fESO}_{wo=}$ with spectra in a way specified in the following definition.

Definition 4.1. Let V be a vocabulary containing unary relation symbols only. Let φ be a sentence of FO or $\mathrm{fESO}_{wo=}$ of the vocabulary V. Let $U \subseteq V$ be the finite set of relation symbols occurring in φ. Let $T = \{\tau_1, ..., \tau_{|T|}\}$ be the finite set of unary U-types, and let $\leq^T$ denote a linear ordering of the types in T defined such that $\tau_i \leq^T \tau_j$ iff $i \leq j$. Define the relation $R_\varphi \subseteq \mathbb{N}^{|T|}$ such that $(n_1, ..., n_{|T|}) \in R_\varphi$ iff there exists a finite U-model $\mathfrak{A}$ of φ such that for all $i \in \{1, ..., |T|\}$, the number of points in the domain of $\mathfrak{A}$ that satisfy τ_i is n_i. We call such a relation R_φ the *spectrum of* φ (with respect to the ordering $\leq^T$).

Notice that the class of finite V-models defined by φ is completely characterized by the spectrum $R_\varphi \subseteq \mathbb{N}^{|T|}$. We next define a special family of spectra and then establish that this family exactly characterizes the expressive power of FO over the class of (finite) models with a vocabulary containing unary relation symbols only. See Figure 2 for an illustration of a spectrum of a sentence of FO with a unary relational vocabulary.

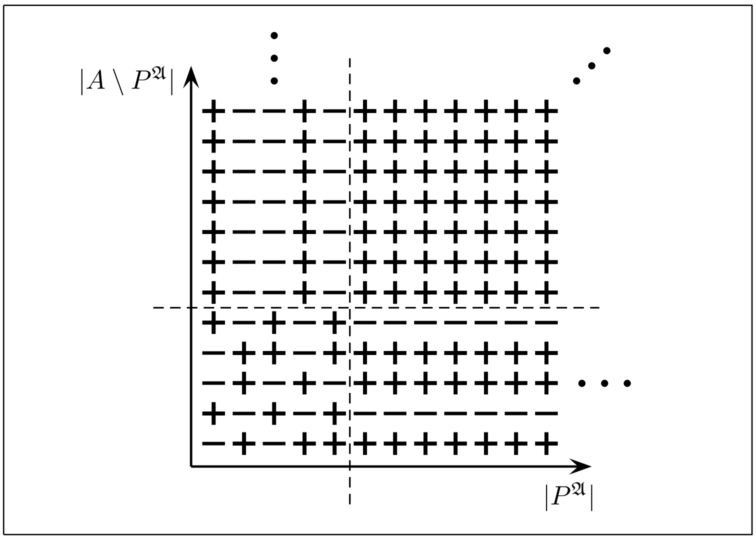

Figure 2: The figure illustrates a stabilizing spectrum that corresponds to some FO sentence φ of the vocabulary $\{P\}$, where P is a unary relation symbol. A plus symbol occurs at the position (i, j) iff there exists a $\{P\}$-model $\mathfrak{A}$ satisfying φ such that $|P^{\mathfrak{A}}| = i$ and $|A \setminus P^{\mathfrak{A}}| = j$, where $A = Dom(\mathfrak{A})$. In other words, the number of points in the domain of $\mathfrak{A}$ satisfying the type $P(x)$ is i and the number of points satisfying the type $\neg P(x)$ is j. The spectra for FO sentences divide the xy-plane into four distinct regions. The upper right region always contains either only plus symbols or only minus symbols. In the bottom left region, any distribution is possible. (The point $(0, 0)$ always contains a minus symbol though since we do not allow for models to have an empty domain.)

Definition 4.2. Let $l = 2^{l'}$ for some $l' \in \mathbb{N}_{\geq 1}$. Let $R \subseteq \mathbb{N}^l$ be a spectrum for which there exists a number $n \in \mathbb{N}_{\geq 1}$ such that for all co-ordinate positions $i \in \{1, ..., l\}$, all integers $k, k' > n$ and all $m_1, ..., m_{i-1}, m_{i+1}, ..., m_l \in \mathbb{N}$, we have

$$(m_1, ..., m_{i-1}, k, m_{i+1}, ..., m_l) \in R$$
$$\Leftrightarrow$$
$$(m_1, ..., m_{i-1}, k', m_{i+1}, ..., m_l) \in R.$$

We call such a number n a *stabilizer* of the spectrum R. A spectrum with a stabilizer is called a *stabilizing spectrum*.

Proposition 4.3. *A spectrum R is a stabilizing spectrum if and only if R is a spectrum of some* FO *sentence.*

Proof. Given a stabilizing spectrum, it is easy to write a corresponding FO sentence by applying the quantifiers $\exists^{=j}$ and $\exists^{\geq j}$ expressible with the use of the equality symbol. (Here $\exists^{=j}x\ \varphi(x)$ states that there exists exactly j elements a such that $\varphi(a)$ holds, and $\exists^{\geq j}$ is defined analogously.)

The fact that each spectrum of an FO sentence is stabilizing follows by a straightforward Ehrenfeucht-Fraïssé argument. □

Next we define some order theoretic concepts and then prove a number of related results that are needed for the proof of the main theorem (Theorem 4.7) of the current section.

A structure $\mathfrak{A} = (A, \leq^{\mathfrak{A}})$ is a *partial order* if $\leq^{\mathfrak{A}} \subseteq A \times A$ is a reflexive, transitive and antisymmetric binary relation. Given a partial order $\mathfrak{A} = (A, \leq^{\mathfrak{A}})$, we let $<^{\mathfrak{A}}$ denote the irreflexive version of the order $\leq^{\mathfrak{A}}$. A partial order is *well-founded* if no strictly decreasing infinite sequence occurs in it. That is, a partial order $\mathfrak{A} = (A, \leq^{\mathfrak{A}})$ is well-founded if for each each sequence $s : \mathbb{N} \longrightarrow A$ there exist numbers $i, j \in \mathbb{N}$ such that $i < j$ and $s(j) \not<^{\mathfrak{A}} s(i)$. An *antichain* $S \subseteq A$ of a partial order $\mathfrak{A} = (A, \leq^{\mathfrak{A}})$ is a set such that for all distinct elements $s, s' \in S$, we have $s \not\leq^{\mathfrak{A}} s'$ and $s' \not\leq^{\mathfrak{A}} s$. In other words, the distinct elements s and s' are incomparable. A well-founded partial order that does not contain an infinite antichain is called a *partial well order*, or a *pwo*.

Let $\mathfrak{A} = (A, \leq^{\mathfrak{A}})$ and $\mathfrak{B} = (B, \leq^{\mathfrak{B}})$ be partial orders. The *Cartesian product* $\mathfrak{A} \times \mathfrak{B}$ of the structures is the partial order defined in the following way.

1. The domain of $\mathfrak{A} \times \mathfrak{B}$ is the Cartesian product $A \times B$.
2. The binary relation $\leq^{\mathfrak{A}\times\mathfrak{B}} \subseteq (A \times B) \times (A \times B)$ is defined in a pointwise fashion as follows.

$$(a, b) \leq^{\mathfrak{A}\times\mathfrak{B}} (a', b') \Leftrightarrow \Big(a \leq^{\mathfrak{A}} a' \text{ and } b \leq^{\mathfrak{B}} b'\Big)$$

For each $k \in \mathbb{N}_{\geq 1}$ and each partial order $\mathfrak{A} = (A, \leq^{\mathfrak{A}})$, we let $\mathfrak{A}^k = (A^k, \leq^{\mathfrak{A}^k})$ denote the partial order where the relation $\leq^{\mathfrak{A}^k} \subseteq A^k \times A^k$ is again defined in the pointwise fashion as follows.

$$(a_1, ..., a_k) \leq^{\mathfrak{A}^k} (a'_1, ..., a'_k) \quad \Leftrightarrow \quad \forall i \in \{1, ..., k\} : a_i \leq^{\mathfrak{A}} a'_i$$

We call the structure $\mathfrak{A}^k$ the k^{th} *Cartesian power* of $\mathfrak{A}$. We let $(\mathbb{N}^k, \leq)$ denote the k^{th} Cartesian power of the linear order $(\mathbb{N}, \leq)$. When $S \subseteq \mathbb{N}^k$, we let $(S, \leq)$ denote the partial order with the domain S and with the ordering relation inherited from the structure $(\mathbb{N}^k, \leq)$. In other words, for all $\overline{s}, \overline{s}' \in S$, we have $\overline{s} \ \leq^{(S,\leq)} \ \overline{s}'$ if and only if $\overline{s} \ \leq^{(\mathbb{N}^k,\leq)} \ \overline{s}'$. We simply write $\overline{u} \leq \overline{v}$ in order to assert that $\overline{u} \leq^{(\mathbb{N}^k,\leq)} \overline{v}$, when $\overline{u}, \overline{v} \in \mathbb{N}^k$.

The following lemma is a paraphrase of Lemma 5 of [12], where the lemma is credited to Higman [6].

Lemma 4.4. *The Cartesian product of any two partial well orders is a partial well order.*

Variations of the following lemma are often attributed to Dickson [3]. The lemma follows immediately from Lemma 4.4.

Lemma 4.5. *Let* $k \in \mathbb{N}_{\geq 1}$. *The structure* $(\mathbb{N}^k, \leq)$ *does not contain an infinite antichain.*

Proof. The structure $(\mathbb{N}, \leq)$ is a pwo, and the property of being a pwo is preserved under taking finite Cartesian products by Lemma 4.4. Therefore the structure $(\mathbb{N}^k, \leq)$ is a pwo. By definition, a pwo does not contain an infinite antichain. □

Let $l \in \mathbb{N}_{\geq 1}$ and let $R \subseteq \mathbb{N}^l$ be a relation such that for all $\overline{u}, \overline{v} \in \mathbb{N}^l$, if $\overline{u} \in R$ and $\overline{u} \leq \overline{v}$, then $\overline{v} \in R$. We call the relation R *upwards closed* with respect to $(\mathbb{N}^l, \leq)$. When the exponent l is irrelevant or known from the context, we simply say that the relation R is upwards closed.

Theorem 4.6. *Let* $l' \in \mathbb{N}_{\geq 1}$ *and* $l = 2^{l'}$. *Let* $R \subseteq \mathbb{N}^l$ *be a relation that is upwards closed with respect to* $(\mathbb{N}^l, \leq)$. *Then* R *is a stabilizing spectrum.*

Proof. We begin the proof by defining a function f that maps each non-empty subset of the set $\{1, ..., l\}$ to a natural number. Let $C \subseteq \{1, ..., l\}$ be a non-empty set. Let $R(C)$ denote the set consisting of exactly those tuples $\overline{w} \in R$ that have a non-zero co-ordinate value at each co-ordinate position $i \in C$ and a zero co-ordinate value at each co-ordinate position $j \in \{1, ..., l\} \setminus C$. Define the value $f(C) \in \mathbb{N}$ as follows.

1. If $R(C) = \emptyset$, let $f(C) = 0$.
2. If $R(C) \neq \emptyset$, choose some $\overline{w} \in R(C)$. Let $W \subseteq R(C)$ be a maximal antichain of $(R(C) \ \leq)$ with $\overline{w} \in W$, i.e., let W be an antichain of $(R(C), \leq)$ such that for all $\overline{u} \in R(C) \setminus W$, there exists some $\overline{v} \in W$ such that $\overline{u} < \overline{v}$ or $\overline{v} > \overline{u}$. By Lemma 4.5, we see that the set W is finite. Thus there exists a maximum co-ordinate value occurring in the tuples in W. Let $f(C)$ to be equal to this value.

(Notice that we have some freedom of *choice* when defining the function f, so there need not be a unique way of defining the function.)

With the function f defined, call

$$n = max(\{\ f(C) \mid C \subseteq \{1, ..., l\},\ C \neq \emptyset\ \}).$$

We establish that n is a stabilizer for the relation R. We assume, for the sake of contradiction, that there there exist integers $k, k' > n$ and $m_1, ..., m_{i-1}, m_{i+1}, ..., m_l \in \mathbb{N}$ such that the equivalence

$$\begin{array}{c} (m_1, ..., m_{i-1}, k, m_{i+1}, ..., m_l) \in R \\ \Leftrightarrow \\ (m_1, ..., m_{i-1}, k', m_{i+1}, ..., m_l) \in R. \end{array}$$

does not hold. Let $k < k'$. As by assumption the relation R is upwards closed, it must be the case that

$$\begin{array}{c} (m_1, ..., m_{i-1}, k, m_{i+1}, ..., m_l) \notin R \\ \text{and} \\ (m_1, ..., m_{i-1}, k', m_{i+1}, ..., m_l) \in R. \end{array}$$

Otherwise we would immediately reach a contradiction. Call

$$\begin{array}{c} \overline{w}_k = (m_1, ..., m_{i-1}, k, m_{i+1}, ..., m_l) \\ \text{and} \\ \overline{w}_{k'} = (m_1, ..., m_{i-1}, k', m_{i+1}, ..., m_l). \end{array}$$

Let $C^* \subseteq \{1, ..., l\}$ be the set of co-ordinate positions where the tuple $\overline{w}_{k'}$ (and therefore also the tuple $\overline{w}_k$) has a non-zero co-ordinate value. Let $W(C^*)$ denote the domain of the maximal antichain of $(R(C^*), \leq)$ chosen when defining the value of the function f on the input C^*. The tuple $\overline{w}_{k'}$ cannot belong to the set $W(C^*)$, since the co-ordinate value k' is greater than n, and therefore greater than any of the co-ordinate values of the tuples in $W(C^*)$. Hence, as $W(C^*)$ is a maximal antichain of $(R(C^*), \leq)$ and $\overline{w}_{k'} \in R(C^*)$, we conclude that there exists a tuple $\overline{u} \in W(C^*)$ such that $\overline{w}_{k'} < \overline{u}$ or $\overline{u} < \overline{w}_{k'}$. Since $k' > f(C^*)$, we must have $\overline{u} < \overline{w}_{k'}$. Therefore, as also $k > f(C^*)$, we conclude that $\overline{u} < \overline{w}_k$. Since R is upwards closed and $\overline{u} \in R$, we have $\overline{w}_k \in R$. This is a contradiction, as desired. $\square$

The following theorem is the main result of the current section.

Theorem 4.7. *Over the class of finite models of a vocabulary V containing only unary relation symbols,* $\mathrm{fESO}_{wo=} < \mathrm{FO}$.

Proof. It is immediate by Theorem 3.3 that $\mathrm{fESO}_{wo=} \neq \mathrm{FO}$ (over finite V-models). It thus suffices to show that $\mathrm{fESO}_{wo=} \leq \mathrm{FO}$ over finite V-models.

To show that $\mathrm{fESO}_{wo=} \leq \mathrm{FO}$, by Proposition 4.3 it suffices to establish that the spectrum R_φ of an arbitrarily chosen $\mathrm{fESO}_{\mathrm{wo=}}$ sentence φ is stabilizing. By Theorem 3.3, the spectrum R_φ is upwards closed. Therefore, by Theorem 4.6, R_φ is a stabilizing spectrum. $\square$

Corollary 4.8. *Over finite models of a vocabulary containing only unary relation symbols,* $\mathrm{IF}_{wo=} < \mathrm{FO}$.

Note that Theorem 4.7 applies not only to $\mathrm{fESO}_{wo=}$ but to any system such that the definable classes of models with a unary vocabulary are closed under bloating. Note also that the method of proof seems nonconstructive in the sense that it seems to leave open the question whether there is an *effective translation* from the system considered into FO.

4.2 $\mathrm{FO}_{wo=} < \mathrm{IF}_{wo=}$ over the Class of Finite Models with a Unary Vocabulary

In this subsection we establish that over the class of finite $\{P\}$-models, where P is a unary relation symbol, we have $\mathrm{FO}_{wo=} < \mathrm{IF}_{wo=}$. The $\mathrm{IF}_{wo=}$ sentence

$$\forall x \exists y \exists z/\{x\}\big(P(y) \wedge (P(x) \leftrightarrow P(z))\big)$$

is true on a model $\mathfrak{M}$ with three points, two of which satisfy P. The sentence is not true on a model $\mathfrak{N}$ with two points, one satisfying P and one not. However, $\mathrm{FO}_{wo=}$ cannot separate the models $\mathfrak{M}$ and $\mathfrak{N}$. This is seen by a straightforward Ehrenfeucht-Fraïssé argument involving a version of the Ehrenfeucht-Fraïssé game that characterizes $\mathrm{FO}_{wo=}$. Instead of the usual partial isomorphism condition, this game involves the following end condition between the pebbles $a_1, ..., a_k \in A = Dom(\mathfrak{A})$ and $b_1, ..., b_k \in B = Dom(\mathfrak{B})$ picked during a play of the k-round game involving models $\mathfrak{A}$ and $\mathfrak{B}$ with a relational vocabulary. A play of the game defines a binary relation $Z = \{(a_1, b_1), ..., (a_k, b_k)\}$. The relation Z is called a *partial relativeness correspondence* between the models $\mathfrak{A}$ and $\mathfrak{B}$ if for all relation symbols R in the vocabulary of the models, the condition $Z(a'_1, b'_1), ..., Z(a'_n, b'_n)$ implies $R^{\mathfrak{A}}(a'_1, ..., a'_n) \Leftrightarrow R^{\mathfrak{B}}(b'_1, ..., b'_n)$. Here n is the arity of the symbol R. The duplicator wins the play of the game if the relation Z defined by the play is a partial relativeness correspondence. A discussion concerning the related Ehrenfeucht-Fraïssé characterization theorem can be found in [2].

Theorem 4.9. *Over finite models of a vocabulary V containing only unary relation symbols,* $\mathrm{FO}_{wo=} < \mathrm{IF}_{wo=}$.

Proof. It suffices to establish that the duplicator has a winning strategy in the Ehrenfeucht-Fraïssé game for $\mathrm{FO}_{wo=}$ for any number k of rounds played on the models $\mathfrak{M}$ and $\mathfrak{N}$ defined above. The duplicator employs a strategy where the reply to each one of the spoiler's moves is simply a pick of any element in the correct model that satisfies exactly the same unary $\{P\}$-type as the element chosen by the spoiler. □

Corollary 4.10. *Over finite models of a vocabulary V containing only unary relation symbols,* $\mathrm{FO}_{wo=} < \mathrm{fESO}_{wo=}$.

5 Concluding Remarks

We have investigated the expressive power of the equality-free version of IF logic without slashed connectives. The results obtained have been established through a study of the logic $\text{fESO}_{wo=}$. Our principal result is that over finite models with a vocabulary containing only unary relation symbols, the logics $\text{IF}_{wo=}$ and $\text{fESO}_{wo=}$ are weaker than FO. We have also identified a model-transformation that preserves the truth of $\text{IF}_{wo=}$ sentences.

