Inclusion and Exclusion Dependencies in Team Semantics

On some logics of imperfect information

Pietro Galliani

Faculteit der Natuurwetenschappen, Wiskunde en Informatica
Institute for Logic, Language and Computation
Universiteit van Amsterdam
P.O. Box 94242, 1090 GE AMSTERDAM, The Netherlands
Phone: +31 020 525 8260
Fax (ILLC): +31 20 525 5206

Abstract

We introduce some new logics of imperfect information by adding atomic formulas corresponding to *inclusion* and *exclusion* dependencies to the language of first order logic. The properties of these logics and their relationships with other logics of imperfect information are then studied. Furthermore, a game theoretic semantics for these logics is developed. As a corollary of these results, we characterize the expressive power of independence logic, thus answering an open problem posed in (Grädel and Väänänen, 2010).

Keywords: dependence, independence, imperfect information, team semantics, game semantics, model theory
2010 MSC: 03B60,
2010 MSC: 03C80,
2010 MSC: 03C85

1. Introduction

The notions of dependence and independence are among the most fundamental ones considered in logic, in mathematics, and in many of their applications. For example, one of the main aspects in which modern predicate

Email address: pgallian@gmail.com (Pietro Galliani)

logic can be thought of as superior to medieval term logic is that the former allows for quantifier alternation, and hence can express certain complex patterns of dependence and independence between variables that the latter cannot easily represent. A fairly standard example of this can be seen in the formal representations of the notions of *continuity* and *uniform continuity*: in the language of first order logic, the former property can be expressed as $\forall x(\forall \epsilon > 0)(\exists \delta > 0)\forall x'(|x - x'| < \delta \rightarrow |f(x) - f(x')| < \epsilon)$, while the latter can be expressed as $(\forall \epsilon > 0)(\exists \delta > 0)\forall x \forall x'(|x - x'| < \delta \rightarrow |f(x) - f(x')| < \epsilon)$. The difference between these two expressions should be clear: in the first one, the value of the variable δ is a function of the values of the variables x and ϵ, while in the second one it is a function of the value of the variable ϵ alone. This very notion of functional dependence also occurs, at first sight rather independently, as one of the fundamental concepts of Database Theory, and in that context it proved itself to be highly useful both for the specification and study of *normal forms* and for that of *constraints* over databases.[1]

Logics of imperfect information are a family of logical formalisms whose development arose from the observation that not all possible patterns of dependence and independence between variables may be represented in first order logic. Among these logics, *dependence logic* [32] is perhaps the one most suited for the analysis of the notion of dependence itself, since it isolates it by means of *dependence atoms* which correspond, in a very exact sense, to functional dependencies of the exact kind studied in Database Theory. The properties of this logic, and of a number of variants and generalizations thereof, have been the object of much research in recent years, and we cannot hope to give here an exhaustive summary of the known results. We will content ourselves, therefore, to recall (in Subsection 2.1) the ones that will be of particular interest for the rest of this work.

Independence logic [15] is a recent variant of dependence logic. In this new logic, the fundamental concept that is being added to the first order language is not *functional dependence*, as for the case of dependence logic proper, but *informational independence*: as we will see, this is achieved by considering *independence atoms* $y \perp_x z$, whose informal meaning corresponds

[1] We will not discuss these issues in any detail in this work; for a handy reference, we suggest [7] or any other database theory textbook.

to the statement "for any fixed value of x, the sets of the possible values for y and z are independent". Just as dependence logic allows us to reason about the properties of functional dependence, independence logic does the same for this notion. Much is not known at the moment about independence logic; in particular, one open problem mentioned in [15] concerns the *expressive power* of this formalism over open formulas. As we will see, a formula in a logic of imperfect information defines, for any suitable model M, the family of its *trumps*, that is, the family of all sets of assignments (*teams*, in the usual terminology of dependence logic) which satisfy the formula. This differs from the case of first order logic, in which formulas satisfy or do not satisfy single assignments, and the intuitive reason for this should be understandable: asking whether a statement such as "the values of the variables x and y are independent" holds with respect of a single variable assignment is meaningless, since such an assertion can be only interpreted with respect to a family of *possible assignments.* A natural question is then *which* families of sets of possible variable assignments may be represented in terms of independence logic formulas.[2] An upper bound for the answer is in [15] already: all classes of sets of assignments which are definable in independence logic correspond to second order relations which are expressible in existential second order logic. In this work, we will show that this is also a *lower bound*: a class of sets of assignments is definable in independence logic if and only if it is expressible in existential second order logic. This result, which we will prove as Corollary 6.3, implies that independence logic is not merely a formalism obtained by adding an arbitrary, although reasonable-looking, new kind of atomic formula to the first order language. It – and any other formalism equivalent to it – is instead a natural upper bound for a general family of logics of imperfect information: in particular, if over finite models an arbitrary logic of imperfect information characterizes only teams which are in NP then, by Fagin's theorem [10], this logic is (again, over finite models) equivalent to some fragment of independence logic.

The way in which we reach this result is also perhaps of some interest. Even though functional dependence and informational independence are certainly very important notions, they are by no means the only ones of their

[2] The analogous question for dependence logic was answered in [26], and we will report that answer as Theorem 2.5 of the present work.

kind that are of some relevance. In the field of database theory, a great variety of other constraints over relations[3] has indeed been studied. Two of the simplest such constraints are *inclusion dependencies* and *exclusion dependencies*, whose definitions and basic properties we will recall in Subsection 4.1; then, in Subsections 4.2 and 4.4, we will develop and study the corresponding logics.[4] As we will see, "exclusion logic" is equivalent, in a strong sense, to dependence logic, while "inclusion logic" is properly contained in independence logic but incomparable with dependence logic. Then, in Subsection 4.5, we will consider *inclusion/exclusion logic*, that is, the logic obtained by adding atoms for inclusion *and* exclusion dependencies to the language of first order logic, and prove that it is equivalent to independence logic.

Section 5 develops a *game theoretic semantics* for inclusion/exclusion logic. A game-theoretic semantics assigns truth values to expressions according to the properties of certain *semantic games* (often, but not always, in terms of the existence of *winning strategies* for these games). Historically, the first semantics for logics of imperfect information were of this kind; and even though, for many purposes, team semantics is a more useful and clearer formalism, we will see that studying the relationship between game semantics and team semantics allows us to better understand certain properties of the semantic rules for disjunction and existential quantification. Then, in Section 6, we examine the classes of teams definable by inclusion/exclusion logic formulas (or equivalently, by independence logic formulas), and we prove that these are precisely the ones corresponding to second order relations definable in existential second order logic.

Finally, in the last section we show that, as a consequence of this, some of the most general forms of dependency studied in database theory are expressible in independence logic. This, in the opinion of the author, suggests that logics of imperfect information (and, in particular, independence logic) may constitute an useful theoretical framework for the study of such dependencies and their properties.

[3] Such constraints are usually called *dependencies*, for historical reasons; but they need not correspond to anything resembling the informal idea of dependency.

[4] Subsection 4.3 briefly considers the case of *equiextension dependencies* and shows that, for our purposes, they are equivalent to inclusion dependencies.

2. Dependence and independence logic

In this section, we will recall a small number of known results about dependence and independence logic. Some of the basic definitions of these logics will be left unsaid, as they will be later recovered in a slightly more general setting in Subsection 3.1. This section and that subsection, taken together, can be seen as a very quick crash course on the field of logics of imperfect information; the reader who is already familiar with such logics can probably skim through most of it, paying however some attention to the discussion of independence logic of Subsection 2.2, the alternative semantic rules of Definition 3.7 and the subsequent discussion.

2.1. Dependence logic

Dependence logic [32] is, together with IF logic ([17], [31]), one of the most widely studied logics of imperfect information. In brief, it can be described as the extension of first order logic obtained by adding *dependence atoms* $=(t_1 \ldots t_n)$ to its language, with the informal meaning of "The value of the term t_n is functionally determined by the values of the terms $t_1 \ldots t_{n-1}$".

This allows us to express patterns of dependence and independence between variables which are not expressible in first order logic: for example, in the formula $\forall x \exists y \forall z \exists w (=(z, w) \wedge \phi(x, y, z, w))$ the choice of the value for the variable w depends only on the value of the variable w, and not from the values of the variables x and y - or, in other words, this expression is equivalent to the branching quantifier ([16]) sentence

$$\left(\begin{array}{cc} \forall x & \exists y \\ \forall z & \exists w \end{array} \right) \phi(x, y, z, w)$$

and the corresponding Skolem normal form is $\exists f \exists g \forall x \forall z \phi(x, f(x), z, g(z))$.

The idea of allowing more general patterns of dependence and independence between quantifiers than the ones permitted in first order logic was, historically, the main reason for the development of logics of imperfect information: in particular, [17] argues that the restriction on these patterns forced by first order logic has little justification, and that hence logics of imperfect information are a more adequate formalism for reasoning about the foundations of mathematics.

No such claim will be made or discussed in this work. But in any case, the idea of allowing more general patterns of dependence and independence between quantifiers seems a very natural one. In IF logic, the notion of dependence is, however, inherently connected with the notion of quantification: for example, the above expression would be written in it as $\forall x \exists y \forall z (\exists z / x, y) \phi(x, y, z, w)$, where $(\exists z / x, y)$ is to be read as "there exists a z, independent from x and y, such that ...". Dependence logic and its variants, instead, prefer to separate the notion of dependency from the notion of quantification: in this second group of logics of imperfect information, dependence patterns between quantifiers are exactly as first order logic and our linguistic intuitions would suggest, but dependence atoms may be used to specify that the value of a certain variable (or, in general, of a certain term) must be a function of certain other values. This corresponds precisely to the notion of *functional dependence* which is one of the central tools of Database Theory; and indeed, as we will recall later in this work, the satisfaction conditions for these atoms are in a very precise relationship with the formal definition of functional dependence.

This, at least in the opinion of the author, makes dependence logic an eminently suitable formalism for the study of the notion of functional dependence and of its properties; and as we will see, one of the main themes of the present work will consist in the development and study of formalisms which have a similar sort of relationship with other notions of dependency.

We will later recall the full definition of the *team semantics* of dependence logic, an adaptation of Hodges' compositional semantics for IF-logic ([21]) and one of the three equivalent semantics for dependence logic described in [32].[5] It is worth noting already here, though, that the key difference between Hodges semantics and the usual Tarskian semantics is that in the former semantics the satisfaction relation $\models$ associates to every first order model[6] M and formula ϕ a set of *teams*, that is, a set of sets of assignments, instead of just a set of assignments as in the latter one.

As discussed in [22], the fundamental intuition behind Hodges' semantics is that a team is a representation of an *information state* of some agent:

[5] The readers interested in a more thorough explanation of the team semantics and of the two game theoretic semantics for dependence logic are referred to [32] itself.

[6] In all this paper, I will assume that first order models have at least two elements in their domain.

given a model M, a team X and a suitable formula ϕ, the expression

$$M \models_X \phi$$

asserts that, from the information that the "true" assignment s belongs to the team X, it is possible to infer that ϕ holds - or, in game-theoretic terms, that the Verifier has a strategy τ which is winning for all plays of the game $G(\phi)$ which start from any assignment $s \in X$.

The satisfaction conditions for the dependence atom is then given by the following semantic rule **TS-dep**:

Definition 2.1 (Dependence atoms). Let M be a first order model, let X be a team over it, let $n \in \mathbb{N}$, and let $t_1 \ldots t_n$ be terms over the signature of M and with variables in $\mathrm{Dom}(X)$. Then

TS-dep: $M \models_X =(t_1 \ldots t_n)$ if and only if, for all $s, s' \in X$ such that $t_i\langle s\rangle = t_i\langle s'\rangle$ for $i = 1 \ldots n-1$, $t_n\langle s\rangle = t_n\langle s'\rangle$.

This rule corresponds closely to the definition of *functional dependency* commonly used in Database Theory ([6]): more precisely, if $X(t_1 \ldots t_n)$ is the relation $\{(t_1\langle s\rangle, \ldots, t_n\langle s\rangle) : s \in X\}$ then

$$M \models_X =(t_1 \ldots t_n) \Leftrightarrow X(t_1 \ldots t_n) \models \{t_1 \ldots t_{n-1}\} \rightarrow t_n$$

where the right hand expression states that, in the relation $X(t_1 \ldots t_n)$, the value of the last term t_n is a function of the values of $t_1 \ldots t_{n-1}$.

Another formulation of the truth condition of a dependence atom $=(t_1 \ldots t_n)$, easily seen to be equivalent to this one, is the following: a team X satisfies such an atom if and only if a rational agent α, whose beliefs about the identity of the "true" assignment s are described by X, would be capable of inferring the value of t_n from the values of $t_1 \ldots t_{n-1}$.[7] A special case of dependence atom, useful to consider in order to clarify our intuitions, is constituted by *constancy atoms* $=(t)$: applying the above definitions, we can observe that $M \models_X =(t)$ if and only if the value $t\langle s\rangle$ is the same for all assignments $s \in X$ - or, using the agent metaphor, if and only if an agent α

[7] Decomposing the notion further, this is equivalent to stating that if the values of $t_1 \ldots t_{n-1}$ for the true assignment $s \in X$ were announced to the agent then he or she would also learn the value of t_n. The properties of this sort of announcement operators for dependence logic are discussed in [14].

as above *knows* the value of t.[8]

The following known results will be of some use for the rest of this work:

Theorem 2.2 (Locality [32]). *Let M be a first order model and let ϕ be a dependence logic formula over the signature of M with free variables in $\vec{v}$. Then, for all teams X with domain $\vec{w} \supseteq \vec{v}$, if X' is the restriction of X to $\vec{v}$ then*

$$M \models_X \phi \Leftrightarrow M \models_{X'} \phi.$$

As an aside, it is worth pointing out that the above property does not hold for most variants of IF-logic: for example, if $\mathrm{Dom}(M) = \{0, 1\}$ and $X = \{(x : 0, y : 0), (x : 1, y : 1)\}$ it is easy to see that $M \models_X (\exists z/y)z = y$, even though for the restriction X' of X to $\mathrm{Free}((\exists z/y)z = y) = \{y\}$ we have that $M \not\models_{X'} (\exists z/y)z = y$.[9]

Theorem 2.3 (Downwards Closure Property [32]). *Let M be a model, let ϕ be a dependence logic formula over the signature of M, and let X be a team over M with domain $\vec{v} \supseteq$ Free(ϕ) such that $M \models_X \phi$. Then, for all $X' \subseteq X$,*

$$M \models_{X'} \phi.$$

Theorem 2.4 (Dependence logic sentences and Σ_1^1 [32]). *For every dependence logic sentence ϕ, there exists a Σ_1^1 sentence Φ such that*

$$M \models_{\{\emptyset\}} \phi \Leftrightarrow M \models \Phi.$$

Conversely, for every Σ_1^1 sentence Φ there exists a dependence logic sentence ϕ such that the above holds.

Theorem 2.5 (Dependence logic formulas and Σ_1^1 [26]). *For every dependence logic formula ϕ and every tuple of variables $\vec{x} \supseteq$ Free(ϕ) there exists a Σ_1^1 sentence $\Phi(R)$, where R is a $|\vec{x}|$-ary relation which occurs only negatively in Φ, such that*

$$M \models_X \phi \Leftrightarrow M \models \Phi(\mathit{Rel}(X))$$

[8] The existence of a relation between these notions and the ones studied in the field of epistemic modal logic is clear, but to the knowledge of the author the matter has not yet been explored in full detail. See [34] for some intriguing reflections about this topic.

[9] This is a typical example of *signalling* ([17], [23]), one of the most peculiar and, perhaps, problematic aspects of IF-logic.

for all teams X with domain $\vec{x}$.[10]

Conversely, for all such Σ_1^1 sentences there exists a dependence logic formula ϕ such that the above holds with respect to all nonempty *teams X.*

2.2. Independence logic

Independence logic [15] is a recently developed logic which substitutes the dependence atoms of dependence logic with *independence atoms* $\vec{t}_2 \perp_{\vec{t}_1} \vec{t}_3$, where $\vec{t}_1 \ldots \vec{t}_3$ are tuples of terms (not necessarily of the same length).

The intuitive meaning of such an atom is that the values of the tuples $\vec{t}_2$ and $\vec{t}_3$ are informationally independent for any fixed value of $\vec{t}_1$; or, in other words, that all information about the value of $\vec{t}_3$ that can be possibly inferred from the values of $\vec{t}_1$ and $\vec{t}_2$ can be already inferred from the value of $\vec{t}_1$ alone.

More formally, the definition of the team semantics for the independence atom is as follows:

Definition 2.6 (Independence atoms). Let M be a first order model, let X be a team over it and let $\vec{t}_1, \vec{t}_2$ and $\vec{t}_3$ be three finite tuples of terms (not necessarily of the same length) over the signature of M and with variables in $\mathrm{Dom}(X)$. Then

TS-indep: $M \models_X \vec{t}_2 \perp_{\vec{t}_1} \vec{t}_3$ if and only if for all $s, s' \in X$ with $\vec{t}_1\langle s\rangle = \vec{t}_1\langle s\rangle$ there exists a $s'' \in X$ such that $\vec{t}_1\langle s''\rangle\vec{t}_2\langle s''\rangle = \vec{t}_1\langle s\rangle\vec{t}_2\langle s\rangle$ and $\vec{t}_1\langle s''\rangle\vec{t}_3\langle s''\rangle = \vec{t}_1\langle s'\rangle\vec{t}_3\langle s'\rangle$.

We refer to [15] for a discussion of this interesting class of atomic formulas and of the resulting logic. Here we only mention a few results, found in that paper, which will be useful for the rest of this work:[11]

Theorem 2.7. *Dependence atoms are expressible in terms of independence atoms: more precisely, for all suitable models M, teams X and terms $t_1 \ldots t_n$*

$$M \models_X =\!(t_1 \ldots t_n) \Leftrightarrow M \models_X t_n \perp_{t_1 \ldots t_{n-1}} t_n.$$

[10] Here $\mathrm{Rel}(X)$ is the relation corresponding to the team X, as in Definition 3.2.

[11] Another interesting result about independence logic, pointed out by Fredrik Engström in [9], is that the semantic rule for independence atoms corresponds to that of *embedded multivalued dependencies*, in the same sense in which the one for dependence atoms corresponds to *functional* ones.

Theorem 2.8. *Independence logic is equivalent to Σ_1^1 (and therefore, by Theorem 2.4, to dependence logic) over sentences: in other words, for every sentence ϕ of independence logic there exists a sentence Φ of existential second order logic such that*

$$M \models_{\{\emptyset\}} \phi \Leftrightarrow M \models \Phi.$$

and for every such Φ there exists a ϕ such that the above holds.

There is no analogue of Theorem 2.3 for independence logic, however, as the classes of teams corresponding to independence atoms are not necessarily downwards closed: for example, according Definition 2.6 the formula $x \perp_{\emptyset} y$ holds in the team

$$\{(x : 0, y : 0), (x : 0, y : 1), (x : 1, y : 0), (x : 1, y : 1)\}$$

but not in its subteam $\{(x : 0, y : 0), (x : 1, y : 1)\}$.

The problem of of finding a characterization similar to that of Theorem 2.5 for the classes of teams definable by formulas of independence logic was left open by Grädel and Väänänen, who concluded their paper by stating that ([15])

> *The main open question raised by the above discussion is the following, formulated for finite structures:*
>
> **Open Problem:** *Characterize the NP properties of teams that correspond to formulas of independence logic.*

In this paper, an answer to this question will be given, as a corollary of an analogous result for a new logic of imperfect information.

3. Team semantics

In this section, we will introduce some of the main concepts that we will need for the rest of this work and then we will test them on a relatively simple case. Subsection 3.1 contains the basic definitions of team semantics, following for the most part the treatment of [32]; and furthermore, in this subsection we introduce two variant rules for disjunction and existential quantification which, as we will later see, will be of significant relevance.

Then, in Subsection 3.2, we will begin our investigations by examining *constancy logic*, that is, the fragment of dependence logic obtained by adding constancy atoms to the language of first order logic. The main result of that subsection will be a proof that constancy logic is expressively equivalent to first order logic over sentences, and, hence, that it is strictly less expressive than the full dependence logic. This particular consequence is a special case of the far-reaching *hierarchy theorem* of [8], which fully characterizes the expressive powers of certain fragments of dependence logic.

3.1. First order (team) logic, in two flavors

In this subsection, we will present and briefly discuss the team semantics for first order logic, laying the groundwork for reasoning about its extensions while avoiding, as far as we are able to do so, all forms of semantical ambiguity.

As we will see, some special care is required here, since certain rules which are equivalent with respect to dependence logic proper will not be so with respect to these new logics. As it often is the case for logics of imperfect information, the game theoretic approach to semantics (which we will discuss in Section 5) will be of support and clarification for our intuitions concerning the intended interpretations of operators.

But let us begin by recalling some basic definitions from [32]:

Definition 3.1 (Team). Let M be a first order model, and let $\vec{v}$ be a tuple of variables.[12] Then a *team* X for M with *domain* $\vec{v}$ is simply a set of assignments with domain $\vec{v}$ over M.

Definition 3.2 (From teams to relations). Let M be a first order model, X be a team for M with domain $\vec{v}$, and let $\vec{t} = t_1 \ldots t_k$ be a tuple of terms with variables in $\vec{v}$. Then we write $X(\vec{t})$ for the relation

$$X(\vec{t}) = \{(t_1\langle s\rangle \ldots t_k\langle s\rangle) : s \in X\}.$$

[12] Or, equivalently, a *set* of variables; but having a fixed ordering of the variables as part of the definition of team will simplify the definition of the correspondence between teams and relations. With an abuse of notation, we will identify this tuple of variables with the underlying set whenever it is expedient to do so.

