Expressivity and Complexity of Dependence Logic

Arnaud Durand, Juha Kontinen, and Heribert Vollmer

1 Introduction

In this article we review recent results on expressivity and complexity of first-order, modal, and propositional dependence logic and some of its variants such as independence and inclusion logic. Dependence logic was introduced by Jouko Väänänen in [56]. On the syntactic side, it extends usual first-order logic by the so-called dependence atoms

$$=(x_1, \ldots, x_n)\,,$$

the meaning of which is that the value of x_n is functionally determined by the values of $x_1, \ldots, x_{n-1}$. The semantics of dependence logic is defined using sets of assignments, teams, rather than single assignments as in first-order logic. Since the introduction of dependence logic in 2007, the area of team semantics has evolved into a general framework for logics in which various notions of dependence and independence can be formalized and studied. In this paper we mainly consider

A. Durand (✉)
IMJ-PRG, CNRS UMR 7586, Université Paris Diderot, Case 7012, 75205 Paris cedex 13, France
e-mail: durand@logique.jussieu.fr

J. Kontinen
Department of Mathematics and Statistics, University of Helsinki, P.O. Box 68,
00014 Helsinki, Finland
e-mail: juha.kontinen@helsinki.fi

H. Vollmer
Institut für Theoretische Informatik, Leibniz Universität Hannover, Appelstr. 4,
30167 Hannover, Germany
e-mail: vollmer@thi.uni-hannover.de

S. Abramsky et al. (eds.), *Dependence Logic*, DOI 10.1007/978-3-319-31803-5_2

variants of dependence logic arising by replacing/supplementing dependence atoms with further dependency notions, and we also study propositional and modal variants.

In Section 2 we review the basic definitions and results on first-order dependence logic and its variants (extensions and fragments). It is divided into three subsections of which the two first ones deal with results related to *expressive power* and *definability*. In particular, results charting the expressive power of certain natural syntactic fragments of dependence logic and its variants will be discussed in Section 2.4. Section 2.5 reviews results on the complexity of *satisfiability* and *model checking* in the (first-order) dependence logic context. In Section 3 we turn to modal and propositional versions of dependence logic. After introducing the basic notions and logics, we will again first touch expressivity questions and then turn to the complexity of algorithmic problems arising in this context, mostly the complexity of satisfiability and model checking. The paper concludes in Section 4 with a list of open questions.

2 First-order Dependence Logic

2.1 Team semantics

In this section we define the basics of the team semantics as presented in the monograph [56] by Väänänen. The origins of this definition go back to a paper by Wilfrid Hodges [35], in which he gave a Tarski-style semantics for Hintikka and Sandu's *independence-friendly logic* $\mathcal{IF}$ [34]. Hodges originally used the term "trump semantics", somewhat reflecting the game-theoretic nature of the previously only known non-compositional semantics for $\mathcal{IF}$.

Definition 1. Let $\mathscr{M}$ be a structure with domain M, and V a finite set of variables. Then

- A *team* X over $\mathscr{M}$ with domain $\mathrm{Dom}(X) = V$ is a finite set of assignments $s: V \to M$.
- For a tuple $\mathbf{v} = (v_1, \ldots, v_n)$, where $v_i \in V$, $X(\mathbf{v}) := \{s(\mathbf{v}) : s \in X\}$ is an n-ary relation of M, where $s(\mathbf{v}) := (s(v_1), \ldots, s(v_n))$.
- For $W \subseteq V$, $X \upharpoonright W$ denotes the team obtained by restricting all assignments of X to W.
- The set of free variables of a formula ϕ is defined analogously as in first-order logic, and is denoted by $\mathrm{Fr}(\phi)$. In particular, all non-first-order atoms considered in this article (see Definition 3) are treated as atomic formulas, and hence all variable occurrences in them are considered to be free.

With the above notions defined, we are now ready to present the semantics of dependence logic. In this article we consider two variants of the semantics called the *strict* and the *lax semantics* introduced in [16]. The original semantics given in [56] is a combination of these variants (with the lax disjunction and the strict existential quantifier). For any logic, e.g., dependence logic, whose formulas have the downwards closure property of Proposition 4, the two variants of the semantics are easily seen to be equivalent. On the other hand, for inclusion and independence logic the semantics are not equivalent [16]. A serious disadvantage of the strict semantics is the failure of the locality property in the case of inclusion and independence logic (see Proposition 1).