In the future we expect to tie up some loose ends that were left undiscussed here. This includes considering infinite models. Furthermore, we wish to identify differences (rather than similarities) in the roles that different logical constructors – such as negation and identity – play in versions of IF logic and other logics of the same family such as dependence logic [16]. The full systems of dependence logic and IF^* coincide in expressive power on the level of sentences, both being able to exactly capture existential second-order logic. However, the systems might perhaps differ in expressive power when a suitable subset of the available logical constructors is uniformly removed from both systems. Another possibility is to restrict the number of available variable symbols to some finite number. The possibilities are endless indeed. Investigations along such lines should lead to a deeper understanding of the strengths and weaknesses different systems have in relation to different applications.

Acknowledgements. The author wishes to thank Lauri Hella, Kerkko Luosto, Allen Mann and Jonni Virtema for valuable feedback.

References

[1] Caicedo X., Dechesne F. and Janssen T. M. V. Equivalence and Quantifier Rules for Logic with Imperfect Information. *Logic Journal of the IGPL*, 17(1):91-129, 2009.

[2] Casanovas E., Dellunde P. and Jansana R. On Elementary Equivalence for Equality-free Logic. *Notre Dame Journal of Formal Logic*, 37(1):506-522, 1996.

[3] Dickson L. E. Finiteness of the Odd perfect and Primitive Abundant Numbers with n Distinct Prime Factors. *American Journal of Mathematics*, 35(4):413-422, 1913.

[4] Galliani P. *Probabilistic Dependence Logic.* M.Sc. thesis, ILLC, 2008.

[5] Henkin, L. Some remarks on infinitely long formulas. In *Infinitistic Methods: Proceedings of the Symposium on Foundations of Mathematics, Warsaw, 29 September 1959*, 167-183, Pergamon Press, 1961.

[6] Higman G. Ordering by Divisibility in Abstract Algebras. In *Proceedings of the London Mathematical Society 1952*, s3-2(1):326-336, 1952.

[7] Hintikka J. Quantifiers vs. Quantification Theory. *Dialectica*, 27:329-358, 1973.

[8] Hintikka J. and Sandu G. Informational Independence as a Semantical Phenomenon. In Fenstad J. E. et al., editors, *Logic, Methodology and Philosophy of Science VIII*, volume 126 of Studies in Logic and the Foundations of Mathematics, 571-589, North-Holland, 1989.

[9] Kontinen Jarmo. *Coherence and Complexity in Fragments of Dependence Logic*. Ph.D. thesis, ILLC, 2010.

[10] Kontinen Juha and Nurmi V. Team logic and second-order logic. In *Proceedings of WoLLIC 2009*, Springer Lecture Notes in Computer Science 5514, 2009.

[11] Kontinen Juha and Väänänen J. On Definability in Dependence Logic. *Journal of Logic, Language and Information*, 18(3):317-332, 2009.

[12] Malicki M. and Rutkowski A. On Operations and Linear Extensions of Well Partially Ordered Sets. *Order* 21(1):7-17, 2004.

[13] Mann A. L. Independence-friendly Cylindric Set Algebras. *Logic Journal of the IGPL*, 17(6):719-754, 2009.

[14] Sevenster M. and Tulenheimo T. Approaches to Independence Friendly Modal Logic. In van Benthem J., Gabbay D. and Löwe B., editors, *Interactive Logic. Selected Papers from the 7th Augustus de Morgan Workshop*, London. Texts in Logic and Games 1, Amsterdam: Amsterdam University Press, 247-280, 2007.

[15] Tulenheimo T. *Independence Friendly Modal Logic*. Ph.D. thesis, Department of Philosophy, University of Helsinki, 2004.

[16] Väänänen J. *Dependence Logic*. Vol. 70 of London Mathematical Society Student Texts. Cambridge University Press, 2007.

COMPLEXITY RESULTS FOR MODAL DEPENDENCE LOGIC

PETER LOHMANN AND HERIBERT VOLLMER

Institut für Theoretische Informatik
Leibniz Universität Hannover
Appelstr. 4, 30167 Hannover, Germany
E-mail address: {lohmann,vollmer}@thi.uni-hannover.de

Abstract. Modal dependence logic was introduced very recently by Väänänen. It enhances the basic modal language by an operator dep. For propositional variables $p_1, \ldots, p_n$, $\mathrm{dep}(p_1, \ldots, p_{n-1}; p_n)$ intuitively states that the value of p_n only depends on those of $p_1, \ldots, p_{n-1}$. Sevenster (J. Logic and Computation, 2009) showed that satisfiability for modal dependence logic is complete for nondeterministic exponential time.

In this paper we consider fragments of modal dependence logic obtained by restricting the set of allowed propositional connectives. We show that satisfibility for *poor man's dependence logic*, the language consisting of formulas built from literals and dependence atoms using $\wedge$, $\Box$, $\Diamond$ (i. e., disallowing disjunction), remains NEXPTIME-complete. If we only allow monotone formulas (without negation, but with disjunction), the complexity drops to PSPACE-completeness. We also extend Väänänen's language by allowing classical disjunction besides dependence disjunction and show that the satisfiability problem remains NEXPTIME-complete. If we then disallow both negation and dependence disjunction, satistiability is complete for the second level of the polynomial hierarchy.

In this way we completely classifiy the computational complexity of the satisfiability problem for all restrictions of propositional and dependence operators considered by Väänänen and Sevenster.

1. Introduction

The concept of extending first-order logic with partially ordered quantifiers, and hence expressing some form of independence between variables, was first introduced by Henkin [Hen61]. Later, Hintikka and Sandu developed independence friendly logic [HS89] which can be viewed as a generalization of Henkin's logic. Recently, Jouko Väänänen introduced the dual notion of functional dependence into the language of first-order logic [Vää07]. In the case of first-order logic, the independence and the dependence variants are expressively equivalent.

Dependence among values of variables occurs everywhere in computer science (databases, software engineering, knowledge representation, AI) but also the social sciences (human history, stock markets, etc.), and thus dependence logic is nowadays a much discussed formalism in the area called *logic for interaction.* Functional dependence of the value of a variable p_n from the values of the variables $p_1, \ldots, p_{n-1}$

Key words and phrases: dependence logic, satisfiability problem, computational complexity, poor man's logic.

Supported in part by the NTH Focused Research School for IT Ecosystems and by DFG VO 630/6-1.

states that there is a function, say f, such that $p_n = f(p_1, \ldots, p_{n-1})$, i. e., the value of p_n only depends on those of $p_1, \ldots, p_{n-1}$. We will denote this in this paper by $\mathrm{dep}(p_1, \ldots, p_{n-1}; p_n)$.

Of course, dependence does not manifest itself in a single world, play, event or observation. Important for such a dependence to make sense is a collection of such worlds, plays, events or observations. These collections are called *teams*. They are the basic objects in the definition of semantics of dependence logic. A team can be a set of plays in a game. Then $\mathrm{dep}(p_1, \ldots, p_{n-1}; p_n)$ intuitively states that in each play, move p_n is determined by moves $p_1, \ldots, p_{n-1}$. A team can be a database. Then $\mathrm{dep}(p_1, \ldots, p_{n-1}; p_n)$ intuitively states that in each line, the value of attribute p_n is determined by the values of attributes $p_1, \ldots, p_{n-1}$, i. e., that p_n is functionally dependent on $p_1, \ldots, p_{n-1}$. In first-order logic, a team formally is a set of assignments; and $\mathrm{dep}(p_1, \ldots, p_{n-1}; p_n)$ states that in each assignment, the value of p_n is determined by the values of $p_1, \ldots, p_{n-1}$. Most important for this paper, in modal logic, a team is a set of worlds in a Kripke structure; and $\mathrm{dep}(p_1, \ldots, p_{n-1}; p_n)$ states that in each of these worlds, the value of the propositional variable p_n is determined by the values of $p_1, \ldots, p_{n-1}$.

Dependence logic is defined by simply adding these dependence atoms to usual first-order logic [Vää07]. Modal dependence logic (MDL) is defined by introducing these dependence atoms to modal logic [Vää08, Sev09]. The semantics of MDL is defined with respect to sets T of worlds in a frame (Kripke structure) W, for example $W, T \models \mathrm{dep}(p_1, \ldots, p_{n-1}; p_n)$ if for all worlds $s, t \in T$, if $p_1, \ldots, p_{n-1}$ have the same values in both s and t, then p_n has the same value in s and t, and a formula

$$\Box \mathrm{dep}(p_1, \ldots, p_{n-1}; p_n)$$

is satisfied in a world w in a Kripke structure W, if in the team T consisting of all successor worlds of w, $W, T \models \mathrm{dep}(p_1, \ldots, p_{n-1}; p_n)$.

MDL was introduced in [Vää08]. Väänänen introduced besides the usual inductive semantics an equivalent game-theoretic semantics. Sevenster [Sev09] considered the expressibility of MDL and proved, that on singleton teams T, there is a translation from MDL to usual modal logic, while on arbitrary sets of teams there is no such translation. Sevenster also initiated a complexity-theoretic study of modal dependence logic by proving that the satisfiability problem for MDL is complete for the class NEXPTIME of all problems decidable nondeterministically in exponential time.

In this paper, we continue the work of Sevenster by presenting a more thorough study on complexity questions related to modal dependence logic. A line of research going back to Lewis [Lew79] and recently taken up in a number of papers [RW00, Hem01, BHSS06, MMTV08] has considered fragments of different propositional logics by restricting the propositional and temporal operators allowed in the language. The rationale behind this approach is that by systematically restricting the language, one might find a fragment with efficient algorithms but still high enough expressibility in order to be interesting for applications. This in turn might lead to better tools for model checking, verification, etc. On the other hand, it is worthwhile to identify the sources of hardness: What exactly makes satisfiability, model checking, or other problems so hard for certain languages?

We follow the same approach here. We consider all subsets of modal operators $\Box, \Diamond$ and propositional operators $\wedge, \vee, \bar{\cdot}$ (atomic negation), $\top, \bot$ (the Boolean constants true and false), i. e., we study exactly those operators considered by Väänänen [Vää08], and examine the satisfiability problem for MDL restricted to the fragment given by these operators. In each case we exactly determine the computational complexity in terms of completeness for a complexity class such as NEXPTIME, PSPACE, coNP, etc., or by showing that the satisfiability problem admits an efficient (polynomial-time) solution. We also extend the logical language of [Vää08] by adding classical disjunction (denoted here by $\varovee$) besides the dependence disjunction. Connective $\varovee$ was already considered by Sevenster (he denoted it by $\bullet$), but not from a complexity point of view. In this way, we obtain a complexity analysis of the satisfiability problem for MDL for all subsets of operators studied by Väänänen and Sevenster.

Our results are summarized in Table 1, where + denotes presence and − denotes absence of an operator, and $*$ states that the complexity does not depend on the operator. One of our main and technically most involved contributions addresses a fragment that has been called *Poor Man's Logic* in the literature on modal logic [Hem01], i. e., the language without disjunction $\vee$. We show that for dependence logic we still have full complexity (Theorem 3.5, first line of the table), i. e., we show that Poor Man's Dependence Logic is NEXPTIME-complete. If we also forbid negation, then the complexity drops down to $\Sigma_2^{\mathrm{p}} (= \mathrm{NP}^{\mathrm{NP}})$; i. e., Monotone Poor Man's Dependence Logic is Σ_2^{p}-complete (Theorem 3.4, but note that we need $\varovee$ here).

$\Box$	$\Diamond$	$\wedge$	$\vee$	$\bar{\cdot}$	$\top$	$\bot$	dep	$\varovee$	**Complexity**	**Reference**
+	+	+	$*$	+	$*$	$*$	+	$*$	NEXPTIME	Theorem 3.5
+	+	+	+	+	$*$	$*$	−	$*$	PSPACE	Corollary 3.3a
+	+	+	+	−	$*$	+	$*$	$*$	PSPACE	Corollary 3.3b
+	+	+	−	+	$*$	$*$	−	+	Σ_2^{p}	Theorem 3.4
+	+	+	−	−	$*$	+	$*$	+	Σ_2^{p}	Theorem 3.4
+	+	+	−	+	$*$	$*$	−	−	coNP	[Lad77], [DLN+92]
+	+	+	−	−	$*$	+	$*$	−	coNP	Corollary 3.3c
+	−	+	+	+	$*$	$*$	$*$	$*$	NP	Corollary 3.7a
−	+	+	+	+	$*$	$*$	$*$	$*$	NP	Corollary 3.7a
+	−	+	−	+	$*$	$*$	$*$	+	NP	Corollary 3.7a
−	+	+	−	+	$*$	$*$	$*$	+	NP	Corollary 3.7a
+	−	+	−	+	$*$	$*$	$*$	−	P	Corollary 3.7b
−	+	+	−	+	$*$	$*$	$*$	−	P	Corollary 3.7b
+	−	+	$*$	−	$*$	$*$	$*$	$*$	P	Corollary 3.7c
−	+	+	$*$	−	$*$	$*$	$*$	$*$	P	Corollary 3.7c
$*$	$*$	−	$*$	$*$	$*$	$*$	$*$	$*$	P	Corollary 3.7d
$*$	$*$	$*$	$*$	−	$*$	−	$*$	$*$	trivial	Corollary 3.3d
−	−	$*$	$*$	$*$	$*$	$*$	$*$	$*$	ordinary propositional logic ($\varovee \equiv \vee$, $\mathrm{dep}(\cdot;\cdot) \equiv \top$)	

+ : operator present − : operator absent $*$: complexity independent of operator

Table 1: Complete classification of complexity for fragments of MDL-SAT
All results are completeness results except for the P cases.

2. Modal Dependence Logic

We will only briefly introduce the syntax and semantics of modal dependence logic here. For a more profound overview consult Väänänen's introduction [Vää08] or Sevenster's analysis [Sev09] which includes a self-contained introduction to MDL.

2.1. Syntax

The formulas of *modal dependence logic* (MDL) are built from a set AP of *atomic propositions* and the MDL *operators* $\Box$, $\Diamond$, $\wedge$, $\vee$, $\overline{\cdot}$ (also denoted $\neg$), $\top$, $\bot$, dep and $\ovee$.

The set of MDL *formulas* is defined by the following grammar

$$\begin{aligned}\varphi \ ::= \ & \top \mid \bot \mid p \mid \neg p \mid \mathrm{dep}(p_1,\ldots,p_{n-1};p_n) \mid \neg\mathrm{dep}(p_1,\ldots,p_{n-1};p_n) \mid \\ & \varphi\wedge\varphi \mid \varphi\vee\varphi \mid \varphi\ovee\varphi \mid \Box\varphi \mid \Diamond\varphi,\end{aligned}$$

where $n \geq 1$.

All formulas in the first row will sometimes be denoted as *atomic* formulas and formulas of the form $\mathrm{dep}(p_1,\ldots,p_{n-1};p_n)$ as *dependence atoms*. We sometimes write ∇^k for $\underbrace{\nabla\ldots\nabla}_{k \text{ times}}$ (with $\nabla \in \{\Box,\Diamond\}$, $k \in \mathbb{N}$).

2.2. Semantics

A *frame* (or *Kripke structure*) is a tuple $W = (S,R,\pi)$ where S is a non-empty set of *worlds*, $R \subseteq S \times S$ is the *accessibility relation* and $\pi : S \to \mathcal{P}(AP)$ is the *labeling function*.

In contrast to usual modal logic, truth of a MDL formula is not defined with respect to a single world of a frame but with respect to a set of worlds, as already pointed out in the introduction. The *truth* of a MDL formula φ in an *evaluation set* T of worlds of a frame $W = (S,R,\pi)$ is denoted by $W,T \models \varphi$ and is defined as follows:

W,T	$\models$	$\top$		always holds
W,T	$\models$	$\bot$	iff	$T = \emptyset$
W,T	$\models$	p	iff	$p \in \pi(s)$ for all $s \in T$
W,T	$\models$	$\neg p$	iff	$p \notin \pi(s)$ for all $s \in T$
W,T	$\models$	$\mathrm{dep}(p_1,\ldots,p_{n-1};p_n)$	iff	for all $s_1,s_2 \in T$ with $\pi(s_1)\cap\{p_1,\ldots,p_{n-1}\} = \pi(s_2)\cap\{p_1,\ldots,p_{n-1}\}$: $p_n \in \pi(s_1)$ iff $p_n \in \pi(s_2)$
W,T	$\models$	$\neg\mathrm{dep}(p_1,\ldots,p_{n-1};p_n)$	iff	$T = \emptyset$
W,T	$\models$	$\varphi\wedge\psi$	iff	$W,T \models \varphi$ and $W,T \models \psi$
W,T	$\models$	$\varphi\vee\psi$	iff	there are sets T_1,T_2 with $T = T_1 \cup T_2$, $W,T_1 \models \varphi$ and $W,T_2 \models \psi$
W,T	$\models$	$\varphi\ovee\psi$	iff	$W,T \models \varphi$ or $W,T \models \psi$
W,T	$\models$	$\Box\varphi$	iff	$W, \{s' \mid \exists s \in T \text{ with } (s,s') \in R\} \models \varphi$
W,T	$\models$	$\Diamond\varphi$	iff	there is a set $T' \subseteq S$ such that $W,T' \models \varphi$ and for all $s \in T$ there is a $s' \in T'$ with $(s,s') \in R$

By $\vee$ we denote dependence disjunction instead of classical disjunction because the semantics of dependence disjunction is an extension of the semantics of usual modal logic disjunction and thus we preserve downward compatibility of our notation in this way. However, we still call the $\ovee$ operator "classical" because in a higher level context – where our sets of states are viewed as single objects themselves – it is indeed the usual disjunction, cf. [AV09]. Note that rationales for the seemingly rather strange definitions of the truth of $\varphi\vee\psi$ as well as $\neg\mathrm{dep}(p_1,\ldots,p_{n-1};p_n)$ were given by Väänänen [Vää08, Vää07].

For each $M \subseteq \{\Box,\Diamond,\wedge,\vee,\overline{\cdot},\top,\bot,\mathrm{dep},\ovee\}$ define the set of $\mathsf{MDL}(M)$ formulas to be the set of MDL formulas which are built from atomic propositions using only operators and constants from M.

We are interested in the parameterized decision problem **MDL-SAT(M)**:

Given: A $\mathsf{MDL}(M)$ formula φ.
Question: Is there a frame W and a non-empty set T of worlds in W such that $W, T \models \varphi$?

