J Log Lang Inf (2012) 21:299–324
DOI 10.1007/s10849-012-9162-4

Generalized Quantifiers in Dependence Logic

Fredrik Engström

Published online: 10 March 2012

Abstract We introduce generalized quantifiers, as defined in Tarskian semantics by Mostowski and Lindström, in logics whose semantics is based on teams instead of assignments, e.g., IF-logic and Dependence logic. Both the monotone and the non-monotone case is considered. It is argued that to handle quantifier scope dependencies of generalized quantifiers in a satisfying way the dependence atom in Dependence logic is not well suited and that the *multivalued dependence atom* is a better choice. This atom is in fact definably equivalent to the *independence atom* recently introduced by Väänänen and Grädel.

Keywords Dependence logic · Independence friendly logic · Generalized quantifiers · Multi valued dependence

1 Introduction

Dependencies appear in many guises in both formal and natural languages. Several logical systems have been constructed bringing such quantifer scope dependencies to the forefront of the syntactical construction, but none of these handle generalized quantifiers, one of the basic tools in logic, descriptive complexity theory, and formal linguistics. The purpose of this paper is to introduce generalized quantifiers in these logical frameworks in such a way that branching, i.e. non linearity, of generalized quantifiers can be handled naturally in the logic itself.

Dependence logic, proposed by Väänänen (2007), is an elegant way of introducing dependencies between variables into the object language. It can also deal with

F. Engström (✉)
Department of Philosophy, Linguistics and Theory of Science,
University of Gothenburg, Box 200, 405 30 Göteborg, Sweden
e-mail: fredrik.engstrom@gu.se

branching of existential and universal quantifiers, but so far it cannot handle generalized quantifiers. In this paper we present a way of extending Dependence logic with generalized quantifiers.

1.1 Generalized Quantifiers and Natural Languages

When giving (parts of written) natural languages, such as English, a formal model theoretic semantics, such as in Montague (1970), several problems naturally surface. One is how to treat *determiners* such as `all`, `some` and `most`. It turns out that Mostowski's (1957) and Lindström's (1966) notions of generalized quantifiers are most useful when formalizing expressions with determiners, see Peters and Westerståhl (2006) for a thorough account of this.

According to Mostowski and Lindström a quantifier of type $\langle n_1, \ldots, n_k \rangle$, where n_i are positive natural numbers, is a class (in most cases a *proper class*) of structures in the finite relational signature $\{ R_1, \ldots, R_k \}$ where R_i is of arity n_i, closed under taking isomorphic images. For example, the meaning of the determiner *most* is commonly the type $\langle 1, 1 \rangle$ quantifier

$$\mathtt{most} = \{ (M, A, B) : |A \cap B| \geq |A \setminus B| \} .$$

Thus, a possible formalization of the proposition "most boys are tall" is

$$\mathtt{most}\, x, y\ (Bx, Ty)$$

where B is the predicate of being a boy and T that one of being tall. The truth condition for this proposition is then

$$(M, B, T) \vDash \mathtt{most}\, x, y\ (Bx, Ty) \ \text{ iff }\ (M, B, T) \in \mathtt{most} \text{ iff } |B \cap T| \geq |B \setminus T|,$$

which seems to coincide with the intuitive truth condition for the proposition. Note that we are subscribing to the sloppy style of not distinguishing between the predicate symbols and the predicates, e.g., in the above truth condition B stands for both the predicate symbol in the formula and a subset of the domain.

Given a generalized quantifier Q of type $\langle n_1, \ldots, n_k \rangle$ and a domain M, let the *local* quantifer Q_M be defined as

$$Q_M = \{ \langle A_0, A_1, \ldots, A_k \rangle \subseteq M^{n_1} \times \cdots \times M^{n_k} \mid (M, A_0, A_1, \ldots, A_k) \in Q \} .$$

Observe that local quantifiers are just sets of relations over the domain M, they are not generalized quantifiers in the strict sense. Generalized quantifiers in the strict sense we sometimes call *global* when need is to distinguish them from *local* quantifiers.

1.2 Dependence and Independence in Natural Languages

In (1974) Hintikka claims that the proposition

(∗) *Some relative of each villager and some relative of each townsmen hate each other.*

ought to be interpreted as

$$\begin{matrix}\forall x \exists y \\ \forall z \exists w\end{matrix} A(x, y, z, w)$$

where $A(x, y, z, w)$ is the quantifier free formula expressing that if x is a villager and z is a townsman then y is a relative of x, w is a relative of z, and y and w hate each other, and $\begin{matrix}\forall x \exists y \\ \forall z \exists w\end{matrix}$ is the partially ordered quantifier studied by Henkin in (1961), whose semantics is easiest expressed by its skolemization:

$$\exists f, g \forall x, z\ A(x, f(x), z, g(z)),$$

thus $y = f(x)$ may only depend on the value of x and $w = g(z)$ only on z.

However this interpretation of (∗) in terms of the branching Henkin quantifier has been strongly objected to (see for example Barwise 1979 and Gierasimczuk and Szymanik 2009) and other more natural examples of branching have been given, such as Barwise's example from (1979)

(†) *Most of the dots and most of the stars are all connected by lines.*

It should be rather clear, we think, that one natural reading of this is that there is a set of stars A which includes most stars, and a set B of dots including most dots, such that each star in A is connected to each dot in B. That is the branching reading of the sentence. Branching here means that the choice of the set of stars may not depend on the choice of any particular dot in the earlier chosen set of dots.

It seems hard to find natural examples in natural languages of branching involving only the first order quantifiers ∃ and ∀. Examples involving generalized quantifiers as in (†) above is easier to find. Another example of when branching reading is natural is with numerical quantifiers as in the following example from Davies (1989).

(+) *Two examiners marked six scripts.*

Maybe the most natural reading of (+) is

$$\begin{matrix}\exists^{=2} x \\ \exists^{=6} y\end{matrix} \big(E(x) \wedge S(y) \wedge M(x, y)\big),$$

where E is the predicate of being an examiner, S that of being a script, and $M(x, y)$ the relation of x marking y. The numerical quantifiers $\exists^{=k}$ are, even though definable in first order logic, proper generalized quantifiers.

To be able to handle branching readings of sentences like (∗) in a coherent logical framework Hintikka developed Independence Friendly logic, or IF-logic for short, in which statements of the form "there exists x, chosen independently of $\bar{y}$, such that" can be expressed by the formal construction

$$\exists x/\bar{y}\ A(x, \bar{y}).$$

Here $\bar{y}$ is a finite sequence of variable $y_0, y_1, \ldots, y_{n-1}$. We say that $\exists x/\bar{y}$ is a *slashed* quantifier. However IF-logic, as it stands, cannot handle generalized quantifiers, the chief example of branching in natural languages. This paper introduces generalized quantifiers in IF-logic, and many of its variants such as Dependence Friendly logic (DF-logic) and Dependence logic.

Barwise (1979), among others, argues that for monotone[1] quantifiers Q_1 and Q_2 of type $\langle 1 \rangle$ the branching of Q_1 and Q_2

$$\begin{matrix} Q_1 x \\ Q_2 y \end{matrix} A(x, y)$$

should be interpreted as

$$\mathrm{Br}(Q_1, Q_2)xy\ A(x, y),$$

where $\mathrm{Br}(Q_1, Q_2)$ is the type $\langle 2 \rangle$ quantifier

$$\{\, (M, R) \mid \exists A \in Q_1, B \in Q_2, A \times B \subseteq R \,\}.$$

We take this as the definition of branching of two monotone quantifiers. The correctness of that definition seems to be rather universally agreed upon. Thus, our definition of quantifiers in DF-logic should reflect upon this.

It could be worth noting that for *monotone* quantifiers Q_1 and Q_2 a formula $Q_1 x\ Q_2 y\, \varphi$ can be translated into existiential second-order logic with Q_1 and Q_2 used as second-order predicates in the following way:

$$\exists X \big(Q_1(X) \wedge \forall x{\in}X \exists Y \big(Q_2(Y) \wedge \forall y{\in}Y \varphi \big)\big).$$

In this formula it is clear that the *second-order* variable Y depends on the *first-order* variable x. By moving the $\exists Y$ outside of the scope of $\forall x \in X$ we can break this dependence. The resulting formula then becomes:

$$\exists X \exists Y \big(Q_1(X) \wedge Q_2(Y) \wedge \forall x{\in}X \forall y{\in}Y \varphi \big),$$

which is equivalent to the branching reading: $\mathrm{Br}(Q_1, Q_2)xy\, \varphi$, giving some evidence on the correctness of the definition of $\mathrm{Br}(Q_1, Q_2)$.

In the next section we will define both IF-logic and DF-logic, but first take a look at another variant of IF-logic developed by Väänänen (2007) called *Dependence logic*.

[1] A quantifier Q is monotone if given $A \subseteq B \subseteq M$ such that $(M, A) \in Q$ then $(M, B) \in Q$.

1.3 Dependence Logic and Related Logics

The syntax of Dependence logic is that of first order logic together with new atoms, the dependence atoms. There is one dependence atom for each arity written $[t_1, \ldots, t_n \to t_{n+1}]$.[2] For simplicity we will assume that all formulas are written in negation normal form, i.e., all negation signs occuring in a formula occur in front of an atomic formula. This is to make some technicalities easier, the downside of this approach is that negation cannot be treated in a compositional way. More on this later. Note also that negation in Dependence logic is not contradictory negation as; for example, we will see later that $\not\models \forall x, y([x \to y] \vee \neg [x \to y])$.

To define a compositional semantics for Dependence logic we need to consider *sets of assignments* called *teams*. Formally, an assignment is a function $s : V \to M$ where V is a finite set of variables and M is the domain under discussion. A team (on the domain M) is a set of assignments of some fixed finite set of variables V, i.e., a subset of $\{ s \mid s : V \to M \}$ for some finite set of variables V. If $V = \emptyset$ there is only one assignment $V \to M$, the empty assignment, denoted by ϵ. Please observe that the team of the empty assignment $\{ \epsilon \}$ is different from the empty team.