We will first define the lax version of the team semantics for first-order formulas in negation normal form. For an assignment s, $\mathcal{M} \models_s \alpha$ below refers to satisfaction in first-order logic. We denote by $s[m/v]$ the assignment such that $s[m/v](x) = m$ if $x = v$, and $s[m/v](x) = s(x)$ otherwise.

Definition 2 (Lax Semantics). Let $\mathcal{M}$ be a structure, X a team over $\mathcal{M}$, and ϕ a formula such that $\mathrm{Fr}(\phi) \subseteq \mathrm{Dom}(X)$. Then X *satisfies* ϕ in $\mathcal{M}$, $\mathcal{M} \models_X \phi$, if

- lit: For a first-order literal α, $\mathcal{M} \models_X \alpha$ if and only if for all $s \in X$, $\mathcal{M} \models_s \alpha$.
- $\vee$: $\mathcal{M} \models_X \psi \vee \theta$ if and only if there are Y and Z such that $Y \cup Z = X$, $\mathcal{M} \models_Y \psi$ and $\mathcal{M} \models_Z \theta$.
- $\wedge$: $\mathcal{M} \models_X \psi \wedge \theta$ if and only if $\mathcal{M} \models_X \psi$ and $\mathcal{M} \models_X \theta$.
- $\exists$: $\mathcal{M} \models_X \exists v \psi$ if and only if there exists a function $F : X \to \mathcal{P}(M) \setminus \{\emptyset\}$ such that $\mathcal{M} \models_{X[F/v]} \psi$, where $X[F/v] = \{s[m/v] : s \in X, m \in F(s)\}$.
- $\forall$: $\mathcal{M} \models_X \forall v \psi$ if and only if $\mathcal{M} \models_{X[M/v]} \psi$, where $X[M/v] = \{s[m/v] : s \in X, m \in M\}$.

A sentence is a formula without free variables. A sentence ϕ is *true* in $\mathcal{M}$ (abbreviated $\mathcal{M} \models \phi$) if $\mathcal{M} \models_{\{\emptyset\}} \phi$. Sentences ϕ and ϕ' are *equivalent*, $\phi \equiv \phi'$, if for all models $\mathcal{M}$, $\mathcal{M} \models \phi \Leftrightarrow \mathcal{M} \models \phi'$.

In the strict semantics, the semantic rule for disjunction is replaced by

> $\mathcal{M} \models_X \psi \vee \theta$ if and only if, there are Y and Z such that $Y \cap Z = \emptyset$, $Y \cup Z = X$, $\mathcal{M} \models_Y \psi$ and $\mathcal{M} \models_Z \theta$,

and the semantic rule for existential quantifier by

> $\mathcal{M} \models_X \exists v \psi$ if and only if, there exists a function $F : X \to \mathcal{P}(M) \setminus \{\emptyset\}$ such that $|F(s)| = 1$ for all $s \in X$, and $\mathcal{M} \models_{X[F/v]} \psi$.

It is worth noting that functions quantified in the strict semantics version of the existential quantifier correspond exactly to functions $F: X \to M$. Hence the notation $X[F/v]$ can be naturally extended to cover also functions $F: X \to M$.

The meaning of first-order formulas is invariant under the choice between the strict and the lax semantics. Furthermore, first-order formulas have the following *flatness property*:

Theorem 1 (Flatness). *Let $\mathscr{M}$ be a structure and X a team of $\mathscr{M}$. Then for a first-order formula ϕ the following are equivalent:*

1. $\mathscr{M} \models_X \phi$,
2. *For all* $s \in X$, $\mathscr{M} \models_s \phi$.

It is worth noting that in [56] also a general notion of flatness of a formula is defined by replacing the second item above by "For all $s \in X$, $\mathscr{M} \models_{\{s\}} \phi$".

Next we will give the semantic clauses for the non-first-order atoms and connectives considered in this paper. We begin with the new atomic formulas:

Definition 3. • Let $\mathbf{x}$ be a tuple of variables and let y be another variable. Then $=(\mathbf{x}, y)$ is a *dependence atom*, with the following semantic rule:

$\mathscr{M} \models_X =(\mathbf{x}, y)$ if and only if for all $s, s' \in X$, if $s(\mathbf{x}) = s'(\mathbf{x})$, then $s(y) = s'(y)$.