Furthermore, if $\vec{w}$ is contained in $\vec{v}$ we will write $\mathrm{Rel}_{\vec{w}}(X)$ for $X(\vec{w})$; and, finally, if $\mathrm{Dom}(X) = \vec{v}$ we will write $\mathrm{Rel}(X)$ for $\mathrm{Rel}_{\vec{v}}(X)$.

Definition 3.3 (Team restrictions). Let X be any team in any model, and let V be a set of variables contained in $\mathrm{Dom}(X)$. Then

$$X_{\upharpoonright V} = \{s_{\upharpoonright V} : s \in X\}$$

where $s_{\upharpoonright V}$ is the restriction of s to V, that is, the only assignment s' with domain V such that $s'(v) = s(v)$ for all $v \in V$.

The team semantics for the first order fragment of dependence logic is then defined as follows:

Definition 3.4 (Team semantics for first order logic ([21], [32])). Let M be a first order model, let ϕ be a first order formula in negation normal form[13] and let X be a team over M with domain $\vec{v} \supseteq \mathrm{Free}(\phi)$. Then

TS-atom: If ϕ is a first order literal, $M \models_X \phi$ if and only if, for all assignments $s \in X$, $M \models_s \phi$ in the usual first order sense;

TS-$\vee_L$: If ϕ is $\psi \vee \theta$, $M \models_X \phi$ if and only if there exist two teams Y and Z such that $X = Y \cup Z$, $M \models_Y \psi$ and $M \models_Z \theta$;

TS-$\wedge$: If ϕ is $\psi \wedge \theta$, $M \models_X \phi$ if and only if $M \models_X \psi$ and $M \models_X \theta$;

TS-$\exists_S$: If ϕ is $\exists x \psi$, $M \models_X \phi$ if and only if there exists a function $F : X \to \mathrm{Dom}(M)$ such that $M \models_{X[F/x]} \psi$, where

$$X[F/x] = \{s[F(s)/x] : s \in X\};$$ [14]

[13] Since the negation is not a semantic operation in dependence logic ([2], [27]), it is useful to assume that all formulas are in negation normal form. It is of course possible to adapt these definitions to formulas not in negation normal form, but in order to do so for the cases of dependence or independence logic it would be necessary to define two distinct relationships $\models^+$ and $\models^-$, as in [32]. Since, for the purposes of this work, this would offer no significant advantage and would complicate the definitions, it was chosen to avoid the issue by requiring all formulas to be in negation normal form instead.

[14] Sometimes, we will write $X[F_1F_2 \ldots F_n/x_1 \ldots x_n]$, or even $X[\vec{F}/\vec{x}]$, as a shorthand for $X[F_1/x_1][F_2/x_2] \ldots [F_n/x_n]$.

TS-$\forall$: If ϕ is $\forall x \psi$, $M \models_X \phi$ if and only if $M \models_{X[M/x]} \psi$, where

$$X[M/x] = \{s[m/x] : s \in X\}.[15]$$

Over singleton teams, this semantics coincides with the usual one for first order logic:

Proposition 3.5 ([32]). *Let M be a first order model, let ϕ be a first order formula in negation normal form over the signature of M, and let s be an assignment with $Dom(s) \supseteq Free(\phi)$. Then $M \models_{\{s\}} \phi$ if and only if $M \models_s \phi$ with respect to the usual Tarski semantics for first order logic.*

Furthermore, as the following proposition illustrates, the team semantics of first order logic is compatible with the intuition, discussed before, that teams represent states of knowledge:

Proposition 3.6 ([32]). *Let M be a first order model, let ϕ be a first order formula in negation normal form over the signature of M, and let X be a team with $Dom(X) \supseteq Free(\phi)$. Then $M \models_X \phi$ if and only if, for all assignments $s \in X$, $M \models_{\{s\}} \phi$.*[16]

On the other hand, these two proposition also show that, for first order logic, all the above machinery is quite unnecessary. We have no need of carrying around such complex objects as teams, since we can consider any assignment in a team individually!

Things, however, change if we add dependence atoms $=\!(t_1 \ldots t_n)$ to our language, with the semantics of rule **TS-dep** (Definition 2.1 here). In the resulting formalism, which is precisely *dependence logic* as defined in [32], not all satisfaction conditions over teams can be reduced to satisfaction conditions over assignments: for example, a "constancy atom" $=\!(x)$ holds in a team X if and only if $s(x) = s'(x)$ for all $s, s' \in X$, and verifying this condition clearly requires to check *pairs* of assignments at least![17]

[15] Sometimes, we will write $X[M/x_1 x_2 \ldots x_n]$, or even $X[M/\vec{x}]$, as a shorthand for $X[M/x_1][M/x_2] \ldots [M/x_n]$.

[16] In other words, first order formulas are *flat* in the sense of [32].

[17] That is, all constancy atoms - and, more in general, all dependence atoms - are 2-*coherent* but not 1-*coherent* in the sense of [24].

When studying variants of dependence logic, similarly, it is necessary to keep in mind that semantic rules which are equivalent with respect to dependence logic proper may not be equivalent with respect to these new formalisms. In particular, two alternative definitions of disjunction and existential quantification exist which are of special interest for this work's purposes:[18]

Definition 3.7 (Alternative rules for disjunctions and existentials). Let M, X, ϕ, ψ and θ be as usual. Then

TS-$\vee_S$: If ϕ is $\psi \vee \theta$, $M \models_X \phi$ if and only if there exist two teams Y and Z such that $X = Y \cup Z$, $Y \cap Z = \emptyset$, $M \models_Y \psi$ and $M \models_Z \theta$;

TS-$\exists_L$: If ϕ is $\exists x \psi$, $M \models_X \phi$ if and only if there exists a function $H : X \to \mathcal{P}(\mathrm{Dom}(M)) \backslash \emptyset$ such that $M \models_{X[H/x]} \psi$, where

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

The subscripts of $\cdot_S$ and $\cdot_L$ of these rules and of the corresponding ones of Definition 3.4 allow us to discriminate between the *lax* operators $\vee_L$ and $\exists_L$ and the *strict* ones $\vee_S$ and $\exists_S$. This distinction will be formally justified in Section 5, and in particular by Theorems 5.10 and 5.11; but even at a glance, this grouping of the rules is justified by the fact that **TS-$\vee_S$** and **TS-$\exists_S$** appear to be stronger conditions than **TS-$\vee_L$** and **TS-$\exists_L$**. We can then define two alternative semantics for first order logic (and for its extensions, of course) as follows:

Definition 3.8 (Lax semantics). The relation $M \models^L_X \phi$, where M ranges over all first order models, X ranges over all teams and ϕ ranges over all formulas with free variables in $\mathrm{Dom}(X)$, is defined as the relation $M \models_X \phi$ of Definition 3.4 (with additional rules for further atomic formulas as required), but substituting Rule **TS-$\exists_S$** with Rule **TS-$\exists_L$**.

Definition 3.9 (Strict semantics). The relation $M \models^S_X \phi$, where M ranges over all first order models, X ranges over all teams and ϕ ranges over all formulas with free variables in $\mathrm{Dom}(X)$, is defined as the relation $M \models_X \phi$ of

[18] The rule **TS-$\exists_L$** is also discussed in [9], in which it is shown that it arises naturally from treating the existential quantifier as a *generalized quantifier* ([30], [28]) for dependence logic.

Definition 3.4 (with additional rules for further atomic formulas as required), but substituting Rule **TS-**$\vee_L$ with Rule **TS-**$\vee_S$.

For the cases of first order and dependence logic, the lax and strict semantics are equivalent:

Proposition 3.10. *Let ϕ be any formula of dependence logic. Then*

$$M \models_X^S \phi \Leftrightarrow M \models_X^L \phi$$

for all suitable models M and teams ϕ.

Proof. This is easily verified by structural induction over ϕ, using the downwards closure property (Theorem 2.3) to take care of disjunctions and existentials (and, moreover, applying the Axiom of Choice for the case of existentials). We verify the case corresponding to existential quantifications, as an example: the one corresponding to disjunctions is similar but simpler, and the the others are trivial.

Suppose that $M \models_X^S \exists x \phi$: then, by rule **TS-**$\exists_S$, there exists a function $F : X \to \mathrm{Dom}(M)$ such that $M \models_{X[F/x]}^S \phi$. Now define the function $H : X \to \mathcal{P}(\mathrm{Dom}(M))\backslash\{\emptyset\}$ so that, for all $s \in X$, $H(s) = \{F(s)\}$: then $X[H/x] = X[F/x]$, and therefore by induction hypothesis $M \models_{X[H/x]}^L \phi$, and hence by rule **TS-**$\exists_L$ $M \models_X^L \exists x \phi$. Conversely, suppose that $M \models_X^L \exists x \phi$: then, by rule **TS-**$\exists_L$, there exists a function $H : X \to \mathcal{P}(\mathrm{Dom}(M))\backslash\{\emptyset\}$ such that $M \models_{X[H/x]}^L \phi$. Then, by the Axiom of Choice, there exists a *choice function* $F : X \to \mathrm{Dom}(X)$ such that, for all $s \in X$, $F(s) \in H(s)$; therefore, $X[F/x] \subseteq X[H/x]$ and, by downwards closure, $M \models_{X[F/x]}^L \phi$. But then by induction hypothesis $M \models_{X[F/x]}^S \phi$ and, by rule **TS-**$\exists_L$, $M \models_X^S \phi$.

As we will argue in Section 4.2, for the logics that we will study for which a difference exists between lax and strict semantics the former will be the most natural choice; therefore, from this point until the end of this work the symbol $\models$ written without superscripts will stand for the relation $\models^L$.

3.2. Constancy logic

In this section, we will present and examine a simple fragment of dependence logic. This fragment, which we will call *constancy logic*, consists of all the formulas of dependence logic in which only dependence atoms of the form

$=(t)$ occur; or, equivalently, it can be defined as the extension of (team) first order logic obtained by adding *constancy atoms* to it, with the semantics given by the following definition:

Definition 3.11 (Constancy atoms). Let M be a first order model, let X be a team over it, and let t be a term over the signature of M and with variables in $\mathrm{Dom}(X)$. Then

TS-const: $M \models_X =(t)$ if and only if, for all $s, s' \in X$, $t\langle s\rangle = t\langle s'\rangle$.

Clearly, constancy logic is contained in dependence logic. Furthermore, over open formulas it is more expressive than first order logic proper, since, as already mentioned, the constancy atom $= (x)$ is a counterexample to Proposition 3.6.

The question then arises whether constancy logic is properly contained in dependence logic, or if it coincides with it. This will be answered through the following results:

Proposition 3.12. *Let ϕ be a constancy logic formula, let z be a variable not occurring in ϕ, and let ϕ' be obtained from ϕ by substituting one instance of $=(t)$ with the expression $z = t$.*

Then

$$M \models_X \phi \Leftrightarrow M \models_X \exists z(=(z) \wedge \phi').$$

Proof. The proof is by induction on ϕ.

1. If the expression $=(t)$ does not occur in ϕ, then $\phi' = \phi$ and we trivially have that $\phi \equiv \exists z(=(z) \wedge \phi)$, as required.
2. If ϕ is $=(t)$ itself then ϕ' is $z = t$, and

   $$M \models_X \exists z(=(z) \wedge z = t) \Leftrightarrow \exists m \in \mathrm{Dom}(M) \text{ s.t. } M \models_{X[m/z]} z = t \Leftrightarrow$$
   $$\Leftrightarrow \exists m \in \mathrm{Dom}(M) \text{ s.t. } t\langle s\rangle = m \text{ for all } s \in X \Leftrightarrow M \models_X =(t)$$

   as required, where we used $X[m/z]$ as a shorthand for $\{s(m/z) : s \in X\}$.
3. If ϕ is $\psi_1 \vee \psi_2$, let us assume without loss of generality that the instance of $=(t)$ that we are considering is in ψ_1. Then $\psi_2' = \psi_2$, and since z

does not occur in ψ_2

$$\begin{aligned}
&M \models_X \exists z(=(z) \wedge (\psi_1' \vee \psi_2)) \Leftrightarrow \exists m \text{ s.t. } M \models_{X[m/z]} \psi_1' \vee \psi_2 \Leftrightarrow\\
&\Leftrightarrow \exists m, X_1, X_2 \text{ s.t. } X_1 \cup X_2 = X, M \models_{X_1[m/z]} \psi_1' \text{ and } M \models_{X_2[m/z]} \psi_2 \Leftrightarrow\\
&\Leftrightarrow \exists m, X_1, X_2 \text{ s.t. } X_1 \cup X_2 = X, M \models_{X_1[m/z]} \psi_1' \text{ and } M \models_{X_2} \psi_2 \Leftrightarrow\\
&\Leftrightarrow X_1, X_2 \text{ s.t. } X_1 \cup X_2 = X, M \models_{X_1} \exists z(=(z) \wedge \psi_1') \text{ and } M \models_{X_2} \psi_2 \Leftrightarrow\\
&\Leftrightarrow X_1, X_2 \text{ s.t. } X_1 \cup X_2 = X, M \models_{X_1} \psi_1 \text{ and } M \models_{X_2} \psi_2 \Leftrightarrow\\
&\Leftrightarrow M \models_X \psi_1 \vee \psi_2
\end{aligned}$$

as required.

4. If ϕ is $\psi_1 \wedge \psi_2$, let us assume again that the instance of $=(t)$ that we are considering is in ψ_1. Then $\psi_2' = \psi_2$, and

$$\begin{aligned}
&M \models_X \exists z(=(z) \wedge \psi_1' \wedge \psi_2) \Leftrightarrow\\
&\Leftrightarrow \exists m \text{ s.t. } M \models_{X[m/z]} \psi_1' \text{ and } M \models_{X[m/z]} \psi_2 \Leftrightarrow\\
&\Leftrightarrow M \models_X \exists z(=(z) \wedge \psi_1') \text{ and } M \models_X \psi_2 \Leftrightarrow\\
&\Leftrightarrow M \models_X \psi_1 \text{ and } M \models_X \psi_2 \Leftrightarrow\\
&\Leftrightarrow M \models_X \psi_1 \wedge \psi_2.
\end{aligned}$$

5. If ϕ is $\exists x \psi$,

$$\begin{aligned}
&M \models_X \exists z(=(z) \wedge \exists x \psi') \Leftrightarrow\\
&\Leftrightarrow \exists m \text{ s.t. } M \models_{X[m/z]} \exists x \psi' \Leftrightarrow\\
&\Leftrightarrow \exists m, \exists H : X[m/z] \to \mathcal{P}(\mathrm{Dom}(M))\backslash\{\emptyset\} \text{ s.t. } M \models_{X[m/z][H/x]} \psi' \Leftrightarrow\\
&\Leftrightarrow \exists H' : X \to \mathcal{P}(\mathrm{Dom}(M))\backslash\{\emptyset\}, \exists m \text{ s.t. } M \models_{X[H'/x][m/z]} \psi' \Leftrightarrow\\
&\Leftrightarrow \exists H' : X \to \mathcal{P}(\mathrm{Dom}(M))\backslash\{\emptyset\} \text{ s.t. } M \models_{X[H'/x]} \exists z(=(z) \wedge \psi') \Leftrightarrow\\
&\Leftrightarrow \exists H' : X \to \mathcal{P}(\mathrm{Dom}(M))\backslash\{\emptyset\}, \text{ s.t. } M \models_{X[H'/x]} \psi \Leftrightarrow\\
&\Leftrightarrow M \models_X \exists x \psi.
\end{aligned}$$

6. If ϕ is $\forall x \psi$,

$$\begin{aligned}
&M \models_X \exists z(=(z) \wedge \forall x \psi') \Leftrightarrow\\
&\Leftrightarrow \exists m \text{ s.t. } M \models_{X[m/z]} \forall x \psi' \Leftrightarrow\\
&\Leftrightarrow \exists m \text{ s.t. } M \models_{X[m/z][M/x]} \psi' \Leftrightarrow\\
&\Leftrightarrow \exists m \text{ s.t. } M \models_{X[M/x][m/z]} \psi' \Leftrightarrow\\
&\Leftrightarrow M \models_{X[M/x]} \exists z(=(z) \wedge \psi') \Leftrightarrow\\
&\Leftrightarrow M \models_{X[M/x]} \psi \Leftrightarrow\\
&\Leftrightarrow M \models_X \forall x \psi.
\end{aligned}$$

As a corollary of this result, we get the following normal form theorem for constancy logic:[19]

Corollary 3.13. *Let ϕ be a constancy logic formula. Then ϕ is logically equivalent to a constancy logic formula of the form*

$$\exists z_1 \ldots z_n \left(\bigwedge_{i=1}^{n} =\!(z_i) \wedge \psi(z_1 \ldots z_n) \right)$$

for some tuple of variables $\vec{z} = z_1 \ldots z_n$ and some first order formula ψ.

Proof. Repeatedly apply Proposition 3.12 to "push out" all constancy atoms from ϕ, thus obtaining a formula, equivalent to it, of the form

$$\exists z_1(=\!(z_1) \wedge \exists z_2(=\!(z_2) \wedge \ldots \wedge \exists z_n(=\!(z_n) \wedge \psi(z_1 \ldots z_n)))$$

for some first order formula $\psi(z_1 \ldots z_n)$. It is then easy to see, from the semantics of our logic, that this is equivalent to

$$\exists z_1 \ldots z_n(=\!(z_1) \wedge \ldots \wedge =\!(z_n) \wedge \psi(z_1 \ldots z_n))$$

as required.

The following result shows that, over sentences, constancy logic is precisely as expressive as first order logic:

Corollary 3.14. *Let $\phi = \exists \vec{z}\, (\bigwedge_i =\!(z_i) \wedge \psi(\vec{z}))$ be a constancy logic sentence in normal form.*

Then ϕ is logically equivalent to $\exists \vec{z} \psi(\vec{z})$.

Proof. By the rules of our semantics, $M \models_{\{\emptyset\}} \psi$ if and only if there exists a family $A_1 \ldots A_n$ of nonempty sets of elements in $\mathrm{Dom}(M)$ such that, for

$$X = \{(z_1 := m_1 \ldots z_n := m_n) : (m_1 \ldots m_n) \in A_1 \times \ldots \times A_n\},$$

[19] This normal form theorem is very similar to the one of dependence logic proper found in [32]. See also [8] for a similar, but not identical result, developed independently, which Arnaud Durand and Juha Kontinen use in that paper in order to characterize the expressive powers of subclasses of dependence logic in terms of the maximum allowed width of their dependence atoms.

it holds that $M \models_X \psi$. But ψ is first-order, and therefore, by Proposition 3.6, this is the case if and only if for all $m_1 \in A_1, \ldots, m_n \in A_n$ it holds that $M \models_{\{(z_1:m_1, \ldots z_n:m_n)\}} \psi$.

But then $M \models_{\{\emptyset\}} \phi$ is and only if there exist $m_1 \ldots m_n$ such that this holds;[20] and therefore, by Proposition 3.5, $M \models_{\{\emptyset\}} \phi$ if and only if $M \models_{\emptyset} \exists z_1 \ldots z_n \psi(z_1 \ldots z_n)$ according to Tarski's semantics, or equivalently, if and only if $M \models_{\{\emptyset\}} \exists z_1 \ldots z_n \psi(z_1 \ldots z_n)$ according to team semantics.

Since, by Theorem 2.4, dependence logic is strictly stronger than first order logic over sentences, this implies that constancy logic is strictly weaker than dependence logic over sentences (and, since sentences are a particular kind of formulas, over formulas too).

The relation between first order logic and constancy logic, in conclusion, appears somewhat similar to that between dependence logic and independence logic - that is, in both cases we have a pair of logics which are reciprocally translatable on the level of sentences, but such that one of them is strictly weaker than the other on the level of formulas. This discrepancy between translatability on the level of sentences and translatability on the level of formulas is, in the opinion of the author, one of the most intriguing aspects of logics of imperfect information, and it deserves further investigation.

4. Inclusion and exclusion in logic

This section is the central part of the present work. We will begin it by recalling two forms of non-functional dependency which have been studied in Database Theory, and some of their known properties. Then we will briefly discuss their relevance in the framework of logics of imperfect information, and then, in Subsection 4.2, we will examine the properties of the logic obtained by adding atoms corresponding to the first sort of non-functional dependency to the basic language of team semantics. Afterward, in Subsection 4.3 we will see that nothing is lost if we only consider a simpler variant of this kind of dependency: in either case, we obtain the same logical formalism, which - as we will see - is strictly more expressive than first order logic,

[20] Indeed, if this is the case we can just choose $A_1 = \{m_1\}, \ldots, A_n = \{m_n\}$, and conversely if $A_1 \ldots A_n$ exist with the required properties we can simply select arbitrary elements of them for $m_1 \ldots m_n$.

strictly weaker than independence logic, but incomparable with dependence logic. In Subsection 4.4, we will then study the other notion of non-functional dependency that we are considering, and see that the corresponding logic is instead equivalent, in a very strong sense, to dependence logic; and finally, in Subsection 4.5 we will examine the logic obtained by adding *both* forms of non-functional dependency to our language, and see that it is equivalent to independence logic.

4.1. Inclusion and exclusion dependencies

Functional dependencies are the forms of dependency which attracted the most interest from database theorists, but they certainly are not the only ones ever considered in that field.