Note that, as Väänänen already pointed out [Vää08, Lemma 4.2.1], the semantics of MDL satisfies the *downward closure property*, i.e., if $W, T \models \varphi$, then $W, T' \models \varphi$ for all $T' \subseteq T$. Hence, to check satisfiability of a formula φ it is enough to check whether there is a frame W and a single world w in W such that $W, \{w\} \models \varphi$.

3. Complexity Results

To state the first lemma we need the following complexity operator. If $\mathcal{C}$ is an arbitrary complexity class then $\exists \cdot \mathcal{C}$ denotes the class of all sets A for which there is a set $B \in \mathcal{C}$ and a polynomial p such that for all x,

$$x \in A \text{ iff there is a } y \text{ with } |y| \leq p(|x|) \text{ and } \langle x, y \rangle \in B.$$

Note that for every class $\mathcal{C}$, $\exists \cdot \mathcal{C} \subseteq \mathrm{NP}^{\mathcal{C}}$. However, the converse does not hold in general. We will only need the following facts: $\exists \cdot \mathrm{coNP} = \Sigma_2^{\mathrm{p}}$, $\exists \cdot \mathrm{PSPACE} = \mathrm{PSPACE}$ and $\exists \cdot \mathrm{NEXPTIME} = \mathrm{NEXPTIME}$.

Our first lemma concerns sets of operators including classical disjunction.

Lemma 3.1. *Let M be a set of* MDL *operators. Then it holds:*

a) Every $\mathsf{MDL}(M \cup \{\varovee\})$ formula φ is equivalent to a formula $\varovee_{i=1}^{2^{|\varphi|}} \psi_i$ with $\psi_i \in \mathsf{MDL}(M)$ for all $i \in \{1, \ldots, 2^{|\varphi|}\}$.
b) If $\mathcal{C}$ is an arbitrary complexity class with $\mathrm{P} \subseteq \mathcal{C}$ and MDL-SAT$(M) \in \mathcal{C}$ *then* MDL-SAT$(M \cup \{\varovee\}) \in \exists \cdot \mathcal{C}$.

Proof. a) follows from the distributivity of $\varovee$ with all other operators. More specifically $\varphi \star (\psi \varovee \sigma) \equiv (\varphi \star \psi) \varovee (\varphi \star \sigma)$ for $\star \in \{\wedge, \vee\}$ and $\nabla(\varphi \varovee \psi) \equiv (\nabla\varphi) \varovee (\nabla\varphi)$ for $\nabla \in \{\Diamond, \Box\}$.[1] b) follows from a) with the observation that $\varovee_{i=1}^{2^{|\varphi|}} \psi_i$ is satisfiable if and only if there is an $i \in \{1, \ldots, 2^{|\varphi|}\}$ such that ψ_i is satisfiable. Note that given $i \in \{1, \ldots, 2^{|\varphi|}\}$ the formula ψ_i can be computed from the original formula φ in polynomial time by choosing (for all $j \in \{1, \ldots, |\varphi|\}$) from the jth subformula of the form $\psi \varovee \sigma$ the formula ψ if the jth bit of i is 0 and σ if it is 1. ∎

We need the following simple property of monotone MDL formulas.

Lemma 3.2. *Let M be a set of* MDL *operators with $\bar{\cdot} \notin M$. Then an arbitrary $\mathsf{MDL}(M)$ formula φ is satisfiable iff the formula generated from φ by replacing every dependence atom and every atomic proposition with the same atomic proposition t is satisfiable.*

Proof. If a frame W is a model for φ, so is the frame generated from W by setting all atomic propositions in all worlds to true. ∎

[1] Interestingly, but not of relevance for our work, $\varphi \varovee (\psi \vee \sigma) \not\equiv (\varphi \varovee \psi) \vee (\varphi \varovee \sigma)$.

We are now able to classify some cases that can be easily reduced to known results.

Corollary 3.3. *a) If* $\{\Box, \Diamond, \wedge, \vee, \bar{\cdot}\} \subseteq M \subseteq \{\Box, \Diamond, \wedge, \vee, \bar{\cdot}, \top, \bot, \ovee\}$ *then* MDL-SAT(M) *is* PSPACE*-complete.*

b) If $\{\Box, \Diamond, \wedge, \vee, \bot\} \subseteq M \subseteq \{\Box, \Diamond, \wedge, \vee, \top, \bot, \mathrm{dep}, \ovee\}$ *then* MDL-SAT(M) *is* PSPACE*-complete.*

c) If $\{\Box, \Diamond, \wedge, \bot\} \subseteq M \subseteq \{\Box, \Diamond, \wedge, \top, \bot, \mathrm{dep}\}$ *then* MDL-SAT(M) *is* coNP*-complete.*

d) If $M \subseteq \{\Box, \Diamond, \wedge, \vee, \top, \mathrm{dep}, \ovee\}$ *then every* MDL(M) *formula is satisfiable.*

Proof. a) follows immediately from Ladner's proof for the case of ordinary modal logic [Lad77], Lemma 3.1 and $\exists \cdot \mathrm{PSPACE} = \mathrm{PSPACE}$. The lower bound for b) was shown by Hemaspaandra [Hem01, Theorem 6.5] and the upper bound follows from a) together with Lemma 3.2. The lower bound for c) was shown by Donini et al. [DLN+92] and the upper bound follows from Ladner's algorithm [Lad77] together with Lemma 3.2. d) follows from Lemma 3.2 together with the fact that every MDL formula with t as the only atomic subformula is satisfied in the transitive singleton, i.e. the frame consisting of only one state which has itself as successor, in which t is true. ∎

3.1. Poor Man's Dependence Logic

We now turn to the Σ_2^{p}-complete cases. These include monotone poor man's logic, with and without dependence atoms.

Theorem 3.4. *If* $\{\Box, \Diamond, \wedge, \bar{\cdot}, \ovee\} \subseteq M \subseteq \{\Box, \Diamond, \wedge, \bar{\cdot}, \top, \bot, \ovee\}$ *or* $\{\Box, \Diamond, \wedge, \bot, \ovee\} \subseteq M \subseteq \{\Box, \Diamond, \wedge, \top, \bot, \mathrm{dep}, \ovee\}$ *then* MDL-SAT(M) *is* Σ_2^{p}*-complete.*

Proof. Proving the upper bound for the second case reduces to proving the upper bound for the first case by Lemma 3.2. For the first case it holds with Lemma 3.1 that MDL-SAT($\Box, \Diamond, \wedge, \bar{\cdot}, \top, \bot, \ovee$) $\in \exists \cdot \mathrm{coNP} = \Sigma_2^{\mathrm{p}}$ since MDL-SAT($\Box, \Diamond, \wedge, \bar{\cdot}, \top, \bot$) $\in$ coNP, which follows directly from Ladner's AP-algorithm for modal logic satisfiability [Lad77].

For the lower bound we consider the quantified constraint satisfaction problem $\mathrm{QCSP}_2(\mathrm{R}_{1/3})$ shown to be Π_2^{p}-complete by Bauland et al. [BBC+ar]. This problem can be reduced to the complement of MDL-SAT($\Box, \Diamond, \wedge, \bar{\cdot}/\bot, \ovee$) in polynomial time.

An instance of $\mathrm{QCSP}_2(\mathrm{R}_{1/3})$ consists of universally quantified Boolean variables $p_1, \ldots, p_k$, existentially quantified Boolean variables $p_{k+1}, \ldots, p_n$ and a set of clauses each consisting of exactly three of those variables. $\mathrm{QCSP}_2(\mathrm{R}_{1/3})$ is the set of all those instances for which for every truth assignment for $p_1, \ldots, p_k$ there is a truth assignment for $p_{k+1}, \ldots, p_n$ such that in each clause exactly one variable evaluates to true.[2]

For the reduction from $\mathrm{QCSP}_2(\mathrm{R}_{1/3})$ to the complement of MDL-SAT($\Box, \Diamond, \wedge, \bar{\cdot}/\bot, \ovee$) we extend a technique from the coNP-hardness proof for MDL-SAT($\Box, \Diamond, \wedge, \bot$) by Donini et al. [DLN+92, Theorem 3.3]. Let $p_1, \ldots, p_k$ be the universally quantified and $p_{k+1}, \ldots, p_n$ the existentially quantified variables of a $\mathrm{QCSP}_2(\mathrm{R}_{1/3})$ instance and let $C_1, \ldots, C_m$ be its clauses (we assume w.l.o.g. that each variable occurs in at least

[2]For our reduction it is necessary that in each clause the variables are pairwise different whereas in $\mathrm{QCSP}_2(\mathrm{R}_{1/3})$ this need not be the case. However, the Π_2^{p}-hardness proof can easily be adapted to account for this.

one clause). Then the corresponding $\mathsf{MDL}(\Box, \Diamond, \wedge, \bot, \ovee)$ formula is

$$
\begin{array}{rcllllll}
\varphi := & \bigwedge\limits_{i=1}^{k} & \Big(& \nabla_{i1}\dots\nabla_{im} & \nabla_{i1}\dots\nabla_{im} & \Box^{i-1}\Diamond\Box^{k-i} & p & \\
 & & \ovee & \Box^{m} & \Box^{m} & \Box^{i-1}\Diamond\Box^{k-i} & p\Big) & \\
\wedge & \bigwedge\limits_{i=k+1}^{n} & & \nabla_{i1}\dots\nabla_{im} & \nabla_{i1}\dots\nabla_{im} & \Box^{k} & p & \\
\wedge & & & \Box^{m} & \Box^{m} & \Box^{k} & \bot &
\end{array}
$$

where p is an arbitrary atomic proposition and $\nabla_{ij} := \begin{cases} \Diamond & \text{if } p_i \in C_j \\ \Box & \text{else} \end{cases}$.

For the corresponding $\mathsf{MDL}(\Box, \Diamond, \wedge, \overline{\cdot}, \ovee)$ formula replace every $\bot$ with $\neg p$.

To prove the correctness of our reduction we will need two claims.

Claim 1. For $r, s \geq 0$ a $\mathsf{MDL}(\Box, \Diamond, \wedge, \overline{\cdot}, \top, \bot)$ formula $\Diamond\varphi_1 \wedge \dots \wedge \Diamond\varphi_r \wedge \Box\psi_1 \wedge \dots \wedge \Box\psi_s$ is unsatisfiable iff there is an $i \in \{1, \dots, r\}$ such that $\varphi_i \wedge \psi_1 \wedge \dots \wedge \psi_s$ is unsatisfiable.

Proof of Claim 1. "$\Leftarrow$": If $\varphi_i \wedge \psi_1 \wedge \dots \wedge \psi_s$ is unsatisfiable, so is $\Diamond\varphi_i \wedge \Box\psi_1 \wedge \dots \wedge \Box\psi_s$ and even more $\Diamond\varphi_1 \wedge \dots \wedge \Diamond\varphi_r \wedge \Box\psi_1 \wedge \dots \wedge \Box\psi_s$.

"$\Rightarrow$: Suppose that $\varphi_i \wedge \psi_1 \wedge \dots \wedge \psi_s$ is satisfiable for all $i \in \{1, \dots, r\}$. Then $\Diamond\varphi_1 \wedge \dots \wedge \Diamond\varphi_r \wedge \Box\psi_1 \wedge \dots \wedge \Box\psi_s$ is satisfiable in a frame that consists of a root state and for each $i \in \{1, \dots, r\}$ a separate branch, reachable from the root in one step, which satisfies $\varphi_i \wedge \psi_1 \wedge \dots \wedge \psi_s$. <<

Note that $\Diamond\varphi_1 \wedge \dots \wedge \Diamond\varphi_r \wedge \Box\psi_1 \wedge \dots \wedge \Box\psi_s$ is always satisfiable if $r = 0$.

Definition. Let $v : \{p_1, \dots, p_k\} \to \{0, 1\}$ be a valuation of $\{p_1, \dots, p_k\}$. Then φ_v denotes the $\mathsf{MDL}(\Box, \Diamond, \wedge, \overline{\cdot}/\bot)$ formula

$$
\begin{array}{rclllll}
 & \bigwedge\limits_{\substack{i\in\{1,\dots,k\},\\ v(p_i)=1}} & \nabla_{i1}\dots\nabla_{im} & \nabla_{i1}\dots\nabla_{im} & \Box^{i-1}\Diamond\Box^{k-i} & p \\
\wedge & \bigwedge\limits_{\substack{i\in\{1,\dots,k\},\\ v(p_i)=0}} & \Box^{m} & \Box^{m} & \Box^{i-1}\Diamond\Box^{k-i} & p \\
\wedge & \bigwedge\limits_{i=k+1}^{n} & \nabla_{i1}\dots\nabla_{im} & \nabla_{i1}\dots\nabla_{im} & \Box^{k} & p \\
\wedge & & \Box^{m} & \Box^{m} & \Box^{k} & \neg p \,/\, \bot
\end{array}
$$

Claim 2. Let $v : \{p_1, \dots, p_k\} \to \{0, 1\}$ be a valuation. Then φ_v is unsatisfiable iff v can be continued to a valuation $v' : \{p_1, \dots, p_n\} \to \{0, 1\}$ such that in each of the clauses $\{C_1, \dots, C_m\}$ exactly one variable evaluates to true under v'.

Proof of Claim 2. By iterated use of Claim 1, φ_v is unsatisfiable iff there are $i_1, \dots, i_{2m}$ with

$$
\begin{array}{rl}
i_j \in & \big\{i \in \{1, \dots, n\} \mid \nabla_{ij'} = \Diamond\big\} \setminus \big\{i \in \{1, \dots, k\} \mid v(p_i) = 0\big\} \\
= & \big\{i \in \{1, \dots, n\} \mid p_i \in C_{j'}\big\} \setminus \big\{i \in \{1, \dots, k\} \mid v(p_i) = 0\big\},
\end{array}
$$

where $j' := \begin{cases} j & \text{if } j \leq m \\ j-m & \text{else} \end{cases}$, such that

$$\begin{aligned}
\varphi_v(i_1,\ldots,i_{2m}) := & \bigwedge_{\substack{i\in\{1,\ldots,k\},\\ i\in\{i_1,\ldots,i_{2m}\},\\ v(p_i)=1}} \Box^{i-1}\Diamond\Box^{k-i} p \\
\wedge & \bigwedge_{\substack{i\in\{1,\ldots,k\},\\ v(p_i)=0}} \Box^{i-1}\Diamond\Box^{k-i} p \\
\wedge & \bigwedge_{\substack{i\in\{k+1,\ldots,n\},\\ i\in\{i_1,\ldots,i_{2m}\}}} \Box^{k} p \\
\wedge & \Box^{k} \neg p \,/\, \bot
\end{aligned}$$

is unsatisfiable (i) and such that there are no $a, b \in \{1,\ldots,2m\}$ with $a < b$, $\nabla_{i_b a'} = \nabla_{i_b b'} = \Diamond$ (this is the case iff $p_{i_b} \in C_{a'}$ and $p_{i_b} \in C_{b'}$) and $i_a \neq i_b$ (ii). The latter condition is already implied by Claim 1 as it simply ensures that no subformula is selected after it has already been discarded in an earlier step. Note that $\varphi_v(i_1,\ldots,i_{2m})$ is unsatisfiable iff for all $i \in \{1,\ldots,k\}$: $v(p_i) = 1$ and $i \in \{i_1,\ldots,i_{2m}\}$ or $v(p_i) = 0$ (and $i \notin \{i_1,\ldots,i_{2m}\}$) ($i'$).

We are now able to prove the claim.

"$\Leftarrow$": For $j = 1,\ldots,2m$ choose $i_j \in \{1,\ldots,n\}$ such that $p_{i_j} \in C_{j'}$ and $v'(p_{i_j}) = 1$. By assumption, all i_j exist and are uniquely determined. Hence, for all $i \in \{1,\ldots,k\}$ we have that $v(p_i) = 0$ (and then $i \notin \{i_1,\ldots,i_{2m}\}$) or $v(p_i) = 1$ and there is a j such that $i_j = i$ (because each variable occurs in at least one clause). Therefore condition (i') is satisfied. Now suppose there are $a < b$ that violate condition (ii). By definition of i_b it holds that $p_{i_b} \in C_{b'}$ and $v'(p_{i_b}) = 1$. Analogously, $p_{i_a} \in C_{a'}$ and $v'(p_{i_a}) = 1$. By the supposition $p_{i_b} \in C_{a'}$ and $p_{i_a} \neq p_{i_b}$. But since $v'(p_{i_a}) = v'(p_{i_b}) = 1$, that is a contradiction to the fact that in clause $C_{a'}$ only one variable evaluates to true.

"$\Rightarrow$": If φ_v is unsatisfiable, there are $i_1,\ldots,i_{2m}$ such that (i') and (ii) hold. Let the valuation $v' : \{p_1,\ldots,p_n\} \to \{0,1\}$ be defined by

$$v'(p_i) := \begin{cases} 1 & \text{if } i \in \{i_1,\ldots,i_{2m}\} \\ 0 & \text{else} \end{cases} .$$

Note that v' is a continuation of v because (i') holds.

We will now prove that in each of the clauses $C_1,\ldots,C_m$ exactly one variable evaluates to true under v'. Therefore let $j \in \{1,\ldots,m\}$ be arbitrarily chosen.

By choice of i_j it holds that $p_{i_j} \in C_j$. It follows by definition of v' that $v'(p_{i_j}) = 1$. Hence, there is at least one variable in C_j that evaluates to true.

Now suppose that besides p_{i_j} another variable in C_j evaluates to true. Then by definition of v' it follows that there is a $\ell \in \{1,\ldots,2m\}$, $\ell \neq j$, such that this other variable is p_{i_ℓ}. We now consider two cases.

Case $j < \ell$: This is a contradiction to (ii) since, by definition of ℓ, p_{i_ℓ} is in $C_{j'}$ as well as, by definition of i_ℓ, in $C_{\ell'}$ and $i_j \neq i_\ell$.

Case $\ell < j$: Since $j \in \{1,\ldots,m\}$ it follows that $\ell \leq m$. Since $C_{\ell'} = C_{(\ell+m)'}$ it holds that $p_{i_{\ell+m}} \in C_{\ell'}$ and $p_{i_{\ell+m}} \in C_{(\ell+m)'}$. Furthermore $\ell < \ell + m$ and thus, by condition (ii), it must hold that $i_\ell = i_{\ell+m}$. Therefore $p_{i_{\ell+m}} \in C_j$ and $v'(p_{i_{\ell+m}}) = 1$. Because $j < \ell + m$ this is a contradiction to condition (ii) as in the first case. <<

The correctness of the reduction now follows with the observation that φ is equivalent to

$\bigovee_{v:\{p_1,\dots,p_k\}\to\{0,1\}} \varphi_v$ and that φ is unsatisfiable iff φ_v is unsatisfiable for all valuations $v : \{p_1, \dots, p_k\} \to \{0, 1\}$.