- Let $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ be tuples of variables (not necessarily of the same length). Then $\mathbf{y} \perp_{\mathbf{x}} \mathbf{z}$ is a *conditional independence atom*, with the semantic rule

  $\mathscr{M} \models_X \mathbf{y} \perp_{\mathbf{x}} \mathbf{z}$ if and only if for all $s, s' \in X$ such that $s(\mathbf{x}) = s'(\mathbf{x})$, there exists a $s'' \in X$ such that $s''(\mathbf{xyz}) = s(\mathbf{xy})s'(\mathbf{z})$.

  Furthermore, we will write $\mathbf{x} \perp \mathbf{y}$ as a shorthand for $\mathbf{x} \perp_{\emptyset} \mathbf{y}$, and call it a *pure independence atom*.
- Let $\mathbf{x}$ and $\mathbf{y}$ be two tuples of variables of the same length. Then $\mathbf{x} \subseteq \mathbf{y}$ is an *inclusion atom*, with the semantic rule

  $\mathscr{M} \models_X \mathbf{x} \subseteq \mathbf{y}$ if and only if $X(\mathbf{x}) \subseteq X(\mathbf{y})$.
- Let $\mathbf{x}$ and $\mathbf{y}$ be two tuples of variables of the same length. Then $\mathbf{x} \mid \mathbf{y}$ is an *exclusion atom*, with the semantic rule

  $\mathscr{M} \models_X \mathbf{x} \mid \mathbf{y}$ if and only if $X(\mathbf{x}) \cap X(\mathbf{y}) = \emptyset$.

We denote the set of all dependence atoms by $=(\ldots)$. Analogously, all independence, inclusion and exclusion atoms are denoted by $\perp_c$, $\subseteq$, and $|$, respectively. For a collection $\mathscr{C} \subseteq \{=(\ldots), \perp_c, \subseteq, |\}$, we write $\mathcal{FO}(\mathscr{C})$ (omitting the set parenthesis of $\mathscr{C}$) for the logic obtained by adding all atoms listed in $\mathscr{C}$ to the syntax of first-order logic. Independence atoms (or independence logic) were first considered in [21], and inclusion atoms go back to [16]. In our notation, dependence logic, independence logic, and inclusion logic are denoted by $\mathcal{FO}(=(\ldots))$, $\mathcal{FO}(\perp_c)$, and $\mathcal{FO}(\subseteq)$, respectively. We also use the notation $\mathcal{D}$ as a shortcut for $\mathcal{FO}(=(\ldots))$. The fragment of independence logic containing only pure independence atoms in denoted $\mathcal{FO}(\perp)$.

Under the lax semantics, all of the above logics satisfy the following locality property [16]:

Proposition 1. *Let $\phi \in \mathcal{FO}(=(\ldots), \perp_c, \subseteq, |)$. Then for all models $\mathcal{M}$ and teams X,*

$$\mathcal{M} \models_X \phi \Leftrightarrow \mathcal{M} \models_{X \upharpoonright Fr(\phi)} \phi.$$

On the other hand, under the strict semantics Proposition 1 fails for inclusion and independence logic [16].

The aforementioned atoms are particular instances of a general notion of *generalized dependence atom* [44]. The semantics of a generalized dependence atom A_Q is determined (essentially) by a class Q of structures and teams over which the atomic formula $A_Q(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ is satisfied (see [44] for details; we will consider $\mathcal{FO}$-definable generalized dependence atoms as depicted in Table 1).