Therefore, studying the effect of substituting the dependence atoms with ones corresponding to other forms of dependency, and examining the relationship between the resulting logics, may be - in the author's opinion, at least - a very promising, and hitherto not sufficiently explored, direction of research in the field of logics of imperfect information.[21] First of all, as previously mentioned, teams correspond to states of knowledge. But often, relations obtained from a database correspond precisely to information states of this kind;[22] and therefore, some of the dependencies studied in database theory may correspond to constraints over the agent's beliefs which often occur in practice, as is certainly the case for functional dependencies.[23]

Moreover, and perhaps more pragmatically, database researchers have already performed a vast amount of research about the properties of many of these non-functional dependencies, and it does not seem unreasonable to

[21] Apart from the present paper, [9], which introduces *multivalued dependence atoms*, is also a step in this direction. The resulting "multivalued dependence logic" is easily seen to be equivalent to independence logic.

[22] As a somewhat naive example, let us consider the problem of finding a spy, knowing that yesterday he took a plane from London's Heathrow airport and that he had at most 100 EUR available to buy his plane ticket. We might then decide to obtain, from the airport systems, the list of the destinations of all the planes which left Heathrow yesterday and whose ticket the spy could have afforded; and this list - that is, the list of all the places that the spy might have reached - would be a state of information of the kind which we are discussing.

[23] For example, our system should be able to represent the assertion that the flight code always determines the destination of the flight.

hope that this might allow us to derive, with little additional effort of our own, some useful results about the corresponding logics.

The present paper will, for the most part, focus on *inclusion* ([11], [3]) and *exclusion* ([4]) dependencies and on the properties of the corresponding logics of imperfect information. Let us start by recalling and briefly discussing these dependencies:

Definition 4.1 (Inclusion Dependencies). Let R be a relation, and let $\vec{x}$, $\vec{y}$ be tuples of attributes of R of the same length. Then $R \models \vec{x} \subseteq \vec{y}$ if and only if $R(\vec{x}) \subseteq R(\vec{y})$, where

$$R(\vec{z}) = \{r(\vec{z}) : r \text{ is a tuple in } R\}.$$

In other words, an inclusion dependency $\vec{x} \subseteq \vec{y}$ states that all values taken by the attributes $\vec{x}$ are also taken by the attributes $\vec{y}$. It is easy to think up practical examples of inclusion dependencies: one might for instance think of the database consisting of the relations (Person, Date_of_Birth), (Father, Children$_F$) and (Mother, Children$_M$).[24] Then, in order to express the statement that every father, every mother and every child in our knowledge base are people and have a date of birth, we may impose the restrictions

$$\left\{ \begin{array}{l} \text{Father} \subseteq \text{Person},\ \text{Mother} \subseteq \text{Person}, \\ \text{Children}_F \subseteq \text{Person},\ \text{Children}_M \subseteq \text{Person} \end{array} \right\}.$$

Furthermore, inclusion dependencies can be used to represent the assertion that every child has a father and a mother, or, in other words, that the attributes Children$_F$ and Children$_M$ take the same values:

$$\{\text{Children}_F \subseteq \text{Children}_M,\ \text{Children}_M \subseteq \text{Children}_F\}.$$

Note, however, that inclusion dependencies do not allow us to express all "natural" dependencies of our example. For instance, we need to use functional dependencies in order to assert that everyone has exactly one birth

[24] Equivalently, one may consider the Cartesian product of these relations, as per the universal relation model ([12]).

date, one father and one mother:[25]

$$\{\text{Person} \to \text{Date\_of\_Birth},\ \text{Children}_F \to \text{Father},\ \text{Children}_M \to \text{Mother}\}.$$

In [3], a sound and complete axiom system for the implication problem of inclusion dependencies was developed. This system consists of the three following rules:

I1: For all $\vec{x}$, $\vdash \vec{x} \subseteq \vec{x}$;

I2: If $|\vec{x}| = |\vec{y}| = n$ then, for all $m \in \mathbb{N}$ and all $\pi : 1 \ldots m \to 1 \ldots n$,

$$\vec{x} \subseteq \vec{y} \vdash x_{\pi(1)} \ldots x_{\pi(m)} \subseteq y_{\pi(1)} \ldots y_{\pi(m)};$$

I3: For all tuples of attributes of the same length $\vec{x}$, $\vec{y}$, and $\vec{z}$,

$$\vec{x} \subseteq \vec{y}, \vec{y} \subseteq \vec{z} \vdash \vec{x} \subseteq \vec{z}.$$

Theorem 4.2 (Soundness and completeness of inclusion axioms [3]). *Let Γ be a set of inclusion dependencies and let $\vec{x}$, $\vec{y}$ be tuples of relations of the same length. Then*

$$\Gamma \vdash \vec{x} \subseteq \vec{y}$$

*can be derived from the axioms **I1**, **I2** and **I3** if and only if all relations which respect all dependencies of Γ also respect $\vec{x} \subseteq \vec{y}$.*

However, the combined implication problem for inclusion and functional dependencies is undecidable ([29], [5]).

Whereas inclusion dependencies state that all values of a given tuple of attributes also occur as values of another tuple of attributes, *exclusion* dependencies state that two tuples of attributes have no values in common:

[25] The simplest way to verify that these conditions are not expressible in terms of inclusion dependencies is probably to observe that inclusion dependencies are *closed under unions*: if the relations R and S respect $\vec{x} \subseteq \vec{y}$, so does $R \cup S$. Since functional dependencies as the above ones are clearly *not* closed under unions, they cannot be represented by inclusions.

Definition 4.3 (Exclusion dependencies). Let R be a relation, and let $\vec{x}$, $\vec{y}$ be tuples of attributes of R of the same length. Then $R \models \vec{x} \mid \vec{y}$ if and only if $R(\vec{x}) \cap R(\vec{y}) = \emptyset$, where

$$R(\vec{z}) = \{r(\vec{z}) : r \text{ is a tuple in } R\}.$$

Exclusion dependencies can be thought of as a way of partitioning the elements of our domain into *data types*, and of specifying which type corresponds to each attribute. For instance, in the example

$$(\text{Person, Date\_of\_birth}) \times (\text{Father, Children}_F) \times (\text{Mother, Children}_M)$$

considered above we have two types, corresponding respectively to *people* (for the attributes Person, Father, Mother, Children$_F$ and Children$_M$) and *dates* (for the attribute Date_of_birth). The requirement that no date of birth should be accepted as a name of person, nor vice versa, can then be expressed by the set of exclusion dependencies

$$\{A \mid \text{Date\_of\_birth} : A = \text{Person}, \text{Father}, \text{Mother}, \text{Children}_M, \text{Children}_F\}.$$

Other uses of exclusion dependencies are less common, but they still exist: for example, the statement that no one is both a father and a mother might be expressed as Father | Mother.

In [4], the axiom system for inclusion dependencies was extended to deal with both inclusion and exclusion dependencies as follows:

1. *Axioms for inclusion dependencies:*

   I1: For all $\vec{x}$, $\vdash \vec{x} \subseteq \vec{x}$;

   I2: If $|\vec{x}| = |\vec{y}| = n$ then, for all $m \in \mathbb{N}$ and all $\pi : 1 \ldots m \to 1 \ldots n$,

   $$\vec{x} \subseteq \vec{y} \vdash x_{\pi(1)} \ldots x_{\pi(m)} \subseteq y_{\pi(1)} \ldots y_{\pi(m)};$$

   I3: For all tuples of attributes of the same length $\vec{x}$, $\vec{y}$ and $\vec{z}$,

   $$\vec{x} \subseteq \vec{y}, \vec{y} \subseteq \vec{z} \vdash \vec{x} \subseteq \vec{z};$$

2. *Axioms for exclusion dependencies:*

   E1: For all $\vec{x}$ and $\vec{y}$ of the same length, $\vec{x} \mid \vec{y} \vdash \vec{y} \mid \vec{x}$;

E2: If $|\vec{x}| = |\vec{y}| = n$ then, for all $m \in \mathbb{N}$ and all $\pi : 1 \dots m \to 1 \dots n$,

$$x_{\pi(1)} \dots x_{\pi(m)} \mid y_{\pi(1)} \dots y_{\pi(m)} \vdash \vec{x} \mid \vec{y};$$

E3: For all $\vec{x}$, $\vec{y}$ and $\vec{z}$ such that $|\vec{y}| = |\vec{z}|$, $\vec{x} \mid \vec{x} \vdash \vec{y} \mid \vec{z}$;

3. *Axioms for inclusion/exclusion interaction:*

IE1: For all $\vec{x}$, $\vec{y}$ and $\vec{z}$ such that $|\vec{y}| = |\vec{z}|$, $\vec{x} \mid \vec{x} \vdash \vec{y} \subseteq \vec{z}$;

IE2: For all $\vec{x}, \vec{y}, \vec{z}, \vec{w}$ of the same length, $\vec{x} \mid \vec{y}, \vec{z} \subseteq \vec{x}, \vec{w} \subseteq \vec{y} \vdash \vec{z} \mid \vec{w}$.

Theorem 4.4 ([4]). *The above system is sound and complete for the implication problem for inclusion and exclusion dependencies.*

It is not difficult to transfer the definitions of inclusion and exclusion dependencies to team semantics, thus obtaining *inclusion atoms* and *exclusion atoms*:

Definition 4.5 (Inclusion and exclusion atoms). Let M be a first order model, let $\vec{t_1}$ and $\vec{t_2}$ be two finite tuples of terms of the same length over the signature of M, and let X be a team whose domain contains all variables occurring in $\vec{t_1}$ and $\vec{t_2}$. Then

TS-inc: $M \models_X \vec{t_1} \subseteq \vec{t_2}$ if and only if for every $s \in X$ there exists a $s' \in X$ such that $\vec{t_1}\langle s \rangle = \vec{t_2}\langle s' \rangle$;

TS-exc: $M \models_X \vec{t_1} \mid \vec{t_2}$ if and only if for all $s, s' \in X$, $\vec{t_1}\langle s \rangle \neq \vec{t_2}\langle s' \rangle$.

Returning for a moment to the agent metaphor, the interpretation of these conditions is as follows.

A team X satisfies $\vec{t_1} \subseteq \vec{t_2}$ if and only if all possible values that the agent believes possible for $\vec{t_1}$ are also believed by him or her as possible for $\vec{t_2}$ - or, by contraposition, that the agent cannot exclude any value for $\vec{t_2}$ which he cannot also exclude as a possible value for $\vec{t_1}$. In other words, from this point of view an inclusion atom is a way of specify a state of *ignorance* of the agent: for example, if the agent is a chess player who is participating to a tournament, we may want to represent the assertion that the agent *does not know* whether he will play against a given opponent using the black pieces or the white ones. In other words, if he believes that he *might* play against a given opponent when using the white pieces, he should also consider it possible that he played against him or her using the black ones, and vice versa; or, in our formalism, that his belief set satisfies the conditions

$$\text{Opponent\_as\_White} \subseteq \text{Opponent\_as\_Black},$$
$$\text{Opponent\_as\_Black} \subseteq \text{Opponent\_as\_White}.$$

This very example can be used to introduce a new dependency atom $\vec{t_1} \bowtie \vec{t_2}$, which might perhaps be called an *equiextension atom*, with the following rule:

Definition 4.6 (Equiextension atoms). Let M be a first order model, let $\vec{t_1}$ and $\vec{t_2}$ be two finite tuples of terms of the same length over the signature of M, and let X be a team whose domain contains all variables occurring in $\vec{t_1}$ and $\vec{t_2}$. Then

TS-equ: $M \models_X \vec{t_1} \bowtie \vec{t_2}$ if and only if $X(\vec{t_1}) = X(\vec{t_2})$.

It is easy to see that this atom is different, and strictly weaker, from the first order formula

$$\vec{t_1} = \vec{t_2} := \bigwedge_i ((\vec{t_1})_i = (\vec{t_2})_i).$$

Indeed, the former only requires that the sets of all possible values for $\vec{t_1}$ and for $\vec{t_2}$ are the same, while the latter requires that $\vec{t_1}$ and $\vec{t_2}$ coincide in all possible states of things: and hence, for example, the team $X = \{(x : 0, y : 1), (x : 1, y : 0)\}$ satisfies $x \bowtie y$ but not $x = y$.

As we will see later, it is possible to recover inclusion atoms from equiextension atoms and the connectives of our logics.

On the other hand, an exclusion atom specifies a state of *knowledge*. More precisely, a team X satisfies $\vec{t_1} \mid \vec{t_2}$ if and only if the agent can confidently exclude all values that he believes possible for $\vec{t_1}$ from the list of the possible values for $\vec{t_2}$. For example, let us suppose that our agent is also aware that a boxing match will be had at the same time of the chess tournament, and that he knows that no one of the participants to the match will have the time to play in the tournament too - he has seen the lists of the participants to the two events, and they are disjoint. Then, in particular, our agent knows that no potential winner of the boxing match is also a potential winner of the chess tournament, even know he is not aware of who these winners will be. In our framework, this can be represented by stating our agent's beliefs respect the exclusion atom

$$\text{Winner\_Boxing} \mid \text{Winner\_Chess}.$$

This is a different, and stronger, condition than the first order expression Winner_Boxing $\neq$ Winner_Chess: indeed, the latter merely requires that, in any possible state of things, the winners of the boxing match and of the chess tournament are different, while the former requires that *no possible* winner of the boxing match is a potential winner for the chess tournament. So, for example, only the first condition excludes the scenario in which our agent does not know whether T. Dovramadjiev, a Bulgarian chessboxing champion, will play in the chess tournament or in the boxing match, represented by the team of the form

$$X = \begin{array}{c|lll} & \text{Winner\_Boxing} & \text{Winner\_Chess} & \ldots \\ \hline s_0 & \text{T. Dovramadjiev} & \text{V. Anand} & \ldots \\ s_1 & \text{T. Woolgar} & \text{T. Dovramadijev} & \ldots \\ \ldots & \ldots & \ldots & \end{array}$$

4.2. Inclusion logic

In this section, we will begin to examine the properties of *inclusion logic* - that is, the logic obtained adding to (team) first order logic the *inclusion atoms* $\vec{t}_1 \subseteq \vec{t}_2$ with the semantics of Definition 4.5.

A first, easy observation is that this logic does not respect the downwards closure property. For example, consider the two assignments $s_0 = (x : 0, y : 1)$ and $s_1 = (x : 1, y : 0)$: then, for $X = \{s_0, s_1\}$ and $Y = \{s_0\}$, it is easy to see by rule **TS-inc** that $M \models_X x \subseteq y$ but $M \not\models_Y x \subseteq y$.

Hence, the proof of Proposition 3.10 cannot be adapted to the case of inclusion logic. The question then arises whether inclusion logic with strict semantics and inclusion logic with lax semantics are different; and, as the next two propositions will show, this is indeed the case.

Proposition 4.7. *There exist a model M, a team X and two formulas ψ and θ of inclusion logic such that $M \models^L_X \psi \vee \theta$ but $M \not\models^S_X \psi \vee \theta$.*

Proof. Let $\mathrm{Dom}(M) = \{0, 1, 2, 3, 4\}$, let X be the team

$X =$

	x	y	z
s_0	0	1	2
s_1	1	0	3
s_2	4	3	0

and let $\psi = x \subseteq y$, $\theta = y \subseteq z$.

- $M \models^L_X \psi \vee \theta$:
  Let $Y = \{s_0, s_1\}$ and $Z = \{s_1, s_2\}$. Then $Y \cup Z = X$, $Y(x) = \{0, 1\} = Y(y)$ and $Z(y) = \{0, 3\} = Z(z)$.

  Hence, $M \models^L_Y x \subseteq y$ and $M \models^L_Z y \subseteq z$, and therefore $M \models^L_X x \subseteq y \vee y \subseteq z$ as required.

- $M \not\models^S_X \psi \vee \theta$:
  Suppose that $X = Y \cup Z$, $Y \cap Z = \emptyset$, $M \models^S_X x \subseteq y$ and $M \models^S_Z y \subseteq z$.

  Now, s_2 cannot belong in Y, since $s_2(x) = 4$ and $s_i(y) \neq 4$ for all assignments s_i; therefore, we necessarily have that $s_2 \in Z$. But since $M \models^S_Z y \subseteq z$, this implies that there exists an assignment $s_i \in Y$ such that $s_i(z) = s_2(y) = 3$. The only such assignment in X is s_1, and therefore $s_1 \in Y$.

  Analogously, s_0 cannot belong in Z: indeed, $s_0(y) = 1 \neq s_i(z)$ for all $i \in 0 \ldots 2$. Therefore, $s_0 \in Y$; and since $M \models^S_Y x \subseteq y$, there exists an $s_i \in Y$ with $s_i(y) = s_0(x) = 0$. But the only such assignment in X is s_1, and therefore $s_1 \in Y$.

  In conclusion, $Y = \{s_0, s_1\}$, $Z = \{s_1, s_2\}$ and $Y \cap Z = \{s_1\} \neq \emptyset$, which contradicts our hypothesis.

Proposition 4.8. *There exist a model M, a team X and a formula ϕ of inclusion logic such that $M \models^L_X \exists x \phi$ but $M \not\models^S_X \exists x \phi$.*

Proof. Let $\mathrm{Dom}(M) = \{0, 1\}$, let X be the team

$X =$

	y	z
s_0	0	1

and let ϕ be $y \subseteq x \wedge z \subseteq x$.

- $M \models^L_X \exists x\phi$:
  Let $H : X \to \mathcal{P}(\mathrm{Dom}(M))$ be such that $H(s_0) = \{0, 1\}$.

  Then
  $$X[H/x] = \begin{array}{c|ccc} & y & z & x \\ \hline s'_0 & 0 & 1 & 0 \\ s'_1 & 0 & 1 & 1 \end{array}$$
  and hence $X[H/x](y), X[H/x](z) \subseteq X[H/x](x)$, as required.

- $M \not\models^S_X \exists x\psi$:
  Let F be any function from X to $\mathrm{Dom}(M)$. Then
  $$X[F/x] = \begin{array}{c|ccc} & y & z & x \\ \hline s''_0 & 0 & 1 & F(s_0) \end{array}$$
  But $F(s_0) \neq 0$ or $F(s_0) \neq 1$; and in the first case $M \not\models^S_{X[F/x]} y \subseteq x$, while in the second one $M \not\models^S_{X[F/x]} z \subseteq x$.

Therefore, when studying the properties inclusion logic it is necessary to specify whether we are are using the strict or the lax semantics for disjunction and existential quantification. However, only one of these choices preserves *locality* in the sense of Theorem 2.2, as the two following results show:

Proposition 4.9. *The strict semantics does not respect locality in inclusion logic (or in any extension thereof). In other words, there exists a a model M, a team X and two formulas ψ and θ such that $M \models^S_X \psi \vee \theta$, but for $X' = X_{\upharpoonright Free(\phi \vee \psi)}$ it holds that $M \not\models^S_{X'} \psi \vee \theta$ instead; and analogously, there exists a model M, a team X and a formula ξ such that $M \models^S_X \exists x\xi$, but for $X' = X_{\upharpoonright Free(\exists x\xi)}$ we have that $M \not\models^S_{X'} \exists\xi$ instead.*

Proof. 1. Let $\mathrm{Dom}(M) = \{0 \ldots 4\}$, let ψ and θ be $x \subseteq y$ and $y \subseteq z$ respectively, and let
$$X = \begin{array}{c|cccc} & x & y & z & u \\ \hline s_0 & 0 & 1 & 2 & 0 \\ s_1 & 1 & 0 & 3 & 0 \\ s_2 & 1 & 0 & 3 & 1 \\ s_3 & 4 & 3 & 0 & 0 \end{array}$$

Then $M \models^S_X \psi \vee \theta$: indeed, for $Y = \{s_0, s_1\}$ and $Z = \{s_2, s_3\}$ we have that $X = Y \cup Z$, $Y \cap Z = \emptyset$, $M \models_Y \psi$ and $M \models_Z \theta$, as required. However, the restriction X' of X to $\text{Free}(\psi \vee \theta) = \{x, y, z\}$ is the team considered in the proof of Proposition 4.7, and - as was shown in that proof - $M \not\models^S_X \psi \vee \theta$.

2. Let $\text{Dom}(M) = \{0, 1\}$, let ξ be $y \subseteq x \wedge z \subseteq x$, and let

$$X = \begin{array}{c|ccc} & y & z & u \\ \hline s_0 & 0 & 1 & 0 \\ s_1 & 0 & 1 & 1 \end{array}$$

Then $M \models^S_X \exists x \xi$: indeed, for $F : X \to \text{Dom}(M)$ defined as

$$F(s_0) = 0;$$
$$F(s_1) = 1;$$

we have that

$$X[F/x] = \begin{array}{c|cccc} & y & z & u & x \\ \hline s'_0 & 0 & 1 & 0 & 0 \\ s'_1 & 0 & 1 & 1 & 1 \end{array}$$

and it is easy to check that this team satisfies ξ. However, the restriction X' of X to $\text{Free}(\exists x \xi) = \{y, z\}$ is the team considered in the proof of Proposition 4.8, and - again, as shown in that proof - $M \not\models^S_X \exists x \psi$.

Theorem 4.10 (Inclusion logic with lax semantics is local). *Let M be a first order model, let ϕ be any inclusion logic formula, and let V be a set of variables with Free(ϕ) $\subseteq V$. Then, for all suitable teams X,*

$$M \models^L_X \phi \Leftrightarrow M \models^L_{X_{\upharpoonright V}} \phi$$

Proof. The proof is by structural induction on ϕ.

In Section 4.5, Theorem 4.24, we will prove the same result for an extension of inclusion logic; so we refer to that theorem for the details of the proof.