The $\mathrm{QCSP}_2(\mathrm{R}_{1/3})$ instance is true iff every valuation $v : \{p_1, \dots, p_k\} \to \{0, 1\}$ can be continued to a valuation $v' : \{p_1, \dots, p_n\} \to \{0, 1\}$ such that in each of the clauses $\{C_1, \dots, C_m\}$ exactly one variable evaluates to true under v' iff, by Claim 2, φ_v is unsatisfiable for all $v : \{p_1, \dots, p_k\} \to \{0, 1\}$ iff, by the above observation, φ is unsatisfiable. ∎

Next we turn to (non-monotone) poor man's logic.

Theorem 3.5. *If* $\{\Box, \Diamond, \wedge, \overline{\cdot}, \mathrm{dep}\} \subseteq M$ *then* MDL-SAT(M) *is* NEXPTIME-*complete.*

Proof. Sevenster showed that the problem is in NEXPTIME in the case of $\bigovee \notin M$ [Sev09, Lemma 14]. Together with Lemma 3.1 and the fact that $\exists \cdot \mathrm{NEXPTIME} = \mathrm{NEXPTIME}$ the upper bound applies.

For the lower bound we reduce 3CNF-DQBF, which was shown to be NEXPTIME-hard by Peterson et al. [PRA01, Lemma 5.2.2][3], to our problem.

An instance of 3CNF-DQBF consists of universally quantified Boolean variables $p_1, \dots, p_k$, existentially quantified Boolean variables $p_{k+1}, \dots, p_n$, dependence constraints $P_{k+1}, \dots, P_n \subseteq \{p_1, \dots, p_k\}$ and a set of clauses each consisting of three (not necessarily distinct) literals. Here, P_i intuitively states that the value of p_i only depends on the values of the variables in P_i. Now, 3CNF-DQBF is the set of all those instances for which there is a collection of functions $f_{k+1}, \dots, f_n$ with $f_i : \{0,1\}^{P_i} \to \{0, 1\}$ such that for every valuation $v : \{p_1, \dots, p_k\} \to \{0, 1\}$ there is at least one literal in each clause that evaluates to true under the valuation $v' : \{p_1, \dots, p_n\} \to \{0, 1\}$ defined by

$$v'(p_i) := \begin{cases} v(p_i) & \text{if } i \in \{1, \dots, k\} \\ f_i(v \upharpoonright P_i) & \text{if } i \in \{k+1, \dots, n\} \end{cases} .$$

The functions $f_{k+1}, \dots, f_n$ act as restricted existential quantifiers, i.e., for an $i \in \{k+1, \dots, n\}$ the variable p_i can be assumed to be existentially quantified dependent on all universally quantified variables in P_i (and, more importantly, independent of all universally quantified variables not in P_i). Dependencies are thus explicitly specified through the dependence constraints and can contain – but are not limited to – the traditional sequential dependencies, e.g. the quantifier sequence $\forall p_1 \exists p_2 \forall p_3 \exists p_4$ can be modeled by the dependence constraints $P_2 = \{p_1\}$ and $P_4 = \{p_1, p_3\}$.

For the reduction from 3CNF-DQBF to MDL-SAT$(\Box, \Diamond, \wedge, \overline{\cdot}, \mathrm{dep})$ we use an idea from the PSPACE-hardness proof of MDL-SAT$(\Box, \Diamond, \wedge, \overline{\cdot})$ over a restricted frame class by Hemaspaandra [Hem01, Theorem 4.2]. Let $p_1, \dots, p_k$ be the universally quantified and $p_{k+1}, \dots, p_n$ the existentially quantified variables of a 3CNF-DQBF instance φ and let $P_{k+1}, \dots, P_n$ be its dependence constraints and $\{l_{11}, l_{12}, l_{13}\}, \dots, \{l_{m1}, l_{m2}, l_{m3}\}$ its

[3]Peterson et al. showed NEXPTIME-hardness for DQBF without the restriction that the formulae must be in 3CNF. However, the restriction does not lower the complexity since every propositional formula is satisfiability-equivalent to a formula in 3CNF whose size is bounded by a polynomial in the size of the original formula.

clauses. Then the corresponding $\mathsf{MDL}(\Box, \Diamond, \wedge, \overline{\cdot}, \mathrm{dep})$ formula is

$$\begin{aligned} g(\varphi) := \quad & \bigwedge_{i=1}^{n} \Box^{i-1}(\Diamond\Box^{n-i}p_i \wedge \Diamond\Box^{n-i}\overline{p_i}) && (i) \\ \wedge \quad & \bigwedge_{i=1}^{m} \Diamond^{n}(\overline{l_{i1}} \wedge \overline{l_{i2}} \wedge \overline{l_{i3}} \wedge f_i) && (ii) \\ \wedge \quad & \bigwedge_{i=1}^{m} \Box^{n}\mathrm{dep}(l'_{i1}, l'_{i2}, l'_{i3}; f_i) && (iii) \\ \wedge \quad & \Box^{k}\Diamond^{n-k}\big(\overline{f_1} \wedge \cdots \wedge \overline{f_m} \ \wedge \ \textstyle\bigwedge_{i=k+1}^{n} \mathrm{dep}(P_i; p_i)\big) && (iv) \end{aligned}$$

where $p_1, \ldots, p_n, f_1, \ldots, f_m$ are atomic propositions and $l'_{ij} := \begin{cases} p & \text{if } l_{ij} = p \\ p & \text{if } l_{ij} = \overline{p} \end{cases}$.

Now if φ is valid, consider the frame which consists of a complete binary tree with n levels (not counting the root) and where each of the 2^n possible labelings of the atomic propositions $p_1, \ldots, p_n$ occurs in exactly one leaf. Additionally, for each $i \in \{1, \ldots, m\}$ f_i is labeled in exactly those leaves in which $l_{i1} \vee l_{i2} \vee l_{i3}$ is false. This frame obviously satisfies (i), (ii) and (iii). And since the modalities in (iv) model the quantors of φ, $\overline{f_i}$ is true exactly in the leaves in which $l_{i1} \vee l_{i2} \vee l_{i3}$ is true and the dep atoms in (iv) model the dependence constraints of φ, (iv) is also true and therefore $g(\varphi)$ is satisfied in the root of the tree.

As an example see Fig. 1 for a frame satisfying $g(\varphi)$ if the first clause in φ is $\{\overline{p_1}, p_n\}$.

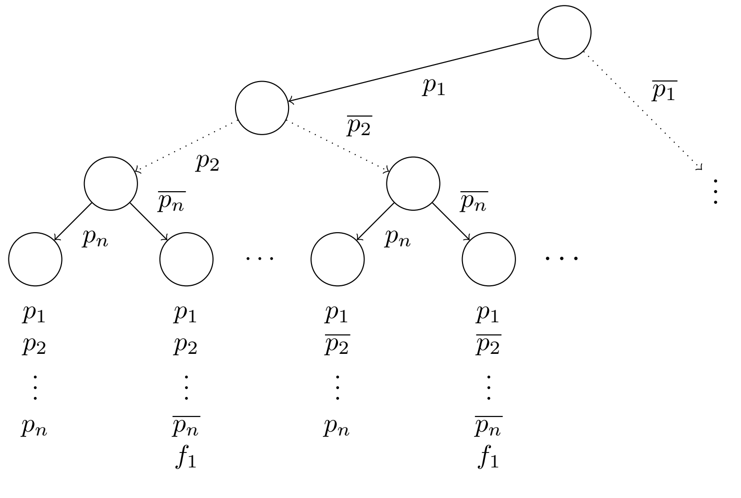

Figure 1: Frame satisfying $g(\varphi)$

If, on the other hand, $g(\varphi)$ is satisfiable, let W be a frame and t a world in W such that $W, \{t\} \models g(\varphi)$. Now (i) enforces W to contain a complete binary tree T with root t such that each labeling of $p_1, \ldots, p_n$ occurs in a leaf of T.

We can further assume w.l.o.g. that W itself is a tree since in MDL different worlds with identical proposition labelings are indistinguishable and therefore every frame can simply be unwinded to become a tree. Since the modal depth of $g(\varphi)$ is n we can assume that the depth of W is at most n. And since (i) enforces that every path in W from t to a leaf has a length of at least n, all leaves of W lie at levels greater or equal to n. Altogether we can assume that W is a tree, that all its leaves lie at level n and that it has the same root as T. The only difference is that the degree of W may be greater than that of T.

But we can nonetheless assume that up to level k the degree of W is 2 $(*)$. This is the case because if any world up to level $k-1$ had more successors than the two lying in T, the additional successors could be omitted and (i), (ii), (iii) and (iv) would still be fulfilled. For (i), (ii) and (iii) this is clear and for (iv) it holds because (iv) begins with $\Box^k$.

We will now show that, although T may be a proper subframe of W, T is already sufficient to fulfill $g(\varphi)$. From this the validity of φ will follow immediately.

Claim. $T, \{t\} \models g(\varphi)$.

Proof of Claim. We consider sets of leaves of W that satisfy $\overline{f_1} \wedge \cdots \wedge \overline{f_m} \wedge \bigwedge_{i=k+1}^{n} \text{dep}(P_i; p_i)$ and that can be reached from the set $\{t\}$ by the modality sequence $\Box^k \Diamond^{n-k}$. Let S be such a set and let S be chosen so that there is no other such set that contains less worlds outside of T than S does. Assume there is a $s \in S$ that does not lie in T.

Let $i \in \{1, \ldots, m\}$ and let s' be the leaf in T that agrees with s on the labeling of $p_1, \ldots, p_n$. Then, with $W, \{s\} \models \overline{f_i}$ and (iii), it follows that $W, \{s'\} \models \overline{f_i}$.

Let $S' := (S \setminus \{s\}) \cup \{s'\}$. Then it follows by the previous paragraph that $W, S' \models \overline{f_1} \wedge \cdots \wedge \overline{f_m}$. Since $W, S \models \bigwedge_{i=k+1}^{n} \text{dep}(P_i; p_i)$ and s' agrees with s on the propositions $p_1, \ldots, p_n$ it follows that $W, S' \models \bigwedge_{i=k+1}^{n} \text{dep}(P_i; p_i)$. Hence, S' satisfies $\overline{f_1} \wedge \cdots \wedge \overline{f_m} \wedge \bigwedge_{i=k+1}^{n} \text{dep}(P_i; p_i)$ and as it only differs from S by replacing s with s' it can be reached from $\{t\}$ by $\Box^k \Diamond^{n-k}$ because s and s' agree on $p_1, \ldots, p_k$ and, by $(*)$, W does not differ from T up to level k. But this is a contradiction to the assumption since S' contains one world less than S outside of T. Thus, there is no $s \in S$ that does not lie in T and therefore (iv) is fulfilled in T. Since (i), (ii) and (iii) are obviously also fulfilled in T, it follows that $T, \{t\} \models g(\varphi)$. $<<$

(ii) ensures that for all $i \in \{1, \ldots, m\}$ there is a leaf in W in which $\neg(l_{i1} \vee l_{i2} \vee l_{i3}) \wedge f_i$ is true. This leaf can lie outside of T. However, (iii) ensures that all leaves that agree on the labeling of l_{i1}, l_{i2} and l_{i3} also agree on the labeling of f_i. And since there is a leaf where $\neg(l_{i1} \vee l_{i2} \vee l_{i3}) \wedge f_i$ is true, it follows that in all leaves, in which $\neg(l_{i1} \vee l_{i2} \vee l_{i3})$ is true, f_i is true. Conversely, if $\overline{f_i}$ is true in an arbitrary leaf of W then so is $l_{i1} \vee l_{i2} \vee l_{i3}$ $(**)$.

The modality sequence $\Box^k \Diamond^{n-k}$ models the quantors of φ and $\bigwedge_{i=k+1}^{n} \text{dep}(P_i; p_i)$ models its dependence constraints. And so there is a bijective correspondence between sets of worlds reachable in T by $\Box^k \Diamond^{n-k}$ from $\{t\}$ and that satisfy $\bigwedge_{i=k+1}^{n} \text{dep}(P_i; p_i)$ on the one hand and truth assignments to $p_1, \ldots, p_n$ generated by the quantors of φ and satisfying its dependence constraints on the other hand. Additionally, by $(**)$ follows that $\overline{f_1} \wedge \cdots \wedge \overline{f_m}$ implies $\bigwedge_{i=1}^{m} (l_{i1} \vee l_{i2} \vee l_{i3})$ and since $T, \{t\} \models g(\varphi)$, φ is valid. ■

3.2. Cases with Only One Modality

We finally examine formulas with only one modality.

Theorem 3.6. *Let* $M \subseteq \{\Box, \Diamond, \wedge, \vee, \bar{\cdot}, \top, \bot, \ovee\}$ *with* $\Box \notin M$ *or* $\Diamond \notin M$. *Then the following hold:*

a) MDL-SAT$(M \cup \{\text{dep}\}) \leq_m^p$ MDL-SAT$(M \cup \{\top, \bot\})$*, i.e., adding the* dep *operator does not increase the complexity if we only have one modality.*

b) For every $\mathsf{MDL}(M \cup \{\text{dep}\})$ *formula* φ *it holds that* $\varovee$ *is equivalent to* $\vee$*, i.e.,* φ *is equivalent to every formula that is generated from* φ *by replacing some or all occurrences of* $\varovee$ *by* $\vee$ *and vice versa.*

Proof. Every negation $\neg$dep of a dependence atom is by definition always equivalent to $\bot$ and can thus be replaced by the latter. For positive dep atoms and the $\varovee$ operator we consider two cases.

Case $\Diamond \notin M$. If an arbitrary $\mathsf{MDL}(\Box, \wedge, \vee, \bar{\cdot}, \top, \bot, \text{dep}, \varovee)$ formula φ is satisfiable then it is so in an intransitive singleton frame, i.e. a frame that only contains one world which does not have a successor, because there every subformula that begins with a $\Box$ is automatically satisfied. In a singleton frame all dep atoms obviously hold and $\varovee$ is equivalent to $\vee$. Therefore the (un-)satisfiability of φ is preserved when substituting every dep atom in φ with $\top$ and every $\varovee$ with $\vee$ (or vice versa).

Case $\Box \notin M$. If an arbitrary $\mathsf{MDL}(\Diamond, \wedge, \vee, \bar{\cdot}, \top, \bot, \text{dep}, \varovee)$ formula φ is satisfiable then, by the downward closure property, there is a frame W with a world s such that $W, \{s\} \models \varphi$. Since there is no $\Box$ in φ, every subformula of φ is also evaluated in a singleton set (because a $\Diamond$ can never increase the cardinality of the evaluation set). And as in the former case we can replace every dep atom with $\top$ and every $\varovee$ with $\vee$ (or vice versa). ∎

Thus we obtain the following consequences – note that with the preceding results this takes care of all cases in Table 1.

Corollary 3.7. *a) If* $\{\wedge, \bar{\cdot}\} \subseteq M \subseteq \{\Box, \Diamond, \wedge, \vee, \bar{\cdot}, \top, \bot, \text{dep}, \varovee\}$*,* $M \cap \{\vee, \varovee\} \neq \emptyset$ *and* $|M \cap \{\Box, \Diamond\}| = 1$ *then* MDL-SAT(M) *is* NP*-complete.*

b) If $\{\wedge, \bar{\cdot}\} \subseteq M \subseteq \{\Box, \Diamond, \wedge, \bar{\cdot}, \top, \bot, \text{dep}\}$ *and* $|M \cap \{\Box, \Diamond\}| = 1$ *then* MDL-SAT$(M) \in$ P.

c) If $\{\wedge\} \subseteq M \subseteq \{\Box, \Diamond, \wedge, \vee, \top, \bot, \text{dep}, \varovee\}$ *and* $|M \cap \{\Box, \Diamond\}| = 1$ *then* MDL-SAT$(M) \in$ P.

d) If $\wedge \notin M$ *then* MDL-SAT$(M) \in$ P*.*

Proof. a) follows from [Hem01, Theorem 6.2(2)] and Theorem 3.6a,b. b) follows from [Hem01, Theorem 6.4(c,d)] and Theorem 3.6a. c) follows from [Hem01, Theorem 6.4(e,f)] and Theorem 3.6a,b.

For d) the proof of [Hem01, Theorem 6.4(b)] can be adapted as follows. Let φ be an arbitrary $\mathsf{MDL}(M)$ formula. By the same argument as in the proof of Theorem 3.6b we can replace all top-level (i.e. not lying inside a modality) occurrences of $\varovee$ in φ with $\vee$ to get the equivalent formula φ'. φ' is of the form $\Box\psi_1 \vee \cdots \vee \Box\psi_k \vee \Diamond\sigma_1 \vee \cdots \vee \Diamond\sigma_m \vee a_1 \vee \cdots \vee a_s$ where every ψ_i and σ_i is a $\mathsf{MDL}(M)$ formula and every a_i is an atomic formula. If $k > 0$ or any a_i is a literal, $\top$ or a dependence atom then φ' is satisfiable. Otherwise it is satisfiable iff one of the σ_i is satisfiable and this can be checked recursively in polynomial time. ∎

4. Conclusion

In this paper we completely classified the complexity of the satisfiability problem for modal dependence logic for all fragments of the language defined by restricting the modal and propositional operators to a subset of those considered by Väänänen and Sevenster. Interestingly, our results show a dichotomy for the dep operator; either the complexity jumps to NEXPTIME-completeness when introducing dep or it does not increase at all – and in the latter case the dep operator does not increase the expressiveness of the logic.

In a number of precursor papers, e. g., [Lew79] on propositional logic or [BHSS06] on modal logic, not only subsets of the classical operators $\{\Box, \Diamond, \wedge, \vee, \bar{\cdot}\}$ were considered but also propositional connectives given by arbitrary Boolean functions. The main result of Lewis, e. g., can be succinctly summarized as follows: Propositional satisfiability is NP-complete if and only if in the input formulas the connective $\varphi \wedge \neg\psi$ is allowed (or can be "implemented" with the allowed connectives).

We consider it interesting to initiate such a more general study for modal dependence logic and determine the computational complexity of satisfiability if the allowed connectives are taken from a fixed class in Post's lattice. Contrary to propositional or modal logic, however, the semantics of such generalized formulas is not clear a priori – for instance, how should exclusive-or be defined in dependence logic? Even for simple implication, there seem to be several reasonable definitions, cf. [AV09].

A further possibly interesting restriction of dependence logic might be to restrict the type of functional dependency. Right now, dependence just means that there is some function whatsoever that determines the value of a variable from the given values of certain other variables. Also here it might be interesting to restrict the function to be taken from a fixed class in Post's lattice, e. g., to be monotone or self-dual.