Next we will define connectives and quantifiers that will also be discussed in the next section. One of the most natural extensions of dependence logic is obtained by the classical negation ($\sim$) with the usual interpretation:

$$\mathcal{M} \models_X \sim \phi \text{ iff } \mathcal{M} \not\models_X \phi.$$

This extension was introduced in [57], and the logic obtained was called *Team Logic* ($\mathcal{TL}$). The classical disjunction $\veebar$ (also sometimes referred to as intuitionistic disjunction) has also been considered especially in the modal team semantics context, see Section 3. The connective $\veebar$ has the expected interpretation

$$\mathcal{M} \models_X \phi \veebar \psi \text{ iff } \mathcal{M} \models_X \phi \text{ or } \mathcal{M} \models_X \psi.$$

In [1] two new connectives called the *intuitionistic* ($\rightarrow$) and the *linear implication* ($-\!\!\circ$) were introduced giving rise to an extension of dependence logic called BID:

$$\mathcal{M} \models_X \phi \rightarrow \psi \quad \text{iff} \quad \text{for all } Y \subseteq X, \text{ if } \mathcal{M} \models_Y \phi \text{ then } \mathcal{M} \models_Y \psi.$$
$$\mathcal{M} \models_X \phi -\!\!\circ \psi \quad \text{iff} \quad \text{for all } Y, \text{ if } \mathcal{M} \models_Y \phi \text{ then } \mathcal{M} \models_{X \cup Y} \psi.$$

Quantifiers, other than the familiar $\exists$ and $\forall$, have also been studied in the team semantics setting [12, 13]. From the complexity theoretic point of view, the following majority quantifier introduced in [7] is interesting:

$$\mathcal{M} \models_X \mathsf{M}x\phi(x) \text{ iff for at least } |M|^{|X|}/2 \text{ many functions } F: X \to M, \text{ we have}$$
$$\mathcal{M} \models_{X[F/x]} \phi(x).$$

2.2 Normal forms

In order to analyse the expressive power of dependence logic and to compare it with other formalisms, it is useful to obtain normal forms such as this one proved in [56].

Theorem 2. *Every dependence logic sentence is equivalent to some sentence of the form:*

$$\phi := \forall \mathbf{x} \exists \mathbf{y} \, (\bigwedge_{i \in I} =(\mathbf{x_i}, y_i) \wedge \theta)$$

where $I \subseteq \mathbb{N}$, $\mathbf{x_i}$ *is a subsequence of* $\mathbf{x}$*, and* y_i *is a member of* $\mathbf{y}$.

Such a result is an analogue of Skolem normal form for first-order logic. It separates clearly the functional dependencies introduced between subsets of variables from the regular part of the formula. It also makes intuitively clear that to be translated into an extension of first-order logic one would need second-order quantification to express these dependencies between variables. Refinements of such a normalization result are at the heart of various characterizations of dependence-like logics and their fragments. For example, the analogue of Theorem 2 for independence logic (with dependence atoms replaced by independence atoms) was shown in [23]. Furthermore, a prenex normal form theorem for formulas of $\mathcal{FO}(=(\ldots), \perp_c, \subseteq)$ was shown for the strict and the lax semantics in [18] and [26], respectively.

2.3 Expressive Power

In this section we review results on the expressive power of the variants of dependence logic of the previous subsections.

As it turns out, the expressive power of sentences of dependence logic corresponds to that of existential second-order logic [56], and hence to the complexity class non-deterministic polynomial time (NP) via the well-known theorem of Fagin [14]. In the following, we will not distinguish in notation between a logic and the classes of models defined by its sentences, and we will use the equality symbol to denote that logics are equivalent for sentences, and we will use equality for logics and complexity classes in the same vein.

Theorem 3. $\mathcal{D} = \mathsf{NP} = \mathcal{ESO}$

The direction $\mathcal{ESO} \leq \mathcal{D}$ is proved by utilizing the fact that every $\mathcal{ESO}$-sentence can be transformed to the so-called Skolem normal form. On the other hand, the direction $\mathcal{D} \leq \mathcal{ESO}$ is proved by essentially simulating the team semantics of dependence logic in $\mathcal{ESO}$ with an extra relation symbol interpreting the team.

An interesting consequence of the team semantics of dependence logic is that Theorem 3 does not immediately settle the question also for open formulas. In fact, all $\mathcal{D}$-formulas have the following *Downwards Closure* property:

Theorem 4 (Downwards Closure). *Let* ϕ *be a* $\mathcal{D}$*-formula. Then for all structures* $\mathcal{M}$ *and teams* X*, if* $\mathcal{M} \models_X \phi$ *and* $Y \subseteq X$*, then* $\mathcal{M} \models_Y \phi$.