Given an assignment $s : V \to M$ and $a \in M$ let $s[a/x] : V \cup \{ x \} \to M$ be the assignment:

$$s[a/x] : y \mapsto \begin{cases} s(y) & \text{if } y \in V \setminus \{ x \}, \text{ and} \\ a & \text{if } x = y. \end{cases}$$

The domain of a (non-empty) team $\mathrm{dom}(X)$ is the set of variables V. The condition $M, X \models \varphi$ means that the formula φ of Dependence logic is satisfied in the structure M by the *team* X. We use the notation $M, s \models \varphi$ for ordinary Tarskian satisfaction of the first order formula φ under the assignment s. We call this type of semantics where a formula is satisfied by a team, not just a single assignment, Hodges semantics[3] to distinguish it from ordinary Tarskian semantics.

The truth conditions for $M, X \models \varphi$ are the following:

$$\begin{aligned}
M, X \models R(\bar{t}) &\text{ iff } \forall s \in X : M, s \models R(\bar{t}) \\
M, X \models \neg R(\bar{t}) &\text{ iff } \forall s \in X : M, s \models \neg R(\bar{t}) \\
M, X \models [t_1, \ldots, t_n \to t_{n+1}] &\text{ iff } \forall s, s' \in X \\
&\qquad \bigwedge_{1 \le i \le n} t_i^{M,s} = t_i^{M,s'} \to t_{n+1}^{M,s} = t_{n+1}^{M,s'} \\
M, X \models \neg [t_1, \ldots, t_n \to t_{n+1}] &\text{ iff } X = \emptyset \\
M, X \models \varphi \wedge \psi &\text{ iff } M, X \models \varphi \text{ and } M, X \models \psi \\
M, X \models \varphi \vee \psi &\text{ iff } \exists Y \cup Z = X : M, Y \models \varphi \text{ and } M, Z \models \psi
\end{aligned}$$

[2] When Väänänen introduced Dependence logic he used the notation $=(t_1, \ldots, , t_n, t_{n+1})$ for $[t_1, \ldots, t_n \to t_{n+1}]$, however we prefer the latter notation.

[3] Hodges in (1997) invented this framework in order to give IF-logic a compositional semantics.

$$M, X \models \exists y\varphi \text{ iff } \exists f : X \to M, \text{ such that } M, X[f/y] \models \varphi$$
$$M, X \models \forall y\varphi \text{ iff } M, X[M/y] \models \varphi.$$

Here $t^{M,s}$ is the interpretation of the term t in the model M under the assignment s,

$$X[M/y] \text{ is the team } \{ s[a/y] \mid s \in X, a \in M \}$$

of assignments, and when ever $f : X \to M$, $X[f/y]$ is $\{ s[f(s)/y] \mid s \in X \}$.

Observe that for some teams X we have $M, X \not\models [x \to y]$ and $M, X \not\models \neg [x \to y]$. In fact this is the case when M has at least two elements and X is the full team of all assignments of x and y. Therefore, $\not\models \forall x, y([x \to y] \vee \neg [x \to y])$. This illustrates that negation is not contradictory negation.

The free variables of a formula is defined in a recursive way, like in first order logic, with the extra base case of the dependence atom: all the variables in $\bar{x}$ and y are free in the formula $[\bar{x} \to y]$. Let $\mathrm{FV}(\varphi)$ be the set of free variables of φ. A sentence is a formula without free variables. We define $M \models \sigma$ for a sentence σ to hold if $M, \{\epsilon\} \models \sigma$.

By just staring at the definition of satisfaction we can make some remarks. First, every formula is satisfied by the empty team, which has as a consequence that for any atomic formula φ we have both $M, \emptyset \models \varphi$ and $M, \emptyset \models \neg\varphi$. Second, satisfaction is preserved under taking subteams:

Proposition 1.1 *If $M, X \models \varphi$ and $Y \subseteq X$ then $M, Y \models \varphi$.*

The next proposition might seem a bit ad hoc at first sight, but its role will later be apparent. It tells us that the truth condition for the existential quantifier is equivalent to the truth condition we later introduce for generalized quantifiers.

Proposition 1.2 *$M, X \models \exists x\varphi$ iff there exists $F : X \to \exists_M$ such that $M, X[F/x] \models \varphi$, where $X[F/x]$ is the team $\{ s[a/x] \mid s \in X, a \in F(s) \}$.*

Recall that $\exists_M$ is the local existential quantifier, i.e., the set of non-empty predicates on M : $\{ A \subseteq M \mid A \neq \emptyset \}$.

Naturally, the semantic value of a formula in Dependence logic is the set of teams satisfying the formula.

Definition 1.3 The semantic value $[\![\varphi]\!]_M$ of a formula φ in the model M is the set of teams satisfying it:

$$[\![\varphi]\!]_M = \{ X \mid \mathrm{dom}(X) = \mathrm{FV}(\varphi) \text{ and } M, X \models \varphi \}.$$

Here we have chosen one of two possible paths, the other one would be to define the semantic value of a formula to be the pair of the set of teams satisfying the formula and the set of teams that satisfy the negation of the formula: $\langle [\![\varphi]\!]_M, [\![\varphi^\neg]\!]_M \rangle$, where $\varphi^\neg$ is the formula in negated normal form that corresponds to $\neg\varphi$. That would have had the advantage of making negation compositional (i.e., a function of semantic values). However, it would also make the theory technically much more involved.

It should also be pointed out that Kontinen and Väänänen in (2009) proved that if φ and ψ are formulas in Dependence logic with the same free variables such that $[\![\varphi]\!]_M \cap [\![\psi]\!]_M = \{\emptyset\}$ then there is a formula σ in Dependence logic such that $[\![\sigma]\!]_M = [\![\varphi]\!]_M$ and $[\![\sigma^\neg]\!]_M = [\![\psi]\!]_M$. Thus the "positive" and the "negative" semantic values, taken to be $[\![\varphi]\!]_M$ and $[\![\varphi^\neg]\!]_M$ respectively, of formulas are independent, in the sense that only knowing the positive (negative) semantic value of a formula does not give any information on the negative (positive) semantic value of the same formula.

DF-logic has a different syntax than Dependence logic but a similar semantics. Instead of introducing dependence atoms we introduce new quantifiers[4] $\exists x \backslash \bar{y}$ where $\bar{y}$ is a finite sequence of variables. We call $\exists x \backslash \bar{y}$ a *backslashed* quantifier. $\exists x \backslash \bar{y}\ \varphi$ has the same truth condition as

$$\exists x([\bar{y} \to x] \wedge \varphi).$$

Independence friendly logic, IF-logic, is syntactically similar to DF-logic but with slashed quantifiers instead of backslashed ones.[5] There is a non-compositional translation of IF-logic into Dependence logic: Given a sentence σ in IF-logic we replace each occurrence of $\exists x/\bar{y}\ \varphi$ by

$$\exists x([\bar{z} \to x] \wedge \varphi)$$

where $\bar{z}$ are the variables occurring in σ but not in $\bar{y}$.

These three logics, IF-, DF- and Depedence logic, are all equivalent in the sense that for each formula in one of the logics there are formulas in the other logics satisfied by the same teams in the same structures. The translations from DF-logic to Dependence logic and back are compositional, but the translations to and from IF-logic is not.

IF-logic has one rather strange property which Dependence logic and DF-logic does not. In IF-logic an extra variable could be used for "signaling" as in the following example:

$$M \not\models \forall x \exists y/x\ x = y$$

if $|M| > 1$, but

$$\models \forall x \exists z \exists y/x\ x = y.$$

Thus quantifying over variables not occurring in a sentence might change the truth value of that sentence. This is rather counterintuitive, which should give us a slight preference for DF-logic and Dependence logic over IF-logic.

[4] Observe that these quantifiers are not *generalized* quantifiers in the sense of Lindström and Mostowski since they are defined using Hodges semantics, not Tarskian semantics.

[5] In fact, what we describe here is, strictly speaking, what Hodges in (2008) calls slash logic and not IF-logic.

2 Generalized Quantifiers

We will now give a rather long argumentation leading up to Definition 2.3 which gives truth conditions for generalized quantifiers in logics whose semantics are given in the framework of teams, such as Dependence logic. As will be apparent later, if a generalized quantifier is definable in existential second order logic, ESO, the result of adding the quantifier to Dependence logic will not change the strength of the logic, it will still be of the same strength as ESO. However, the translation into ESO will not be compositional, see the discussion in Sect. 4. The main reason for introducing generalized quantifiers in this framework is not to gain strength, but to give a compositional explanation of branching.

In the following we fix a structure and let M ambiguously denote it and its domain. We will ambiguously use Q to denote both a global quantifier and the local version on M, which really should be denoted by Q_M. We write $[\![\varphi]\!]$ as a shorthand for $[\![\varphi]\!]_M$.

Teams are sets of assignments, and thus not relations, however if X is a team with $\mathrm{dom}(X) = \{ x_1, \dots, x_k \}$ let

$$X(x_1, \dots, x_k) = \{ \langle s(x_1), \dots, s(x_k) \rangle \mid s \in X \}$$

be the relation on M we get by applying the assignments in X to the tuple $\langle x_1, \dots, x_k \rangle$. Furthermore, if $R \subseteq M^k$ let $[R/x_1, \dots, x_k]$ be the team

$$\{ \{ \langle x_1, a_1 \rangle, \dots, \langle x_k, a_k \rangle \} \mid \langle a_1, \dots, a_k \rangle \in R \} .$$

We will be quite sloppy in distinguishing between teams and relations, instead identifying the team X with the relation $X(\bar{x})$ where $\bar{x}$ is $\mathrm{dom}(X)$ listed with the indices in increasing order, and R with $[R/x_0, \dots, x_{k-1}]$ where k is the arity of R.

In (2009) Abramsky and Väänänen give an argument for the correctness of the truth conditions for $\forall$ and $\exists$ in Hodges semantics. In short the argument goes as follows: First they show that Hodges semantics is a special case of a more general construction, that of the free commutative quantale. Second, they show that the truth conditions of the quantifiers in Hodges semantics are the image under this general construction of the usual Tarskian truth conditions. Let us see how this works.