Since, as we saw, inclusion logic is not downwards closed, by Theorem 2.3 it is not contained in dependence logic. It is then natural to ask whether dependence logic is contained in inclusion logic, or if dependence and inclusion logic are two incomparable extensions of first order logic.

This is answered by the following result, and by its corollary:

Theorem 4.11 (Union closure for inclusion logic). *Let ϕ be any inclusion logic formula, let M be a first order model and let $(X_i)_{i \in I}$ be a family of teams with the same domain such that $M \models_{X_i} \phi$ for all $i \in I$. Then, for $X = \bigcup_{i \in I} X_i$, we have that $M \models_X \phi$.*

Proof. By structural induction on ϕ.

1. If ϕ is a first order literal, this is obvious.
2. Suppose that $M \models_{X_i} \vec{t_1} \subseteq \vec{t_2}$ for all $i \in I$. Then $M \models_X \vec{t_1} \subseteq \vec{t_2}$. Indeed, let $s \in X$: then $s \in X_i$ for some $i \in I$, and hence there exists another $s' \in X_i$ with $s'(\vec{t_2}) = s(\vec{t_1})$. Since $X_i \subseteq X$ we then have that $s' \in X$, as required.
3. Suppose that $M \models_{X_i} \psi \vee \theta$ for all $i \in I$. Then each X_i can be split into two subteams Y_i and Z_i with $M \models_{Y_i} \psi$ and $M \models_{Z_i} \theta$. Now, let $Y = \bigcup_{i \in I} Y_i$ and $Z = \bigcup_{i \in I} Z_i$: by induction hypothesis, $M \models_Y \psi$ and $M \models_Z \theta$. Furthermore, $Y \cup Z = \bigcup_{i \in I} Y_i \cup \bigcup_{i \in I} Z_i = \bigcup_{i \in I}(Y_i \cup Z_i) = X$, and hence $M \models_X \psi \vee \theta$, as required.
4. Suppose that $M \models_{X_i} \psi \wedge \theta$ for all $i \in I$. Then for all such i, $M \models_{X_i} \psi$ and $M \models_{X_i} \theta$; but then, by induction hypothesis, $M \models_X \psi$ and $M \models_X \theta$, and therefore $M \models_X \psi \wedge \theta$.
5. Suppose that $M \models_{X_i} \exists x \psi$ for all $i \in I$, that is, that for all such i there exists a function $H_i : X_i \to \mathcal{P}(\mathrm{Dom}(M)) \backslash \{\emptyset\}$ such that $M \models_{X_i[H_i/x]} \psi$. Then define the function $H : X \to \mathcal{P}(\mathrm{Dom}(M)) \backslash \{\emptyset\}$ so that, for all $s \in X$, $H(s) = \bigcup \{H_i(s) : s \in X_i\}$. Now, $X[H/x] = \bigcup_{i \in I}(X_i[H_i/x])$, and hence by induction hypothesis $M \models_{X[H/x]} \psi$, and therefore $M \models_X \exists x \psi$.
6. Suppose that $M \models_{X_i} \forall x \psi$ for all $i \in I$, that is, that $M \models_{X_i[M/x]} \psi$ for all such i. Then, since $\bigcup_{i \in I}(X_i[M/x]) = \left(\bigcup_{i \in I} X_i\right)[M/x] = X[M/x]$, by induction hypothesis $M \models_{X[M/x]} \psi$ and therefore $M \models_X \forall x \psi$, as required.

Corollary 4.12. *There exist constancy logic formulas which are not equivalent to any inclusion logic formula.*

Proof. This follows at once from the fact that the constancy atom $=\!(x)$ is not closed under unions.

Indeed, let M be any model with two elements 0 and 1 in its domain, and consider the two teams $X_0 = \{(x : 0)\}$ and $X_1 = \{(x : 1)\}$: then $M \models_{X_0} =\!(x)$ and $M \models_{X_1} =\!(x)$, but $M \not\models_{X_0 \cup X_1} =\!(x)$.

Therefore, not only inclusion logic does not contain dependence logic, it does not even contain constancy logic!

Now, by Theorem 2.7 we know that dependence logic is properly contained in independence logic. As the following result shows, inclusion logic is also (properly, because dependence atoms are expressible in independence logic) contained in independence logic:

Theorem 4.13. *Inclusion atoms are expressible in terms of independence logic formulas. More precisely, an inclusion atom $\vec{t_1} \subseteq \vec{t_2}$ is equivalent to the independence logic formula*

$$\phi := \forall v_1 v_2 \vec{z}((\vec{z} \neq \vec{t_1} \wedge \vec{z} \neq \vec{t_2}) \vee (v_1 \neq v_2 \wedge \vec{z} \neq \vec{t_2}) \vee ((v_1 = v_2 \vee \vec{z} = \vec{t_2}) \wedge \vec{z} \perp v_1 v_2)).$$

where v_1, v_2 and $\vec{z}$ do not occur in $\vec{t_1}$ or $\vec{t_2}$ and where, as in [15], $\vec{z} \perp v_1 v_2$ is a shorthand for $\vec{z} \perp_\emptyset v_1 v_2$.

Proof. Suppose that $M \models_X \vec{t_1} \subseteq \vec{t_2}$. Then split the team $X' = X[M/v_1 v_2 \vec{z}]$ into three teams Y, Z and W as follows:

- $Y = \{s \in X' : s(\vec{z}) \neq \vec{t_1}\langle s\rangle$ and $s(\vec{z}) \neq \vec{t_2}\langle s\rangle\}$;
- $Z = \{s \in X' : s(v_1) \neq s(v_2)$ and $s(\vec{z}) \neq \vec{t_2}\langle s\rangle\}$;
- $W = X' \backslash (Y \cup Z) = \{s \in X' : s(\vec{z}) = \vec{t_2}\langle s\rangle$ or $(s(\vec{z}) = \vec{t_1}\langle s\rangle$ and $s(v_1) = s(v_2))\}$.

Clearly, $X' = Y \cup Z \cup W$, $M \models_Y z \neq t_1 \wedge z \neq t_2$ and $M \models_Z v_1 \neq v_2 \wedge z \neq t_2$; hence, if we can prove that

$$M \models_W ((v_1 = v_2 \vee \vec{z} = \vec{t_2})) \wedge \vec{z} \perp v_1 v_2$$

we can conclude that $M \models_X \phi$, as required.

Now, suppose that $s \in W$ and $s(v_1) \neq s(v_2)$: then necessarily $s(\vec{z}) = \vec{t}_2$, since otherwise we would have that $s \in Z$ instead. Hence, the first conjunct $v_1 = v_2 \vee \vec{z} = \vec{t}_2$ is satisfied by W.

Now, consider two assignments s and s' in W: in order to conclude this direction of the proof, we need to show that there exists a $s'' \in W$ such that $s''(\vec{z}) = s(\vec{z})$ and $s''(v_1 v_2) = s'(v_1 v_2)$. There are two distinct cases to examine:

1. If $s(\vec{z}) = \vec{t}_2\langle s \rangle$, consider the assignment
$$s'' = s[s'(v_1)/v_1][s'(v_2)/v_2] :$$
by construction, $s'' \in X'$. Furthermore, since $s''(\vec{z}) = \vec{t}_2\langle s \rangle = \vec{t}_2\langle s'' \rangle$, s'' is neither in Y nor in Z. Hence, it is in W, as required.
2. If $s(\vec{z}) \neq \vec{t}_2\langle s \rangle$ and $s \in W$, then necessarily $s(\vec{z}) = \vec{t}_1\langle s \rangle$ and $s(v_1) = s(v_2)$.
Since $s \in W \subseteq X[M/v_1 v_2 \vec{z}]$, there exists an assignment $o \in X$ such that
$$\vec{t}_1\langle o \rangle = \vec{t}_1\langle s \rangle = s(\vec{z});$$
and since $M \models_X \vec{t}_1 \subseteq \vec{t}_2$, there also exist an assignment $o' \in X$ such that
$$\vec{t}_2\langle o' \rangle = \vec{t}_1\langle o \rangle = s(\vec{z}).$$
Now consider the assignment $s'' = o'[s'(v_1)/v_1][s'(v_2)/v_2][s(\vec{z})/\vec{z}]$: by construction, $s'' \in X'$, and since
$$s''(\vec{z}) = s(\vec{z}) = \vec{t}_2\langle o' \rangle = \vec{t}_2\langle s'' \rangle$$
we have that $s'' \in W$, that $s''(\vec{z}) = s(\vec{z})$ and that $s''(v_1 v_2) = s'(v_1 v_2)$, as required.

Conversely, suppose that $M \models_X \phi$, let 0 and 1 be two distinct elements of the domain of M, and let $s \in X$.

By the definition of ϕ, the fact that $M \models_X \phi$ implies that the team $X[M/v_1 v_2 \vec{z}]$ can be split into three teams Y, Z and W such that

$$\begin{aligned}
&M \models_Y \vec{z} \neq \vec{t}_1 \wedge \vec{z} \neq \vec{t}_2;\\
&M \models_Z v_1 \neq v_2 \wedge \vec{z} \neq \vec{t}_2;\\
&M \models_W (v_1 = v_2 \vee \vec{z} = \vec{t}_2) \wedge \vec{z} \perp v_1 v_2.
\end{aligned}$$

Then consider the assignments

$$h = s[0/v_1][0/v_2][\vec{t}_1\langle s\rangle/\vec{z}]$$

and

$$h' = s[0/v_1][1/v_2][\vec{t}_2\langle s\rangle/\vec{z}]$$

Clearly, h and h' are in $X[M/v_1v_2\vec{z}]$. However, neither of them is in Y, since $h(\vec{z}) = \vec{t}_1\langle h\rangle$ and $h'(\vec{z}) = \vec{t}_2\langle h'\rangle$, nor in Z, since $h(v_1) = h(v_2)$ and, again, since $h'(\vec{z}) = \vec{t}_2\langle h'\rangle$. Hence, both of them are in W.

But we know that $M \models_W \vec{z} \perp v_1v_2$, and thus there exists an assignment $h'' \in W$ with

$$h''(\vec{z}) = h(\vec{z}) = \vec{t}_1\langle s\rangle$$

and

$$h''(v_1v_2) = h'(v_1v_2) = 01.$$

Now, since $h''(v_1) \neq h''(v_2)$, since $h'' \in W$ and since

$$M \models_W v_1 = v_2 \vee \vec{z} = \vec{t}_2,$$

it must be the case that $h''(\vec{z}) = \vec{t}_2\langle h''\rangle$.

Finally, this h'' corresponds to some $s'' \in X$; and for this s'',

$$\vec{t}_2\langle s''\rangle = \vec{t}_2\langle h''\rangle = h''(\vec{z}) = h(\vec{z}) = \vec{t}_1\langle s\rangle.$$

This concludes the proof.

The relations between first order (team) logic, constancy logic, dependence logic, inclusion logic and independence logic discovered so far are then represented by Figure 1.

However, things change if we take in consideration the the expressive power of these logics with respect to their sentences only. Then, as we saw, first order logic and constancy logic have the same expressive power, in the sense that every constancy logic formula is equivalent to some first order formula and vice versa, and so do dependence and independence logic. What about inclusion logic sentences?

At the moment, relatively little is known by the author about this. In essence, all that we know is the following result:

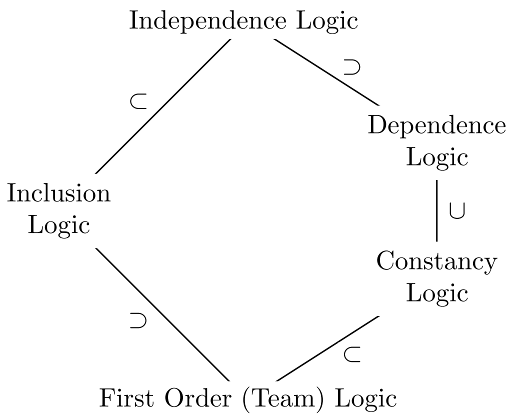

Figure 1: Translatability relations between logics (wrt formulas)

Proposition 4.14. *Let $\psi(\vec{x}, \vec{y})$ be any first order formula, where $\vec{x}$ and $\vec{y}$ are tuples of disjoint variables of the same arity. Furthermore, let $\psi'(\vec{x}, \vec{y})$ be the result of writing $\neg\psi(\vec{x}, \vec{y})$ in negation normal form. Then, for all suitable models M and all suitable pairs $\vec{a}$, $\vec{b}$ of constant terms of the model,*

$$M \models_{\{\emptyset\}} \exists \vec{z}(\vec{a} \subseteq \vec{z} \wedge \vec{z} \neq \vec{b} \wedge \forall \vec{w}(\psi'(\vec{z}, \vec{w}) \vee \vec{w} \subseteq \vec{z}))$$

if and only if $M \models \neg[TC_{\vec{x},\vec{y}}\ \psi](\vec{a}, \vec{b})$, that is, if and only if the pair of tuples of elements corresponding to $(\vec{a}, \vec{b})$ is not in the transitive closure of $\{(\vec{m}_1, \vec{m}_2) : M \models \psi(\vec{m}_1, \vec{m}_2)\}$.

Proof. Suppose that $M \models_{\{\emptyset\}} \exists \vec{z}(\vec{a} \subseteq \vec{z} \wedge \vec{z} \neq \vec{b} \wedge \forall \vec{w}(\psi'(\vec{z}, \vec{w}) \vee \vec{w} \subseteq \vec{z}))$. Then, by definition, there exists a tuple of functions $\vec{H} = H_1 \ldots H_n$ such that

1. $M \models_{\{\emptyset\}[\vec{H}/\vec{z}]} \vec{a} \subseteq \vec{z}$, that is, $\vec{a} \in \vec{H}(\{\emptyset\})$;
2. $M \models_{\{\emptyset\}[\vec{H}/\vec{z}]} \vec{z} \neq \vec{b}$, and therefore $\vec{b} \not\in \vec{H}(\{\emptyset\})$;
3. $M \models_{\{\emptyset\}[\vec{H}/\vec{z}][\vec{M}/\vec{w}]} \psi'(\vec{z}, \vec{w}) \vee \vec{w} \subseteq \vec{z}$.

Now, the third condition implies that whenever $M \models \psi(\vec{m}_1, \vec{m}_2)$ and $\vec{m}_1$ is in $\vec{H}(\{\emptyset\})$, $\vec{m}_2$ is in $\vec{H}(\{\emptyset\})$ too. Indeed, let $Y = \{\emptyset\}[\vec{H}/\vec{z}][\vec{M}/\vec{w}]$: then, by the semantics of our logic, we know that $Y = Y_1 \cup Y_2$ for two subteams Y_1 and Y_2 such that $M \models_{Y_1} \psi'(\vec{z}, \vec{w})$ and $M \models_{Y_2} \vec{w} \subseteq \vec{z}$. But ψ' is logically equivalent to the negation of ψ, and therefore we know that, for all $s \in Y_1$,

$M \not\models \psi(s(\vec{z}), s(\vec{w}))$ in the usual Tarskian semantics.

Suppose now that $\vec{m}_1 \in \vec{H}(\{\emptyset\})$ and that $M \models \psi(\vec{m}_1, \vec{m}_2)$. Then $s = (\vec{z} := \vec{m}_1, \vec{w} := \vec{m}_2)$ is in Y; but it cannot be in Y_1, as we saw, and hence it must belong to Y_2. But $M \models_{Y_2} \vec{w} \subseteq \vec{z}$, and therefore there exists another assignment $s' \in Y_2$ such that $s'(\vec{z}) = s(\vec{w}) = \vec{m}_2$. But we necessarily have that $s'(\vec{z}) \in \vec{H}(\{\emptyset\})$, and therefore $\vec{m}_2 \in \vec{H}(\{\emptyset\})$, as required.

So, $\vec{H}(\{\emptyset\})$ is an set of tuples of elements of our models which contains the interpretation of $\vec{a}$ but not that of $\vec{b}$ and such that

$$\vec{m}_1 \in H(\{\emptyset\}), M \models \psi(\vec{m}_1), \vec{M}_2 \Rightarrow \vec{m}_2 \in H(\{\emptyset\}).$$

This implies that $M \models \neg[\mathrm{TC}_{\vec{x},\vec{y}}\ \psi](\vec{a}, \vec{b})$, as required.

Conversely, suppose that $M \models \neg[\mathrm{TC}_{\vec{x},\vec{y}}\ \psi](\vec{a}, \vec{b})$: then there exists a set A of tuples of elements of the domain of M which contains the interpretation of $\vec{a}$ but not that of $\vec{b}$, and such that it is closed by transitive closure for $\psi(\vec{x}, \vec{y})$. Then, by choosing the functions $\vec{H}$ so that $\vec{h}(\{\emptyset\}) = A$, it is easy to verify that M satisfies our inclusion logic sentence.

As a corollary, we have that inclusion logic is strictly more expressive than first order logic over sentences: for example, for all finite linear orders $M = (\mathrm{Dom}(M), <, S, 0, e)$, where S is the successor function, 0 is the first element of the linear order and e is the last one, we have that

$$M \models \exists z(0 \subseteq z \wedge z \neq e \wedge \forall w(w \neq S(S(z)) \vee w \subseteq z))$$

if and only if $|M|$ is odd. It is not difficult to see, for example through the Ehrenfeucht-Fraïssé method ([20]), that this property is not expressible in first order logic.

4.3. Equiextension logic

Let us now consider *equiextension logic*, that is, the logic obtained by adding to first order logic (with the lax team semantics) equiextension atoms $\vec{t}_1 \bowtie \vec{t}_2$ with the semantics of Definition 4.6.

It is easy to see that equiextension logic is contained in inclusion logic:

Proposition 4.15. *Let $\vec{t}_1$ and $\vec{t}_2$ be any two tuples of terms of the same length. Then, for all suitable models M and teams X,*

$$M \models_X \vec{t}_1 \bowtie \vec{t}_2 \Leftrightarrow M \models_X \vec{t}_1 \subseteq \vec{t}_2 \wedge \vec{t}_2 \subseteq \vec{t}_1.$$

Proof. Obvious.

Translating in the other direction, however, requires a little more care:

Proposition 4.16. *Let $\vec{t}_1$ and $\vec{t}_2$ be any two tuples of terms of the same length. Then, for all suitable models M and teams X, $M \models_X \vec{t}_1 \subseteq \vec{t}_2$ if and only if*

$$M \models_X \forall u_1 u_2 \exists \vec{z}(\vec{t}_2 \bowtie \vec{z} \wedge (u_1 \neq u_2 \vee \vec{z} = \vec{t}_1))$$

where u_1, u_2 and $\vec{z}$ do not occur in $\vec{t}_1$ and $\vec{t}_2$.

Proof. Suppose that $M \models_X \vec{t}_1 \subseteq \vec{t}_2$. Then let $X' = X[M/u_1 u_2]$, and pick the tuple of functions $\vec{H}$ used to choose $\vec{z}$ so that

$$\vec{H}(s) = \begin{cases} \{\vec{t}_1\langle s\rangle\}, & \text{if } s(\vec{u}_1) = s(\vec{u}_2); \\ \{\vec{t}_2\langle s\rangle\}, & \text{otherwise} \end{cases}$$

for all $s \in X'$.[26]

Then, for $Y = X'[\vec{H}/\vec{z}]$, by definition we have that $M \models_Y u_1 \neq u_2 \vee \vec{z} = \vec{t}_1$, and it only remains to verify that $M \models_Y \vec{t}_2 \bowtie \vec{z}$, that is, that $Y(\vec{t}_2) = Y(\vec{z})$.

- $Y(\vec{t}_2) \subseteq Y(\vec{z})$:
Let $h \in Y$. Then there exists an assignment $s \in X$ with $\vec{t}_2\langle s\rangle = \vec{t}_2\langle h\rangle$. Now let 0 and 1 be two distinct elements of M, and consider the assignment $h' = s[0/u_1][1/u_2][\vec{H}/\vec{z}]$. By construction, $h' \in Y$; and furthermore, by the definition of $\vec{H}$ we have that $h'(\vec{z}) = \vec{t}_2\langle s\rangle = \vec{t}_2\langle h\rangle$, as required.

- $Y(\vec{z}) \subseteq Y(\vec{t}_2)$:
Let $h \in Y$. Then, by construction, $h(\vec{z})$ is $\vec{t}_1\langle h\rangle$ or $\vec{t}_2\langle h\rangle$. But since $X(\vec{t}_1) \subseteq X(\vec{t}_2)$, in either case there exists an assignment $s \in X$ such $\vec{t}_2\langle s\rangle = h(\vec{z})$. Now consider $h' = s[0/u_1][1/u_2][\vec{H}/\vec{z}]$: again, $h' \in Y$ and $h'(\vec{z}) = \vec{t}_2\langle h'\rangle = \vec{t}_2\langle s\rangle = h(\vec{z})$, as required.

[26] As an aside, it can be observed that, since $\vec{H}$ always selects singletons, this whole argument can be adapted to the case of strict semantics without any difficulties. Therefore, strict equiextension logic is equivalent to strict inclusion logic and, by Proposition 4.9, does not satisfy locality either.