It was shown in [39] that the open formulas of dependence logic can define exactly the downward closed properties of teams expressible in $\mathcal{ESO}$ (again with an extra relation symbol for the team). Furthermore, already dependence atoms combined with disjunction give rise to NP-complete decision problems [37]. Define the formulas ϕ_1 and ϕ_2 as follows:

- $\phi_1 := \ =(x, y) \vee =(u, v)$,
- $\phi_2 := \ =(x, y) \vee =(u, v) \vee =(u, v)$.

Then the question of deciding whether a finite team X satisfies ϕ_1 is NL-complete, and for ϕ_2, already NP-complete.

As one might expect, the expressive power of dependence logic with the classical negation ($\mathcal{TL}$) increases to full second-order logic, and hence to the complexity class Polynomial Hierarchy (PH).

Theorem 5. $\mathcal{TL} = \mathcal{SO} = \mathsf{PH}$

This result is already shown in [56], but a direct translation of $\mathcal{SO}$ sentences into $\mathcal{TL}$-sentences was later given in [48]. Furthermore, in [38] it was shown that any property of teams definable in second-order logic can be expressed in team logic. It is worth noting that, for example, in general $\mathcal{TL}$-formulas are not closed downwards, e.g., the formula

$$\sim =(x) \tag{1}$$

expresses that x has at least two distinct values.

Interestingly, the two new connectives (implications) introduced in [1] preserve downwards closure when added to dependence logic. It was observed in [1] that any sentence of BID-logic can be translated into second-order logic. In fact, by the result of [62], already the intuitionistic implication alone increases the expressive power of dependence logic to full second-order logic.

Theorem 6. $\mathcal{D}(\to) = \mathcal{SO} = \mathsf{PH}$

This result utilizes the universal quantification implicit in the semantic rule of the intuitionistic implication. On the other hand, in [7, 8] the extension $\mathcal{D}(\mathsf{M})$ of dependence logic by the majority quantifier M was defined and studied. The main result of that paper is stated as follows:

Theorem 7. $\mathcal{D}(\mathsf{M}) = \mathsf{CH}$.

Above CH refers to the complexity class the counting hierarchy $\mathsf{CH} \supseteq \mathsf{PH}$. Theorems 7 and 5 imply that, for sentences, $\mathcal{D}(\mathsf{M})$ is at least as expressive as $\mathcal{TL}$ over finite structures. On the other hand, this result does not extend to open formulas since $\mathcal{D}(\mathsf{M})$-formulas have the downward closure property unlike $\mathcal{TL}$-formulas (see, e.g., formula (1)).

The aforementioned results show that dependence logic and its extensions allow us to logically characterize NP and some of its super classes. In [11] the question whether PTIME corresponds to a natural fragment of dependence

logic was considered. For $\mathcal{ESO}$ (also $\mathcal{SO}$) it is known that the so-called Horn fragment $\mathcal{SO}\exists$-Horn of $\mathcal{ESO}$ captures PTIME over successor structures [19]. In [11] a fragment $\mathcal{D}^*$-Horn equivalent to $\mathcal{SO}\exists$-Horn was identified. The formulas of $\mathcal{D}^*$-Horn have the form

$$\forall \mathbf{x} \exists \mathbf{y} (\bigwedge_i =(\mathbf{z}_i, y_i) \wedge \bigwedge_j C_j),$$

where $\mathbf{z}_i$ is subsequence of $\mathbf{x}$, the clauses C_j (i.e. disjunctions of $\mathcal{FO}$-literals) are assumed to satisfy a certain Horn condition, and the existentially quantified variables y_i are only allowed to appear in certain identity atoms of C_j (see [11] for the exact definition). The main result of [11] shows that

Theorem 8. *Over finite successor structures, $\mathcal{D}^*$-Horn $= \mathcal{SO}\exists$-Horn.*

Theorem 8 implies that

$$\mathcal{D}^*\text{-Horn} = \text{PTIME}$$

over finite successor structures. In the article [11] the expressive power of open formulas of $\mathcal{D}^*$-Horn is also characterized.

All of the results discussed in this section use the original semantics of dependence logic. It is easy to check that the results hold also for both variants of the semantics. Next we will consider the expressive power of inclusion logic. It turns out that the expressive power of inclusion logic is not invariant under the choice between the strict and the lax semantics.