Related is the more general problem of finding interesting fragments of modal dependence logic where adding the dep operator does not let the complexity of satisfiability testing jump up to NEXPTIME but still increases the expressiveness of the logic.

Finally, it seems natural to investigate the possibility of enriching classical temporal logics as LTL, CTL or CTL* with dependence as (at least some of them) they are extensions of classical modal logic. The questions here are of the same kind as for MDL: expressivity, complexity, fragments, etc.

References

[AV09] Samson Abramsky and Jouko Väänänen. From IF to BI. *Synthese*, 167(2):207–230, 2009.

[BBC+ar] Michael Bauland, Elmar Böhler, Nadia Creignou, Steffen Reith, Henning Schnoor, and Heribert Vollmer. The complexity of problems for quantified constraints. *Theory of Computing Systems*, To appear.

[BHSS06] Michael Bauland, Edith Hemaspaandra, Henning Schnoor, and Ilka Schnoor. Generalized modal satisfiability. In *STACS*, pages 500–511, 2006. Revised version: [HSS08].

[DLN+92] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Bernhard Hollunder, Werner Nutt, and Alberto Marchetti-Spaccamela. The complexity of existential quantification in concept languages. *Artif. Intell.*, 53(2-3):309–327, 1992.

[Hem01] Edith Hemaspaandra. The complexity of poor man's logic. *Journal of Logic and Computation*, 11(4):609–622, 2001. Corrected version: [Hem05].

[Hem05] Edith Hemaspaandra. The complexity of poor man's logic. *CoRR*, cs.LO/9911014v2, 2005.

[Hen61] L. Henkin. Some remarks on infinitely long formulas. In *Infinitistic Methods*, Proceedings Symposium Foundations of Mathematics, pages 167–183, Warsaw, 1961. Pergamon.

[HS89] J. Hintikka and G. Sandu. Informational independence as a semantical phenomenon. In J. E. Fenstad, I. T. Frolov, and R. Hilpinen, editors, *Logic, Methodology and Philosophy of Science*, volume 8, pages 571–589. Elsevier, Amsterdam, 1989.

[HSS08] Edith Hemaspaandra, Henning Schnoor, and Ilka Schnoor. Generalized modal satisability. *CoRR*, abs/0804.2729, 2008.

[Lad77] Richard E. Ladner. The computational complexity of provability in systems of modal propositional logic. *Siam Journal on Computing*, 6(3):467–480, 1977.

[Lew79] H. Lewis. Satisfiability problems for propositional calculi. *Mathematical Systems Theory*, 13:45–53, 1979.

[MMTV08] Arne Meier, Martin Mundhenk, Michael Thomas, and Heribert Vollmer. The complexity of satisfiability for fragments of CTL and CTL*. *Electronic Notes in Theoretical Computer Science*, 223:201 – 213, 2008. Proceedings of the Second Workshop on Reachability Problems in Computational Models (RP 2008).

[PRA01] G. Peterson, J. Reif, and S. Azhar. Lower bounds for multiplayer noncooperative games of incomplete information. *Computers & Mathematics with Applications*, 41(7-8):957 – 992, 2001.

[RW00] Steffen Reith and Klaus W. Wagner. The complexity of problems defined by boolean circuits. In *Proceedings International Conference Mathematical Foundation of Informatics, (MFI99)*, pages 25–28. World Science Publishing, 2000.

[Sev09] Merlijn Sevenster. Model-theoretic and computational properties of modal dependence logic. *Journal of Logic and Computation*, 19(6):1157–1173, 2009.

[Vää07] Jouko Väänänen. *Dependence logic: A new approach to independence friendly logic.* Number 70 in London Mathematical Society student texts. Cambridge University Press, 2007.

[Vää08] Jouko Väänänen. Modal dependence logic. In Krzysztof R. Apt and Robert van Rooij, editors, *New Perspectives on Games and Interaction*, volume 4 of *Texts in Logic and Games*, pages 237–254. Amsterdam University Press, 2008.

Quantifier Independence and Downward Monotonicity

Denis Paperno paperno@ucla.edu

University of California, Los Angeles

1 Introduction

Game theoretical semantics interprets formulas of logical languages as games with two players. Each player behaves according to one of the two roles, Verifier and Falsifier (sometimes shortened as V and F). The roles can be traded at certain points in the game (which is assumed to be the game-theoretical denotation of negation), so that the Verifier becomes a Falsifier and vice versa. The player who acts as the Verifier at the beginning of a game is called Eloise, and the one who starts as the Falsifier is named Abelard. Truth is defined in terms of existence of a winning strategy: a formula is true iff Eloise has a winning strategy in the game it defines, and false if Abelard has a winning strategy.

Semantic games defined by formulas of first order predicate logic are games with perfect information, i.e. moves by players are totally ordered and at each move a player has complete information about every preceding move; in games of perfect information one of the players always has a winning strategy, so closed formulas are always true or false in a model. There are different approaches to incorporate imperfect information into semantic games. For example, the language of Independence Friendly Logic directly incorporates the requirement to 'forget' certain previous moves at a particular point in the game. Another approach, offering less flexibility in quantifier independence, is based on the natural metaphor of two games running in parallel.

Let us adopt the logical syntax proposed by Samson Abramsky, who pursues the latter option. Abramsky defines the syntax for n-player multi-agent logic as follows:

$\phi ::= 1 \mid A \mid Q_\alpha x \mid \hat{\pi}\phi \mid \phi \oplus_\alpha \psi \mid \phi \cdot \psi \mid \phi \,||\, \psi$

where A ranges over literals, α ranges over agents, so that Q_α are quantifiers and $\oplus_\alpha$ are connectives indexed by agents: the agent α makes a move at Q_α and $\oplus_\alpha$; $\hat{\pi}$ ranges over permutations of the agent set (correspond to role exchange in the game), and x ranges over variables. For the classic 2-player games with players $\{V, F\}$ one can identify $\oplus_V = \vee$, $\oplus_F = \wedge$, $Q_V = \exists$, $Q_F = \forall$, and the only non-trivial permutation is (V, F) denoted by the negation symbol $\sim$, which exchanges the roles of the Verifier and the Falsifier.

The crucial feature of this language is that it allows for parallel composition of two games $\phi \mid\mid \psi$, which can be thought of as encoding fully parallel processing (Hintikka and Sandu 1995).

For the purposes of 2-player games the syntax can be defined as

$\phi ::= 1 \mid A \mid Qx \mid \sim \phi \mid \phi \vee \psi \mid \phi \wedge \psi \mid \phi \cdot \psi \mid \phi \mid\mid \psi$

where Q ranges over quantifiers $\exists, \forall$.

2 Towards a Natural Language Application: Branching Quantification

This language just described is capable of expressing quantifier branching, including the Henkin quantifier:

(1) $[\forall x \cdot \exists y \mid\mid \forall z \cdot \exists w] \cdot R(x, y, z, w)$,

as expressed by the English sentence

(2) *Some relative of every townsman and some relative of every villager hate each other*, where $R(x, y, z, w)$ =‘if x is a townsman and z is a villager then y is a relative of x and w is a relative of z and y and w hate each other’.

The Henkin quantifier is perhaps the best known example of branching quantification in natural language, and also the simplest example of *essential* branching quantification. But there are much less elaborate examples in natural language that can’t be treated compositionally as simple iteration of quantifiers involved, and are rather interpreted as involving quantifier independence (cf. Barwise 1979):

(3) a. *No man and no woman date each other.* (synonymous to *No man dates any woman*)

b. *Two men and three women date each other.*

c. *Few men and few women are dating each other.*

d. *Not every man and not every woman know each other.* (synonymous to *It's not the case that every man and every woman know each other.*)

As these examples show, both upward monotone and downward monotone quantifiers can be scopally independent, and it is desirable to have a representation of them in a logic that allows for quantifier independence. Abramsky's logical language comes close to this; for instance, it is tempting to represent the semantics of the last English sentence as $[\sim \forall x \, || \sim \forall y] \cdot H(x, y)$ where H stands for the binary relation of mutual hate. However, the semantics that Abramsky proposes for the formal language does not match the meaning of the English sentence in consideration. In fact, since $\sim$ in game theoretical semantics stands for role exchange, $\sim \forall x \equiv \exists x$, and $[\sim \forall x \, || \sim \forall y] \cdot K(x, y) \equiv [\exists x \, || \, \exists y] \cdot K(x, y)$. But of course 3d in English is not made true by the mere fact that some man and some woman know each other.

The problem here lies in the fact that downward entailing quantifiers like those expressed by English determiners *not every* or *no* **can not be translated in Abramsky's language as syntactic units**, unlike upward entailing quantifiers *some* ($\exists x$), *every* ($\forall x$), or even the Henkin quantifier ($[\forall x \exists y \, || \, \forall z \exists w]$). For example, the quantifier *not every* is expressible in the language by formulas of the kind $\sim (\forall x \cdot P(x))$, but crucially this formula does not have an immediate constituent excluding the formula $P(x)$.

Note that $(\sim \forall x) \cdot P(x)$ is also a formula of Abramsky's language, but the game it defines isn't isomorphic to $\sim (\forall x \cdot P(x))$; instead, since $\sim \forall x \equiv \exists x$, $(\sim \forall x) \cdot P(x) \equiv \exists x \cdot P(x)$.

As we have seen, in order to capture branching quantification with the device of parallel processing, we need to combine syntactically units of the language that express the quantifiers involved. Once it comes to downward monotone quantifiers, this move is no longer possible. We will now attempt to alter the interpretation of the language minimally, in order to avoid this issue. I will define a game-theoretic interpretation of the very same language where in fact $\sim (\forall x \cdot P(x)) \equiv (\sim \forall x) \cdot P(x)$ (and no longer equivalent to $\exists x \cdot P(x)$). The major change to the rules will be that role exchange (denoted by $\sim$) will not just affect the game that $\sim$ is prefixed to, but also the games that follow it in sequential composition.

3 Game Semantics Revised

The game definition given below is applicable to the general case of n-player games that Abramsky discusses; winning rules specific for two player semantic games are introduced later.

Notational remarks. We will write ${}_{\hat{\pi}}\psi$ to denote a game ψ played with roles exchanged by permutation $\hat{\pi}$ (which may be any permutation including the identity function). $\pi(\psi)$ will stand for the permutation of roles at the end of the game denoted by ψ, i.e. the function assigning each player her role at the end of the game ψ.

Formally, the rules of a game ϕ are defined as follows:

> if $\phi = 1$, or ϕ is atomic, the players make no move;
>
> in $\phi = \psi_1 \oplus_\alpha \psi_2$, the player α chooses one of ψ_1 and ψ_2, which is then played;
>
> in $\phi = \pi\psi$, the game ${}_{\pi}\psi$ is played.
>
> in $\phi = Q_\alpha x$, the player α assigns a value to the variable x.
>
> in $\phi = \psi_1 \cdot \psi_2$, ψ_1 is played, then ${}_{\pi(\psi_1)}\psi_2$;
> For instance, $\pi(1) = Id$, $\pi(\sim \exists x) = (V, F)$. In the game $1 \cdot (\sim \exists x) \cdot P(x)$, first the empty game 1 is played; then ${}_{\pi(1)}(\sim \exists x) \cdot P(x) = (\sim \exists x) \cdot P(x)$. Then $\sim \exists x = \sim Q_V x$ is played: Abelard and Eloise exchange roles, then Abelard, acting as the Verifier, chooses a value for x. The last portion of the game is ${}_{\pi(\sim\exists x)}P(x) = {}_{(V,F)}\, P(x)$, i.e. $P(x)$ with roles exchanged; in Game Theoretical Semantics the standard interpretation is that the truth value of $P(x)$ is assessed and whoever plays the Verifier role (in this example, that is Abelard), wins iff $P(x)$ is true.
>
> in $\phi = \psi_1 \mid\mid \psi_2$, ψ_1 and ψ_2 are played in parallel, without information flow between the two games.

Let us now refine winning conditions of the classic 2-player semantic games. For the first order subset of the language we will keep the interpretation equivalent to the standard game theoretic one. But since the language in question goes well beyond the language of first-order logic (its proper subset), we will use extra terms to describe the game compared to the usual game theoretical semantics of first order languages. Thus, in addition to winning and losing, we will also talk about *failing* a game. The underlying intuition is this: some games like those defined by first-order formulas end with a robust win; for instance, falsification of an atomic formula makes Abelard win and Eloise *fail* the game; no matter what may follow in a longer game, Eloise can not make up for this failure and always loses. But there are also games where no substantial claims have been verified or falsified. The simplest example is the empty game 1; of course Eloise wins it by default (since no claim she defended was refuted), but we won't say that Abelard failed in this game; in a sense, every game may be seen as starting with the empty game, so in a continuation of 1, either one of the players has the potential to win. In the game ~ 1 the players' roles change and Eloise loses by default; but again the case isn't lost completely, and if the game goes on, she may still, acting as a Falsifier, win it. Winning conditions for the two-player game are defined as follows:

in $\phi = 1$, the Verifier wins and nobody fails;

if ϕ is a true atomic formula, the Verifier wins and the Falsifier fails, otherwise the Falsifier wins and the Verifier fails;

in $\phi = \psi_1 \wedge \psi_2$, the Falsifier chooses one of ψ_1 and ψ_2, which is then played; if a player wins, loses or fails that game, he also wins, loses or fails ϕ correspondingly.

in $\phi = \psi_1 \vee \psi_2$, the Verifier chooses one of ψ_1 and ψ_2, which is then played; if a player wins, loses or fails that game, he also wins, loses or fails ϕ correspondingly.

in $\phi = \sim \psi$, the outcome is as in game ${}_{(V,F)}\psi$.

in $\phi = \psi_1 \cdot \psi_2$, a player loses and fails and the other wins if she fails in ψ_1; if neither player fails ψ_1, the outcome is determined by the subsequent ${}_{\pi(\psi_1)}\psi_2$;

in $\phi = \psi_1 \mid\mid \psi_2$, the Verifier wins iff she wins both ψ_1 and ψ_2, and fails iff she fails at least one of ψ_1 and ψ_2.

The game theoretic interpretation proposed here differs from Abramsky's original proposal. Some of his original equivalences still hold, e.g. for any ϕ, $\phi \equiv 1 \cdot \phi \equiv \phi \cdot 1$; but equivalences involving role permutations no longer hold, so e.g. $\hat{\pi}(\phi \cdot \psi) \equiv \hat{\pi}(\phi) \cdot \hat{\pi}(\psi)$ is valid in Abramsky's interpretation but not ours.

4 On the Notion of Downward Entailment

Entailment relation can be defined in game theoretical semantics analogously to classical logical entailment. A game of partial information ϕ can give rise to three possible truth values: $\mid \phi \mid = \top$ iff Eloise has a winning strategy, $\mid \phi \mid = \bot$ iff Abelard has one, and $\mid \phi \mid = 0$ otherwise. Entailment of closed formulas in three valued logic is standardly defined as:

Definition. ϕ entails ψ iff $\mid \phi \mid \preceq \mid \psi \mid$, assuming the order relation to be $\bot \prec 0 \prec \top$.

Open formulas do not have truth values; still, we would like to apply the notion of entailment to them as well. Let us write ϕ^β for the game ϕ played with free variables in ϕ assigned values by the assignment function β. This is a standard move; in each model, every formula / assignment pair has a truth value. Now we can give the following definition:

Definition. ϕ entails ψ iff for every assignment β, $\mid \phi^\beta \mid \preceq \mid \psi^\beta \mid$, assuming the order relation to be $\bot \prec 0 \prec \top$. $\phi \simeq \psi$ iff ϕ entails ψ and ψ entails ϕ

So far we have been talking about monotonicity without a proper definition, assuming that a game-theoretic quantifier is downward monotone iff it translates a downward monotone generalized quantifier. In our current perspective quantifiers are formulas of the language that function as prefixes to larger formulas. Modulo this syntactic status of quantifiers, we can give completely standard definitions of downward and upward monotonicity:

Definition. A formula prefix ϕ is *upward monotone* iff for all ψ, ξ, if ψ entails ξ then $\phi \cdot \psi$ entails $\phi \cdot \xi$. ϕ is *downward monotone* iff for all ψ, ξ, if ψ entails ξ then $\phi \cdot \xi$ entails $\phi \cdot \psi$

This definition covers both negation (or technically the game ~ 1 which, when prefixed to a formula, is equivalent to the negation thereof) and downward monotone quantifiers like $\sim \exists x$, and can extend to other downward monotone operators (e.g. an operator that makes a formula antecedent of an implication) if the language is enriched with them.

Definition. A formula prefix ϕ is trivial iff for all ψ, $\phi \simeq \phi \cdot \psi$.

The definitions of entailment and monotonicity cited above are standard. It turns out to be possible, however, to give a purely game-theoretic definition to downward monotonicity: a game prefix is downward monotone if it exchanges the players' roles. This is justified by the theorem:

Theorem on Monotonicity. A non-trivial formula prefix ϕ is downward monotone iff it denotes a game in which the players' roles are permuted, $\pi(\phi) = (V, F)$.

5 Parallel Games vs. IF Notation

Parallel composition of two games $\phi \mid\mid \psi$ is so defined that there is no temporal or causal order between choices or events in ϕ and ψ. This has one naturally appealing consequence, namely that parallel composition is commutative, $\phi \mid\mid \psi$ is equivalent to $\psi \mid\mid \phi$. Thus, two independent quantifiers can be reordered without changing the truth value of the formula. This also holds sometimes for IF logic (with necessary adjustment of independence marking); indeed, $\forall x(\exists y/\forall x).R(x, y) \equiv \exists y(\forall x/\exists y).R(x, y)$.

Once downward monotone quantifiers come into play, commutativity may become problematic. As we will see below, this can be captured in the approach that uses partially ordered quantifiers (or games with parallel composition), but it does not seem possible in the language with a total linear order of quantifiers and special marking for independence like Hintikka's slash notation. Indeed, the choice of variable assignment can be independent of previous choices but switching the role of the player makes a big difference. Take the quantifier NOx to be equivalent to $\forall x \sim$ or $\sim \exists x$, and consider for example the formulas

$(NOx).(NOy/NOx).Pxy \equiv \forall x. \sim (\forall y/\forall x). \sim Pxy \equiv \forall x.(\exists y/\forall x). \sim\sim Pxy \equiv \forall x.(\exists y/\forall x).Pxy$

and

$(NOy).(NOx/NOy).Pxy \equiv \forall y. \sim (\forall x/\forall y). \sim Pxy \equiv \forall y.(\exists x/\forall y). \sim\sim Pxy \equiv \forall y.(\exists x/\forall y).Pxy$.

The former is true iff the first order formula $\exists y \forall x.Pxy$ is true, and the latter is true whenever $\exists x \forall y.Pxy$ is true. But these two first order formulas obviously are not logically equivalent, hence independent downward monotone expressions are not permutable in IF.