Conversely, suppose that $M \models_X \forall u_1 u_2 \exists \vec{z}(\vec{t}_2 \bowtie \vec{z} \wedge (u_1 \neq u_2 \vee \vec{z} = \vec{t}_1))$, and that therefore there exists a tuple of functions $\vec{H}$ such that, for $Y = X[M/u_1u_2][\vec{H}/\vec{z}]$, $M \models_Y \vec{t}_2 \bowtie \vec{z} \wedge (u_1 \neq u_2 \vee \vec{z} = \vec{t}_1)$. Then consider any assignment $s \in X$, and let $h = s[0/u_1][0/u_2][\vec{H}/\vec{z}]$. Now, $h \in Y$ and $h(\vec{z}) = \vec{t}_1\langle s\rangle$; but since $M \models_Y \vec{t}_2 \bowtie \vec{z}$, this implies that there exists an assignment $h' \in Y$ such that $\vec{t}_2\langle h'\rangle = h(\vec{z}) = \vec{t}_1\langle s\rangle$. Finally, h' derives from some assignment $s' \in X$, and for this assignment we have that $\vec{t}_2\langle s\rangle = \vec{t}_2\langle h'\rangle = \vec{t}_1\langle s\rangle$ as required.

As a consequence, inclusion logic is precisely as expressive as equiextension logic:

Corollary 4.17. *Any formula of inclusion logic is equivalent to some formula of equiextension logic, and vice versa.*

4.4. Exclusion logic

With the name of *exclusion logic* we refer to (lax, team) first order logic supplemented with the *exclusion atoms* $\vec{t}_1 \mid \vec{t}_2$, with the satisfaction condition given in Definition 4.5.

As the following results show exclusion logic is, in a very strong sense, equivalent to dependence logic:

Theorem 4.18. *For all tuples of terms $\vec{t}_1$ and $\vec{t}_2$, of the same length, there exists a dependence logic formula ϕ such that*

$$M \models_X \phi \Leftrightarrow M \models_X \vec{t}_1 \mid \vec{t}_2$$

for all suitable models M and teams X.

Proof. This follows immediately from Theorem 2.5, since the satisfaction condition for the exclusion atom is downwards monotone and expressible in Σ^1_1.

For the sake of completeness, let us write a direct translation of exclusion atoms into dependence logic anyway.

Let $\vec{t}_1$ and $\vec{t}_2$ be as in our hypothesis, let $\vec{z}$ be a tuple of new variables, of the same length of $\vec{t}_1$ and $\vec{t}_2$, and let u_1, u_2 be two further unused variables. Then $M \models_X \vec{t}_1 \mid \vec{t}_2$ if and only if

$$M \models_X \forall \vec{z} \exists u_1 u_2 (=(\vec{z}, u_1) \wedge =(\vec{z}, u_2) \wedge ((u_1 = u_2 \wedge \vec{z} \neq \vec{t}_1) \vee (u_1 \neq u_2 \wedge \vec{z} \neq \vec{t}_2))).$$

Indeed, suppose that $M \models_X \vec{t}_1 \mid \vec{t}_2$, let $X' = X[M/\vec{z}]$, and let $0, 1$ be two distinct elements in $\mathrm{Dom}(M)$.

Then define the functions H_1 and H_2 as follows:

- For all $s' \in X'$, $H_1(s') = \{0\}$;
- For all $s'' \in X'[H_1/u_1]$, $H_2(s'') = \begin{cases} \{0\} & \text{if } s''(\vec{z}) \not\in X(\vec{t}_1); \\ \{1\} & \text{if } s''(\vec{z}) \in X(\vec{t}_1). \end{cases}$

Then, for $Y = X'[H_1H_2/u_1u_2]$, we have that $M \models_Y =\!(\vec{z}, u_1)$ and that $M \models_Y =\!(\vec{z}, u_2)$, since the value of u_1 is constant in Y and the value of u_2 in Y is functionally determined by the value of $\vec{z}$.

Now split Y into the two subteams Y_1 and Y_2 defined as

$$Y_1 = \{s \in Y : s(u_2) = 0\};$$
$$Y_2 = \{s \in Y : s(u_2) = 1\}.$$

Clearly, $M \models_{Y_1} u_1 = u_2$ and $M \models_{Y_2} u_1 \neq u_2$; hence, we only need to verify that $M \models_{Y_1} \vec{z} \neq \vec{t}_1$ and that $M \models_{Y_2} \vec{z} \neq \vec{t}_2$.

For the first case, let h be any assignment in Y_1: then, by definition, $h(\vec{z}) \neq \vec{t}_1\langle s\rangle$ for all $s \in X$. But then $h(\vec{z}) \neq \vec{t}_1\langle h'\rangle$ for all $h' \in Y_1$, and since this is true for all $h \in Y_1$ we have that $M \models_{Y_1} \vec{z} \neq \vec{t}_1$, as required.

For the second case, let h be in Y_2 instead: then, again by definition, $h(\vec{z}) = \vec{t}_1\langle s\rangle$ for some $s \in X$. But $M \models_X \vec{t}_1 \mid \vec{t}_2$, and hence $h(\vec{z}) \neq \vec{t}_2\langle s'\rangle$ for all $s' \in X$; and as in the previous case, this implies that $h(\vec{z}) \neq \vec{t}_2(h')$ for all $h' \in Y_2$ and, since this argument can be made for all $h \in Y_2$, $M \models_{Y_2} \vec{z} \neq \vec{t}_2$.

Conversely, suppose that

$$M \models_X \forall \vec{z} \exists u_1 u_2 (=\!(\vec{z}, u_1) \wedge =\!(\vec{z}, u_2) \wedge ((u_1 = u_2 \wedge \vec{z} \neq \vec{t}_1) \vee (u_1 \neq u_2 \wedge \vec{z} \neq \vec{t}_2))).$$

Then there exist two functions H_1 and H_2 such that, for $Y = X[M/\vec{z}][H_1H_2/u_1u_2]$,

$$M \models_Y =\!(\vec{z}, u_1) \wedge =\!(\vec{z}, u_2) \wedge ((u_1 = u_2 \wedge \vec{z} \neq \vec{t}_1) \vee (u_1 \neq u_2 \wedge \vec{z} \neq \vec{t}_2)).$$

Now, let s_1 and s_2 be any two assignments in X: in order to conclude the proof, I only need to show that $\vec{t}_1\langle s_1\rangle \neq \vec{t}_2\langle s_2\rangle$. Suppose instead that $\vec{t}_1\langle s_1\rangle =$

$\vec{t}_2\langle s_2\rangle = \vec{m}$ for some tuple of elements $\vec{m}$, and consider two assignments h_1, h_2 such that

$$h_1 \in \{s_1[\vec{m}/\vec{z}]\}[H_1H_2/u_1u_2];$$[27]

and

$$h_2 \in \{s_2[\vec{m}/\vec{z}]\}[H_1H_2/u_1u_2].$$

Then $h_1, h_2 \in Y$; and furthermore, since $h_1(\vec{z}) = h_2(\vec{z})$ and $M \models =\!(\vec{z}, u_1) \wedge =\!(\vec{z}, u_2)$, it must hold that $h_1(\vec{u}_1) = h_2(\vec{u}_1)$ and $h_1(\vec{u}_2) = h_2(\vec{u}_2)$.

Moreover, $M \models_Y (u_1 = u_2 \wedge \vec{z} \neq \vec{t}_1) \vee (u_1 \neq u_2 \wedge \vec{z} \neq \vec{t}_2)$, and therefore Y can be split into two subteams Y_1 and Y_2 such that

$$M \models_{Y_1} (u_1 = u_2 \wedge \vec{z} \neq \vec{t}_1)$$

and

$$M \models_{Y_2} (u_1 \neq u_2 \wedge \vec{z} \neq \vec{t}_2).$$

Now, as we saw, the assignments h_1 and h_2 coincide over u_1 and u_2, and hence either $\{h_1, h_2\} \subseteq Y_1$ or $\{h_1, h_2\} \subseteq Y_2$. But neither case is possible, because

$$h_1(\vec{z}) = \vec{m} = \vec{t}_1\langle s_1\rangle = \vec{t}_1\langle h_1\rangle$$

and therefore h_1 cannot be in Y_1, and because

$$h_2(\vec{z}) = \vec{m} = \vec{t}_2\langle s_2\rangle = \vec{t}_2\langle h_2\rangle$$

and therefore h_2 cannot be in Y_2.

So we reached a contradiction, and this concludes the proof.

Theorem 4.19. *Let $t_1 \ldots t_n$ be terms, and let z be a variable not occurring in any of them. Then the dependence atom $=\!(t_1 \ldots t_n)$ is equivalent to the exclusion logic expression*

$$\phi = \forall z(z = t_n \vee (t_1 \ldots t_{n-1}z \mid t_1 \ldots t_{n-1}t_n)),$$

for all suitable models M and teams X.

[27] This team and the next one are actually singletons, because H_1 and H_2 must satisfy the dependency conditions.

Proof. Suppose that $M \models_X =(t_1 \dots t_n)$, and consider the team $X[M/z]$. Now, let $Y = \{s \in X[M/z] : s(z) = t_n\langle s\rangle\}$ and let $Z = X[M/z]\backslash Y$.

Clearly, $Y \cup Z = X[M/x]$ and $M \models_Y z = t_n$; hence, if we show that $Z \models t_1 \dots t_{n-1}z \mid t_1 \dots t_{n-1}t_n$ we can conclude that $M \models_X \phi$, as required.

Now, consider any two $s, s' \in Z$, and suppose that $t_i\langle s\rangle = t_i\langle s'\rangle$ for all $i = 1 \dots n-1$. But then $s(z) \neq t_n\langle s'\rangle$: indeed, since $M \models_X =(t_1 \dots t_n)$, by the locality of dependence logic and by the downwards closure property we have that $M \models_Z =(t_1 \dots t_n)$ and hence that $t_n\langle s\rangle = t_n\langle s'\rangle$.

Therefore, if we had that $s(z) = t_n\langle s'\rangle$, it would follow that $s(z) = t_n\langle s'\rangle = t_n\langle s\rangle$ and s would be in Y instead.

So $s(z) \neq t_n\langle s'\rangle$, and since this holds for all s and s' in Z which coincide over $t_1 \dots t_{n-1}$ we have that

$$M \models_Z t_1 \dots t_{n-1}z \mid t_1 \dots t_{n-1}t_n,$$

as required.

Conversely, suppose that $M \models_X \phi$, and let $s, s' \in X$ assign the same values to $t_1 \dots t_{n-1}$. Now, by the definition of ϕ, $X[M/z]$ can be split into two subteams Y and Z such that $M \models_Y z = t_n$ and
$M \models_Z (t_1 \dots t_{n-1}z \mid t_1 \dots t_{n-1}t_n)$.

Now, suppose that $t_n\langle s\rangle = m$ and $t_n\langle s'\rangle = m'$, and that $m \neq m'$: then $s[m'/z]$ and $s'[m/z]$ are in $s[M/z]$ but not in Y, and hence they are both in Z. But then, since $\vec{t_i}\langle s\rangle = \vec{t_i}\langle s'\rangle$ for all $i = 1 \dots n-1$,

$$t_n\langle s'\rangle = m' = s[m'/z](z) \neq t_n\langle s'[m/z]\rangle = t_n\langle s'\rangle$$

which is a contradiction. Therefore, $m = m'$, as required.

Corollary 4.20. *Dependence logic is precisely as expressive as exclusion logic, both with respect to definability of sets of teams and with respect to sentences.*

4.5. Inclusion/exclusion logic

Now that we have some information about inclusion logic and about exclusion logic, let us study *inclusion/exclusion logic* (I/E logic for short), that is, the formalism obtained by adding both inclusion and exclusion atoms to the language of first-order logic.

By the results of the previous sections, we already know that inclusion atoms are expressible in independence logic and that exclusion atoms are expressible in dependence logic; furthermore, by Theorem 2.7, dependence atoms are expressible in independence logic.

Then it follows at once that I/E logic is contained in independence logic:

Corollary 4.21. *For every inclusion/exclusion logic formula ϕ there exists an independence logic formula ϕ^* such that*

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

for all suitable models M and teams X.

Now, is I/E logic properly contained in independence logic?

As the following result illustrates, this is not the case:

Theorem 4.22. *Let $\vec{t}_2 \perp_{\vec{t}_1} \vec{t}_3$ be an independence atom, and let ϕ be the formula*

$$\forall \vec{p}\vec{q}\vec{r}\ \exists u_1 u_2 u_3 u_4 \left(\bigwedge_{i=1}^{4} {=}(\vec{p}\vec{q}\vec{r}, u_i) \wedge ((u_1 \neq u_2 \wedge (\vec{p}\vec{q} \mid \vec{t}_1\vec{t}_2)) \vee \right.$$
$$\left. \vee (u_1 = u_2 \wedge u_3 \neq u_4 \wedge (\vec{p}\vec{r} \mid \vec{t}_1\vec{t}_3)) \vee (u_1 = u_2 \wedge u_3 = u_4 \wedge (\vec{p}\vec{q}\vec{r} \subseteq \vec{t}_1\vec{t}_2\vec{t}_3))) \right)$$

where the dependence atoms are used as shorthands for the corresponding exclusion logic expressions, which exist because of Theorem 4.19, and where all the quantified variables are new.

Then, for all suitable models M and teams X,

$$M \models_X \vec{t}_2 \perp_{\vec{t}_1} \vec{t}_3 \Leftrightarrow M \models_X \phi.$$

Proof. Suppose that $M \models_X \vec{t}_2 \perp_{\vec{t}_1} \vec{t}_3$, and consider the team $X' = X[M/\vec{p}\vec{q}\vec{r}]$.

Now, let 0 and 1 be two distinct elements of the domain of M, and let the functions $F_1 \ldots F_4$ be defined as follows:

- For all $s \in X'$, $F_1(s) = 0$;
- For all $s \in X'[F_1/u_1]$,

$$F_2(s) = \begin{cases} 0 & \text{if there exists a } s' \in X \text{ such that } \vec{t}_1\langle s'\rangle\vec{t}_2\langle s'\rangle = s(\vec{p})s(\vec{q}); \\ 1 & \text{otherwise;} \end{cases}$$

- For all $s \in X'[F_1/u_1][F_2/u_2]$, $F_3(s) = 0$;
- For all $s \in X'[F_1/u_1][F_2/u_2][F_3/u_3]$,

$$F_4(s) = \begin{cases} 0 & \text{if there exists a } s' \in X \text{ such that } \vec{t}_1\langle s'\rangle\vec{t}_3\langle s'\rangle = s(\vec{p})s(\vec{r}); \\ 1 & \text{otherwise.} \end{cases}$$

Now, let $Y = X'[F_1/u_1][F_2/u_2][F_3/u_3][F_4/u_4]$: by the definitions of $F_1 \ldots F_4$, it holds that all dependencies are respected. Let then Y be split into Y_1, Y_2 and Y_3 according to:

- $Y_1 = \{s \in Y : s(u_1) \neq s(u_2)\}$;
- $Y_2 = \{s \in Y : s(u_3) \neq s(u_4)\} \backslash Y_1$;
- $Y_3 = Y \backslash (Y_1 \cup Y_2)$.

Now, let s be any assignment of Y_1: then, since $s(u_1) \neq s(u_2)$, by the definitions of F_1 and F_2 we have that

$$\forall s' \in Y, s(\vec{p})s(\vec{q}) \neq \vec{t}_1\langle s'\rangle\vec{t}_2\langle s'\rangle$$

and, in particular, that the same holds for all the $s' \in Y_1$. Hence,

$$M \models_{Y_1} u_1 \neq u_2 \wedge (\vec{p}\vec{q} \mid \vec{t}_1\vec{t}_2),$$

as required.

Analogously, let s be any assignment of Y_2: then $s(u_1) = s(u_2)$, since otherwise s would be in Y_1, $s(u_3) \neq s(u_4)$ and

$$\forall s' \in Y, s(\vec{p})s(\vec{r}) \neq \vec{t}_1\langle s'\rangle\vec{t}_3\langle s'\rangle$$

and therefore

$$M \models_{Y_2} u_1 = u_2 \wedge u_3 \neq u_4 \wedge (\vec{p}\vec{r} \mid \vec{t}_1\vec{t}_3).$$

Finally, suppose that $s \in Y_3$: then, by definition, $s(u_1) = s(u_2)$ and $s(u_3) = s(u_4)$. Therefore, there exist two assignments s' and s'' in X such that

$$\vec{t}_1\langle s'\rangle\vec{t}_2\langle s'\rangle = s(\vec{p})s(\vec{q})$$

and

$$\vec{t}_1\langle s''\rangle\vec{t}_3\langle s''\rangle = s(\vec{p})s(\vec{r})$$

But by hypothesis we know that $M \models_X \vec{t}_2 \perp_{\vec{t}_1} \vec{t}_3$, and s' and s'' coincide over $\vec{t}_1$, and therefore there exists a new assignment $h \in X$ such that

$$\vec{t}_1\langle h\rangle \vec{t}_2\langle h\rangle \vec{t}_3\langle h\rangle = s(\vec{p})s(\vec{q})s(\vec{r}).$$

Now, let o be the assignment of Y given by

$$o = h[\vec{t}_1\langle h\rangle \vec{t}_2\langle h\rangle \vec{t}_3\langle h\rangle / \vec{p}\vec{q}\vec{r}][F_1 \ldots F_4/u_1 \ldots u_4] :$$

by the definitions of $F_1 \ldots F_4$ and by the construction of o, we then get that

$$o(u_1) = o(u_2) = o(u_3) = o(u_4) = 0$$

and therefore that $o \in Y_3$.

But by construction,

$$\vec{t}_1\langle o\rangle \vec{t}_2\langle o\rangle \vec{t}_3\langle o\rangle = \vec{t}_1\langle h\rangle \vec{t}_2\langle h\rangle \vec{t}_3\langle h\rangle = s(\vec{p})s(\vec{q})s(\vec{r}),$$

and hence

$$M \models_{Y_3} \vec{p}\vec{q}\vec{r} \subseteq \vec{t}_1\vec{t}_2\vec{t}_3$$

as required.

Conversely, suppose that $M \models_X \phi$, and let $s, s' \in X$ be such that $\vec{t}_1\langle s\rangle = \vec{t}_1\langle s'\rangle$. Now, consider the two assignments $h, h' \in X' = X[M/\vec{p}\vec{q}\vec{r}]$ given by

$$h = s[\vec{t}_1\langle s\rangle/\vec{p}][\vec{t}_2\langle s\rangle/\vec{q}][\vec{t}_3\langle s'\rangle/\vec{r}]$$

and

$$h' = s'[\vec{t}_1\langle s\rangle/\vec{p}][\vec{t}_2\langle s\rangle/\vec{q}][\vec{t}_3\langle s'\rangle/\vec{r}].$$

Now, since $M \models_X \phi$, there exist functions $F_1 \ldots F_4$, depending only on $\vec{p}$, $\vec{q}$ and $\vec{r}$, such that
$Y = X'[F_1/u_1][F_2/u_2][F_3/u_3][F_4/u_4]$ can be split into three subteams Y_1, Y_2 and Y_3 and

$$\begin{aligned}
&M \models_{Y_1} (u_1 \neq u_2 \wedge (\vec{p}\vec{q} \mid \vec{t}_1\vec{t}_2));\\
&M \models_{Y_2} (u_1 = u_2 \wedge u_3 \neq u_4 \wedge (\vec{p}\vec{r} \mid \vec{t}_1\vec{t}_3));\\
&M \models_{Y_3} (u_1 = u_2 \wedge u_3 = u_4 \wedge (\vec{p}\vec{q}\vec{r} \subseteq \vec{t}_1\vec{t}_2\vec{t}_3)).
\end{aligned}$$

Now, let

$$o = h[F_1/u_1][F_2/u_2][F_3/u_3][F_4/u_4]$$

and

$$o' = h'[F_1/u_1][F_2/u_2][F_3/u_3][F_4/u_4] :$$

since the F_i depend only on $\vec{p}\vec{q}\vec{r}$ and the values of these variables are the same for h and for h', we have that o and o' have the same values for $u_1 \ldots u_4$, and therefore that they belong to the same Y_i.

But they cannot be in Y_1 nor in Y_2, since

$$o(\vec{p})o(\vec{q}) = o'(\vec{p})o'(\vec{q}) = \vec{t}_1\langle s\rangle\vec{t}_2\langle s\rangle = \vec{t}_1\langle o\rangle\vec{t}_2\langle o\rangle$$

and

$$o(\vec{p})o(\vec{r}) = o'(\vec{p})o'(\vec{r}) = \vec{t}_1\langle s'\rangle\vec{t}_3\langle s'\rangle = \vec{t}_1\langle o'\rangle\vec{t}_3\langle o'\rangle;$$

therefore, o and o' are in Y_3, and there exists an assignment $o'' \in Y_3$ with

$$\vec{t}_1\langle o''\rangle\vec{t}_2\langle o''\rangle\vec{t}_3\langle o''\rangle = o(\vec{p})o(\vec{q})o(\vec{r}) = \vec{t}_1\langle s\rangle\vec{t}_2\langle s\rangle\vec{t}_3\langle s'\rangle$$

and, finally, there exists a $s'' \in X$ such that $\vec{t}_1\langle s''\rangle\vec{t}_2\langle s''\rangle\vec{t}_3\langle s''\rangle = \vec{t}_1\langle s\rangle\vec{t}_2\langle s\rangle\vec{t}_3\langle s'\rangle$, as required.

Independence logic and inclusion/exclusion logic are therefore equivalent:

Corollary 4.23. *Any independence logic formula is equivalent to some inclusion/exclusion logic formula, and any inclusion/exclusion logic formula is equivalent to some independence logic formula.*

Figure 2 summarizes the translatability[28] relations between the logics of imperfect information which have been considered in this work.

Let us finish this section verifying that I/E logic (and, as a consequence, also inclusion logic, equiextension logic and independence logic) with the lax semantics is local:

Theorem 4.24 (Inclusion/exclusion logic with lax semantics is local). *Let M be a first order model, let ϕ be any I/E logic formula and let V be a set of variables such that Free(ϕ) $\subseteq V$. Then, for all suitable teams X,*

$$M \models_X \phi \Leftrightarrow M \models_{X_{\upharpoonright V}} \phi$$

[28] To be more accurate, Figure 2 represents the translatability relations between the logics which we considered, *with respect to all formulas.* Considering sentences only would lead to a different graph.