The expressive power of inclusion logic under the lax semantics was studied in [17]. The main result of that paper shows that

Theorem 9. *Over the lax semantics,*

$$\mathcal{FO}(\subseteq) = \text{GFP}^+,$$

where GFP^+ is the so-called Positive Greatest Fixed Point Logic. It is known that over finite structures GFP^+ is equi-expressive with Least Fixed Point Logic (LFP), and furthermore for ordered finite structures LFP $=$ PTIME by the famous result of Immerman [36] and Vardi [60]. Therefore, it follows that

$$\mathcal{FO}(\subseteq) = \text{PTIME}$$

over ordered finite structures. In drastic contract with Theorem 9, it was observed in [18] that, over the strict semantics, inclusion logic is equivalent to $\mathcal{ESO}$ and hence captures NP.

Theorem 10. *Over the strict semantics,*

$$\mathcal{FO}(\subseteq) = \mathcal{ESO}.$$

This result is based on a simulation of dependence atoms in a dependence logic sentence (in the $\forall^*\exists^*$-normal form) by certain inclusion logic formulas. This simulation is not possible in general but only over teams that are generated by evaluating a $\forall^*\exists^*$-block of quantifiers with the strict semantics.

2.4 Refining the correspondence with $\mathcal{ESO}$

In this part we investigate how the correspondence between existential second-order logic and $\mathcal{FO}(\mathscr{C})$ for subsets $\mathscr{C}$ of dependence-like atoms can be refined. In particular we examine what is the effect of bounding the number of variables and the so-called arity of atoms which roughly concerns the number of distinct variables involved in them.

By relating fragments of $\mathcal{FO}(\mathscr{C})$ to fragments of existential second-order logic, one may hope to obtain separation results in dependence logics through hierarchy theorems in complexity or to give evidence that such results would have non-trivial consequences in complexity theory. In either way, this provides interesting insight on the expressive power of these logics.

Let us first define the notion of arity of an atom.

Definition 1. Let $k \in \mathbb{N}$.

- A dependence atom $=(\mathbf{x}, y)$ is of arity k if the length of $\mathbf{x}$ is k.
- An independence atom $\mathbf{y} \perp_{\mathbf{x}} \mathbf{z}$ is of arity k if $\mathbf{xyz}$ contains $k+1$ distinct variables.
- An inclusion atom $\mathbf{x} \subseteq \mathbf{y}$ is of arity k if the length of $\mathbf{x}$ and $\mathbf{y}$ is k.

We now define the corresponding fragments of $\mathcal{FO}(\mathscr{C})$ and existential second-order logic.

Definition 2. Let $\mathscr{C}$ be a subset of $\{=(\ldots), \perp_c, \subseteq, |\}$. Let $k \in \mathbb{N}$. Then:

- $\mathcal{FO}(\mathscr{C})(k\text{−ary})$ is the class of sentences of $\mathcal{FO}(\mathscr{C})$ in which all atoms of $\mathscr{C}$ are of arity bounded by k.
- $\mathcal{FO}(\mathscr{C})(k\forall)$ is the class of sentences of $\mathcal{FO}(\mathscr{C})$ in which every variable is quantified exactly once and at most k universal quantifiers occur.
- For convenience, we set by $\mathcal{D}(k\text{−ary})$ the class $\mathcal{FO}(= (\ldots))(k\text{−ary})$ and by $\mathcal{D}(k\text{−ary})$ the class $\mathcal{FO}(=(\ldots))(k\forall)$.

Definition 3. Let $k \in \mathbb{N}$.

- $\mathcal{ESO}(k\text{−ary})$ is the class of $\mathcal{ESO}$-sentences

$$\exists X_1 \ldots \exists X_n \psi,$$

in which the relation symbols X_i are at most k-ary and ψ is a first-order formula.

- $\mathcal{ESO}_f(k\text{−ary})$ is the class of $\mathcal{ESO}$-sentences

$$\exists f_1 \ldots \exists f_n \psi,$$

  in which the function symbols f_i are at most k-ary and ψ is a first-order formula.
- $\mathcal{ESO}_f(m\forall)$ is the class of $\mathcal{ESO}$-sentences in Skolem normal form

$$\exists f_1 \ldots \exists f_n \forall x_1 \ldots \forall x_r \psi,$$

  where $r \leq m$.