Start off by letting the Hodges space be

$$\mathcal{H}(M^n) = \mathcal{L}(\mathcal{P}(M^n))$$

where $\mathcal{L}(X)$ is the set of order ideals,[6] or *down sets*, of the ordered set X and $\mathcal{P}(M^n)$ is the power set of M^n ordered by set inclusion. Given a formula with n free variables in Dependence logic the set of relations corresponding to the teams satisfying the formula is an element of $\mathcal{H}(M^n)$, we therefore think of $\mathcal{H}(M^n)$ as the set of possible

[6] Order ideals are sets closed downwards, i.e., $I \subseteq \mathcal{P}(M^n)$ is an order ideal if for every $A \subseteq M^n$ and any $B \in I$ such that $B \subseteq AB \in I$.

semantic values of formulas.[7] Since the elements of $\mathcal{H}(M^n)$ are all closed downwards we restrict ourselves, at the moment, to logics where satisfaction is closed under taking subteams. Note that $\emptyset$ is a down set and thus an element of $\mathcal{H}(M^n)$.

If we reformulate the truth conditions for $\exists$ and $\forall$ in algebraic terms as operations mapping semantic values in $\mathcal{H}(M^{n+1})$ to semantics values in $\mathcal{H}(M^n)$ we get the following. The Hodges quantifiers $\exists_{\mathcal{H}}$ and $\forall_{\mathcal{H}}$ are families of functions

$$\begin{aligned} &\forall_{\mathcal{H}}, \exists_{\mathcal{H}} : \mathcal{H}(M^{n+1}) \to \mathcal{H}(M^n), \\ &\exists_{\mathcal{H}}(\mathscr{X}) = \{\, R \mid \exists f : R \to M \text{ s.t. } R[f] \in \mathscr{X} \,\}, \\ &\forall_{\mathcal{H}}(\mathscr{X}) = \{\, R \mid R[M] \in \mathscr{X} \,\}, \end{aligned}$$

where $R[f] = \{\, \langle \bar{a}, f(\bar{a}) \rangle \mid \bar{a} \in R \,\}$ and $R[M] = \{\, \langle \bar{a}, b \rangle \mid \bar{a} \in R, b \in M \,\}$.

The truth condition for the existential quantifier can now be restated as:

$$[\![\exists x\varphi]\!] = \big[\exists_{\mathcal{H}}\big([\![\varphi]\!][\bar{y}, x]\big)/\bar{y}\big],$$

where $\bar{y}$ are the free variables of $\exists x\varphi$. The corresponding equality is of course true also for the universal quantifier. Let us now see that these truth conditions are *forced* upon us by the operation $\mathcal{L}$.

Given a function $h : \mathcal{P}(A) \to \mathcal{P}(B)$ we define the Hodges lift of that function as:

$$\mathcal{L}(h) : \mathcal{H}(A) \to \mathcal{H}(B), \;\; \mathscr{X} \mapsto \downarrow\{\, h(X) \mid X \in \mathscr{X} \,\},$$

where $\downarrow\mathscr{X}$ is the downward closure of $\mathscr{X}$, i.e.,

$$\downarrow\mathscr{X} = \{\, X \mid \exists Y \in \mathscr{X}, X \subseteq Y \,\}.$$

To every generalized quantifier Q of type $\langle 1 \rangle$ there is a corresponding function on the Tarskian semantic values:

$$h_Q : \mathcal{P}(M^{n+1}) \to \mathcal{P}(M^n), \;\; R \mapsto \{\, \bar{a} \mid R_{\bar{a}} \in Q \,\},$$

where $R_{\bar{a}} = \{\, b \mid \langle \bar{a}, b \rangle \in R \,\}$. Now the truth condition for $\exists$ and $\forall$ in the Hodges setting is just the image under $\mathcal{L}$ of the truth conditions for $\exists$ and $\forall$ in the Tarskian setting, in the sense that: $\exists_{\mathcal{H}} = \mathcal{L}(h_\exists)$, and $\forall_{\mathcal{H}} = \mathcal{L}(h_\forall)$. These facts follow easily from the definitions, but see Proposition 2.4.

We do not have to stop here. Let us see what happens if we start with some other quantifier Q of type $\langle 1 \rangle$ and argue in the same way that led us to the truth conditions for $\exists$ and $\forall$ in the Hodges setting. Thus, for a generalized quantifier Q, let us write $Q_{\mathcal{H}}$, or $\mathcal{L}(Q)$, for $\mathcal{L}(h_Q)$. Let $Y[F] = \{\, \langle \bar{a}, b \rangle \mid \bar{a} \in Y, b \in F(\bar{a}) \,\}$.

[7] Observe that not all elements of $\mathcal{H}(M^n)$ are semantic values of formulas in Dependence logic, see Kontinen and Väänänen (2009) for a complete characterization of elements of $\mathcal{H}(M^n)$ which are.

Lemma 2.1 *Suppose $\mathscr{X} \subseteq \mathscr{P}(M^k)$ and Q a monadic quantifier.* (a) *If Q is such that $\emptyset \notin Q$, then*

$$\{ h_Q(X) \mid X \in \mathscr{X} \} = \{ Y \mid \exists F : Y \to Q \text{ s.t. } Y[F] \in \mathscr{X} \}.$$

(b) *Furthermore, for any Q, if $\mathscr{X}$ is a down set then*

$$\{ \{ \bar{a} \mid X_{\bar{a}} \in Q \} \mid X \in \mathscr{X} \}$$

is also a down set.

Proof (a) Follows from the fact that $Y[F]_{\bar{a}} = F(\bar{a})$ if $\bar{a} \in Y$ and $Y[F]_{\bar{a}} = \emptyset$ otherwise.
(b) Is immediate. □

Proposition 2.2 *For any Q of type $\langle 1 \rangle$ and any $\mathscr{X} \in \mathcal{H}(M^k)$ we have*

$$Q_{\mathcal{H}}(\mathscr{X}) = \{ Y \mid \exists F : Y \to Q \text{ s.t. } Y[F] \in \mathscr{X} \}.$$

Proof Follows directly from the lemma whenever $\emptyset \notin Q$. On the other hand if $\emptyset \in Q$ then $Q_{\mathcal{H}}(\mathscr{X}) = \mathcal{P}(M^n)$ whenever $\mathscr{X} \neq \emptyset$ and $Q_{\mathcal{H}}(\emptyset) = \emptyset$. Also if F is the constant function $F(\bar{a}) = \emptyset$ then $Y[F] = \emptyset$, thus $\{ Y \mid \exists F : Y \to Q \text{ s.t. } Y[F] \in \mathscr{X} \}$ is $\emptyset$ or $\mathcal{P}(M^n)$ depending on whatever $\mathscr{X}$ is the empty set or not. □

This all leads up to the following truth condition:

Definition 2.3 Let Q be a monotone generalized quantifiers Q of type $\langle 1 \rangle$ and φ some formula in a logic whose semantics is based on teams. We define what it means for the team X to satisfy the formula φ by the following truth condition.

$$M, X \models Qx\varphi \text{ iff there exists } F : X \to Q \text{ such that } M, X[F/x] \models \varphi.$$

This applies even for non-monotone quantifiers but for those quantifiers Q the truth condition above does not make a whole lot of sense as the following example shows. Let $M = \mathbb{N}$ and $Q = \{ A \}$ where A is the set of even numbers. According to the truth condition above $M, \{ \epsilon \} \models Qx(x = x)$ since there is a team $X = A(x)$ such that $M, X \models x = x$.

For this reason let us, for now, restrict the definitions to monotone quantifiers.

We do not however need to restrict to type $\langle 1 \rangle$ as the definition easily can be extended to all quantifiers of type $\langle k \rangle$:

$$M, X \models Q\bar{x}\varphi \text{ iff there exists } F : X \to Q \text{ such that } M, X[F/\bar{x}] \models \varphi.$$

Here $X[F/\bar{x}]$ is the team $\{ s[a_1/x_1, \dots a_k/x_k] \mid s \in X, \langle a_1, \dots a_k \rangle \in F(s) \}$.

The following easy proposition suggests that we indeed have the right truth condition, at least for monotone quantifiers of type $\langle 1 \rangle$. Given some language L let $L(Q)$ the set of first order formulas in that language extended with the generalized quantifier Q.

Proposition 2.4 *Below, let Q be a monotone quantifier of type $\langle 1 \rangle$.*

(1) $\mathcal{L}(\exists)(\mathscr{X}) = \{ Y \mid \exists f : Y \to M \text{ s.t. } Y[f] \in \mathscr{X} \}$.
(2) $\mathcal{L}(\forall)(\mathscr{X}) = \{ Y \mid Y[M] \in \mathscr{X} \}$.
(3) *For L(Q)-formulas φ and teams X, $M, X \vDash \varphi$ iff for all $s \in X$, $M, s \vDash \varphi$.*
(4) *For L(Q)-sentences σ, $M, \epsilon \vDash \sigma$ iff $M, \{ \epsilon \} \vDash \sigma$.*[8]
(5) $\mathcal{L}(Q_1 Q_2) = \mathcal{L}(Q_1) \circ \mathcal{L}(Q_2)$, *where $Q_1 Q_2$ is the iteration (product) of Q_1 and Q_2 and $\mathcal{L}(Q_1) \circ \mathcal{L}(Q_2)$ is just the ordinary composition of functions (observe that this equality is really an infinite conjunction of equalities since $\mathcal{L}(Q)$ is a family of functions).*

Proof (1) The right-to-left inclusion is immediate using Proposition 2.2. The other inclusion follows from the fact that if $Y[F] \in \mathscr{X}$ and $f : Y \to M$ is such that $f(\bar{a}) \in F(\bar{a})$ for each $\bar{a} \in Y$ then $Y[f] \subseteq Y[F] \in \mathscr{X}$ and thus $Y[f] \in \mathscr{X}$.

(2) Any function $F : X \to \forall$ has to be the constant function taking s to M. Thus $\mathcal{L}(\forall)(\mathscr{X}) = \{ Y \mid Y[M] \in \mathscr{X} \}$ follows from Proposition 2.2.