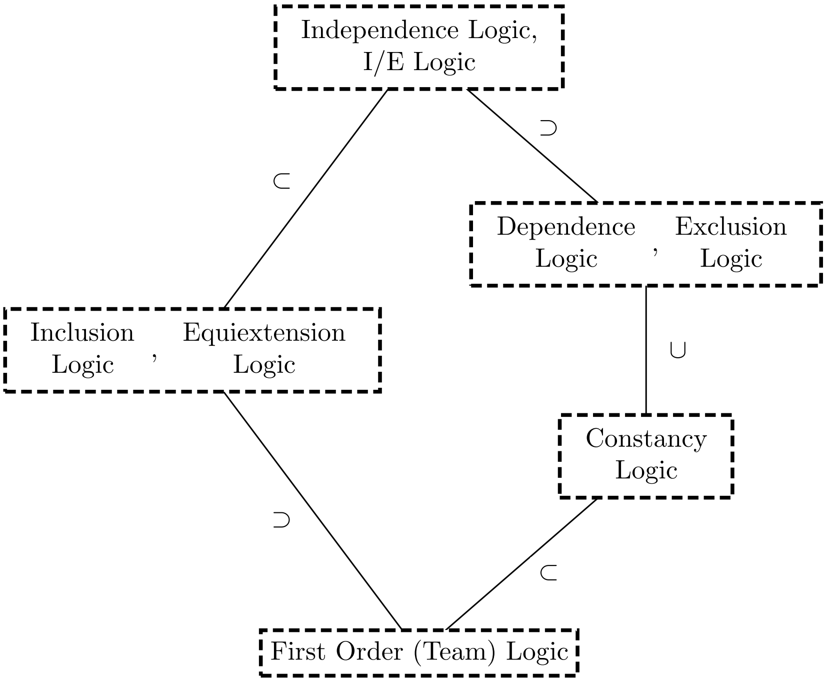

Figure 2: Relations between logics of imperfect information (wrt formulas)

Proof. The proof is by structural induction on ϕ.

1. If ϕ is a first order literal, an inclusion atom or an exclusion atom then the statement follows trivially from the corresponding semantic rule;
2. Let ϕ be of the form $\psi \vee \theta$, and suppose that $M \models_X \psi \vee \theta$. Then, by definition, $X = Y \cup Z$ for two subteams Y and Z such that $M \models_Y \psi$ and $M \models_Z \theta$. Then, by induction hypothesis, $M \models_{Y_{\upharpoonright V}} \psi$ and $M \models_{Z_{\upharpoonright V}} \theta$. But $X_{\upharpoonright V} = Y_{\upharpoonright V} \cup Z_{\upharpoonright V}$: indeed, $s \in X$ if and only if $s \in Y$ or $s \in Z$, and hence $s_{\upharpoonright V} \in X_{\upharpoonright V}$ if and only if it is in $Y_{\upharpoonright V}$ or in $Z_{\upharpoonright V}$. Hence, $M \models_{X_{\upharpoonright V}} \psi \vee \theta$, as required.
   Conversely, suppose that $M \models_{X_{\upharpoonright V}} \psi \vee \theta$, that is, that $X_{\upharpoonright V} = Y' \cup Z'$ for two subteams Y' and Z' such that $M \models_{Y'} \psi$ and $M \models_{Z'} \theta$. Then define $Y = \{s \in X : s_{\upharpoonright V} \in Y'\}$ and $Z = \{s \in X : s_{\upharpoonright V} \in Z'\}$. Now, $X = Y \cup Z$: indeed, if $s \in X$ then $s_{\upharpoonright V}$ is in $X_{\upharpoonright V}$, and hence it is in Y' or in Z', and on the other hand if s is in Y or in Z then it is in X by definition. Furthermore, $Y_{\upharpoonright V} = Y'$ and $Z_{\upharpoonright V} = Z'$,[29] and hence by induction hypothesis $M \models_Y \psi$ and $M \models_Z \theta$, and finally $M \models_X \psi \vee \theta$.
3. Let ϕ be of the form $\psi \wedge \theta$. Then $M \models_X \psi \wedge \theta$ if and only if $M \models_X \psi$ and $M \models_X \theta$, that is, by induction hypothesis, if and only if $M \models_{X_{\upharpoonright V}} \psi$ and $M \models_{X_{\upharpoonright V}} \theta$. But this is the case if and only if $M \models_{X_{\upharpoonright V}} \psi \wedge \theta$, as required.
4. Let ϕ be of the form $\exists x \psi$, and suppose that $M \models_X \exists x \psi$. Then there exists a function $H : X \to \mathcal{P}(\mathrm{Dom}(M)) \backslash \{\emptyset\}$ such that $M \models_{X[H/x]} \psi$. Then, by induction hypothesis, $M \models_{(X[H/x])_{\upharpoonright V \cup \{x\}}} \psi$.

   Now consider the function $H' : X_{\upharpoonright V} \to \mathcal{P}(\mathrm{Dom}(M)) \backslash \emptyset$ which assigns to every $s' \in X_{\upharpoonright V}$ the set
   $$H'(s') = \bigcup \{H(s) : s \in X, s' = s_{\upharpoonright V}\}.$$
   Then H' assigns a nonempty set to every $s' \in X_{\upharpoonright V}$, as required; and furthermore, $X_{\upharpoonright V}[H'/x]$ is precisely $(X[H/x])_{\upharpoonright V \cup \{x\}}$.[30] Therefore, $M \models_{X_{\upharpoonright V}} \exists x \psi$, as required.

[29] By definition, $Y_{\upharpoonright V} \subseteq Y'$ and $Z_{\upharpoonright V} \subseteq Z'$. On the other hand, if $s' \in Y'$ then $s' \in X_{\upharpoonright V}$, and hence s' is of the form $s_{\upharpoonright V}$ for some $s \in X$, and therefore this s is in Y too, and finally $s' = s_{\upharpoonright V} \in Y_{\upharpoonright V}$. The same argument shows that $Z' \subseteq Z_{\upharpoonright V}$.

[30] Indeed, suppose that $s' \in X[H/x]$: then there exists a $s \in X$ such that $s' = s[m/x]$ for some $m \in H(s)$. Then $s_{\upharpoonright V} \in X_{\upharpoonright V}$, and moreover $m \in H'(s_{\upharpoonright V})$ by the definition of H',

Conversely, suppose that $M \models_{X_{\restriction V}} \exists x\psi$, that is, that $M \models_{X_{\restriction V}[H'/x]} \psi$ for some H'. Then define the function $H : X \to \mathcal{P}(\mathrm{Dom}(M))\backslash\{x\}$ so that $H(s) = H'(s_{\restriction V})$ for all $s \in X$; now, $X_{\restriction V}[H'/x] = (X[H/x])_{\restriction V \cup \{x\}}$,[31] and hence by induction hypothesis $M \models_X \exists x\psi$.

5. For all suitable teams X, $X[M/x]_{\restriction V \cup \{x\}} = X_{\restriction V}[M/x]$; and hence, $M \models_{X_{\restriction V}} \forall x\psi \Leftrightarrow M \models_{X[M/x]_{\restriction V\cup\{x\}}} \psi \Leftrightarrow M \models_{X[M/x]} \psi \Leftrightarrow M \models_X \forall x\psi$, as required.

5. Game theoretic semantics

By this point, we have developed a team semantics for inclusion/exclusion logic and we have examined the relations between it and other logics of imperfect information. In this section, an equivalent game theoretic semantics for inclusion/exclusion logic will be developed; once this is done, the semantics for inclusion logic and for exclusion logic will simply be the restrictions of this semantics to the corresponding sublanguages. The connection between game semantics and team semantics, moreover, will allow us to revisit and further justify the distinction between lax and strict connectives introduced in Section 3.1. However, we will not discuss here the history or the motivations of game theoretic semantics, nor its connections to other game-theoretical approaches to formal semantics. The interested reader is referred to [18] and [19] for a more philosophically oriented discussion of game theoretic semantics; in the rest of this section, we will content ourselves to present such a semantics for the case of I/E logic and prove its equivalence to team semantics.

Definition 5.1 (Semantic games for I/E logic). Let ϕ be an I/E logic formula, let M be a first order model over a signature containing that of ϕ and let X be a team over M whose domain contains all free variables of ϕ. Then the game $G^M_X(\phi)$ is defined as follows:

and hence $s'_{\restriction V\cup\{x\}} = s_{\restriction V}[m/x] \in X_{\restriction V}[H'/x]$.

Conversely, suppose that $h' \in X_{\restriction V}[H'/x]$: then there exists a $h \in X_{\restriction V}$ such that $h' = h[m/x]$ for some $m \in H'(h)$. But then there exists a $s \in X$ such that $h = s_{\restriction V}$ and such that $m \in H(s)$; and therefore, $s[m/x] \in X[H/x]$, and finally $h' = h[m/x] = (s[m/x])_{\restriction V\cup\{x\}} \in (X[H/x])_{\restriction V\cup\{x\}}$.

[31] In brief, for all $s \in X$ and all $m \in \mathrm{Dom}(M)$ we have that $m \in H'(s_{\restriction V})$ if and only if $m \in H(s)$, by definition. Hence, for all such s and m, $s_{\restriction V}[m/x] \in X_{\restriction V}[H'/x]$ if and only if $s[m/x] \in X[H/x]$.

- There are two players, called I and II;[32]
- The positions of the game are expressions of the form (ψ, s), where ψ is an instance of a subformula of ϕ and s is an assignment whose domain contains all free variables of ψ;
- The initial positions are all those of the form (ϕ, s) for $s \in X$;
- The terminal positions are those of the form (α, s), where α is a first order literal, an inclusion atom, or an exclusion atom;
- If $p = (\psi, s)$ is not a terminal position, the set $S(p)$ of its *successors* is defined according to the following rules:
  1. If ψ is of the form $\theta_1 \vee \theta_2$ or $\theta_1 \wedge \theta_2$ then $S(p) = \{(\theta_1, s), (\theta_2, s)\}$;
  2. If ψ is of the form $\exists x\theta$ or $\forall x\theta$ then $S(p) = \{(\theta, s[m/x]) : m \in \mathrm{Dom}(M)\}$;
- If $p = (\psi, s)$ is not a terminal position, the *active player* $T(p) \in \{I, II\}$ is defined according to the following rules:
  1. If ψ is of the form $\theta_1 \vee \theta_2$ or $\exists x\theta$ then $T(p) = II$;
  2. If ψ is of the form $\theta_1 \wedge \theta_2$ or $\forall x\theta$ then $T(p) = I$;
- A terminal position $p = (\alpha, s)$ is *winning* for Player II if and only if

  - α is a first order literal and $M \models_s \alpha$ in the usual first order sense, or
  - α is an inclusion atom $\vec{t_1} \subseteq \vec{t_2}$ and s is any assignment, or
  - α is an exclusion atom $\vec{t_1} \mid \vec{t_2}$ and s is any assignment.

  If a terminal position is not winning for Player II, it is winning for Player I.

The definitions of *play*, *complete play* and *winning play* are straightforward:

Definition 5.2. Let $G_X^M(\phi)$ be a semantic game as above. Then a *play* for $G_X^M(\phi)$ is a finite sequence of positions $p_1 \ldots p_n$ such that

[32] These players can also be named Falsifier and Verifier, or Abelard and Eloise.

- p_1 is an initial position;
- For all $i = 2 \ldots n$, $p_i \in S(p_{i-1})$.

Such a play is said to be *complete* if, furthermore, p_n is a terminal position; and it is *winning* for Player II $[I]$ if and only if p_n is a winning position for II $[I]$.

However, it will be useful to consider *non-deterministic strategies* rather than deterministic ones only:

Definition 5.3. Let $G_X^M(\phi)$ be a semantic game as above. Then a *strategy* for Player II $[I]$ in $G_X^M(\phi)$ is a function τ sending each position $p = (\psi, s)$ with $T(p) = II$ $[I]$ into some $\tau(p) \in \mathcal{P}(S(p))\backslash\emptyset$.

Such a strategy is said to be *deterministic* if, for all such p, $\tau(p)$ is a singleton.

A play $p_1 \ldots p_n$ is said to *follow* a strategy τ for II $[I]$ if and only if, for all $i \in 1 \ldots n-1$,

$$T(p_i) = II\ [I] \Rightarrow p_{i+1} \in \tau(p_i).$$

A strategy τ for II $[I]$ is *winning* for Player II $[I]$ if and only if all complete plays $\vec{p}$ which follow τ are winning for II $[I]$.

The set of all plays of $G_X^M(\phi)$ in which Player $\rho \in \{I, II\}$ follows strategy τ will be written as $P(G_X^M(\phi), \rho : \tau)$.

So far, inclusion and exclusion atoms play little role in our semantics, as they always correspond to winning positions for Player II. Similarly to dependence atoms in [32], however, inclusion and exclusion atoms restrict the *set of strategies* available to Player II. This is modeled by the following definition of *uniform strategy*:

Definition 5.4. Let $G_X^M(\phi)$ be a semantic game as above. Then a strategy τ for Player II is said to be *uniform* if and only if, for all complete plays $p_1 \ldots p_n = \vec{p} \in P(G_X^M(\phi), II : \tau)$,

1. If p_n is of the form $p_n = (\vec{t}_1 \subseteq \vec{t}_2, s)$ then there exists a play $q_1 \ldots q_{n'} \in P(G_X^M(\phi), II : \tau)$ such that $q_{n'} = (\vec{t}_1 \subseteq \vec{t}_2, s')$ for the same instance of the inclusion atom and such that $\vec{t}_2\langle s' \rangle = \vec{t}_1\langle s \rangle$;

2. If p_n is of the form $p_n = (\vec{t_1} \mid \vec{t_2}, s)$ then for all plays $q_1 \dots q_{n'} \in P(G^M_X(\phi), II : \tau)$ such that $q_{n'} = (\vec{t_1} \mid \vec{t_2}, s')$ for the same instance of the exclusion atom it holds that $\vec{t_1}\langle s\rangle \neq \vec{t_2}\langle s'\rangle$.

This notion of uniformity also makes it clear why in inclusion logic there is a difference between working with non-deterministic and with deterministic strategies: whereas the uniformity condition for dependence atoms restrict the information available to Player II thorough the game, the one for inclusion atoms requires that the set of *possible plays*, given a strategy for Player II, is *closed* with respect to certain monotonically increasing operators. This phenomenon does not occur for the uniformity conditions of exclusion atoms, whose form is more similar to the conditions of the dependence atom in [32]. The next definition and the lemmas following it will be of some use in order to prove the main result of this section:

Definition 5.5. Let $G^M_X(\phi)$ be a game as in our previous definitions and let τ be a strategy for Player II in it. Furthermore, let ψ be an instance of a subformula of ϕ and let

$$Y = \{s : \text{ there is a play in } P(G^M_X, II : \tau) \text{ passing through } (\psi, s)\}.$$

Furthermore, let τ' be the restriction of τ to $G^M_Y(\psi)$, in the sense that $\tau'(\theta, s) = \tau(\theta, s)$ for all θ contained in ψ and for all assignments s. Then we say that (Y, ψ, τ') is a *M-successor* of (X, ϕ, τ), and we write

$$(Y, \psi, \tau') \leq_M (X, \phi, \tau).$$

From a game-theoretical perspective, the notion of M-successor can be seen as a generalization of the notion of the concepts of *subgame* and *substrategy* to multiple initial positions and to games of imperfect information.

Lemma 5.6. *Let $G^M_X(\phi)$ be a semantic game for I/E logic, and let ψ be an instance of a subformula in ϕ. Then there exists precisely one team Y and precisely one strategy τ' for $G^M_Y(\psi)$ such that $(Y, \psi, \tau') \leq_M (X, \phi, \tau)$.*

Proof. Obvious from definition.

Lemma 5.7. *Let $G^M_X(\phi)$ be a semantic game as usual, and let τ be a strategy for Player II in it. Furthermore, let ψ be an instance of a subformula of ϕ and let Y, τ' be such that $(Y, \psi, \tau') \leq_M (X, \phi, \tau)$. Then*

1. *For any play $p_1 \dots p_n = \vec{p} \in P(G^M_X(\phi), II : \tau)$ passing through the subformula ψ there exist a $k \in 1 \dots n$ such that $p_k \dots p_n$ is a play in $P(G^M_Y(\psi), II : \tau')$;*
2. *For any play $q_1 \dots q_m = \vec{q} \in P(G^M_Y(\psi), II : \tau')$ there exists a $k \in 1 \dots n$ and positions $p_1 \dots p_k$ of the game $G^M_X(\phi)$ such that $p_1 \dots p_k q_1 \dots q_m$ is a play in $(G^M_X(\psi), II : \tau)$.*

Proof.

1. Consider any play $p_1 \dots p_n$ as in our hypothesis, and let $k \in 1 \dots n$ be such that $p_k = (\psi, s)$ for some assignment s. Then, by definition of M-successors, $s \in Y$ and p_k is a possible initial position of $G^M_Y(\psi)$; furthermore, again by the definition of M-successor, we have that, for all $i = k \dots n-1$, $\tau'(p_i) = \tau(p_i) \ni p_{i+1}$.

   Hence, $p_k \dots p_n$ is a play in $P(G^M_Y(\psi), II : \tau')$, as required.
2. Consider any play $q_1 \dots q_m$ as in our hypothesis, and hence let $q_1 = (\psi, s)$ for some $s \in Y$. Then, by definition, there exists a play $p_1 \dots p_n$ in $P(G^M_X(\psi), II : \tau)$ such that $p_{k+1} = q_1 = (\psi, s)$ for some $k \in 0 \dots n-1$. But τ' behaves like τ, and hence $\tau(q_i) = \tau'(q_i) \ni q_{i+1}$ for all $i = 1 \dots m-1$. Thus, $p_1 \dots p_k q_1 \dots q_m$ is a play in $(G^M_X(\psi), II : \tau)$, as required.

Lemma 5.8. *Let $G^M_X(\phi)$ be a semantic game as usual, and let τ be a strategy for Player II in it. Furthermore, let ψ be an instance of a subformula of ϕ and let Y, τ' be such that $(Y, \psi, \tau') \leq_M (X, \phi, \tau)$.*

Then

1. *If τ is winning for II in $G^M_X(\phi)$ then τ' is winning for II in $G^M_Y(\psi)$;*
2. *If τ is uniform in $G^M_X(\phi)$ then τ' is uniform in $G^M_Y(\psi)$;*
3. *If τ is deterministic in $G^M_X(\phi)$ then τ' is deterministic in $G^M_Y(\psi)$.*

Proof.

1. Suppose that τ is winning[33], and consider any play $q_1 \dots q_m = \vec{q} \in P(G^M_Y(\psi), II : \tau')$. Then, by Lemma 5.7, there exists a play $p_1 \dots p_n \in P(G^M_X(\phi), II : \tau)$ such that $p_k \dots p_n = q_1 \dots q_m$ for some $k \in 1 \dots m$. But τ is a winning strategy for II in $G^M_X(\phi)$ and therefore p_n is a winning position, as required.

[33] Here and in the rest of the work, when we write "winning" without specifying the player we mean "winning for Player II".

2. Suppose that τ is uniform, and consider any play $q_1 \ldots q_m = \vec{q} \in P(G^M_Y(\psi), II : \tau')$.

   Then, again, there exists a play $p_1 \ldots p_n = \vec{p} \in P(G^M_X(\psi), II : \tau)$ such that $p_k \ldots p_n = q_1 \ldots q_m$ for some k.

   Now suppose that $p_n = q_m = (\vec{t_1} \subseteq \vec{t_2}, s)$: then, since τ is a uniform strategy, there exists another play $p'_1 \ldots p'_{n'}$ in $(G^M_X(\phi), II : \tau)$ such that $p'_{n'} = (\vec{t_1} \subseteq \vec{t_2}, s')$ for the same instance of the inclusion atom and for a s' such that $t_2\langle s' \rangle = t_1 \langle s \rangle$.

   Since p_n and $p'_{n'}$ correspond the same dependency atom of $\vec{p}$, it must be the case that the play $p'_1 \ldots p'_{n'}$ passes through ψ; and therefore, by Lemma 5.7, there exists some $j \in 1 \ldots n'$ such that $p'_j \ldots p'_{n'}$ is a play in $P(G^M_Y(\psi), \tau')$, thus satisfying the uniformity condition for τ'.

   Now suppose that $p_n = q_m = (\vec{t_1} \mid \vec{t_2}, s)$ instead, and consider any other play $q'_1 \ldots q'_{m'} \in P(G^M_Y(\psi), \tau')$ such that $q'_m = (\vec{t_1} \mid \vec{t_2}, s')$ for the same instance of the exclusion atom. Then there exist positions $p'_1 \ldots p'_{k'}$ such that $p'_1 \ldots p'_{k'} q'_1 \ldots q'_{m'}$ is a play in $P(G^M_X(\phi), II : \tau)$. But τ is uniform, and therefore $s(\vec{t_1}) \neq s'(\vec{t_2})$, as required.
3. This follows trivially by the definition of M-successor.