Such fragments of $\mathcal{ESO}$ have been widely studied and in particular their relationship with complexity classes. Roughly speaking, controlling the number of first-order variables in existential second-order logic amounts to control the polynomial degree in non-deterministic polynomial-time computations. To be more precise, it is known (see[22]) that, for $k \geq 1$:

$$\mathcal{ESO}_f(k\text{-ary}, k\forall) = \mathcal{ESO}_f(k\forall) = \mathsf{NTIME}_{\mathrm{RAM}}(n^k).$$

where $\mathsf{NTIME}_{\mathrm{RAM}}(n^k)$ is the class of problems decidable by a non-deterministic random access machine in time $O(n^k)$. We can now relate the expressive power of these various fragments.

First, reusing of variables is a key issue in team semantics. It turns out that the following result is true (see [6]).

Proposition 2. *Any sentence of dependence logic is logically equivalent to a sentence in which at most one variable is universally quantified (possibly several times).*

For what concerns dependence logic, the correspondence with existential second-order logic for the fragments with bounded arity and bounded number of universal variables is as follows [6]:

Theorem 11. *For all integers $k \geq 1$,*

- $\mathcal{D}(k\text{−ary}) = \mathcal{ESO}_f(k\text{−ary})$,
- $\mathcal{D}(k\forall) \leq \mathcal{ESO}_f(k\forall) \leq \mathcal{D}(2k\forall)$.

Roughly speaking, dependence logic can be seen as existential second-order logic with functions (dependence atoms) but "without proper names" for these functions. Hence composition of functions, that can be done freely in existential second-order logic, can be simulated only by using intermediate variables in dependence logic. So, as long as, only the arity of dependence atoms is fixed, one can obtain an exact correspondence between the fragments (as stated in Theorem 11). By contrast an exact correspondence between the fragments $\mathcal{D}(k\forall)$ and $\mathcal{ESO}_f(k\forall)$ seems unlikely (see the second item of Theorem 11). However, it is possible to establish an exact correspondence between $\mathcal{D}(k\forall)$ and some syntactically restricted fragment of $\mathcal{ESO}_f(k\forall)$ (see [6]).

Because inclusion and independence logic do not have the downward closure property, the situation is drastically different depending on whether the lax or the strict semantics are used (see [18, 24–26]).

Theorem 12. *Let $k \geq 1$. The following holds in the lax semantics:*

- $\mathcal{FO}(\subseteq)(k\text{−ary}) < \mathcal{FO}(\subseteq)(k+1\text{−ary})$,
- $\mathcal{FO}(\subseteq)(k\text{−ary}) \leq \mathcal{ESO}_f(k\text{−ary}) = \mathcal{FO}(\perp_c)(k\text{−ary})$,
- $\mathcal{FO}(\perp)(2\forall) = \mathcal{FO}(\perp)$,
- $\mathcal{FO}(\subseteq)(1\forall) = \mathcal{FO}(\subseteq)$.

For the strict semantics, the following results are true:

- $\mathcal{FO}(\subseteq)(k\forall) = \mathcal{ESO}_f(k\forall) = \mathsf{NTIME}_{\mathrm{RAM}}(n^k)$,
- $\mathcal{FO}(\perp_c)(k\forall) \leq \mathcal{ESO}_f((k+1)\forall)$,
- $\mathcal{ESO}_f(k\forall) \leq \mathcal{FO}(\perp_c)(2k\forall)$.

The results above imply that there is an infinite expressivity hierarchy for $\mathcal{D}(k\forall)$, $\mathcal{FO}(\subseteq)(k\forall)$, and $\mathcal{FO}(\perp_c)(k\forall)$. Indeed, it is well known (by a slight adaptation of classical non-deterministic time hierarchy [5]) that, for any integer $k \geq 1$, $\mathsf{NTIME}_{\mathrm{RAM}}(n^k)$ is strictly included in $\mathsf{NTIME}_{\mathrm{RAM}}(n^{k+1})$. Hence, for example, $\mathcal{D}(k\forall)$ is strictly less expressive than $\mathcal{D}((k+1)\forall)$.