(3) The argument is an induction on the formula φ, the only non trivial case being when φ is $Qx\psi$. If $M, X \vDash Qx\psi$ then there is a function $F : X \to Q$ such that $M, X[F/x] \vDash \psi$, which means that for each $s \in X$ there is a set $F(s) \in Q$ such that $F(s) \subseteq \psi^{M,s}$, where $\psi^{M,s} = \{ a \in M \mid M, s[a/x] \vDash \psi \}$. By the monotonicity of Q we have $\psi^{M,s} \in Q$ and thus that $M, s \vDash Qx\psi$ for every $s \in X$. For the other direction suppose $M, s \vDash Qx\psi$ for every $s \in X$, and let $F(s)$ be $\psi^{M,s}$.

(4) Follows directly from (3).

(5) By unwinding the definitions we get

$$\begin{aligned} \mathcal{L}(Q_1) \circ \mathcal{L}(Q_2)(\mathscr{X}) &= \mathcal{L}(Q_1)(\{ Y \mid \exists F : Y \to Q_2 \text{ s.t. } Y[F] \in \mathscr{X} \}) \\ &= \{ Z \mid \exists G : Z \to Q_1 \text{ s.t. } \exists F : Z[G] \to Q_2, Z[G][F] \in \mathscr{X} \} \\ &= \{ Z \mid \exists H : Z \to Q_1 Q_2 \text{ s.t. } Z[H] \in \mathscr{X} \}. \end{aligned}$$

The left to right inclusion in the last equality comes from the fact that if such F and G exist then we can define $H(s)$ to be

$$\{ \langle a, b \rangle \mid a \in G(s), b \in F(s, a) \} \in Q_1 Q_2$$

and then $Z[H] = Z[G][F]$. For the other inclusion assume that such an H is given and let $a \in G(s)$ if $H(s)_a \in Q_2$ and $F(s, a) = H(s)_a$, then $F : Z \to Q_1, G : Z[F] \to Q_2$ and $Z[G][F] \subseteq Z[H] \in \mathscr{X}$. □

2.1 Quantifiers and Dependence

Proposition 2.4 states that the truth condition for generalized quantifiers in the Hodges setting behaves nicely when applied to formulas without dependence atoms. The reason for introducing generalized quantifiers in Hodges semantics however is to use

[8] Observe that $M, \epsilon \vDash \sigma$ uses the ordinary Tarskian truth conditions, but $M, \{ \epsilon \} \vDash \sigma$ Hodges semantics.

them with dependencies. Let us see how well they handle relations of dependences and independences.

First let us try to see what happens if we introduce the dependence atom of Dependence logic into our logic. If Q contains no singletons and not the empty set then

$$M \nvDash Qx([\to x] \wedge x = x)$$

as long as X is non-empty.[9] This is counterintuitive since the sentence $Qx([\to x] \wedge x = x)$ should be equivalent to $Qx\, x = x$. There are also problems with the notion of definability as the next example shows

Assume that Q is definable by a first order sentence σ with P as the only unary predicate. This means that

$$M \vDash Qx\varphi \text{ iff } M \vDash \sigma[\varphi/P]$$

for all first order formulas φ such that no free variables of φ occur in σ.[10] It would be natural to think that σ also defines Q in Dependence logic. However if $Q = \exists^{\geq 2}$, σ is

$$\exists x \exists y (x \neq y \wedge Px \wedge Py),$$

and φ is $[\to z]$, then $\nvDash \exists^{\geq 2} z\varphi$. But for any $|M| > 1$, $M \vDash \sigma[\varphi/P]$. Thus, introducing dependence atoms into the language seems to destroy nice properties of the logic. However, we think that the dependence atom should take the blame for this, and not the truth condition for the generalized quantifier.

There are two different solutions for handling dependencies in the setting with generalized quantifiers. Either one redefines the dependence atom, or one skips dependence as an atomic property altogether and define slashed and/or backslashed versions of the generalized quantifiers, very much as is done in DF- and IF-logic. Let us start with the latter suggestion and postpone the definition of a new dependence atom until Sect. 3.

$$\begin{aligned} M, X \vDash Qx\backslash\bar{y}\,\varphi \text{ iff there exists } F : X \to Q \text{ such that} \\ M, X[F/x] \vDash \varphi \text{ and } F \text{ depends only on the values of } \bar{y}. \end{aligned}$$

Or slightly more formally:

Definition 2.5 $M, X \vDash Qx\backslash\bar{y}\varphi$ iff there exists

$$G : X\restriction\bar{y} \to Q \text{ such that } M, X[G/x] \vDash \varphi,$$

where $s\restriction\bar{y} = \{\langle v, a\rangle \mid \langle v, a\rangle \in s, v \in \bar{y}\}$, $X\restriction\bar{y} = \{s\restriction\bar{y} : s \in X\}$ and $X[G/x] = X[F/x]$ where $F(s) = G(s\restriction\bar{y})$.

[9] $[\to x]$ is a short-hand for $[\emptyset\to x]$, i.e., the statement that x is *constant*.

[10] $\sigma[\varphi/P]$ is σ with all occurences of $P(x_1, \ldots, x_k)$ replaced by $\varphi(x_1, \ldots, x_k)$.

We also define $M, X \models Qx/\bar{y}\,\varphi$ in the obvious way: iff there exists

$$F : X{\restriction}(\mathrm{dom}(X) \setminus \bar{y}) \to Q$$

such that $M, X[F/x] \models \varphi$.

Let us denote first order logic with the generalized quantifiers Q by SBL(Q) when we allow both slashed and backslashed versions of the quantifiers $\exists$, $\forall$, and Q. Note that we can translate backslashed quantifiers into formulas where we only allow slashed ones, and vice versa. However, these translations are not compositional and therefore we include both slashed and backslashed quantifiers in the logic SBL(Q).

Proposition 2.6 *Let Q be a monotone quantifier of type $\langle k\rangle$.* SBL(Q) *is closed under taking subteams, i.e., if $M, X \models \varphi$ and $Y \subseteq X$ then $M, Y \models \varphi$ for all formulas φ in* SBL(Q).

Proof Easily seen by checking the truth conditions of the slashed and the backslashed quantifiers. □

Observe that

$$\exists F : X \to \exists_M \text{ s.t. } X[F/x] \in \mathscr{X} \text{ and } F \text{ only depends on } \bar{y}$$
$$\text{iff}$$
$$\exists f : X \to M \text{ s.t. } X[f/x] \in \mathscr{X} \text{ and } f \text{ only depends on } \bar{y},$$

for every down set $\mathscr{X}$, making this new definition of $\exists x\backslash\bar{y}$ compatible with the old one. Also, it is easy to see that both conditions are equivalent to the corresponding sentence in Dependence logic:

Proposition 2.7 *Let Q be a monotone quantifier of type $\langle k\rangle$ and φ a formula in* SBL(Q) *then*

$$M, X \models \exists x([\bar{y}{\to}x] \wedge \varphi) \;\textit{ iff }\; M, X \models \exists x\backslash\bar{y}\,\varphi.$$

Let Q_1 and Q_2 be monotone quantifiers of type $\langle k\rangle$ and $\langle l\rangle$ respectively, then

$$\mathrm{Br}(Q_1, Q_2) = \{\, (M, R) \mid R \subseteq M^{k+l}, \exists A{\in}Q_1\, \exists B{\in}Q_2 : A \times B \subseteq R \,\}.$$

The next proposition states that $Q_1\bar{x}\,Q_2\bar{y}/\bar{x}$ has the intended meaning $\mathrm{Br}(Q_1, Q_2)$

Proposition 2.8 *If Q_1 and Q_2 are monotone quantifiers of type $\langle k\rangle$ and $\langle l\rangle$ respectively, then*

$$M, X \models \mathrm{Br}(Q_1, Q_2)\bar{x}\bar{y}\,\varphi \;\textit{ iff }\; M, X \models Q_1\bar{x}\,Q_2\bar{y}/\bar{x}\,\varphi,$$

where φ is in SBL(Q_1, Q_2).

Proof To simplify notation we only prove this in the case that $k = l = 1$. Assume that $M, X \vDash \mathrm{Br}(Q_1, Q_2)xy\,\varphi$, i.e., there if $H : X \to \mathrm{Br}(Q_1, Q_2)$ such that $M, X[H] \vDash \varphi$. For $s \in X$ define $F(s) = A$ and $G(s) = B$ where $A \in Q_1$ and $B \in Q_2$ are such that $A \times B \subseteq H(s)$. Now $X[F][G] \subseteq X[H]$ and since $\mathrm{SBL}(Q_1, Q_2)$ is closed under taking subteams we have that $X[F][G] \vDash \varphi$. The functions F and G witness that $M, X \vDash Q_1 x Q_2 y/x\,\varphi$.

On the other hand if there are such F and G witnessing that $M, X \vDash Q_1 x Q_2 y/x\,\varphi$ let $H(s) = F(s) \times G(s)$. Then $X[H] = X[F][G]$ and so H witnesses that $M, X \vDash \mathrm{Br}(Q_1, Q_2)xy\,\varphi$. □

Thus, the slashed and backslashed quantifiers seem to have the intended meanings. One oddity arises with the universal quantifier: In Dependence logic we have that $M \nvDash \forall x([\to x] \wedge \varphi)$ for every structure M with at least two elements. However with the backslashed universal quantifier we have that

$$M \vDash \forall x \backslash \epsilon\, \varphi \leftrightarrow \forall x \varphi.$$

Thus, the constructions $\exists x \backslash \epsilon$ and $\exists x([\to x] \wedge \cdot)$ are equivalent, however the analogous constructions $\forall x \backslash \epsilon$ and $\forall x([\to x] \wedge \cdot)$ are *not* equivalent.

2.2 Non-monotone Quantifiers

The truth condition for monotone quantifiers cannot be extended to non-monotone quantifiers as we have previously seen. However, with a slight twist the truth condition can actually be extended.