In a game with parallel composition, in contrast, $(NOx \mid\mid NOy).Pxy \equiv (NOy \mid\mid NOx).Pxy$ if we interpret the game as follows: at NOx and NOy the player F ($\forall$belard) assigns values to the two variables (independently, although the choice independence requirement is vacuous in this case), and each negative quantifier also indicates the role switch for whatever follows. Since neither of the NO quantifiers follows the other, the roles switch just for the literal Pxy, and the formula is actually equivalent to the first order $\forall x \forall y \sim Pxy$. Interestingly, this is precisely the interpretation of natural language sentences with scopally independent negative quantifiers, e.g. *No man and no woman are a perfect couple.*

6 Identity of Permutations in Parallel Composition

Let me make explicit and justify the assumption made in the last paragraph. We have been treating quantifier independence as parallel processing of two games. We have also been assuming that negation is allowed outside literals (contra some approaches languages with quantifier independence), sometimes giving rise to downward monotone quantifiers. My central proposal was that the role switch introduced by a negation in ϕ persists in a subsequent part ψ of the game $\phi \cdot \psi$. But then, if we take parallel processing into account, the game semantics risks being not well-defined. Indeed, consider $(NOx \mid\mid \exists y).Pxy$. The downward monotone quantifier NOx requires a role switch for the atomic formula Pxy, while the upward monotone $\exists x$ maintains the roles constant for the same formula. Parallel composition of the two places incompatible requirements on players' roles for whatever follows. I see only one way to avoid this issue without giving up the symmetry between two parallel games, by adopting the natural condition that *subgames under parallel composition must have the same monotonicity.* Generalizing for the multiple player setting, we may say that two parallel subgames must permute the players in the same way:

> **Parallel Processing Well-formedness Condition**. A game $\phi \mid\mid \psi$ is well-formed iff $\pi(\phi) = \pi(\psi)$, i.e. the role assignment at the end of the games denoted by π and ψ must be identical.

One may think of this condition graph-theoretically. If we represent games with (ordered) graphs, parallel games can be encoded as cycles (in the spirit of the usual graph representation of branching). Role assignments may be seen as colors of edges and nodes of the graph, while role permutation symbols mark borders beween parts of the graph with different colors. The Parallel Processing Well-formedness Condition makes sure that border marking in cycles is coherent, and that two branches have the same color at the point of convergence.

7 Applications to Natural Language Semantics

This correctness condition evokes an analogy in natural language semantics: as has been noticed already by Jon Barwise (1979), branching quantifiers produce a coherent interpretation only if their monotonicities match:

(4) a. *More than five men and more than six women are dating each other*

b. *Fewer than five men and fewer than six women are dating each other*

c. *Fewer than five men and more than six women are dating each other*

In 4a both quantifiers involved are upward monotone, and in 4b they are both downward monotone. In 4c one is downward monotone and the other is upward monotone; Barwise's observation is that while examples like 4c is perfectly syntactically well-formed, speakers of English have much weaker intuitions about the truth conditions of 4c and similar utterances than they have about utterances like 4a or 4b. The strength of this contrast has been challenged (Sher 1990), but the contrast itself appears to be real.

Apart from this observation, the logical language under discussion provides a natural compositional semantics for a class of independent quantifier expressions in natural language. Indeed, take parallel composition to be the interpretation of the conjunction *and*. Combinations of independent basic upward monotone quantifiers then receive the correct interpretation:

(5) a. *Some line and some plane are parallel* $[\exists x \mid\mid \exists y] \cdot P(x, y)$.

b. *Every line and every plane are parallel* $[\forall x \mid\mid \forall y] \cdot P(x, y)$.

Game semantics correctly predicts these to be equivalent to sentences without *and*, which, we assume, translate as an ordinary first order formulas without parallel composition:

(6) a. *Some line is parallel to some plane* $\exists x \cdot \exists y \cdot P(x, y)$.

b. *Every line is parallel to every plane* $\forall x \cdot \forall y \cdot P(x, y)$.

Indeed, if one of the players has a winning strategy in 5a and 5b, she can apply it in 6a and 6b, and vice versa, and will still be guaranteed to win.

Consider now analogous examples with downward monotone quantifiers. This paper proposed to apply the same formal language to represent these:

(7) a. *No line and no plane are parallel* $[\sim \exists x \mid\mid \sim \exists y] \cdot P(x, y)$.

b. *Not every line and not every plane are parallel* $[\sim \forall x \mid\mid \sim \forall y] \cdot P(x, y)$.

The game theoretic semantics proposed here gives the formal expressions just the truth conditions of the corresponding English sentences. For example, 7a is true iff Abelard, acting as the Verifier, can not pick x and y to make $P(x, y)$ true (Eloise makes no moves in this game). For Abelard to be incapable of doing so, the relation denoted by P must be empty; and indeed, the English sentence in 7a is true just in case the relation of being a parallel for a line and a plane is empty. Analogously, 7b is true iff Eloise, acting as the Falsifier, can pick x and y and win $P(x, y)$, making it false (Abelard makes no moves in this game). For Eloise to do this, the complement of the relation denoted by P must be nonempty; and indeed, the English sentence in 7a is true just in case the relation of intersecting (being non-parallel) for a line and a plane is nonempty.

Other generalized quantifiers can also be expressed as syntactic units of the language, take ≥ 2 'two or more':

$\geq 2x.\phi(x)$ is equivalent to $\exists y \exists z.y \neq z \wedge \forall x.\phi(x) \vee \neg(x = y \vee x = z)$ in first order logic; but there is an equivalent formula $[\exists y \cdot \exists z \cdot y \neq z \wedge \forall x.1 \vee \sim (x = y \vee x = z)] \cdot \phi(x)$ where the part in square brackets corresponds to the quantifier and is a syntactic constituent. Similarly, 'at most one' ($\leq 1x.\phi$) is expressible as $\exists y.\forall x.y = x \vee \neg\phi$, which can be restated in our language as $[\exists y \cdot \forall x \cdot (y = x) \vee \sim 1] \cdot \phi$. Branching combinations of two upward monotone quantifiers are given precisely the truth conditions described by Jon Barwise:

(8) a. *At least two men and at least two women are dating each other.*

b. $([\exists y \cdot \exists z \cdot y \neq z \wedge \forall x.1 \vee \sim (x = y \vee x = z)]$
$|| [\exists y' \cdot \exists z' \cdot y' \neq z' \wedge \forall x'.1 \vee \sim (x' = y' \vee x' = z')]) \cdot D(x, x')$

The last formula has truth value *true* iff Eloise can find two pairs z, y and z', y', so that when Abelard picks a member x,x' from each pair, D(x,x') is true, i.e. each member of one pair dates each member of the other pair.

8 A Remaining Issue

The only case when branching quantification in natural language does not fit neatly with the game theoretical truth conditions is when two non-upward monotone quantifiers like ≤ 1 are combined:

(9) a. At most one man and at most one woman date each other.

b. $([\exists y \cdot \forall x \cdot (y = x) \vee \sim 1] \ || \ [\exists y' \cdot \forall x' \cdot (y' = x') \vee \sim 1]) \cdot D(x, x')$.

In the game described by the formula, Eloise has a winning strategy even in some cases when the corresponding English sentence seems to be false[1].

[1] I must note, though, that truth condition judgments on such sentences are somewhat shaky.

Take a model where there is exactly one man m who dates anyone, and multiple women $w_1, ..., w_k$ that he dates. Then Eloise may pick $y = m$ and $y' = w_1$. For a chance to win the game, Abelard has to pick $x \neq y$ and $x' \neq y'$. But if both members of the pair x, x' are distinct from their counterparts in m, w_1, then by assumption $D(x, x')$ must be false in the model, and Abelard (who is a Verifier at this point) loses.

A further refinement of game semantics for downward monotone operators may be needed to accomodate such cases.

References

Abramsky, Samson (2007). A Compositional Game Semantics for Multi-Agent Logics of Partial Information. Vol. 1 of Texts in Logic and Games, 11-48. Amsterdam University Press.

Barwise, Jon (1979). 'On Branching Quantifiers in English', Journal of Philosophical Logic 8, 47-80.

Jaakko Hintikka, Gabriel Sandu (1995). What is the Logic of Parallel Processing? Int. J. Found. Comput. Sci. 6(1): 27-49.

Sher, Gila. (1990), 'Ways of Branching Quantifiers,' Linguistics and Philosophy 13: 393-422.

Expressing Second-order Sentences in Intuitionistic Dependence Logic

Fan Yang
Department of Mathematics and Statistics
University of Helsinki
fan.yang@helsinki.fi

1 Introduction

Dependence Logic (**DL**), as a new approach to independence friendly logic (IF-logic) [Hintikka and Sandu 1989], was introduced in [Väänänen 2007]. Hodges gave a compositional semantics for IF-logic in [Hodges 1997a], [Hodges 1997b]. A recent research by Abramsky, Väänänen [Abramsky and Väänänen 2009] generalized Hodges' construction for team semantics and introduced BID-logic, which extends dependence logic and includes both intuitionistic implication and linear implication. We call the intuitionistic fragment of BID-loigc "intuitionistic dependence logic (**IntDL**)". In this paper, we study basic properties of intuitionistic dependence logic. It is known that dependence logic is equivalent to second order Σ_1^1 fragment. We will show that sentences of **DL** (or Σ_1^1 sentences) are all expressible in **IntDL**. Moreover, sentences of the whole second-order logic are expressible in **IntDL**. Together with the result in [Abramsky and Väänänen 2009] that **IntDL** sentences are translatable into second-order logic, we conclude that **IntDL** is equivalent to second-order logic, in the sense that there is a translation from sentences of one logic into another.

We remark that intuitionistic dependence logic is not "intuitionistic" in the usual sense, as the above-mentioned properties indicate already. When restricted to first-order formulas, IntDL is in fact classical. It is only in the dependence formulas, the intuitionistic implication plays a role.

Throughout the paper, we assume readers are familiar with the standard Tarskian semantics of first-order logic and the standard semantics of second-order logic. We denote the length of a sequence $\bar{x} = \langle x_1, \cdots, x_n \rangle$ of variables by $len(\bar{x})$; similarly for sequences of constants and elements of models. For any assignment for $\bar{x}$, we write $s(\bar{x})$ for $\langle s(x_1), \cdots, s(x_n) \rangle$. We use the standard abbreviation $\forall \bar{x}$ to stand for a sequence of universal quantifiers $\forall x_1 \cdots \forall x_n$ (the length of $\bar{x}$ is always clear from the context or does not matter); similarly for existential quantifiers.

2 BID-logic, Dependence logic and Intuitionistic Dependence Logic

In this section, we introduce intuitionistic dependence logic, which is an intuitionistic fragment of BID-logic introduced in [Abramsky and Väänänen 2009]. We will also recall the dependence logic introduced in [Väänänen 2007] and its basic properties in this general framework.

BID-logic is obtained from a general consideration of Hodges' team semantics [Hodges 1997a], [Hodges 1997b]. Well-formed formulas of BID-logic are given by the rule

$$\begin{aligned}\phi := &\alpha \mid =\!(t_1,\cdots,t_n) \mid \neg =\!(t_1,\cdots,t_n) \mid \bot \mid \psi\wedge\chi \mid \psi\otimes\chi \mid \psi\vee\chi \mid \\ &\psi\to\chi \mid \psi \multimap \chi \mid \forall x\psi \mid \exists x\psi\end{aligned}$$

where α is a first-order literal, $t_1,\cdots,t_n$ are terms. For the semantics for BID-logic, we adopt and generalize Hodges' team semantics. For any structure M, a *team* X of M is a set of assignments with domain M. We define two operations on teams. For any team X of M, and any function $F: X\to M$, the *supplement team* $X(F/x_n)=\{s(F(s)/x_n): s\in X\}$ and the *duplicate team* $X(M/x_n)=\{s(a/x_n): a\in M,\ s\in X\}$. To give the team semantics for BID-logic, we assume all formulas of BID-logic to be evaluated are in negation normal form. Now, for any model M and any team X of M,

- $M\models_X \alpha$ with α first-order literal iff $M\models_s \alpha$ for all $s\in X$ in the usual Tarskian semantics sense;
- $M\models_X =\!(t_1,\cdots,t_n)$ iff for all $s,s'\in X$ such that $t_1(s)=t_1(s'),\cdots,t_{n-1}(s)=t_{n-1}(s')$, we have $t_n(s)=t_n(s')$;
- $M\models_X \neg =\!(t_1,\cdots,t_n)$ iff $X=\emptyset$;
- $M\models_X \bot$ iff $X=\emptyset$;
- $M\models_X \phi\wedge\psi$ iff $M\models_X \phi$ and $M\models_X \psi$;
- $M\models_X \phi\otimes\psi$ iff there exist $Y,Z\subseteq X$ such that $X=Y\cup Z$, $M\models_Y \phi$ and $M\models_Z \psi$;
- $M\models_X \phi\vee\psi$ iff $M\models_X \phi$ and $M\models_X \psi$;
- $M\models_X \phi\to\psi$ iff for any $Y\subseteq X$, if $M\models_Y \phi$ then $M\models_Y \psi$;
- $M\models_X \phi\multimap\psi$ iff for any team Y, if $M\models_Y \phi$ then $M\models_{X\cup Y} \psi$;
- $M\models_X \exists x\phi$ iff $M\models_{X(F/x)} \phi$ for some function $F: X\to M$;
- $M\models_X \forall x\phi$ iff $M\models_{X(M/x)} \phi$.

The intuitionistic implication and linear implication are adjoints of conjunctions; that is

$$\phi \wedge \psi \models \chi \Longleftrightarrow \phi \models \psi \to \chi,$$

$$\phi \otimes \psi \models \chi \Longleftrightarrow \phi \models \psi \multimap \chi.$$

The propositional fragment without dependence formulas of BID-logic is the BI logic, the "logic of Bunched Implications" introduced in [O'Hearn and Pym 1999], [Pym 2002]. The fragment with connectives $\wedge$, $\otimes$ and quantifiers is the usual dependence logic, while the intuitionistic fragment of BID-logic is called *intuitionistic dependence logic*. More precisely, well-formed formulas of **DL** are formed by the following rule

$$\phi := \alpha \mid =\!(t_1, \cdots, t_n) \mid \neg =\!(t_1, \cdots, t_n) \mid \psi \wedge \chi \mid \psi \otimes \chi \mid \forall x \psi \mid \exists x \psi$$

where α is a first-order literal and $t_1, \cdots, t_n$ are terms.; and well-formed formulas of **IntDL** are formed by the following rule

$$\phi := \alpha \mid =\!(t) \mid \bot \mid \psi \wedge \chi \mid \psi \vee \chi \mid \psi \to \chi \mid \forall x \psi \mid \exists x \psi$$

where α is a first-order atom and t is a term. Note that the dependence atoms of IntDL have only single variables, the disjunction $\vee$ is classical and the implication $\to$ is intuitionistic.

The most important property of BID-logic is the *downwards closure* property that for any formula ϕ, if $M \models_X \phi$ and $Y \subseteq X$, then $M \models_Y \phi$. A formula ϕ is said to be *flat* if for all models M and teams X

$$M \models_X \phi \Longleftrightarrow (M \models_{\{s\}} \phi \text{ for all } s \in X).$$

The left-to-right implication in the above clause follows from downwards closure property of BID logic, while the other direction is non-trivial. However, two classes of BID formulas are easily proved to be flat, as the following lemma and theorem shows. We call the **DL** formulas with no occurrence of dependence subformulas *first-order* formulas (of BID-logic).

Theorem 2.1. *First-order formulas are flat.*

Lemma 2.2. *Sentences of BID-logic are flat.*

3 First-order Formulas are Expressible in Intuitionistic Dependence Logic

In this section, we show that every first-order formulas is logically equivalent to a formula in intuitionistic dependence logic. First, we give a definition. Two **BID** formulas ϕ and ψ are said to be *logically equivalent* to each other (in symbols $\phi \equiv \psi$) if for any model M and any team X it holds that

$$M \models_X \phi \Longleftrightarrow M \models_X \psi.$$

Lemma 3.1. *We have the following logical equivalences in* ***BID****-logic*

(1) $=(t_1,\cdots,t_n) \equiv =(t_1)\wedge\cdots\wedge =(t_{n-1}) \to =(t_n)$ *for any terms* $t_1,\cdots,t_n$*;*

(2) $\neg\phi \equiv \phi \to \bot$ *whenever* ϕ *is an atom;*

(3) $(\phi \to \bot) \to \bot \equiv \phi$ *whenever* ϕ *is a flat formula;*

(4) $\phi \otimes \psi \equiv (\phi \to \bot) \to \psi$ *whenever both* ϕ *and* ψ *are flat formulas;*

Proof. Items (1)-(3) are easily proved. We will only show (4). That is to show that for any model M and any team X with $dom(X) \supseteq Fr(\phi) \cup Fr(\psi)$ it holds that

$$M \models_X \phi \otimes \psi \Longleftrightarrow M \models_X (\phi \to \bot) \to \psi.$$

$\Longrightarrow$: Suppose $M \models_X \phi \otimes \psi$. Then there exist Y, Z such that $X = Y \cup Z$, $M \models_Y \phi$ and $M \models_Z \psi$. For any $U \subseteq X$ with $M \models_U \phi \to \bot$, downwards closure gives that for any $s \in U$, $M \models_{\{s\}} \phi \to \bot$, i.e. $M \not\models_{\{s\}} \phi$. Since $M \models_Y \phi$, in view of the flatness of ϕ we conclude that $s \notin Y$, thus $U \subseteq Z$, which implies $M \models_U \psi$ by downwards closure.

$\Longleftarrow$: Suppose $M \models_X (\phi \to \bot) \to \psi$. Define

$$Y = \{s \in X \mid M \models_{\{s\}} \phi\} \text{ and } Z = \{s \in X \mid M \not\models_{\{s\}} \phi\}.$$

Clearly, $X = Y \cup Z$. For any $s \in Z \subseteq X$, we have that $M \models_{\{s\}} \phi \to \bot$, thus since $M \models_{\{s\}} (\phi \to \bot) \to \psi$, we obtain that $M \models_{\{s\}} \psi$. Now both $M \models_Y \phi$ and $M \models_Z \psi$ follow from the flatness of ϕ and ψ. □

Remark 3.2. *When restricted to singleton teams, connectives of* ***IntDL*** *behave as first-order connectives. This explains why double negation law holds for flat formulas.*

Now we define expressibility.

Definition 3.3. Let $\mathscr{L}$ be a sublogic of BID-logic. We say that a formula ϕ of BID-logic is *expressible* in $\mathscr{L}$, if there exists an $\mathscr{L}$ formula ψ such that $\phi \equiv \psi$.