Lemma 5.9. *Let $G^M_X(\phi)$ be a semantic game for I/E logic and let τ be a strategy for II in it. Furthermore, let $\psi_1 \ldots \psi_t$ be an enumeration of all immediate subformulas of ϕ, and let $Y_1 \ldots Y_t$, $\tau_1 \ldots \tau_t$ be such that $(Y_i, \psi_i, \tau_i) \leq_M (X, \phi, \tau)$ for all $i \in 1 \ldots t$. Then*

1. *If all τ_i are winning in $G^M_{Y_i}(\psi_i)$ then τ is winning in $G^M_X(\phi)$;*
2. *If all τ_i are uniform in $G^M_{Y_i}(\psi_i)$ then τ is uniform in $G^M_X(\phi)$;*
3. *If all τ_i are deterministic in $G^M_{Y_i}(\psi_i)$ and $T(\phi) = I$[34] then τ is deterministic;*
4. *If all τ_i are deterministic in $G^M_{Y_i}(\psi_i)$, $T(\phi) = II$ and $|\tau(\phi, s)| = 1$ for all $s \in Y$ then τ is deterministic.*

[34] With a slight abuse of notation, we say that $T(\psi) = \alpha$ if $T(\psi, s) = \alpha$ for all suitable assignments s. In other words, $T(\psi) = I$ if ψ is of the form $\psi_1 \wedge \psi_2$ or of the form $\forall v \psi_1$, and $T(\psi) = II$ if ψ is of the form $\psi_1 \vee \psi_2$ or $\exists v \psi_1$.

Proof. 1. Suppose that all τ_i are winning for the respective games, and consider any play $p_1 \ldots p_n = \vec{p} \in P(G^M_X(\phi), II : \tau)$. Then p_2 is of the form (ψ_i, s) for some $i \in 1 \ldots t$ and some $s \in Y_i$; and therefore, $p_2 \ldots p_n \in P(G^M_{Y_i}(\psi), II : \tau_i)$. But τ_i is winning, and hence p_n is a winning position for Player II, as required.

2. Suppose that all τ_i are uniform, and consider any play $p_1 \ldots p_n = \vec{p} \in P(G^M_X(\phi), II : \tau)$: then, once again, $p_2 \ldots p_n \in P(G^M_{Y_i}(\psi_i), II : \tau_i)$ for some i.

   Suppose now that p_n is $(\vec{t_1} \subseteq \vec{t_2}, s)$: since τ_i is uniform, there exists another play $q_1 \ldots q_m = \vec{q} \in P(G^M_{Y_i}(\psi_i), II : \tau_i)$ such that $q_m = (\vec{t_1} \subseteq \vec{t_2}, s')$ for the same instance of the inclusion atom and

$$\vec{t_1}\langle s \rangle = \vec{t_2}\langle s' \rangle.$$

   Finally, $\vec{q}$ is contained in a play of $(G^M_X(\phi), II : \tau)$ and hence the uniformity condition is respected for τ.

   Suppose instead that p_n is $(\vec{t_1} \mid \vec{t_2}, s)$, and consider any other play $p'_1 \ldots p'_{n'}$ of $P(G^M_X(\phi), II : \tau)$ such that $p'_{n'}$ is $(\vec{t_1} \mid \vec{t_2}, s')$ for the same instance of $\vec{t_1} \mid \vec{t_2}$. Now, since the same exclusion atom is reached, it must be the case that $p'_2 \ldots p'_{n'}$ is in $P(G^M_Y(\psi_i), II : \tau_i)$ too, for the same i; but then, since τ_i is uniform, $\vec{t_1}\langle s \rangle \neq \vec{t_2}\langle s' \rangle$, as required.

3. Let p be any position in $G^M_X(\phi)$ such that $T(p) = II$. Then p corresponds to a subformula of some ψ_i, and hence $|\tau(p)| = |\tau_i(p)| = 1$.
4. Let p be any position in $G^M_X(\phi)$ such that $T(p) = I$. If p is (ϕ, s) for some $s \in Y$, then $|\tau(p)| = 1$ by hypothesis; and otherwise, p corresponds to a subformula of some ψ_i, and as in the previous case $|\tau(p)| = |\tau_i(p)| = 1$.

Finally, the connection between semantic games and team semantics is given by the following theorem:

Theorem 5.10. *Let M be a first order model, let ϕ be an inclusion logic formula over the signature of M and let X be a team over M whose domain contains all free variables of ϕ. Then Player II has a uniform winning strategy in $G^M_X(\phi)$ if and only if $M \models_X \phi$ (with respect to the lax semantics).*

Proof. The proof is by structural induction on ϕ.

1. If ϕ is a first order literal then the only strategy available to II in $G^M_X(\phi)$ is the empty one. This strategy is always uniform, and the plays which follow it are of the form $\vec{p} = p_1 = (\phi, s)$, where s ranges over X. Such a play is *winning* for II if and only if $M \models_s \phi$ in the usual first-order sense; and hence, the strategy is winning for II if and only if $M \models_s \phi$ for all $s \in X$, that is, if and only if $M \models_X \phi$.
2. If ϕ is an inclusion atom $\vec{t_1} \subseteq \vec{t_2}$ then, again, the only strategy available to Player II is the empty one and the plays which follow it are those of the form $\vec{p} = p_1 = (\vec{t_1} \subseteq \vec{t_2}, s)$ for some $s \in X$.
   By the definition of the winning positions of $G^M_X(\phi)$, this strategy is winning; hence, it only remains to check whether it is uniform.
   Now, in order for the strategy to be uniform it must be the case that for all plays $\vec{p} = p_1 = (\vec{t_1} \subseteq \vec{t_2}, s)$ where $s \in X$ there exists a play $\vec{q} = q_1 = (\vec{t_1} \subseteq \vec{t_2}, s')$, again for $s' \in X$, such that $\vec{t_2}\langle s' \rangle = \vec{t_1}\langle s \rangle$. But this can be the case if and only if $\forall s \in X \exists s' \in X$ s.t. $\vec{t_1}\langle s \rangle = \vec{t_2}\langle s' \rangle$, that is, if and only if $M \models_X \vec{t_1} \subseteq \vec{t_2}$.
3. If ϕ is an exclusion atom $\vec{t_1} \mid \vec{t_2}$, the only strategy for II in $G^M_X(\phi)$ is, once again, the empty one. This strategy is always winning, and it is uniform if and only if for all plays $\vec{p} = p_1 = (\vec{t_1} \mid \vec{t_2}, s)$ and $\vec{q} = q_1 = (\vec{t_1} \mid \vec{t_2}, s')$ (for $s, s' \in X$) it holds that $\vec{t_1}\langle s \rangle \neq \vec{t_2}\langle s' \rangle$.
   But this is the case if and only if $M \models_X \vec{t_1} \mid \vec{t_2}$, as required.
4. If ϕ is a disjunction $\psi \vee \theta$, suppose that τ is a uniform winning strategy for II in $G^M_X(\psi \vee \theta)$. Then define the teams $Y, Z \subseteq X$ as follows:

   $$Y = \{s \in X : (\psi, s) \in \tau(\psi \vee \theta, s)\};$$
   $$Z = \{s \in X : (\theta, s) \in \tau(\psi \vee \theta, s)\}.$$

   Then $Y \cup Z = X$: indeed, for all $s \in X$ it must be the case that $\emptyset \neq \tau(\psi \vee \theta, s) \subsetneq \{(\psi, s), (\theta, s)\}$. Furthermore, $Y \cap Z = \emptyset$.

   Now consider the following two strategies for II in $G^M_Y(\psi)$ and $G^M_Z(\theta)$ respectively:

   - $\tau_1(p) = \tau(p)$ for all positions p of $G^M_Y(\psi)$;
   - $\tau_2(p) = \tau(p)$ for all positions p of $G^M_Z(\theta)$.

   Since all positions of $G^M_Y(\psi)$ and of $G^M_Z(\theta)$ are also positions of $G^M_X(\psi \vee \theta)$, τ_1 and τ_2 are well-defined.

Furthermore, $(Y, \psi, \tau_1) \leq_M (X, \phi, \tau)$ and $(Z, \psi, \tau_2) \leq_M (X, \phi, \tau)$; therefore, by Lemma 5.8, τ_1 and τ_2 are uniform and winning for $G^M_Y(\psi)$ and $G^M_Z(\theta)$. By induction hypothesis, this implies that $M \models_Y \psi$ and $M \models_Z \theta$, and by the definition of the semantics for disjunction, this implies that $M \models_X \psi \vee \theta$.

Conversely, suppose that $M \models_X \psi \vee \theta$: then, by definition, there exist teams Y and Z such that $X = Y \cup Z$, $M \models_Y \psi$ and $M \models_Z \theta$. Then, by induction hypothesis, there exist uniform winning strategies τ_1 and τ_2 for II in $G^M_Y(\psi)$ and $G^M_Z(\theta)$ respectively. Then define the strategy τ for II in $G^M_X(\psi \vee \theta)$ as follows:

- $\tau(\psi \vee \theta, s) = \begin{cases} \{(\psi, s)\} & \text{if } s \in Y \backslash Z; \\ \{(\theta, s)\} & \text{if } s \in Z \backslash Y; \\ \{(\psi, s), (\theta, s)\} & \text{if } s \in Y \cap Z; \end{cases}$
- If p is (χ, s) for some s and some formula χ contained in ψ, then $\tau(p) = \tau_1(p)$;
- If p is (χ, s) for some s and some χ contained in θ, then $\tau(p) = \tau_2(p)$.

Then, by construction, we have that $(Y, \psi, \tau_1), (Z, \theta, \tau_2) \leq_M (X, \psi \vee \theta, \tau)$; furthermore, ψ and θ are all the immediate subformulas of $\psi \vee \theta$, and τ_1 and τ_2 are winning and uniform by hypothesis. Therefore, by Lemma 5.9, τ is a uniform winning strategy for $G^M_X(\psi \vee \theta)$, as required.

5. If ϕ is $\psi \wedge \theta$, suppose again that τ is a uniform winning strategy for II in $G^M_X(\psi \wedge \theta)$. Then consider the two strategies for II in $G^M_X(\psi)$ and $G^M_Z(\theta)$, respectively, defined as

   - $\tau_1(p) = \tau(p)$ for all positions p of $G^M_X(\psi)$;
   - $\tau_2(p) = \tau(p)$ for all positions p of $G^M_X(\theta)$.

   Then $(X, \psi, \tau_1), (X, \theta, \tau_2) \leq_M (X, \psi \wedge \theta, \tau)$, and therefore by Lemma 5.8 ψ and θ are uniform winning strategies. Hence, by induction hypothesis, $M \models_X \psi$ and $M \models_X \theta$, and therefore $M \models_X \psi \wedge \theta$.

   Conversely, suppose that $M \models_X \psi \wedge \theta$. Then $M \models_X \psi$ and $M \models_X \theta$, and therefore II has uniform winning strategies τ_1 and τ_2 for $G^M_X(\psi)$ and $G^M_X(\theta)$ respectively. Now define the strategy τ for II in $G^M_X(\psi \wedge \theta)$ as follows:

$$\text{for all } s \in X, \tau(\chi, s) = \begin{cases} \tau_1(\chi, s) & \text{if } \chi \text{ is contained in } \psi; \\ \tau_2(\chi, s) & \text{if } \chi \text{ is contained in } \theta. \end{cases}$$

Then $(X, \psi, \tau_1), (X, \theta, \tau_2) \leq_M (X, \psi \wedge \theta, \tau)$ and ψ, θ are all immediate subformulas of $\psi \wedge \theta$; hence, by Lemma 5.9, τ is a uniform winning strategy for II in $G^M_X(\phi)$, as required.

6. If ϕ is $\exists x\psi$, suppose that τ is a uniform winning strategy for II in $G^M_X(\exists x\psi)$. Then define the function $H : X \to \mathcal{P}(\text{Dom}(M))\backslash\emptyset$ as $H(s) = \{m \in M : (\psi, s[m/x]) \in \tau(\exists x\psi, s)\}$ and consider the following strategy τ' for II in $G^M_{X[H/x]}(\psi)$:

$$\tau'(p) = \tau(p) \text{ for all suitable } p.$$

τ' is well-defined, because any position of $G^M_{X[H/x]}(\psi)$ is also a possible position of $G^M_X(\exists x\psi)$. Furthermore, $(X[H/x], \psi, \tau') \leq_M (X, \exists x\psi, \tau)$, and therefore τ' is a uniform winning strategy for II in $G^M_{X[H/x]}(\psi)$. By induction hypothesis, this implies that $M \models_{X[H/x]} \psi$, and hence that $M \models_X \exists x\psi$.

Conversely, suppose that $M \models_X \exists x\psi$; then, there exists a function H such that $M \models_{X[H/x]} \psi$. By induction hypothesis, this means that there exists a winning strategy τ' for II in $G^M_{X[H/x]}(\psi)$. Now consider the following strategy τ for II in $G^M_X(\exists x\psi)$:

$$\tau(\exists x\psi, s) = \{(\psi, s[m/x]) : m \in H(s)\};$$
$$\tau(\theta, s) = \tau'(\tau, s) \text{ for all } \tau \text{ contained in } \psi \text{ and all } s.$$

Then $(X[H/x], \psi, \tau') \leq_M (X, \exists x\psi, \tau)$, and ψ is the only direct subformula of $\exists x\psi$; hence, τ is uniform and winning, as required.

7. If ϕ is $\forall x\psi$, suppose that τ is a uniform winning strategy for II in $G^M_X(\forall x\psi)$. Then consider the strategy τ' for II in $G^M_{X[M/x]}(\psi)$ given by

$$\tau'(\theta, s) = \tau(\theta, s) \text{ for all } \theta \text{ contained in } \psi \text{ and all } s.$$

Then $(X[M/x], \psi, \tau') \leq_M (X, \forall x\psi, \tau)$, and hence τ' is uniform and winning. By induction hypothesis, this means that $M \models_{X[M/x]} \psi$, and hence that $M \models_X \forall x\psi$.

Conversely, suppose that $M \models_X \forall x\psi$. Then $M \models_{X[M/x]} \psi$, and hence there exists a uniform winning strategy τ' for II in $G^M_{X[M/x]}(\psi)$. Then consider the strategy τ for II in $G^M_X(\psi)$ given by

$$\tau(\theta, s) = \tau'(\theta, s) \text{ for all } \theta \text{ contained in } \psi \text{ and all } s.$$

This strategy is well-defined, since the first move of $G^M_X(\forall x \psi)$ is Player I's; furthermore,

$$(X[M/x], \psi, \tau') \leq_M (X, \forall x \psi, \tau)$$

and therefore τ is uniform and winning, as required.

Hence, we have a game theoretic semantics which is equivalent to the lax team semantics for inclusion/exclusion logic; and of course, the game theoretic semantics for inclusion and exclusion logic are simply the restrictions of this semantics to the corresponding languages. As was argued previously, the strict team semantics for disjunction and existential quantification is somewhat less natural when it comes to inclusion logic or I/E logic. However, there exists a link between strict team semantics and *deterministic* strategies:

Theorem 5.11. *Let M be a first order model, let ϕ be an inclusion logic formula over the signature of M and let X be a team over M whose domain contains all free variables of ϕ. Then Player II has a uniform,* deterministic *winning strategy in $G^M_X(\phi)$ if and only if $M \models_X \phi$ (with respect to the* strict *semantics).*

PROOF. The proof is by structural induction over ϕ, and it runs exactly as for the lax case. The only differences occur in the cases of disjunction and existential quantification, in which the determinism of the strategies poses a restriction on the choices available to Player II and for which the proof runs as follows:

- If ϕ is a disjunction $\psi \vee \theta$, suppose that τ is a uniform, deterministic winning strategy for II in $G^M_X(\psi \vee \theta)$. Then define the teams $Y, Z \subseteq X$ as follows:

  $$\begin{aligned} Y &= \{s \in X : \tau(\psi \vee \theta, s) = \{(\psi, s)\}\};\\ Z &= \{s \in X : \tau(\psi \vee \theta, s) = \{(\theta, s)\}\}. \end{aligned}$$

  Then $Y \cup Z = X$: indeed, for all $s \in X$ it must be the case that $\emptyset \neq \tau(\psi \vee \theta, s) \subseteq \{(\psi, s), (\theta, s)\}$, and hence s is in Y or in Z (or in both). Furthermore, $Y \cap Z = \emptyset$.

  Now consider the following two strategies for II in $G^M_Y(\psi)$ and $G^M_Z(\theta)$ respectively:

- $\tau_1(p) = \tau(p)$ for all positions p of $G^M_Y(\psi)$;
- $\tau_2(p) = \tau(p)$ for all positions p of $G^M_Z(\theta)$.

Since all positions of $G^M_Y(\psi)$ and of $G^M_Z(\theta)$ are also positions of $G^M_X(\psi \vee \theta)$, τ_1 and τ_2 are well-defined. Furthermore, they are deterministic, since τ is so, and $(Y, \psi, \tau_1), (Z, \theta, \tau_2) \leq_M (X, \phi, \tau)$; therefore, τ_1 and τ_2 are uniform and winning for $G^M_Y(\psi)$ and $G^M_Z(\theta)$. By induction hypothesis, this implies that $M \models_Y \psi$ and $M \models_Z \theta$; and by the definition of the (strict) semantics for disjunction, this implies that $M \models_X \psi \vee \theta$.

Conversely, suppose that $M \models_X \psi \vee \theta$, according to the strict semantics: then, by definition, there exist teams Y and Z such that $X = Y \cup Z$, $Y \cap Z = \emptyset$, $M \models_Y \psi$ and $M \models_Z \theta$. Then, by induction hypothesis, there exist uniform, deterministic winning strategies τ_1 and τ_2 for II in $G^M_Y(\psi)$ and $G^M_Z(\theta)$ respectively. Then define the strategy τ for II in $G^M_X(\psi \vee \theta)$ as follows:

- $\tau(\psi \vee \theta, s) = \begin{cases} \{(\psi, s)\} & \text{if } s \in Y; \\ \{(\theta, s)\} & \text{if } s \in Z. \end{cases}$
- If p is (χ, s) and χ is contained in ψ then $\tau(p) = \tau_1(p)$;
- If p is (χ, s) and χ is contained in θ then $\tau(p) = \tau_2(p)$.

Then, by construction, we have that

$$(Y, \psi, \tau_1), (Z, \theta, \tau_2) \leq_M (X, \psi \vee \theta, \tau);$$

and furthermore, ψ and θ are all the immediate subformulas of $\psi \vee \theta$, and τ_1 and τ_2 are winning and uniform by hypothesis. Therefore τ is a uniform, deterministic winning strategy for $G^M_X(\psi \vee \theta)$, as required.

- If ϕ is $\exists x \psi$, suppose that τ is a uniform, deterministic winning strategy for II in $G^M_X(\exists x \psi)$. Then define the function $F : X \to \mathrm{Dom}(M)$ so that, for every $s \in X$, $F(s)$ is the unique element m of the model such that $\tau(\exists x \psi, s) = \{(\psi, s[m/x])\}$ and consider the following strategy τ' for II in $G^M_{X[F/x]}(\psi)$:

$$\tau'(p) = \tau(p) \text{ for all suitable } p.$$

τ' is well-defined, because any position of $G^M_{X[F/x]}(\psi)$ is also a possible position of $G^M_X(\exists x\psi)$. Furthermore, $(Y[F/x], \psi, \tau') \leq_M (Y, \exists x \psi, \tau)$, and therefore τ' is a uniform, deterministic winning strategy for II in $G^M_{X[F/x]}(\psi)$. By induction hypothesis, this implies that $M \models_{X[F/x]} \psi$, and hence that $M \models_X \exists x\psi$ (with respect to the strict semantics).

Conversely, suppose that $M \models_X \exists x\psi$ according to the strict semantics; then, there exists a F such that $M \models_{X[F/x]} \psi$. By induction hypothesis, this means that there exists a uniform, deterministic winning strategy τ' for II in $G^M_{X[F/x]}(\psi)$. Now consider the following strategy τ for II in $G^M_X(\exists x\psi)$:

$$\tau(\exists x\psi, s) = \{(\psi, s[F(s)/x])\};$$
$$\tau(\theta, s) = \tau'(\tau, s) \text{ for all } \tau \text{ contained in } \psi.$$

Then $(X[F/x], \psi, \tau') \leq_M (X, \exists x\psi, \tau)$, and ψ is the only direct subformula of $\exists x\psi$; hence, τ is uniform, deterministic and winning, as required.

In [13], Thomas Forster considers the distinction between deterministic and nondeterministic strategies for the case of the logic of branching quantifiers and points out that, in the absence of the Axiom of Choice, different truth conditions are obtained for these two cases. In the same paper, he then suggests that

> Perhaps advocates of branching quantifier logics and their descendents will tell us which semantics [**that is, the deterministic or nondeterministic one**] they have in mind.

Dependence logic, inclusion logic, inclusion/exclusion logic and independence logic can certainly be seen as descendents of branching quantifier logic, and the present work strongly suggests that the semantics that we "have in mind" is the nondeterministic one. As we just saw, the deterministic/nondeterministic distinction in game theoretic semantics corresponds precisely to the strict/lax distinction in team semantics; and indeed, as seen in Subsection 3.1, for dependence logic proper (which is expressively equivalent to branching quantifier logic), the lax and strict semantics are equivalent modulo the Axiom of Choice (Proposition 3.10).

But for inclusion logic and its extensions, we have that lax and strict (and, hence, nondeterministic and deterministic) semantics are not equivalent, even in the presence of the Axiom of Choice (Propositions 4.7 and 4.8), and that only the lax one satisfies Locality in the sense of Theorem 2.2 (see Proposition 4.9 and Theorems 4.10, 4.24 for the proof).

Furthermore, as stated before, Fredrik Engström showed in [9] that the lax semantics for existential quantification arises naturally from his treatment of generalized quantifiers in dependence logic.