In the Tarskian setting we say that $M \vDash Qx\varphi$ iff $[\![\varphi]\!] \in Q$. When Q is monotone (increasing) this is equivalent to demanding that there is some set $A \subseteq [\![\varphi]\!]$ such that $A \in Q$, a fact we used for the truth conditions in the Hodges setting. When dealing with non-monotone quantifiers Q we need, apart from that there is a set $A \subseteq [\![\varphi]\!]$ such that $A \in Q$, also that it is the *largest* set satisfying $A \subseteq [\![\varphi]\!]$.

Translating this into the Hodges setting we need to say not only that $M, X[F/x] \vDash \varphi$ but also that F is a *maximal* function for which this holds. However, there might not be a maximal F such that $X[F/x]$ satisfies φ. Instead we demand that there should be one F such that $X[F/x]$ satisfies φ and such that every larger F such that $X[F/x]$ also satisfies φ is mapping X into Q. Let us try to formalize this in the following definition:

Definition 2.9 Given $F, F' : X \to \mathscr{P}(M)$ let $F \leq F'$ if for every $s \in X : F(s) \subseteq F'(s)$. Let Q be a type $\langle 1 \rangle$ quantifier. Then $M, X \vDash Qx\varphi$ if there is $F : X \to \mathscr{P}(M)$ such that

(1) $M, X[F/x] \vDash \varphi$ and
(2) for each $F' \geq F$ if $M, X[F'/x] \vDash \varphi$ then for all $s \in X : F(s) \in Q$.

We call the second condition on F the *largeness condition* since it forces the function F to take large sets as values.

This definition generalizes to other types of quantifiers as in the case of monotone quantifiers. If Q is of type $\langle k \rangle$ then $M, X \vDash Q\bar{x}\varphi$ iff there is $F : X \to \mathscr{P}(M^k)$ satisfying (slight variants of) the two conditions above.

This largeness condition we have added is quite similar to Sher's maximality principle for the branching of non-monotone quantifiers, see Sher (1990). However, as we will see, the above condition gives rise to a slightly stronger notion of branching than the one proposed by Sher.

We can clearly add this largeness condition to slashed and backslashed non-monotone quantifiers: $M, X \vDash Qx/\bar{y}\,\varphi$ if there is a witness $F : X \to \mathscr{P}(M)$ to $M, X \vDash Qx\varphi$ where F is independent of the values of $\bar{y}$. Observe here that demanding F to be a witness for the statement $M, X \vDash Qx\,\varphi$ means that F is large with respect to *all* functions $F' : X \to \mathscr{P}(M)$ and not only those that are determined by the values of $\bar{y}$. We define $M, X \vDash Qx\backslash\bar{y}\,\varphi$ in a similar manner.

Proposition 2.10 *For Q monotone the truth condition with the largeness condition in Definition 2.9 is equivalent to the old condition without the largeness condition.*

Proof Obvious since if there is $F : X \to Q$ and $F' \geq F$ then $F'(s) \in Q$ for all $s \in X$ by the monotonicity of Q. □

Proposition 2.11 *If φ is an $L(Q)$-formula then*

$$M, X \vDash \varphi \textit{ iff for all } s \in X : M, s \vDash \varphi.$$

Proof The proof is by induction. We only need to check the induction step for the quantifier Q. Assume $M, X \vDash Qx\varphi$. Let F be such that $X[F/x] \vDash \varphi$ and satisfying the largeness condition, and let $s \in X$. By the induction hypothesis we have that for all $a \in F(s) : M, s[a/x] \vDash \varphi$. Therefore $F(s) \subseteq [\![\varphi]\!]_{M,s}$ and by the largeness condition on F we know that $[\![\varphi]\!]_{M,s} \in Q_M$ and therefore $M, s \vDash Qx\varphi$.

On the other hand if for all $s \in X : M, s \vDash \varphi$, then we let $F(s) = [\![\varphi]\!]_{M,s}$. It is clear that $F(s) \in Q_M$ for all $s \in X$ and also that there cannot be any $F' > F$ such that $M, X[F'/x] \vDash \varphi$ since that would violate the definition of F. Therefore $M, X \vDash Qx\varphi$. □

Definition 2.12 (*Sher 1990*) A cartesian product $A \times B$ is *maximal in* R if $A \times B \subseteq R$, no $A' \supsetneq A$ satisfies $A' \times B \subseteq R$ and no $B' \supsetneq B$ satisfies $A \times B' \subseteq R$.

Let the branching of two type $\langle 1 \rangle$ quantifiers Q_1 and Q_2, $\mathrm{Br}^S(Q_1, Q_2)$ be the type $\langle 2 \rangle$ quantifier

$$\{\,(M, R) \mid R \subseteq M^2, \exists A{\in}Q_1 \exists B{\in}Q_2 : A \times B \text{ is maximal in } R\,\}.$$

Lemma 2.13 *If $R \notin \mathrm{Br}^S(Q_1, Q_2)$ then for every $A \in Q_1$ and $B \in Q_2$ if $A \times B \subseteq R$ there is either*

(1) *$A' \supseteq A$ such that $A' \notin Q_1$ and $A' \times B \subseteq R$, or*
(2) *$B' \supseteq B$ such that $B' \notin Q_2$ and $A \times B' \subseteq R$.*

Proof Suppose not and let A_0 be the union of all $A' \supseteq A$ such that $A' \times B \subseteq R$. Then $A_0 \times B \subseteq R$, and by the assumption $A_0 \in Q_1$. Let B_0 be the union of all $B' \supseteq B$ such that $A_0 \times B' \subseteq R$. Then $A_0 \times B_0 \subseteq R$ and thus $A \times B_0 \subseteq R$ and by the assumption $B_0 \in Q_2$.

$A_0 \times B_0$ is maximal in R by construction, hence $R \in \mathrm{Br}^S(Q_1, Q_2)$ contradicting the assumption. □

Proposition 2.14 *Suppose that φ is such that $[\![\varphi]\!]$ is closed downwards (φ could for example be a formula of $L(Q)$ or of Dependence logic). If $M, X \vDash Q_1 x Q_2 y/x\varphi$ then $M, X \vDash \mathrm{Br}^S(Q_1, Q_2)xy\varphi$.*

Proof Assume that $M, X \vDash Q_1 x Q_2 y/x\varphi$, i.e., that there are $F, G : X \to \mathscr{P}(M)$ satisfying the relevant largeness condition. To prove that $M, X \vDash \mathrm{Br}^S(Q_1, Q_2)xy\varphi$ we need to find an $H : X \to \mathscr{P}(M^2)$ witnessing the truth condition. Let $H(s) = F(s) \times G(s)$. We need to prove that (1) $M, X[H/xy] \vDash \varphi$ and (2) that for any $H' \geq H$ such that $M, X[H'/xy] \vDash \varphi$ we have that $H'(s) \in \mathrm{Br}^S(Q_1, Q_2)$.

(1) Since $X[H/xy] = X[F/x][G/y]$; $M, X[H/xy] \vDash \varphi$ follows from the assumption that

$$M, X[F/x][G/y] \vDash \varphi.$$

(2) Assume that $H' \geq H$, $M, X[H'/xy] \vDash \varphi$ and $H'(s_0) \notin \mathrm{Br}^S(Q_1, Q_2)$ for some $s_0 \in X$. By the lemma we either have $A \notin Q_1$ such that

$$F(s_0) \times G(s_0) \subseteq A \times G(s_0) \subseteq H'(s_0)$$

or $B \notin Q_2$ such that

$$F(s_0) \times G(s_0) \subseteq F(s_0) \times B \subseteq H'(s_0).$$

First assume that we have such a B. Then G does not satisfy the largeness condition, i.e., that for every $G' \geq G$ if $M, X[F/x][G'/y] \vDash \varphi$ then $G'(s) \in Q_2$ for all $s \in X$, this is because we could define G' as G except that $G'(s_0) = B$.

On the other hand suppose we have such an A and let F' be as F except that $F'(s) = A$. Then $F' \geq F$ and if we prove that $M, X[F'/x] \vDash Q_2 y/x\,\varphi$ that contradicts the largeness of F since $F'(s_0) \notin Q_1$. Since $X[F'/x][G/y] \subseteq X[H'/xy]$ and $M, X[H'/xy] \vDash \varphi$ we have that $M, X[F'/x][G/y] \vDash \varphi$. We also need to prove that G satisfies the largeness condition. Let $G' \geq G$ be such that $M, X[F'/x][G'/y] \vDash \varphi$, then $M, [F/x][G'/y] \vDash \varphi$ and since G satisfies the largeness condition for $X[F/x]$ we know that $G'(s) \in Q_2$ for all $s \in X$, proving the largeness condition for G with $X[F'/x]$. □

We leave the question as wether the proposition holds for general formulas φ open.

Question 1 Is Proposition 2.14 also true for formulas φ such that $[\![\varphi]\!]$ is not a down set?

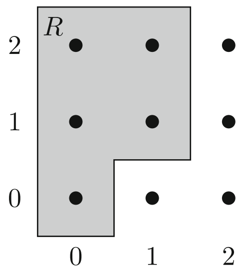


Fig. 1 Example of a relation R satisfying $\mathrm{Br}^S(\exists^{=1}, \exists)xy$ and $\exists y \exists^{=1} x/y$ but not $\exists^{=1} x \exists y/x$

The implication in the other direction is in general false as the following example shows. Let

$$R = \{\, \langle 0, 0 \rangle \,\} \cup \big(\{\, 0, 1 \,\} \times \{\, 1, 2 \,\}\big)$$

where $0, 1, 2 \in M$, see Fig. 1. Then

$$(M, R) \models \mathrm{Br}^S(\exists^{=1}, \exists)xy\; R(x, y)$$

since $\{\, 0 \,\} \times \{\, 0, 1, 2 \,\}$ is maximal in R and $\{\, 0 \,\}$ is in $\exists^{=1}$ and $\{\, 0, 1, 2 \,\}$ is in $\exists$. However,

$$(M, R) \not\models \exists^{=1} x \exists y/x\; R(x, y)$$

since to get $(M, R), A \times B \models R(x, y)$, $A \in \exists^{=1}$ and $B \neq \emptyset$, we are forced to choose $A = \{\, 0 \,\}$ or $A = \{\, 1 \,\}$. But then the largeness condition for A is not satisfied.