Theorem 3.4. *Every first-order formula is expressible in* ***IntDL****.*

Proof. Assuming that every first-order formula is in conjunctive normal form, the theorem follows immediately from Items (2) and (4) of Lemma 3.1. In order to demonstrate that the order of replacement is important in the translation, we give an example as follows. The first-order formula $((\neg\alpha \otimes \beta) \otimes \gamma) \wedge \delta$ in conjunctive normal form, where $\alpha, \beta, \gamma, \delta$ are first-order atoms, can be translated into **IntDL** in the following order:

$$\begin{aligned}((\neg\alpha \otimes \beta) \otimes \gamma) \wedge \delta &\Longrightarrow (((\neg\alpha \otimes \beta) \to \bot) \to \gamma) \wedge \delta \\ &\Longrightarrow (((((\neg\alpha) \to \bot) \to \beta) \to \bot) \to \gamma) \wedge \delta \\ &\Longrightarrow (((((\alpha \to \bot) \to \bot) \to \beta) \to \bot) \to \gamma) \wedge \delta.\end{aligned}$$

□

4 Sentences of Dependence Logic and Σ_1^1 are Expressible in Intuitionistic Dependence Logic

It is known that IF-logic is equivalent to Σ_1^1 fragment of second-order logic. Dependence logic, which is equivalent to IF-logic is therefore also equivalent to Σ_1^1 fragment. Väänänen in [Väänänen 2007] gave a translation from one logic into another.

Definition 4.1. Let $\mathscr{L}_{SO}$ and $\mathscr{L}_{BID}$ be sublogics of second-order logic and of BID-logic, respectively.

1. We say that a sentence ϕ of $\mathscr{L}_{BID}$ is *expressible* in $\mathscr{L}_{SO}$, if there exists an $\mathscr{L}_{SO}$ sentence ψ such that for any model M

$$M \models \psi \Longleftrightarrow M \models_{\{\emptyset\}} \phi.$$

2. We say that a sentence ψ of $\mathscr{L}_{SO}$ is *expressible* in $\mathscr{L}_{BID}$, if there exists an $\mathscr{L}_{BID}$ sentence ϕ such that for any model M

$$M \models \psi \Longleftrightarrow M \models_{\{\emptyset\}} \phi.$$

Theorem 4.2 (Väänänen 2007)**.** ***DL** sentences are expressible in Σ_1^1 fragment of second-order logic.*

We sketch the proof of the next theorem. In the next section, we will generalize this translation to translate all second order sentences first into BID-logic, and in the end into **IntDL**.

Theorem 4.3 (Väänänen 2007)**.** *Σ_1^1 sentences are expressible in **DL**.*

Proof. (sketch) Without loss of generality, we may assume every Σ_1^1 sentence ϕ is of the following special Skolem normal form

$$\exists f_1 \cdots \exists f_n \forall x_1 \cdots \forall x_m \psi,$$

where every occurrence of f_i ($1 \leq i \leq n$) is of the form $f_i x_{i_1} \cdots x_{i_{m_i}}$. The sentence

$$\begin{aligned} \phi^* = \forall x_1 \cdots \forall x_m \exists y_1 \cdots \exists y_n (=(x_{1_1}, \cdots, x_{1_{m_1}}, y_1) \wedge \\ \cdots \wedge =(x_{n_1}, \cdots, x_{n_{m_n}}, y_n) \wedge \psi'), \end{aligned} \tag{1}$$

where ψ' is obtained from ψ by replacing everywhere $f_i x_{i_1} \cdots x_{i_{m_i}}$ by y_i, is the sentence of **DL** expressing ϕ. □

Remark 4.4. *Equation (1) with ψ' in conjunctive normal form is a normal form of **DL** sentences.*

Note that in the normal form of a **DL** sentence, the only subformulas that are not in the language of **IntDL** are dependence atoms and first-order formulas. As we have proved in the previous section, these two kinds of formulas are both expressible in **IntDL**, therefore we have the next theorem.

Theorem 4.5. *For any **DL** sentence ϕ, there exists a sentence ϕ^* of **IntDL** which is logically equivalent to ϕ.*

Proof. Without loss of generality, we may assume the **DL** sentence ϕ is of the normal form (1). The **IntDL** sentence ϕ^* is obtained by replacing the subformulas of the form $=(x_{i_1}, \cdots, x_{i_{m_i}}, y_i)$ by the formula $=(x_{i_1}) \wedge \cdots \wedge =(x_{i_{m_i}}) \to =(y_i)$ and the first-order formula ψ' in (1) by its equivalent **IntDL** formula described in Theorem 3.4. □

Theorem 4.6. Σ^1_1 *sentences are expressible in **IntDL**.*

Proof. Follows from Theorem 4.3 and Theorem 4.5. □

Negation in dependence logic, as well as BID-logic does not satisfy the Law of Excluded Middle and is therefore not classical. Dependence logic with classical negation are called *team logic*, see Chapter 8 in [Väänänen 2007] for further discussions on team logic. However, for sentences of intuitionistic dependence logic, as shown in the next lemma we do get the classical negation. This is basically because for sentences we only consider singleton teams (the team $\{\emptyset\}$ with empty assignment), and when restricted to singleton teams, the semantics of **IntDL** is in fact classical, as it is pointed out in Remark 3.2.

Lemma 4.7. *For any sentence ϕ, we have that for any model M*

$$M \models_{\{\emptyset\}} \phi \to \bot \iff M \not\models_{\{\emptyset\}} \phi.$$

Using the intuitionistic ("classical") negation for sentences, we are able to express Π^1_1 sentences as well.

Corollary 4.8. Π^1_1 *sentences are expressible in **IntDL**.*

Proof. Let ψ be a Π^1_1 sentence. Note that ψ is equivalent to some $\neg\phi$, where ϕ is a Σ^1_1 sentence. By Theorem 4.6, there exists an **IntDL** sentence ϕ^* such that

$$M \models \phi \iff M \models_{\{\emptyset\}} \phi^*$$

for all model M. Since ϕ^* is a sentence, by Lemma 4.7, we have that

$$M \models_{\{\emptyset\}} \phi^* \to \bot \iff M \not\models_{\{\emptyset\}} \phi^* \iff M \not\models \phi \iff M \models \neg\phi \iff M \models \psi,$$

thus $\phi^* \to \bot$ is the sentence of **IntDL** expressing ψ. □

5 Second-order Sentences are Expressible in Intuitionistic Dependence Logic

In this section, we will generalize the proofs of Theorem 4.3 and Theorem 4.6 to show that all sentences of second-order logic are expressible in **IntDL**. Together with the result of the next theorem proved in [Abramsky and Väänänen 2009], we will be able to conclude that the expressive power of **IntDL** is so strong that it in fact is equivalent to the whole second-order logic, in the sense that there is a translation from sentences of one logic into another.

Theorem 5.1 (Abramsky, Väänänen 2009)**.** ***IntDL*** *sentences are expressible in second-order logic.*

In order to proceed to the main theorem of this paper (Theorem 5.13), we first recall the normal form of second order formulas.

Theorem 5.2. *[Normal Form of second order formulas]*

1. *Every Σ^1_n formula is equivalent to a formula*
   - *either of the form $\exists\overline{f^1}\forall\overline{f^2}\cdots\forall\overline{f^{n-1}}\exists\overline{f^n}\forall\bar{x}\psi$, where ψ is quantifier-free, whenever n is odd,*
   - *or of the form $\exists\overline{f^1}\forall\overline{f^2}\cdots\exists\overline{f^{n-1}}\forall\overline{f^n}\exists\bar{x}\psi$, where ψ is quantifier-free, whenever n is even;*
2. *Every Π^1_n formula is equivalent to a formula*
   - *either of the form $\forall\overline{f^1}\exists\overline{f^2}\cdots\exists\overline{f^{n-1}}\forall\overline{f^n}\exists\bar{x}\psi$, where ψ is quantifier-free, whenever n is odd,*
   - *or of the form $\forall\overline{f^1}\exists\overline{f^2}\cdots\forall\overline{f^{n-1}}\exists\overline{f^n}\forall\bar{x}\psi$, where ψ is quantifier-free, whenever n is even.*

In the main proofs, we will first focus on Π^1_{2n} and Σ^1_{2n+1} sentences; later, we will give translations of Σ^1_{2n} and Π^1_{2n+1} sentences out of those of Π^1_{2n} and Σ^1_{2n+1}, respectively.

The first step of the proofs is to turn every Π^1_{2n} and Σ^1_{2n+1} sentences into equivalent sentences of a nice form. To this end, we need some lemmas.

Lemma 5.3. *For any first-order $L(f)$ formula $\phi(ft_1\cdots t_n)$, where f has an occurrence in ϕ of the form $ft_1\cdots t_n$ for some terms $t_1\cdots t_n$, we have that*

$$\models \phi(ft_1\cdots t_n)\leftrightarrow\forall x_1\cdots\forall x_n((t_1=x_1)\wedge\cdots\wedge(t_n=x_n)\to\phi(fx_1\cdots x_n)),$$

where $x_1,\cdots,x_n$ are new variables and $\phi(fx_1\cdots x_n)$ is the formula obtained from $\phi(ft_1\cdots t_n)$ by replacing everywhere $ft_1\cdots t_n$ by $fx_1\cdots x_n$.

Proof. Easy. □

Lemma 5.4. *If $\phi(fx_{i_1}\cdots x_{i_m}, fx_{j_1}\cdots x_{j_m})$ is a first-order $L(f)$ formula, such that in ϕ f has an occurrence of the form $fx_{i_1}\cdots x_{i_m}$ and an occurrence of the form $fx_{j_1}\cdots x_{j_m}$ with $\{x_{i_1}\cdots x_{i_m}\}\cap\{x_{j_1}\cdots x_{j_m}\}=\emptyset$, then we have that*

$$\begin{aligned}&\models \forall x_1\cdots\forall x_n\phi(fx_{i_1}\cdots x_{i_m}, fx_{j_1}\cdots x_{j_m})\\&\quad\leftrightarrow \exists g\forall x_1\cdots\forall x_n(\phi(fx_{i_1}\cdots x_{i_m}, gx_{j_1}\cdots x_{j_m})\\&\quad\wedge((x_{i_1}=x_{j_1})\wedge\cdots\wedge(x_{i_m}=x_{j_m})\to(fx_{i_1}\cdots x_{i_m}=gx_{j_1}\cdots x_{j_m}))),\end{aligned}$$

where $\phi(fx_{i_1}\cdots x_{i_m}, gx_{j_1}\cdots x_{j_m})$ is the formula obtained from $\phi(fx_{i_1}\cdots x_{i_m}, fx_{j_1}\cdots x_{j_m})$ by replacing everywhere $fx_{j_1}\cdots x_{j_m}$ by $gx_{j_1}\cdots x_{j_m}$.

Proof. Easy. □

Lemma 5.5. *Every Π^1_{2n} formula is equivalent to a formula ϕ of the form*

$$\forall f^1_1\cdots\forall f^1_{p_1}\exists g^1_1\cdots\exists g^1_{q_1}\cdots\forall f^n_1\cdots\forall f^n_{p_n}\exists g^n_1\cdots\exists g^n_{q_n}\forall\bar{z}\forall\bar{x}\forall\bar{y}\psi,$$

where

- *ψ is quantifier free;*
- *every occurrence of f^γ_i ($1\le\gamma\le n$, $1\le i\le p_\gamma$) in ψ is of the form $f^\gamma_i\mathbf{x}^{\gamma,i}$, where*
$$\mathbf{x}^{\gamma,i}=\langle x_{\gamma,i_1},\cdots,x_{\gamma,i_{o(f^\gamma_i)}}\rangle$$
is a subsequence of $\overline{x}$;
- *every occurrence of g^δ_j ($1\le\delta\le n$, $1\le j\le q_\delta$) in ψ is of the form $g^\delta_j\mathbf{y}^{\delta,j}$, where*
$$\mathbf{y}^{\delta,j}=\langle y_{\delta,j_1},\cdots,y_{\delta,j_{o(g^\delta_j)}}\rangle$$
is a subsequence of $\overline{y}$.

Proof. By Theorem 5.2, we may assume that every Π^1_{2n} formula χ is in the following normal form:

$$\forall f^1_1\cdots\forall f^1_{p_1}\exists g^1_1\cdots\exists g^1_{q_1}\cdots\forall f^n_1\cdots\forall f^n_{p_n}\exists g^n_1\cdots\exists g^n_{q'_n}\forall\bar{z}\psi,$$

where ψ is quantifier-free. First, apply Lemma 5.3 to every function symbol occurred in χ in such a way that the first-order part of the resulting formula χ' satisfies the condition of Lemma 5.4 for each occurred function symbol not in the required form. Then several applications of Lemma 5.4 will give the formula ϕ of the required form. □

The next technical lemma will play a role in the proof of Lemmas 5.7 and 5.9.

Lemma 5.6. *Let $\phi(\bar{f},\bar{x})$ be any $L(\bar{f})$ first-order formula, where the occurrences of f_i ($1 \le i \le p$) in ϕ is of the form*

$$f_i x_{i_1} \cdots x_{i_{m_i}},$$

where $\langle x_{i_1} \cdots x_{i_{m_i}} \rangle$ ($1 \le i \le p$) is a subsequence of $\bar{x}$. Let $(M,\overline{F})$ be any $L(\bar{f})$ model and s any assignment for $\bar{x}$. Let $y_1,\cdots,y_p$ be new variables. Define an assignment $\bar{s}$ for $\bar{x},y_1,\cdots,y_p$ extended from s by taking

$$\bar{s}(y_i) = F_i(s(x_{i_1}),\cdots,s(x_{i_{m_i}}) \textit{ for all } 1 \le i \le p$$

and $\bar{s}(x_j) = s(x_j)$ for all $1 \le j \le n$. Then

$$(M,\overline{F},s(\bar{x})) \models \phi(\bar{f},\bar{x}) \iff M \models_{\{\bar{s}\}} \phi',$$

*where ϕ' is the **DL** formula obtained from ϕ by replacing everywhere $f_i x_{i_1} \cdots x_{i_{m_i}}$ by y_i for each $1 \le i \le p$.*

Proof. It is easy to show by induction that for any term t, $s(t) = \bar{s}(t')$. Next, we show the lemma by induction on ϕ. The only interesting case is the case that $\phi \equiv \psi \vee \chi$. In this case, we have that

$$\begin{aligned}(M,\overline{F},s(\bar{x})) \models \psi \vee \chi &\iff (M,\overline{F},s(\bar{x})) \models \psi \text{ or } (M,\overline{F},s(\bar{x})) \models \chi \\ &\iff M \models_{\{\bar{s}\}} \psi' \text{ or } M \models_{\{\bar{s}\}} \chi' \quad \text{(by induction hypothesis)} \\ &\iff M \models_{\{\bar{s}\}} \psi' \otimes \chi'.\end{aligned}$$

□

Let $o(f)$ denote the arity of a function symbol f. Let $\mathbf{x} = \langle x_1,\cdots,x_n\rangle$ be a sequence of variables. The formula $=\!(x_1,\cdots,x_n,y)$ is abbreviated as $=\!(\mathbf{x},y)$. The duplicate team $X(M/x_1)\cdots(M/x_n)$ of a team X of M is abbreviated as $X(M/x_1,\cdots,x_n)$. A sequence of functions with both superscripts and subscripts $F^i_1,\cdots,F^i_n$ is abbreviated as $\overline{F^i}$; similarly for function-variables.

Lemma 5.7. *Π^1_{2n} sentences are expressible in BID-logic.*

Proof. Without loss of generality, we may assume that every Π^1_{2n} sentence ϕ is of the form described in Lemma 5.5. Let

$$\begin{aligned}\phi^* =& \forall u_{1,1}\cdots\forall u_{1,p_1}\cdots\forall u_{n,1}\cdots\forall u_{n,p_n}\forall\bar{z}\forall\bar{x}\forall\bar{y} \\ &(\Psi_1 \to \exists v_{1,1}\cdots\exists v_{1,q_1}(\Theta_1 \wedge (\Psi_2 \to \exists v_{2,1}\cdots\exists v_{2,q_2}(\Theta_2 \wedge (\Psi_3 \cdots \\ &\to \exists v_{n-1,1}\cdots\exists v_{n-1,q_{n-1}}(\Theta_{n-1} \wedge (\Psi_n \to \exists v_{n,1}\cdots\exists v_{n,q_n}(\Theta_n \wedge \psi' \underbrace{)))\cdots)))))}_{2n},\end{aligned}$$

where

$$\Psi_\gamma = \bigwedge_{i=1}^{p_\gamma} =\!(\mathbf{x}^{\gamma,i},u_{\gamma,i}) \text{ for every } 1 \le \gamma \le n,$$

$$\Theta_\delta = \bigwedge_{j=1}^{q_\delta} =\!(\mathbf{y}^{\delta,j}, v_{\delta,j}) \text{ for every } 1 \leq \delta \leq n$$

and ψ' is the first-order formula obtained from ψ by replacing everywhere

- $f_i^\gamma \mathbf{x}^{\gamma,i}$ by a new variable $u_{\gamma,i}$ $(1 \leq \gamma \leq n,\ 1 \leq i \leq p_\gamma)$,
- $g_j^\delta \mathbf{y}^{\delta,j}$ by a new variable $v_{\delta,j}$ $(1 \leq \gamma \leq n,\ 1 \leq j \leq q_\delta)$.