All of this, in the opinion of the author at least, makes a convincing case for the adoption of the nondeterministic semantics (or, in terms of team semantics, of the lax one) as the natural semantics for the study of logics of imperfect information, thus suggesting an answer to Thomas Forster's question.

6. Definability in I/E logic (and in independence logic)

In [26], Kontinen and Väänänen characterized the expressive power of dependence logic formulas (Theorem 2.5 here), and, in [25], Kontinen and Nurmi used a similar technique to prove that a class of teams is definable in team logic ([33]) if and only if it is expressible in full second order logic.

In this section, I will attempt to find an analogous result for I/E logic (and hence, through Corollary 4.23, for independence logic). One direction of the intended result is straightforward:

Theorem 6.1. *Let $\phi(\vec{v})$ be a formula of I/E logic with free variables in $\vec{v}$. Then there exists an existential second order logic formula $\Phi(A)$, where A is a second order variable with arity $|\vec{v}|$, such that*

$$M \models_X \phi(\vec{v}) \Leftrightarrow M \models \Phi(Rel_{\vec{v}}(X))$$

for all suitable models M and teams X.

Proof. The proof is an unproblematic induction over the formula ϕ, and follows closely the proof of the analogous results for dependence logic ([32]) or independence logic ([15]).

The other direction, instead, requires some care:[35]

[35] The details of this proof are similar to the ones of [26] and [25].

Theorem 6.2. *Let $\Phi(A)$ be a formula in Σ_1^1 such that Free$(\Phi) = \{A\}$, and let $\vec{v}$ be a tuple of distinct variables with $|\vec{v}| =$ Arity(A). Then there exists an I/E logic formula $\phi(\vec{v})$ such that*

$$M \models_X \phi(\vec{v}) \Leftrightarrow M \models \Phi(\mathit{Rel}_{\vec{v}}(X))$$

for all suitable models M and nonempty teams X.

Proof. It is easy to see that any $\Phi(A)$ as in our hypothesis is equivalent to the formula

$$\Phi^*(A) = \exists B(\forall \vec{x}(A\vec{x} \leftrightarrow B\vec{x}) \wedge \Phi(B)),$$

in which the variable A occurs only in the conjunct $\forall \vec{x}(A\vec{x} \leftrightarrow B\vec{x})$. Then, as in [26], it is possible to write $\Phi^*(A)$ in the form

$$\exists \vec{f}\, \forall \vec{x}\vec{y}((A\vec{x} \leftrightarrow f_1(\vec{x}) = f_2(\vec{x})) \wedge \psi(\vec{x}, \vec{y}, \vec{f})),$$

where $\vec{f} = f_1 f_2 \dots f_n$, $\psi(\vec{f}, x, y)$ is a quantifier-free formula in which A does not appear, and each f_i occurs only as $f(\vec{w}_i)$ for some fixed tuple of variables $\vec{w}_i \subseteq \vec{x}\vec{y}$.

Now define the formula $\phi(\vec{v})$ as

$$\forall \vec{x}\vec{y}\, \exists \vec{z} \left(\bigwedge_i {=}(\vec{w}_i, z_i) \wedge (((\vec{v} \subseteq \vec{x} \wedge z_1 = z_2) \vee (\vec{v} \mid \vec{x} \wedge z_1 \neq z_2)) \wedge \psi'(\vec{x}, \vec{y}, \vec{z})) \right),$$

where $\psi'(\vec{x}, \vec{y}, \vec{z})$ is obtained from $\psi(\vec{x}, \vec{y}, \vec{f})$ by substituting each $f_i(\vec{w}_i)$ with z_i, and the dependence atoms are used as shorthands for the corresponding expressions of I/E logic.

Now we have that $M \models_X \phi(\vec{v}) \Leftrightarrow M \models \Phi^*(\mathrm{Rel}_{\vec{v}}(X))$:

Indeed, suppose that $M \models_X \phi(\vec{v})$. Then, by construction, for each $i = 1 \dots n$ there exists a function F_i, depending only on $\vec{w}_i$, such that for $Y = X[M/\vec{x}\vec{y}][\vec{F}/\vec{z}]$

$$M \models_Y ((\vec{v} \subseteq \vec{x} \wedge z_1 = z_2) \vee (\vec{v} \mid \vec{x} \wedge z_1 \neq z_2)) \wedge \psi'(\vec{x}, \vec{y}, \vec{z}).$$

Therefore, we can split Y into two subteams Y_1 and Y_2 such that $M \models_{Y_1} \vec{v} \subseteq \vec{x} \wedge z_1 = z_2$ and $M \models_{Y_2} \vec{v} \mid \vec{x} \wedge z_1 \neq z_2$.

Now, for each i define the function f_i so that, for every tuple $\vec{m}$ of the required arity, $f_i(\vec{m})$ corresponds to $F_i(s)$ for an arbitrary $s \in X[M/\vec{x}\vec{y}]$ with $s(\vec{w}_i) = \vec{m}$, and let o be any assignment with domain $\vec{x}\vec{y}$.

Thus, if we can prove that $M \models_o ((\text{Rel}_{\vec{v}}(X))\vec{x} \leftrightarrow f_1(\vec{x}) = f_2(\vec{x})) \wedge \psi(\vec{x}, \vec{y}, \vec{f})$ then the left-to-right direction of our proof is done.

First of all, suppose that $M \models_o (\text{Rel}_{\vec{v}}(X))\vec{x}$, that is, that $o(\vec{x}) = \vec{m} = s(\vec{v})$ for some $s \in X$.

Then choose an arbitrary tuple of elements $\vec{r}$ and consider the assignment $h = s[\vec{m}/\vec{x}][\vec{r}/\vec{y}][\vec{F}/\vec{z}] \in Y$. This h cannot belong to Y_2, since $h(\vec{v}) = s(\vec{v}) = \vec{m} = h(\vec{x})$, and therefore it is in Y_1 and $h(z_1) = h(z_2)$.

By the definition of the f_i, this implies that $f_1(\vec{m}) = f_2(\vec{m})$, as required.

Analogously, suppose that $M, \not\models_o (\text{Rel}_{\vec{v}}(X))\vec{x}$, that is, that $o(\vec{x}) = \vec{m} \neq s(\vec{v})$ for all $s \in X$. Then pick an arbitrary such $s \in X$ and an arbitrary tuple of elements $\vec{r}$, and consider the assignment

$$h = s[\vec{m}/\vec{x}][\vec{r}/\vec{y}][\vec{F}/\vec{z}] \in Y.$$

If h were in Y_1, there would exist an assignment $h' \in Y_1$ such that $h'(\vec{v}) = h(\vec{x}) = \vec{m}$; but this is impossible, and therefore $h \in Y_2$. Hence $h(z_1) \neq h(z_2)$, and therefore $f_1(\vec{m}) \neq f_2(\vec{m})$.

Putting everything together, we just proved that

$$M \models_o R\vec{x} \Leftrightarrow f_1(\vec{x}) = f_2(\vec{x})$$

for all assignments o with domain $\vec{x}\vec{y}$, and we still need to verify that $M \models_o \psi(\vec{x}, \vec{y}, f)$ for all such o.

But this is immediate: indeed, let s be an arbitrary assignment of X, and construct the assignment

$$h = s[o(\vec{x}\vec{y})/\vec{x}\vec{y}][\vec{F}/\vec{z}] \in X[M/\vec{x}\vec{y}][\vec{F}/\vec{z}].$$

Then, since $M \models_{X[M/\vec{x}\vec{y}][\vec{F}/\vec{z}]} \psi'(\vec{x}, \vec{y}, \vec{z})$ and $\psi'(\vec{x}, \vec{y}, \vec{z})$ is first order, $M \models_{\{h\}} \psi'(\vec{x}, \vec{y}, \vec{z})$; but $\psi'(\vec{x}, \vec{y}, \vec{f}(\vec{x}\vec{y}))$ is equivalent to $\psi(\vec{x}, \vec{y}, \vec{f})$ and $h(z_i) = f(h(\vec{w}_i)) = f(o(\vec{w}_i))$, and therefore

$$M \models_o \psi(\vec{x}, \vec{y}, \vec{f})$$

as required.

Conversely, suppose that $M \models_s (\text{Rel}_{\vec{v}}(X))\vec{x} \leftrightarrow (f_1(\vec{x}) = f_2(\vec{x})) \wedge \psi(\vec{x}, \vec{y}, \vec{f})$ for all assignments s with domain $\vec{x}\vec{y}$ and for some fixed choice of the tuple of functions $\vec{f}$.

Then let $\vec{F}$ be such that, for all assignments h and for all i,

$$F_i(h) = f_i(h(\vec{w}_i))$$

and consider $Y = X[M/\vec{x}\vec{y}][F/\vec{z}]$.

Clearly, Y satisfies the dependency conditions; furthermore, it satisfies $\psi'(\vec{x}, \vec{y}, \vec{z})$, because for every assignment $h \in Y$ and every $i \in 1 \ldots n$ we have that $h(z_i) = F_i(h) = f_i(h(\vec{w}_i))$.

Finally, we can split Y into two subteams Y_1 and Y_2 as follows:

$$\begin{aligned} Y_1 &= \{o \in Y : o(\vec{z}_1) = o(\vec{z}_2)\}; \\ Y_2 &= \{o \in Y : o(\vec{z}_1) \neq o(\vec{z}_2)\}. \end{aligned}$$

It is then trivially true that $M \models_{Y_1} z_1 = z_2$ and $M \models_{Y_2} z_1 \neq z_2$, and all that is left to do is proving that $M \models_{Y_1} \vec{v} \subseteq \vec{x}$ and $M \models_{Y_2} \vec{v} \mid \vec{x}$.

As for the former, let $o \in Y_1$: then, since $o(z_1) = o(z_2)$, $f_1(o(\vec{x})) = f_2(o(\vec{x}))$.

This implies that $o(\vec{x}) \in \mathrm{Rel}_{\vec{v}}(X)$, and hence that there exists an assignment $s' \in X$ with $s'(\vec{v}) = o(\vec{x})$.

Now consider the assignment

$$o' = s'[o(\vec{x}\vec{y})/\vec{x}\vec{y}][\vec{F}/\vec{z}] :$$

since in Y the values of $\vec{z}$ depend only on the values of $\vec{x}\vec{y}$ and since $o(z_1) = o(z_2)$, we have that $o'(z_1) = o'(z_2)$ and hence $o' \in Y_1$ too. But $o'(\vec{v}) = s'(\vec{v}) = o(\vec{x})$, and since o was an arbitrary assignment of Y_1, this implies that $M \models_{Y_1} \vec{v} \subseteq \vec{x}$.

Finally, suppose that $o \in Y_2$. Then, since $o(z_1) \neq o(z_2)$, we have that $f_1(o(\vec{x})) \neq f_2(o(\vec{x}))$; and therefore, $o(\vec{x}) \notin \mathrm{Rel}_{\vec{v}}(X)$, that is, for all assignments $s \in X$ it holds that $s(\vec{v}) \neq o(\vec{x})$. Then the same holds for all $o' \in Y_2$.

This concludes the proof.

Since by Corollary 4.23 we already know independence logic and I/E logic have the same expressive power, this has the following corollary:

Corollary 6.3. *Let $\Phi(A)$ be an existential second order formula with Free(Φ) = A, and let $\vec{v}$ be any set of variables such that $|\vec{v}| = Arity(A)$. Then there exists an independence logic formula $\phi(\vec{v})$ such that*

$$M \models_X \phi(\vec{v}) \Leftrightarrow M \models \Phi(Rel_{\vec{v}}(X))$$

for all suitable models M and teams X.

Finally, by Fagin's Theorem ([10]) this gives an answer to Grädel and Väänänen's question:

Corollary 6.4. *All NP properties of teams are expressible in independence logic.*

This result has far-reaching consequences. First of all, it implies that independence logic (or, equivalently, I/E logic) is the most expressive logic of imperfect information which only deals with existential second order properties. Extensions of independence logic can of course be defined; but unless they are capable of expressing some property which is not existential second order (as, for example, is the case for the intuitionistic dependence logic of [35], or for the *BID* logic of [1]), they will be expressively equivalent to independence logic proper. As (Jouko Väänänen, private communication) pointed out, this means that independence logic is *maximal* among the logics of imperfect information which always generate existential second order properties of teams. In particular, *any* dependency condition which is expressible as an existential second order property over teams can be expressed in independence logic: and as we will see in the next section, this entails that such a logic is capable of expressing a great amount of the notions of dependency considered by database theorists.

7. Equality generating dependencies, tuple generating dependencies and independence logic

In Database Theory, two of the most general notions of dependence are *tuple generating* and *equality generating* dependencies.

In brief, a *tuple generating dependency* over a database relation R is a sentence of the form

$$\Delta(A) = \forall x_1 \ldots x_n (\phi(x_1 \ldots x_n) \rightarrow \exists z_1 \ldots z_k \psi(x_1 \ldots x_n, z_1 \ldots z_k))$$

where A is a second order variable with arity equal to the number of attributes of R.[36] and ϕ and ψ are conjunctions of atoms of the form $A\vec{t}$ or $\vec{t_1} = \vec{t_2}$ for some terms $\vec{t}$, $\vec{t_1}$ and $\vec{t_2}$ in the empty vocabulary and with free variables in $x_1 \ldots x_n$.

[36] In other words, if we consider R as a relation in first order logic then Arity(A) = Arity(R).

An *equality generating dependency* is defined much in the same way, except that ψ is a single equality atom instead.

Then, given a domain of predication M, a relation R is said to satisfy a (tuple-generating or equality-generating) dependency Δ if and only if $M \models \Delta(R)$ in the usual first order sense.

As an example of the expressive power of tuple-generating and equality-generating dependencies, let us observe that dependency atoms correspond to equality generating dependencies and that independence atoms correspond to tuple generating dependencies: indeed, for example, $M \models_X =\!(x, y)$ if and only if

$$M \models \forall x y_1 y_2 \vec{z}_1 \vec{z}_2 ((\mathrm{Rel}(X)) x y_1 \vec{z}_1 \wedge (\mathrm{Rel}(X)) x y_2 \vec{z}_2 \rightarrow y_1 = y_2)$$

where $|\vec{z}_1| = |\vec{z}_2| = |\mathrm{Dom}(X) \backslash \{x, y\}|$, and $M \models_X y \perp_x z$ if and only if

$$\begin{aligned} M \models &\forall x y_1 y_2 z_1 z_2 \vec{w}_1 \vec{w}_2 (((\mathrm{Rel}(X)) x y_1 z_1 \vec{w}_1 \wedge (\mathrm{Rel}(X)) x y_2 z_2 \vec{w}_2) \rightarrow \\ &\rightarrow \exists \vec{w}_3 (\mathrm{Rel}(X)) x y_1 z_2 \vec{w}_3). \end{aligned}$$

From the main result of the previous section, it is easy to see that I/E logic (and, as a consequence, independence logic) is capable to express all tuple and equality generating dependencies:

Proposition 7.1. *Let $\Delta(A)$ be a tuple generating or equivalent generating dependency, and let $\vec{v}$ be a tuple of distinct variables with $|\vec{v}| = Arity(A)$. Then there exists an I/E logic (or independence logic) formula $\phi(\vec{v})$ such that*

$$M \models_X \phi(\vec{v}) \Leftrightarrow M \models \Delta(Rel_{\vec{v}}(X))$$

for all suitable models M and all teams X with $\vec{v} \subseteq Dom(X)$.

Proof. $\Delta(A)$ is definable by a first order formula, and hence by Theorem 6.2 it is expressible in I/E logic (and therefore by independence logic too, by Corollary 4.23).

Hence, many of the properties which are discussed in the context of Database Theory can be expressed through independence logic. The vast expressive power of this formalism comes with a very high computational cost, of course; but it is the hope of the author that the result of this work may provide a justification to the study of this logic (and, more in general, of logics of imperfect information) as a general theoretic framework for reasoning about knowledge bases.

8. Acknowledgements

The author wishes to thank Jouko Väänänen for many valuable insights and suggestions. Furthermore, he thanks Erich Grädel for having suggested a better notation for inclusion and exclusion dependencies, and Allen Mann for having mentioned Thomas Forster's paper. Finally, the author thankfully acknowledges the support of the EUROCORES LogICCC LINT programme.

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

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

[3] Marco A. Casanova, Ronald Fagin, and Christos H. Papadimitriou. Inclusion dependencies and their interaction with functional dependencies. In *Proceedings of the 1st ACM SIGACT-SIGMOD symposium on Principles of database systems*, PODS '82, pages 171–176, New York, NY, USA, 1982. ACM.

[4] Marco A. Casanova and Vânia M. P. Vidal. Towards a sound view integration methodology. In *Proceedings of the 2nd ACM SIGACT-SIGMOD symposium on Principles of database systems*, PODS '83, pages 36–47, New York, NY, USA, 1983. ACM.

[5] Ashok K. Chandra and Moshe Y. Vardi. The implication problem for functional and inclusion dependencies is undecidable. *SIAM Journal on Computing*, 14(3):671–677, 1985.

[6] Edgar F. Codd. Further normalization of the data base relational model. In R. Rustin, editor, *Data Base Systems*, pages 33–64. Prentice-Hall, 1972.

[7] C.J. Date. *Introduction to Database Systems.* Addison Wesley, 8th edition, 2003.

[8] Arnaud Durand and Juha Kontinen. Hierarchies in dependence logic. *CoRR*, abs/1105.3324, 2011.

[9] Fredrik Engström. Generalized Quantifiers in Dependence Logic. Draft, 2010.

[10] Ronald Fagin. Generalized first-order spectra and polynomial-time recognizable sets. In *Complexity of Computation, SIAM-AMS Proceedings,*, volume Vol. 7, pages 43–73, 1974.

[11] Ronald Fagin. A normal form for relational databases that is based on domains and keys. *ACM Trans. Database Syst.*, 6:387–415, September 1981.

[12] Ronald Fagin and Moshe Vardi. The theory of data dependencies An overview. In *Automata, Languages and Programming*, pages 1–22. Springer Berlin / Heidelberg, 1984.

[13] Thomas Forster. Deterministic and nondeterministic strategies for hintikka games in first-order and branching-quantifier logic. *Logique et Analyse*, 49(195), 2006.

[14] Pietro Galliani. Epistemic operators and uniform definability in dependence logic. In Juha Kontinen and Jouko Väänänen, editors, *Proceedings of Dependence and Independence in Logic*, pages 4–29. ESSLLI 2010, 2010.

[15] Erich Grädel and Jouko Väänänen. Dependence and Independence. *Studia Logica (to appear)*, 2010.

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

[17] Jaakko Hintikka. *The Principles of Mathematics Revisited.* Cambridge University Press, 1996.

[18] Jaakko Hintikka and Jack Kulas. *The Game of Language: Studies in Game-Theoretical Semantics and Its Applications.* D. Reidel Publishing Company, 1983.

[19] Jaakko Hintikka and Gabriel Sandu. Game-Theoretical Semantics. In Johan van Benthem and Alice T. Meulen, editors, *Handbook of Logic and Language*, pages 361–410. Elsevier, 1997.

[20] Wilfrid Hodges. *A Shorter Model Theory.* Cambridge University Press, 1997.

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

[22] Wilfrid Hodges. Logics of imperfect information: why sets of assignments? In D.; van Benthem, J.; Gabbay and B. Löwe, editors, *Interactive Logic*, Texts in Logic and Games, pages 117–133. Amsterdam University Press, 2007.

[23] Theo M.V. Janssen and Francien Dechesne. Signaling in IF-Games: A Tricky Business. In J. van Benthem, G. Heinzmann, M. Rebuschi, and H. Visser, editors, *The age of alternative logics*, pages 221–241. Springer, 2006.

[24] Jarmo Kontinen. Coherence and computational complexity of quantifier-free dependence logic formulas. In Juha Kontinen and Jouko Väänänen, editors, *Proceedings of Dependence and Independence in Logic*, pages 58–77. ESSLLI 2010, 2010.

[25] Juha Kontinen and Ville Nurmi. Team logic and second-order logic. In Hiroakira Ono, Makoto Kanazawa, and Ruy de Queiroz, editors, *Logic, Language, Information and Computation*, volume 5514 of *Lecture Notes in Computer Science*, pages 230–241. Springer Berlin / Heidelberg, 2009.

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

[27] Juha Kontinen and Jouko Väänänen. A Remark on Negation of Dependence Logic. *Notre Dame Journal of Formal Logic*, 52(1):55–65, 2011.

[28] Per Lindström. First order predicate logic with generalized quantifiers. *Theoria*, 32(3):186–195, 1966.

[29] John C. Mitchell. The implication problem for functional and inclusion dependencies. *Information and Control*, 56(3):154–173, 1983.

[30] Andrzej Mostowski. On a generalization of quantifiers. *Fundamenta mathematicae*, 44:12–36, 1957.

[31] Tero Tulenheimo. Independence Friendly Logic. Stanford Encyclopedia of Philosophy, 2009.

[32] Jouko Väänänen. *Dependence Logic*. Cambridge University Press, 2007.

[33] Jouko 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.

[34] Johan van Benthem. The epistemic logic of IF games. In R.E. Auxier and L.E. Hahn, editors, *The Philosophy of Jaakko Hintikka*, volume 30 of *Library of living philosophers*, chapter 13, pages 481–513. Open Court Publishers, 2006.

[35] Fan Yang. Expressing second-order sentences in intuitionistic dependence logic. In Juha Kontinen and Jouko Väänänen, editors, *Proceedings of Dependence and Independence in Logic*, pages 118–132. ESSLLI 2010, 2010.