This example also shows that in general the two quantifier prefixes $Q_1 x Q_2 y/x$ and $Q_2 y Q_1 x/y$ are not equivalent:

$$(M, R) \not\models \exists^{=1} x \exists y/x\; R(x, y)$$

but

$$(M, R) \models \exists y \exists^{=1} x/y\; R(x, y).$$

We end this section with two open questions regarding this pathology.

Question 2 Are there natural conditions under which the prefixes $Q_1 x Q_2 y/x$ and $Q_2 y Q_1 x/y$ are equivalent?

Question 3 Are there other truth conditions for non-monotone quantifiers (and slashed versions) such that

(1) for monotone quantifiers the truth conditions coincide with the ones for the monotone case,
(2) for formulas of $L(Q)$ we have $M, X \models \varphi$ iff for all $s \in X : M, s \models \varphi$, and
(3) the prefixes $Q_1 x Q_2 y/x$ and $Q_2 y Q_1 x/y$ are equivalent?

Our proposed truth conditions satisfy (1) and (2), but not (3).

3 Dependence as an Atom

Let us now investigate the possibility of defining a new dependence atom $D(\bar{x}, y)$ giving the intended meaning $Qy\backslash\bar{x}\,\varphi$ to expressions of the form $Qy(D(\bar{x}, y) \wedge \varphi)$. First we observe that there is no way of doing this if we want to keep the property of the logic being closed under taking subteams. The following argument shows this.

Assume that $D(x, y)$ is an atom closed under subteams satisfying that

$$\models \forall x \exists^{\geq 3} y \big(D(x, y) \wedge R(x, y)\big) \;\leftrightarrow\; \forall x \exists^{\geq 3} y \backslash\epsilon\ R(x, y).$$

Fix $M = \{0, 1, 2\}$, then

$$(M, M^2) \models \forall x \exists^{\geq 3} y \backslash\epsilon\ R(x, y).$$

Thus, $X = [M^2/x, y]$ has to satisfy $D(x, y)$. By the downward closure of D, we have that the team

$$X = [S/x, y], \text{ where } S = (\{0, 1\} \times \{0, 1\}) \cup (\{2\} \times \{1, 2\}), \tag{1}$$

satisfies the atom D and thus that

$$(M, S) \models \forall x \exists^{\geq 2} y (D(x, y) \wedge R(x, y)),$$

however

$$(M, S) \not\models \forall x \exists^{\geq 2} y \backslash\epsilon\ R(x, y).$$

This argument shows that no atom $D(x, y)$ closed under taking subteams can have the intended effect on both the quantifiers $\exists^{\geq 2}$ and $\exists^{\geq 3}$.

However, by abandoning the property that truth is closed under subteams we can define an atom satisfying the right equivalences. To get the mind on the right track let us go back and take a look at the formula $Q_1 x\ Q_2 y\ \varphi$ and its translation into existential second order logic with Q_1 and Q_2 as second order predicates:

$$\exists X \big(Q_1(X) \wedge \forall x{\in}X \exists Y \big(Q_2(Y) \wedge \forall y{\in}Y \varphi\big)\big).$$

In this translation it is clear that the variable Y depends on the variable x. Thus a quantifier prefix like $Q_1 x\ Q_2 y$ gives rise to a dependence in which the value of x determines the *set* Y of possible values for y. The new dependence atom, which we denote by $\big[x_1 \ldots, x_k \twoheadrightarrow x_{k+1}\big]$, tries to capture this type of dependence in which the *set of possible values* of x_{k+1} is determined by the values of the variables $x_1, \ldots, x_k$. We formalize this idea, but first we need a little bit of notation to work with.

Let $X_s^{\bar{y}}$, for $s \subseteq s' \in X$ and $\bar{y} \in \mathrm{dom}(X)$, be the set of possible values of $\bar{y}$ given s, in other words:

Definition 3.1 Given a team X, variables $\bar{y} \in \mathrm{dom}(X)$ and $s \subseteq s'$ for some $s' \in X$, let

$$X_s^{\bar{y}} = \{\, s'(\bar{y}) \mid \exists s' \in X, s \subseteq s' \,\}\,.$$

As an example let X be as in (1) and $s : x \mapsto 1$ then $X_s^y = \{\,0, 1\,\}$ and if $s' : x \mapsto 2$ then $X_{s'}^y = \{\,1, 2\,\}$.

Definition 3.2 Assume $\bar{x}, \bar{y} \in \mathrm{dom}(X)$, then

- $M, X \models [\bar{x} \twoheadrightarrow \bar{y}]$ iff for all $s \in X$, $X^{\bar{y}}_{s\restriction\bar{x}} = X^{\bar{y}}_{s\restriction\bar{x}\bar{z}}$, where $\bar{z} = \mathrm{dom}(X) \setminus \{\,\bar{x}, \bar{y}\,\}$.
- $M, X \models \neg\,[\bar{x} \twoheadrightarrow \bar{y}]$ iff $X = \emptyset$.

In fact, this is functional dependence for set-valued functions: Assume that we want to check whether $M, X \models [\bar{x} \twoheadrightarrow \bar{y}]$ or not. Let F map $s \in X\restriction\bar{x}, \bar{z}$ to the set of possible values of $\bar{y}$, $X_s^{\bar{y}}$. Then $M, X \models [\bar{x} \twoheadrightarrow \bar{y}]$ iff $F(s)$ is determined by the values $s(\bar{x})$.

Next we give an equivalent definition for $[\bar{x} \twoheadrightarrow \bar{y}]$ this time as a first order property of X.

Proposition 3.3 *$M, X \models [\bar{x} \twoheadrightarrow \bar{y}]$ iff*

$$\forall s, s' {\in} X \Big(s(\bar{x}) = s'(\bar{x}) \to \exists s_0 {\in} X \big(s_0(\bar{x}, \bar{y}) = s(\bar{x}, \bar{y}) \wedge s_0(\bar{z}) = s'(\bar{z}) \big) \Big),$$

where $\{\,\bar{z}\,\} = dom(X) \setminus \{\,\bar{x}, \bar{y}\,\}$.

Proof Assume $M, X \models [\bar{x} \twoheadrightarrow \bar{y}]$ and $s, s' \in X$ such that $s(\bar{x}) = s'(\bar{x})$. Then $X^{\bar{y}}_{s\restriction\bar{x}\bar{z}} = X^{\bar{y}}_{s'\restriction\bar{x}\bar{z}}$. Clearly $s(\bar{y}) \in X^{\bar{y}}_{s\restriction\bar{x}\bar{z}}$ and thus $s(\bar{y}) \in X^{\bar{y}}_{s'\restriction\bar{x}\bar{z}}$ which means that there is $s_0 \in X$ such that $s_0 \supseteq s'\restriction\bar{x}\bar{z}$ and $s_0(\bar{y}) = s(\bar{y})$, i.e., that $s_0(\bar{x}, \bar{z}) = s'(\bar{x}, \bar{z})$ and $s_0(\bar{y}) = s(\bar{y})$.

For the other implication let $s \in X$, we show that $X^{\bar{y}}_{s\restriction\bar{x}} = X^{\bar{y}}_{s\restriction\bar{x},\bar{z}}$. It should be clear that $X^{\bar{y}}_{s\restriction\bar{x}} \supseteq X^{\bar{y}}_{s\restriction\bar{x},\bar{z}}$, so let $\bar{a} \in X^{\bar{y}}_{s\restriction\bar{x}}$. Then there is $s' \in X$ such that $s' \supseteq s\restriction\bar{x}$ and $s'(\bar{y}) = \bar{a}$. By assumption there is a $s_0 \in X$ such that $s_0(\bar{x}, \bar{y}) = s'(\bar{x}, \bar{y})$ and $s_0(\bar{z}) = s(\bar{z})$, or in other words $s_0 \supseteq s\restriction\bar{x}, \bar{z}$ and $s_0(\bar{y}) = \bar{a}$, i.e., that $\bar{a} \in X^{\bar{y}}_{s\restriction\bar{x},\bar{z}}$. □

The dependence relation $[\bar{x} \twoheadrightarrow \bar{y}]$ is what database theorists call *multivalued dependence*, see Beeri et al. (1977).

By some easy calculations we find that any X satisfying both $[\bar{x} \twoheadrightarrow y]$ and $[\bar{x}, y \twoheadrightarrow z]$ also satisfies $[\bar{x} \twoheadrightarrow y, z]$, i.e.,

$$[\bar{x} \twoheadrightarrow y]\,, [\bar{x}, y \twoheadrightarrow z] \models [\bar{x} \twoheadrightarrow y, z]\,.$$ [11]

[11] Here, by $\Gamma \models \varphi$ we mean that for every model M and team X, whose domain includes at least all free variable of Γ and φ, if $M, X \models \gamma$ for all $\gamma \in \Gamma$ then also $M, X \models \varphi$.

However it is not in general the case that an X satisfying $[\bar{x} \twoheadrightarrow y, z]$ satisfies $[\bar{x} \twoheadrightarrow y]$, cf., the case of functional dependence where $[\bar{x} \to y] \wedge [\bar{x} \to z]$ is equivalent to $[\bar{x} \to y, z]$. Here by $M, X \vDash [\bar{x} \to \bar{y}]$ we mean

$$\forall s, s' \in X\big(s(\bar{x}) = s'(\bar{x}) \to s(\bar{y}) = s'(\bar{y})\big).$$

It should also be noted that in the case of functional dependence the dependence atom is *not dependent on context* in the sense that

$$M, X \vDash [\bar{x} \to \bar{y}] \text{ iff } M, X \restriction \bar{x}, \bar{y} \vDash [\bar{x} \to \bar{y}]\,,$$

where $X \restriction \bar{x}$ is the team $\{ s \restriction \bar{x} \mid s \in X \}$. However multivalued dependencies are dependent on context as the following easy examples shows. $M, X \restriction x \vDash [\twoheadrightarrow x]$ is always true disregarding what X is. On the other hand if $s(x) = s(y) = 0$ and $s'(x) = s'(y) = 1$ then $M, \{ s, s' \} \nvDash [\twoheadrightarrow x]$.