The rest of the proof is devoted to show that the sentence ϕ^* of BID-logic expresses the Π_{2n}^1 sentence ϕ, i.e. to show that for any model M it holds that

$$M \models \phi \Longleftrightarrow M \models_{\{\emptyset\}} \phi^*.$$

$\Longrightarrow$: Suppose $M \models \phi$. Then for any sequence of functions

$$F_1^1 : M^{o(f_1^1)} \to M, \cdots, F_{p_1}^1 : M^{o(f_{p_1}^1)} \to M,$$

there exists a sequence of functions

$$G_1^1(\overline{F^1}) : M^{o(g_1^1)} \to M, \cdots, G_{q_1}^1(\overline{F^1}) : M^{o(g_{q_1}^1)} \to M$$

such that

$$\cdots\ \cdots$$

for any sequence of functions

$$F_1^n : M^{o(f_1^n)} \to M, \cdots, F_{p_n}^n : M^{o(f_{p_n}^n)} \to M,$$

there exists a sequence of functions

$$G_1^n(\overline{F^1}, \cdots, \overline{F^n}) : M^{o(g_1^n)} \to M, \cdots, G_{q_n}^n(\overline{F^1}, \cdots, \overline{F^n}) : M^{o(g_{q_n}^n)} \to M$$

such that

$$(M, \overline{F^1}, \overline{G^1}, \cdots, \overline{F^n}, \overline{G^n}) \models \forall \bar{z} \forall \bar{x} \forall \bar{y} \psi(\overline{f^1}, \overline{g^1}, \cdots, \overline{f^n}, \overline{g^n}). \tag{2}$$

Let Y_1 be a subteam of

$$X = \{\emptyset\}(M/u_{1,1}, \cdots, u_{1,p_1}, \cdots, u_{n,1}, \cdots u_{n,p_n})(M/\bar{z}, \bar{x}, \bar{y})$$

such that $M \models_{Y_1} \Psi_1$. It suffices to show that

$$\begin{aligned} M \models_{Y_1} &\exists v_{1,1} \cdots \exists v_{1,q_1} (\Theta_1 \wedge (\Psi_2 \to \exists v_{2,1} \cdots \exists v_{2,q_2} (\Theta_2 \wedge (\Psi_3 \cdots \\ &\to \exists v_{n-1,1} \cdots \exists v_{n-1,q_{n-1}} (\Theta_{n-1} \wedge (\Psi_n \to \exists v_{n,1} \cdots \exists v_{n,q_n} (\Theta_n \wedge \psi' \underbrace{))) \cdots))))}_{2n-1}. \end{aligned}$$

For any $1 \leq i \leq p_1$ and any $\bar{d} \in M^{o(f_i^1)}$, if there exists $s \in Y_1$ such that $s(\mathbf{x}^{1,i}) = \bar{d}$, then choose an $s^1_{i,\bar{d}} \in Y_1$ that has this property. Pick a point $a_0 \in M$. Define functions $F_i^1(Y_1) : M^{o(f_i^1)} \to M$ for all $1 \leq i \leq p_1$ by taking

$$F_i^1(Y_1)(\bar{d}) = \begin{cases} s^1_{i,\bar{d}}(u_{1,i}), & \text{if there exists } s \in Y_1 \text{ such that } s(\mathbf{x}^{1,i}) = \bar{d}; \\ a_0, & \text{otherwise.} \end{cases}$$

Inductively define for each $1 \leq j \leq q_1$ a function

$$\beta_{1,j} : Y_1(\beta_{1,1}/v_{1,1}) \cdots (\beta_{1,j-1}/v_{1,j-1}) \to M$$

by taking

$$\beta_{1,j}(s) = G_j^1(\overline{F^1(Y_1)})(s(\mathbf{y}^{1,j})).$$

Put

$$Y_1' = Y_1(\beta_{1,1}/v_{1,1}) \cdots (\beta_{1,q_1}/v_{1,q_1}).$$

It suffices to show that $M \models_{Y_1'} \Theta_1$ and

$$M \models_{Y_1'} \Psi_2 \to \exists v_{2,1} \cdots \exists v_{2,q_2} (\Theta_2 \wedge (\Psi_3 \cdots \to \exists v_{n-1,1} \cdots \exists v_{n-1,q_{n-1}} (\Theta_{n-1} \wedge (\Psi_n \to \exists v_{n,1} \cdots \exists v_{n,q_n} (\Theta_n \wedge \psi' \underbrace{))) \cdots))}_{2n-3}.$$

The former is obvious by the definitions of Y_1' and $\beta_{1,1}, \cdots, \beta_{1,q_1}$. To show the latter, it suffices to show that for any subteams Y_2 of Y_1', ... , Y_n of Y_{n-1}' such that $M \models_{Y_2} \Psi_2$, ... , $M \models_{Y_n} \Psi_n$, it holds that

$$M \models_{Y_2'} \Theta_2, \cdots, M \models_{Y_n'} \Theta_n \tag{3}$$

and $M \models_{Y_n'} \psi'$, where $Y_1', \cdots, Y_n'$ are defined in the same way as above. By the definitions of $Y_2', \cdots, Y_n'$ and $\beta_{2,1}, \cdots, \beta_{2,q_2}, \cdots \beta_{n,1}, \cdots, \beta_{n,q_n}$, (3) hold obviously. To show $M \models_{Y_n'} \psi'$, since ψ' is flat, it suffices to show $M \models_{\{s\}} \psi'$ for any $s \in Y_n'$.

Indeed, for the functions $F_i^\gamma(Y_\gamma) : M^{o(f_i^\gamma)} \to M$ ($1 \leq \gamma \leq n$, $1 \leq i \leq p_\gamma$) defined as above, by (2) we have that

$$(M, \overline{F^1(Y_1)}, \overline{G^1(\overline{F^1(Y_1)})}, \cdots, \overline{F^n(Y_n)}, \overline{G^n(\overline{F^1(Y_1)}, \cdots, \overline{F^n(Y_n)})}, s(\bar{z}), s(\bar{x}), s(\bar{y})) \models \psi(\overline{f^1}, \overline{g^1}, \cdots, \overline{f^n}, \overline{g^n}, \bar{z}, \bar{x}, \bar{y}),$$

Note that for any sequence $\mathbf{x}^{\gamma,i}$ ($1 \leq \gamma \leq n$, $1 \leq i \leq p_\gamma$), by definition, we have that $s^\gamma_{i,s(\mathbf{x}^{\gamma,i})}(\mathbf{x}^{\gamma,i}) = s(\mathbf{x}^{\gamma,i})$, from which and the fact that $M \models_{Y_\gamma} \Psi_\gamma$, it follows that

$$F_i^\gamma(Y_\gamma)(s(\mathbf{x}^{\gamma,i})) = s^\gamma_{i,s(\mathbf{x}^{\gamma,i})}(u_{\gamma,i}) = s(u_{\gamma,i}).$$

Hence, applying Lemma 5.6 gives $M \models_{\{s\}} \psi'$.

$\Longleftarrow$: Suppose $M \models_{\{\emptyset\}} \phi^*$. Then

$$M \models_X \boldsymbol{\Psi}_1 \to \exists v_{1,1} \cdots \exists v_{1,q_1} (\boldsymbol{\Theta}_1 \wedge (\boldsymbol{\Psi}_2 \to \exists v_{2,1} \cdots \exists v_{2,q_2} (\boldsymbol{\Theta}_2 \wedge (\boldsymbol{\Psi}_3 \cdots \to \exists v_{n-1,1} \cdots \exists v_{n-1,q_{n-1}} (\boldsymbol{\Theta}_{n-1} \wedge (\boldsymbol{\Psi}_n \to \exists v_{n,1} \cdots \exists v_{n,q_n} (\boldsymbol{\Theta}_n \wedge \psi' \underbrace{)))\cdots))))}_{2n-1}.$$

For any sequence of functions $F_1^1 : M^{o(f_1^1)} \to M, \cdots, F_{p_1}^1 : M^{o(f_{p_1}^1)} \to M$, consider the following subteam Y_1 of X:

$$Y_1 = \{s \in \{\emptyset\}(M/u_{1,1}, \cdots, u_{1,p_1}, \cdots, u_{n,1}, \cdots u_{n,p_n})(M/\bar{z}, \bar{x}, \bar{y}) \mid s(u_{1,i}) = F_i^1(s(\mathbf{x}^{1,i})) \text{ for each } 1 \leq i \leq p_1.\}$$

Clearly, $M \models_{Y_1} \boldsymbol{\Psi}_1$ holds, thus we have that

$$M \models_{Y_1} \exists v_{1,1} \cdots \exists v_{1,q_1} (\boldsymbol{\Theta}_1 \wedge (\boldsymbol{\Psi}_2 \to \exists v_{2,1} \cdots \exists v_{2,q_2} (\boldsymbol{\Theta}_2 \wedge (\boldsymbol{\Psi}_3 \cdots \to \exists v_{n-1,1} \cdots \exists v_{n-1,q_{n-1}} (\boldsymbol{\Theta}_{n-1} \wedge (\boldsymbol{\Psi}_n \to \exists v_{n,1} \cdots \exists v_{n,q_n} (\boldsymbol{\Theta}_n \wedge \psi' \underbrace{)))\cdots))))}_{2n-1}.$$

So there exist functions

$$\beta_{1,j} : Y_1(\beta_{1,1}/v_{1,1}) \cdots (\beta_{1,j-1}/v_{1,j-1}) \to M$$

for all $1 \leq j \leq q_1$ such that $M \models_{Y_1'} \boldsymbol{\Theta}_1$ and

$$M \models_{Y_1'} \boldsymbol{\Psi}_2 \to \exists v_{2,1} \cdots \exists v_{2,q_2} (\boldsymbol{\Theta}_2 \wedge (\boldsymbol{\Psi}_3 \cdots \to \exists v_{n-1,1} \cdots \exists v_{n-1,q_{n-1}} (\boldsymbol{\Theta}_{n-1} \wedge (\boldsymbol{\Psi}_n \to \exists v_{n,1} \cdots \exists v_{n,q_n} (\boldsymbol{\Theta}_n \wedge \psi' \underbrace{)))\cdots))}_{2n-3},$$

where $Y_1' = Y_1(\beta_{1,1}/v_{1,1}) \cdots (\beta_{1,q_1}/v_{1,q_1})$.

For any $1 \leq j \leq q_1$ and any $\bar{d} \in M^{o(g_j^1)}$, by the definition of Y_1', there must exist $s \in Y_1'$ such that $s(\mathbf{y}^{1,j}) = \bar{d}$; choose an $s_{j,\bar{d}}^1 \in Y_1'$ that has this property. Define for each $1 \leq j \leq q_1$ a function $G_j^1 : M^{o(g_j^1)} \to M$ by taking

$$G_j^1(\bar{d}) = s_{j,\bar{d}}^1(v_{1,j}).$$

Repeat the same procedure to define inductively for any sequence of functions $\overline{F^2}, \cdots, \overline{F^n}$ the subteams Y_2 of Y_1', ... , Y_n of Y_{n-1}' such that $M \models_{Y_\gamma} \boldsymbol{\Psi}_\gamma$, where the extension Y_γ' of Y_γ satisfies $M \models_{Y_\gamma'} \boldsymbol{\Theta}_\gamma$, for all $2 \leq \gamma \leq n$, and to define inductively functions

$$G_1^2 : M^{o(g_1^2)} \to M, \cdots, G_{q_2}^2 : M^{o(g_{q_2}^2)} \to M, \cdots,$$
$$G_1^n : M^{o(g_1^n)} \to M, \cdots, G_{q_n}^n : M^{o(g_{q_n}^n)} \to M.$$

In the last step, we have that $M \models_{Y_n'} \boldsymbol{\Theta}_n \wedge \psi'$ and the sequence of functions $G_1^n, \cdots, G_{q_n}^n$ is defined. Now, for any sequences $\bar{c}, \bar{a}, \bar{b}$ in M with

$$len(\bar{c}) = len(\bar{z}),\ len(\bar{a}) = len(\bar{x}),\ len(\bar{b}) = len(\bar{y}),$$

there exists $s \in Y'_n$ such that

$$s(\bar{z}) = \bar{c},\ s(\bar{x}) = \bar{a},\ s(\bar{y}) = \bar{b}.$$

We have that $M \models_{\{s\}} \psi'$. Note that for each $1 \leq \delta \leq n$ and each $1 \leq j \leq q_\delta$, we have that $s^\delta_{j,s(\mathbf{y}^{\delta,j})}(\mathbf{y}^{\delta,j}) = s(\mathbf{y}^{\delta,j})$, from which and the fact that $M \models_{Y'_\delta} \mathbf{\Theta}_\delta$, we obtain that

$$G^\delta_j(s(\mathbf{y}^{\delta,j})) = s^\delta_{j,s(\mathbf{y}^{\delta,j})}(v_{\delta,j}) = s(v_{\delta,j}).$$

Hence, applying Lemma 5.6 gives

$$(M, \overline{F^1}, \overline{G^1}, \cdots, \overline{F^n}, \overline{G^n}, \bar{c}, \bar{a}, \bar{b}) \models \psi,$$

therefore $M \models \phi$. □

Observe that in the sentence ϕ^* of the proof of Theorem 5.7, the only subformulas that are not in the language of **IntDL** are dependence atoms and first-order formulas, both of which are expressible in **IntDL**, therefore we have the following theorem.

Theorem 5.8. Π^1_{2n} *sentences are expressible in* ***IntDL****.*

Proof. Follows from Lemma 5.7, Lemma 3.1 and Theorem 3.4. □

Lemma 5.9. Σ^1_{2n+1} *sentences are expressible in BID-logic.*

Proof. By a similar argument with that in the proof of Lemma 5.5, one can turn every Σ^1_{2n+1} sentences into a sentence ϕ of the form

$$\exists f^1_1 \cdots \exists f^1_{p_1} \forall g^1_1 \cdots \forall g^1_{q_1} \cdots \exists f^n_1 \cdots \exists f^n_{p_n} \forall g^n_1 \cdots \forall g^n_{q_n}$$
$$\exists f^{n+1}_1 \cdots \exists f^{n+1}_{p_{n+1}} \forall \bar{z} \forall \bar{x} \forall \bar{y} \psi,$$

where

- ψ is quantifier free;
- every occurrence of f^γ_i ($1 \leq \gamma \leq n+1$, $1 \leq i \leq p_\gamma$) in ψ is of the form $f^\gamma_i \mathbf{x}^{\gamma,i}$, where
$$\mathbf{x}^{\gamma,i} = \langle x_{\gamma,i_1}, \cdots, x_{\gamma,i_{o(f^\gamma_i)}} \rangle$$
is a subsequence of $\bar{x}$;
- every occurrence of g^δ_j ($1 \leq \delta \leq n$, $1 \leq j \leq q_\delta$) in ψ is of the form $g^\delta_j \mathbf{y}^{\delta,j}$, where
$$\mathbf{y}^{\delta,j} = \langle y_{\delta,j_1}, \cdots, y_{\delta,j_{o(g^\delta_j)}} \rangle$$
is a subsequence of $\bar{y}$.

Now, by a similar argument with that in the proof of Lemma 5.7, one can show the following sentence ϕ^* of BID logic expresses the Σ^1_{2n+1} sentence ϕ:

$$\begin{aligned}\phi^* =& \forall v_{1,1}\cdots\forall v_{1,q_1}\cdots\forall v_{n,1}\cdots\forall v_{n,q_n}\forall\bar{z}\forall\bar{x}\forall\bar{y}\\ &\exists u_{1,1}\cdots\exists u_{1,p_1}(\Psi_1\wedge(\Theta_1\to\exists u_{2,1}\cdots\exists u_{2,p_2}(\Psi_2\wedge(\Theta_2\to\cdots\\ &\to\exists u_{n,1}\cdots\exists u_{n,p_n}(\Psi_n\wedge(\Theta_n\to\exists u_{n+1,1}\cdots\exists u_{n+1,p_{n+1}}(\Psi_{n+1}\wedge\psi'\underbrace{)))\cdots))))}_{2n+1},\end{aligned}$$

where

$$\Psi_\gamma = \bigwedge_{i=1}^{p_\gamma} =\!(\mathbf{x}^{\gamma,i},u_{\gamma,i}) \text{ for every } 1\le\gamma\le n+1,$$

$$\Theta_\delta = \bigwedge_{j=1}^{q_\delta} =\!(\mathbf{y}^{\delta,j},v_{\delta,j}) \text{ for every } 1\le\delta\le n$$

and ψ' is the first-order formula obtained from ψ by replacing everywhere

- $f_i^\gamma\mathbf{x}^{\gamma,i}$ by a new variable $u_{\gamma,i}$ ($1\le\gamma\le n+1$, $1\le i\le p_\gamma$),
- $g_j^\delta\mathbf{y}^{\delta,j}$ by a new variable $v_{\delta,j}$ ($1\le\gamma\le n$, $1\le j\le q_\delta$).

□

Theorem 5.10. Σ^1_{2n+1} *sentences are expressible in* ***IntDL****.*

Proof. Follows from Lemma 5.9, Lemma 3.1 and Theorem 3.4. □

Remark 5.11. *The proof of Theorem 4.6 is a special case of the proof of Theorem 5.10.*

By adding dummy quantifiers, one may transform a Σ^1_{2n} sentence to an equivalent Σ^1_{2n+1} sentence in the normal form, and a Π^1_{2n+1} sentence to an equivalent Π^1_{2n+2} sentence in the normal form. It then follows immediately from Theorem 5.8 and Theorem 5.10 that Σ^1_{2n} and Π^1_{2n+1} sentences are expressible in **IntDL**.

Alternatively, one may also obtain translations for Σ^1_{2n} and Π^1_{2n+1} sentences by using intuitionistic negation as it is described in the next corollary.

Corollary 5.12. Σ^1_{2n} *and* Π^1_{2n+1} *sentences are expressible in* ***IntDL****.*

Proof. Note that any Σ^1_{2n} sentence ϕ is equivalent to a sentence $\neg\psi$ with ψ a Π^1_{2n} sentence. Let ψ^* be an **IntDL** sentence expressing ψ, thus by a similar argument with that in the proof of Corollary 4.8, $\psi^*\to\bot$ is the **IntDL** sentence that expresses ϕ. A similar argument applies to Π^1_{2n+1} sentences as well. □

Finally, we arrive at the following theorem.

Theorem 5.13. *Second order sentences are expressible in* ***IntDL****.*

Proof. Follows from Theorem 5.8, Theorem 5.10 and Corollary 5.12. □

References

[Abramsky and Väänänen 2009] S. Abramsky, J Väänänen, *From IF to BI*, Synthese 167(2): 207-230

[Hintikka and Sandu 1989] J. Hintikka ,and G. Sandu *Informational Independence as a Semantical Phenomenon*, in Logic, Methodology and Philosophy of Science, Vol. 8, J. E. Fenstad, I. T. Frolov, and R. Hilpinen (eds.), Amsterdam: Elsevier, pp. 571C589.

[Hodges 1997a] W. Hodges, *Compositional Semantics for a Langauge of Imperfect Information*, Logic Journal of the IGPL, 5: pp. 539C563.

[Hodges 1997b] W. Hodges, *Some Strange Quantifiers*, in Structures in Logic and Computer Science: A Selection of Essays in Honor of A. Ehrenfeucht, (Lecture Notes in Computer Science, Volume 1261) J. Mycielski, G. Rozenberg, and A. Salomaa (eds.), London: Springer, pp. 51C65.

[O'Hearn and Pym 1999] P. O'Hearn, and D. Pym, *The Logic of Bunched Implications*, Bulletin of Symbolic Logic 5(2), pp. 215-244.

[Pym 2002] D. Pym, *The Semantics and Proof Theory of the Logic of Bunched Implications*, Kluwer Academic Publishers.

[Väänänen 2007] J. Väänänen, Dependence Logic: A New Approach to Independence Friendly Logic, Cambridge: Cambridge University Press.