There is a close connection between lossless decomposition of databases and multivalued dependencies: Let $X \bowtie Y$ be the *natural join* of the teams X and Y, i.e.,

$$X \bowtie Y = \{ s : \text{dom}(X) \cup \text{dom}(Y) \to M \mid s \restriction \text{dom}(X) \in X \text{ and } s \restriction \text{dom}(Y) \in Y \}\,.$$

Proposition 3.4 (Fagin 1977)

$$X \vDash [\bar{x} \twoheadrightarrow \bar{y}] \textit{ iff } X = (X \restriction \bar{x}\bar{y}) \bowtie (X \restriction \bar{x}\bar{z}),$$

where $\bar{z}$ is dom$(X) \setminus \{ \bar{x}, \bar{y} \}$.

Observe that it follows that $M, X \vDash [\twoheadrightarrow \bar{y}]$ iff there are teams Y and Z such that

$$\text{dom}(Y) = \{ \bar{y} \}\,, \text{dom}(Z) = \text{dom}(X) \setminus \{ \bar{x}, \bar{y} \}\,, \text{ and } X = Y \bowtie Z.$$

In this case, when dom(Y) and dom(Z) are disjoint, the natural join of Y and Z is nothing more than the cartesian product.

Next we prove that the functional dependence may be replaced be multivalued dependence in a certain well-behaved syntactical fragment of Dependence logic. This fragment is as expressive as full Dependence logic at the level of sentences.

Proposition 3.5 *Let Q be monotone of type $\langle 1 \rangle$ and σ a sentence in* SBL(Q) *with no slashed quantifiers, then the resulting sentence $\sigma^{\twoheadrightarrow}$ in which all occurrences of $Qy \backslash \bar{x}\ \varphi$ are replaced by $Qy([\bar{x} \twoheadrightarrow y] \wedge \varphi)$ is equivalent to σ.*

Proof We prove the more general statement that for every formula ψ of SBL(Q) with no slashed quantifiers, the resulting formula $\psi^{\twoheadrightarrow}$ in which all occurrences of $Qy \backslash \bar{x}\ \varphi$ are replaced by $Qy([\bar{x} \twoheadrightarrow y] \wedge \varphi)$ is equivalent to ψ. The proof is by induction on the formula ψ. The only non trivial case is when ψ is $Qy \backslash \bar{x}\ \varphi$. Then $M, X \vDash \psi$

iff there is $F : X \to Q$ such that $M, X[F/y] \models \varphi$ and $F(s)$ is determined by the values $s(\bar{x})$.

$$M, X \models Qy([\bar{x} \twoheadrightarrow y] \wedge \varphi^{\twoheadrightarrow}) \text{ iff } \exists F : X \to Q \text{ s.t. } M, X[F/y] \models \varphi^{\twoheadrightarrow},$$
$$\text{and } M, X[F/y] \models [\bar{x} \twoheadrightarrow y] .$$

Now $M, X[F/y] \models [\bar{x} \twoheadrightarrow y]$ iff $X[F/y]^{y}_{s \upharpoonright \bar{x}} = X[F/y]^{y}_{s \upharpoonright \bar{x}\bar{z}}$ for every $s \in X$. However

$$X[F/y]^{y}_{s \upharpoonright \bar{x}\bar{z}} = F(s),$$

so the result follows from the induction hypothesis. □

It should be clear that if for all $s \neq s' \in X$ there is $x \in \mathrm{dom}(X) \setminus \{ y \}$ such that $s(x) \neq s'(x)$, i.e., $X(\bar{x}, y)$ is (the graph of) a partial function $M^k \to M$, then

$$M, X \models [\bar{x} \to y] \text{ iff } M, X \models [\bar{x} \twoheadrightarrow y] .$$

Thus, if y is existentially quantified in a sentence of Dependence logic σ then the resulting team X can be assumed to have this property and thus $[\bar{x} \to y]$ and $[\bar{x} \twoheadrightarrow y]$ are interchangeable in the following restricted way:

Definition 3.6 A Dependence logic formula φ is *normal* if $[\bar{x} \to y]$ only occurs as $\exists y([\bar{x} \to y] \wedge \psi)$.

Proposition 3.7 *If φ is normal and φ' is the result of replacing atoms $[\bar{x} \to y]$ by $[\bar{x} \twoheadrightarrow y]$ in φ, then for every M and X*

$$M, X \models \varphi \ \textit{iff} \ M, X \models \varphi' .$$

Proof Easy induction. □

This means that under restricted use of the dependence atom we can use either $[\to]$ or $[\twoheadrightarrow]$. Since every sentence of Dependence logic can be expressed by a sentence in DF-logic and those in turn can be expressed by a normal sentence of Dependence logic, we know that the fragment of normal sentences is as strong as full Dependence logic.

Let us call Dependence logic in which $[\twoheadrightarrow]$ is used instead of $[\to]$ for *Multivalued Dependence logic* or MVDL for short.

The truth definition of $M, X \models [\bar{x} \twoheadrightarrow y]$ is first order in X, see Proposition 3.3, and thus for every formula φ in MVDL there is a sentence $\sigma(R)$ in ESO such that

$$M, X \models \varphi \text{ iff } (M, X) \models \sigma(R).$$

That means that MVDL is at most as strong as existential second order logic (when it comes to sentences) and thus as Dependence logic. Also, by translating sentences of Dependence logic into normal sentences and then replacing the functional dependence atom with the multivalued dependence atom we get a sentence of MVDL which is equivalent to the original Dependence logic sentence. Thus, MVDL, Dependence logic and ESO are all of the same strength on the level of sentences.

Recently Galliani proved that MVDL is exactly as strong as existential second order logic also on the level of formulas:

Proposition 3.8 (Galliani 2011) *Let $\mathscr{X}$ be a set of teams on a model M, then the following are equivalent:*

- *There is a formula φ of MVDL such that $\mathscr{X} = [\![\varphi]\!]^M$.*
- *There is a sentence of existential second order logic, ESO, σ such that $X \in \mathscr{X}$ iff $(M, X(\bar{x})) \models \sigma$.*

Remember that $X(\bar{x})$ is the relation corresponding to the team X.

3.1 Multivalued Dependence, Independence and Completeness

In a recent paper by Grädel and Väänänen (2011) *independence atoms* are introduced:

$$M, X \models \bar{y} \perp_{\bar{x}} \bar{z} \text{ iff}$$
$$\forall s, s' \in X \Big(s(\bar{x}) = s'(\bar{x}) \to \exists s_0 \in X \big(s_0(\bar{x}, \bar{y}) = s(\bar{x}, \bar{y}) \wedge s_0(\bar{z}) = s'(\bar{z}) \big) \Big).$$

This atom also applies to terms $\bar{t} \perp_{\bar{s}} \bar{t}'$ by a slight change of the definition.

As easily seen, we have

$$M, X \models [\bar{x} \twoheadrightarrow \bar{y}] \text{ iff } M, X \models \bar{y} \perp_{\bar{x}} \bar{z}$$

where $\bar{z} = \text{dom}(X) \setminus \{\bar{x}, \bar{y}\}$. The logic we get when adding independence atoms to first order logic is called *Independence logic*.

The independence relation introduced by Grädel and Väänänen is in the database theory community known as the *embedded multivalued depencency*. It is usually denoted by $[\bar{x} \twoheadrightarrow \bar{y} | \bar{z}]$.

Let us use the notation $D \models \varphi$, where D is a (finite) set of dependence atoms (functional, multivalued or embedded multivalued) and φ is a single dependence atom (of the same kind) to mean that any team X (over any domain) satisfying all the dependencies in D also satisfies φ. It is well known that functional dependence is axiomatizable:

Proposition 3.9 (Armstrong 1974) *If $D \cup \{\varphi\}$ is a finite set of functional dependence atoms then $D \models \varphi$ iff φ is derivable from D with the following inference rules:*

- *Reflexivity: If $\bar{y} \subseteq \bar{x}$ then $[\bar{x} \to \bar{y}]$.*
- *Augmentation: If $[\bar{x} \to \bar{y}]$ then $[\bar{x}, \bar{z} \to \bar{y}, \bar{z}]$.*
- *Transitivity: If $[\bar{x} \to \bar{y}]$ and $[\bar{y} \to \bar{z}]$ then $[\bar{x} \to \bar{z}]$.*

A complete axiomatization of multivalued dependence is also possible as was shown by Beeri et al.:

Proposition 3.10 (Beeri et al. 1977) *Let U be a finite set of variables, $D \cup \{\varphi\}$ a finite set of multivalued dependence atoms over the variables in U. Then $D \models \varphi$ iff φ is derivable from D with the following inference rules:*

- *Complementation: If $\bar{x} \cup \bar{y} \cup \bar{z} = U$, $\bar{y} \cap \bar{z} \subseteq \bar{x}$, and $[\bar{x} \twoheadrightarrow \bar{y}]$ then $[\bar{x} \twoheadrightarrow \bar{z}]$*
- *Reflexivity: If $\bar{y} \subseteq \bar{x}$ then $[\bar{x} \twoheadrightarrow \bar{y}]$.*
- *Augmentation: If $[\bar{x} \twoheadrightarrow \bar{y}]$ then $[\bar{x}, \bar{z} \twoheadrightarrow \bar{y}, \bar{z}]$.*
- *Transitivity: If $[\bar{x} \twoheadrightarrow \bar{y}]$ and $[\bar{y} \twoheadrightarrow \bar{z}]$ then $[\bar{x} \twoheadrightarrow \bar{z} \setminus \bar{y}]$.*[12]

We are assuming that all $\bar{x}$, $\bar{y}$, and $\bar{z}$ are variables in U.

However the embedded multivalued dependency is not axiomatizable as was shown by Sagiv and Walecka in the following sense:

Proposition 3.11 (Sagiv and Walecka 1982) *There is no finite set of inference rules, where each inference rule is a recursive set of k-tuples of embedded multivalued dependencies, axiomatizing the consequence relation $D \models \varphi$ for embedded multivalued dependencies.*

This answers an open question stated in Grädel and Väänänen (2011).
Galliani recently observed that $X \models \bar{t} \perp_{\bar{s}} \bar{t}'$ iff

$$X \models \exists \bar{x} \bar{y} \bar{z} \big(\bar{x} = \bar{s} \wedge \bar{y} = \bar{t} \wedge \bar{z} = \bar{t}' \wedge \forall \bar{u}\, [\bar{x} \twoheadrightarrow \bar{y}]\big),$$

where $\bar{u}$ is the domain of X. Thus, we get the following proposition.

Proposition 3.12 (Galliani 2011) *The multivalued Dependence logic has the same strength as Independence logic, even at the level of formulas.*

Thus the definable sets of teams of both Independence logic and multivalued Dependence logic is exactly the sets of teams definable by existential second order sentences.

4 Conclusion and Discussion

In this paper we have given truth conditions for monotone generalized quantifiers in logics using team semantics in such a way that the meaning of $L(Q)$-formulas remain the same when moving to team semantics, i.e., a team satisfies a formula of $L(Q)$ iff every assignment in the team satisfies the formula. It is also shown that the truth conditions in a natural way can be extended to deal with relations of dependence and independence between the quantifiers, in such a way that branching of two quantifiers Q_1 and Q_2 can be expressed by a linear quantifier prefix: $Q_1 x Q_2 y/x$.

We also gave truth conditions for non-monotone quantifiers by using an idea from Sher in (1990) to add a largeness or maximality condition. For this condition the quantifier prefix $Q_1 x Q_2 y/x$ comes close to the branching $\mathrm{Br}^S(Q_1, Q_2)xy$

12 Here $\bar{z} \setminus \bar{y}$ is the set difference, i.e., the set of all variables in $\bar{z}$ not in $\bar{y}$.

defined in Sher (1990), but they are not equivalent: In the prefix $Q_1 x Q_2 y/x$ the second quantifier depends on the first in a weak sense, however in the case of $\mathrm{Br}^S(Q_1, Q_2)xy$ there is full symmetry in the sense that $\mathrm{Br}^S(Q_1, Q_2)xy$ is equivalent to $\mathrm{Br}^S(Q_2, Q_1)yx$.

Is there some way of treating the maximaility principle of Sher, $\mathrm{Br}^S(Q_1, Q_2)$, in a compositional way in the framework used in this paper? Can other proposed principles, e.g. the one in Westerståhl (1987), of branching in the non-monotone case be handled compositional in the same way?

The question of whether the notion of dependence and independence of (monotone) quantifiers can be handled on the atomic level is answered positively in the paper. However, the notion of dependence is not the functional dependence of Dependence logic, but rather a new kind of dependence atom, called multivalued dependence. This atom is not closed under taking subteams, but can be used to express branching of generalized quantifiers, which the functional dependence atom cannot: $\mathrm{Br}(Q_1, Q_2)xy$ is equivalent to $Q_1 x Q_2 y([\twoheadrightarrow y] \wedge \ldots)$.

If a monotone quantifier Q is definable in ESO, i.e., there is an ESO sentence σ such that $M \vDash \sigma$ iff $M \in Q$, then it is easy to see that the strength on sentence level of the logic $\mathrm{SBL}(Q)$ is just the strength of existential second order logic, ESO. This comes from the fact that SBL $\equiv$ ESO and by observing that for any formula φ of $\mathrm{SBL}(Q)$ we can find a sentence σ of ESO such that

$$M, X \vDash \varphi \text{ iff } (M, X(\bar{x})) \vDash \sigma.$$

This is done by coding the truth conditions of φ into the sentence σ. Thus, $\mathrm{SBL}(Q) \equiv$ SBL but the translation of $\mathrm{SBL}(Q)$ sentences into SBL is non-compositional.

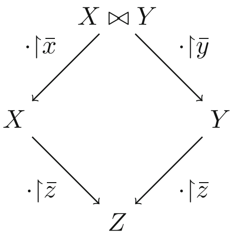
For which quantifiers Q are there compositional translations of $\mathrm{SBL}(Q)$ into SBL? In particular, is there a compositional translation of $\mathrm{SBL}(Q_0)$ into SBL, where Q_0 is the quantifier "there exists infinitely many"?

Of course we have not answered one of the basic questions regarding our definition of generalized quantifiers: When introducing a monotone quantifier, which may not be definable in ESO, into Dependence logic, what is the strength of the resulting logic?

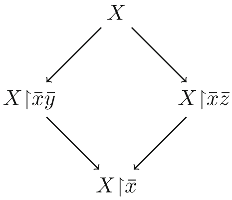
In connection with investigating the strength of these kinds of logics it might be worth mentioning Krynicki's result in (1993) saying that there is a single quantifier Q of type $\langle 4 \rangle$ such that every IF-logic sentence is equivalent to a sentence of $L(Q)$ over every structure with a pairing function. Is this also true for $\mathrm{SBL}(Q)$, i.e., is there a single quantifier Q' such that any sentence of $\mathrm{SBL}(Q)$ is equivalent to a sentence of $L(Q')$ over any structure with a pairing function?

There is a connection of multivalued dependence with category theory through pullbacks, or fibered products. Proposition 3.4 gives a characterization of multivalued dependence in terms of natural join, which in turn has a characterization in terms of pullbacks:

In the category of teams, where the objects are teams and the morphisms are functions between teams, the natural join of X and Y is the pullback of X and Y over $Z = (X \upharpoonright \bar{z}) \cap (Y \upharpoonright \bar{z})$, where $\bar{z}$ is $\mathrm{dom}(X) \cap \mathrm{dom}(Y)$. More precisely; let $\bar{x}$ and $\bar{y}$ be $\mathrm{dom}(X)$ and $\mathrm{dom}(Y)$ respectively, then the following is a pullback diagram:



Thus, by using Proposition 3.4, we see that $X \vDash [\bar{x} \twoheadrightarrow \bar{y}]$ holds iff the commuting diagram



where $\bar{z}$ is $\mathrm{dom}(X) \setminus \{\bar{x}, \bar{y}\}$, is a pullback. This suggests that there might be more, and deeper, connections between team semantics and category theory.

Acknowledgments The author was partially supported by the EUROCORE LogICCC LINT program and the Swedish Research Council. The author would like to thank the anonymous referee who in several ways improved this paper.

References

Abramsky, S., & Väänänen, J. (2009). From IF to BI: A tale of dependence and separation. *Synthese, 167*(2, Knowledge, Rationality & Action), 207–230.

Armstrong, W. W. (1974). Dependency structures of data base relationships. In: *Information processing 74, proceedings of IFIP congress, Stockholm, 1974* (pp. 580–583). Amsterdam: North-Holland.

Barwise, J. (1979). On branching quantifiers in English. *Journal of Philosophical Logic, 8*(1), 47–80.

Beeri, C., Fagin, R., & Howard, J. (1977). A complete axiomatization for functional and multivalued dependencies in database relations. In: *Proceedings of the 1977 ACM SIGMOD international conference on Management of data* (p. 61).

Davies, M. (1989). Two examiners marked six scripts. Interpretations of numerically quantified sentences. *Linguistics and Philosophy 12*(3), 293–323.

Fagin, R. (1977). Multivalued dependencies and a new normal form for relational databases. *ACM Transactions on Database Systems (TODS), 2*(3), 262–278.

Galliani, P. (2011). Inclusion and exclusion dependencies in team semantics: On some logics of imperfect information. *ArXiv e-prints*.

Gierasimczuk, N., & Szymanik, J. (2009). Branching quantification vs. Two-way quantification. *Journal of Semantics, 26*(4), 329–366.

Grädel, E., & Väänänen, J. (2011). *Dependence and independence*. To be published in Studia Logica.

Henkin, L. (1961). Some remarks on infinitely long formulas. In: *Infinitistic methods. Proceedings of the Symposium. Foundations of Mathematics, Warsaw, 1959* (pp. 167–183). Oxford: Pergamon.

Hintikka, J. (1974). Quantifiers vs quantification theory. *Linguistic Inquiry, V*, 153–177.

Hodges, W. (1997). Compositional semantics for a language of imperfect information. *Logic Journal of IGPL, 5*(4), 539–563.

Hodges, W. (2008). Logics of imperfect information: Why sets of assignments?. In J. van Benthem, D. Gabbay, & B. Löwe (Eds.), *Interactive Logic, selected papers from the 7th Augustus de Morgan Workshop, London* (pp. 117–133).

Kontinen, J., & Väänänen, J. (2009). On definability in dependence logic. *Journal of Logic, Language and Information, 18*(3), 317–332.
Krynicki, M. (1993). Hierarchies of partially ordered connectives and quantifiers. *Mathematical Logic Quarterly, 39*(1), 287–294.
Lindström, P. (1966). First order predicate logic with generalized quantifiers. *Theoria, 32*, 186–195.
Montague, R. (1970). English as a formal language. In: *Linguaggi nella società e nella tecnica* (pp. 189–223). Mailand.
Mostowski, A. (1957). On a generalization of quantifiers. *Fundamenta Mathematicae, 44*, 12–36.
Peters, S., & Westerståhl, D. (2006). *Quantifiers in Language and Logic*. : Oxford University Press.
Sagiv, Y., & Walecka, S.F. (1982). Subset dependencies and a completeness result for a subclass of embedded multivalued dependencies. *Journal of the Association for Computing Machinery, 29*(1), 103–117.
Sher, G. (1990). Ways of branching quantifers. *Linguistics and Philosophy, 13*(4), 393–422.
Väänänen, J. (2007). *Dependence logic: A new approach to independence friendly logic, London Mathematical Society Student Texts* (Vol. 70). Cambridge: Cambridge University Press.
Westerståhl D. (1987). Branching generalized quantifiers and natural language. *Generalized quantifiers: Linguistic and logical approaches* (pp. 109–150).