Similarly, one might ask whether there is a strict hierarchy based on arity for $\mathcal{D}$ or $\mathcal{FO}(\perp_c)$. For example, is $\mathcal{D}(k+1\text{−ary})$ strictly more expressive than $\mathcal{D}(k\text{−ary})$ for all (or some) $k \geq 1$. Such a hierarchy would imply the existence of a similar hierarchy for $\mathcal{ESO}_f(k\text{−ary})$ which is a long-standing open question (for empty signature, this is known as the *Spectrum Arity Hierarchy Conjecture* [15]).

Finally, let us examine the situation when exclusion atoms are allowed in the syntax. It turns out that none of the approach above helps to control the arity correspondence between the corresponding fragments. By introducing mainly two new concepts, namely inclusion quantifier (an adaptation of the idea of quantifier relativization applied to inclusion atoms) and term value preserving disjunction, the following result is obtained in [51]:

Theorem 13. *For all integers $k \geq 1$, $\mathcal{FO}(\subseteq, |)(k\text{−ary}) = \mathcal{ESO}(k\text{−ary})$.*

Note that in this result the correspondence is with the relational fragment of existential second-order logic: no quantification on functions is allowed.

2.5 Satisfiability and Model Checking

In this section we briefly review results on satisfiability and model checking in the first-order dependence logic context.

We begin by recalling these problems for a logic $\mathscr{L}$:

- The *Satisfiability Problem* $\mathrm{SAT}[\mathscr{L}]$ is defined as

$$\mathrm{SAT}[\mathscr{L}] := \{\phi \in \mathscr{L} \mid \text{there is a structure } \mathscr{M} \text{ such that } \mathscr{M} \models \phi\}.$$

- The *Model-Checking Problem* is defined as

$$\mathrm{MC}[\mathscr{L}] := \{(\phi, \mathbb{M}) \mid \mathbb{M} \models \phi\}.$$

Since, for sentences, dependence logic is equivalent to $\mathcal{ESO}$ the classical results [4, 55] on the Decision Problem (Entscheidungsproblem) of $\mathcal{FO}$ imply that SAT[$\mathcal{D}$] is undecidable (Π^0_1-complete). In the case of first-order logic, these negative results spurred an extensive investigation of decidable fragments of $\mathcal{FO}$. Henkin [33] was the first to consider fragments of first-order logic with a fixed number of variables. Satisfiability for the fragments with three or more variables is easily seen to be undecidable, but for two variables, it is decidable (see, e.g., [3]). The two-variable fragment of $\mathcal{FO}$ is denoted by $\mathcal{FO}^2$.

In [40], the two-variable fragment $\mathcal{D}^2$ (i.e. the sentences of $\mathcal{D}$ in which only variables x and y appear) of dependence logic was studied and the following result was obtained:

Theorem 14. SAT[$\mathcal{D}^2$] *is* NEXPTIME*-complete.*

The complexity of the problem remains the same also for finite satisfiability, although there are sentences of $\mathcal{D}^2$ having only infinite models (the so-called infinity axioms). The proof of Theorem 14 is based on a polynomially bounded translation of sentences of $\mathcal{D}^2$ to $\Sigma^1_1(\mathcal{FOC}^2)$- sentences, i.e., sentences of the form

$$\exists R_1 \ldots \exists R_m \phi,$$

where ϕ is a $\mathcal{FOC}^2$-sentence. The logic $\mathcal{FOC}^2$ extends $\mathcal{FO}^2$ by counting quantifiers $\exists^{\geq i} x \phi(x)$ expressing that $\phi(x)$ is satisfied by at least i distinct elements. Theorem 14 then follows from the fact that satisfiability is NEXPTIME-complete for $\mathcal{FOC}^2$ [50]. It is worth noting that since $\mathcal{D}^2$ is not closed under classical negation, Theorem 14 does not solve the complexity of the validity problem for $\mathcal{D}^2$. In fact, this problem is still open.

Recently Theorem 14 has been generalized to logics of the form $\mathcal{FO}^2(\mathscr{A})$ that replace dependence atoms of $\mathcal{D}^2$ by a certain collections $\mathscr{A}$ of generalized dependence atoms [41]. The result therein shows that SAT[$\mathcal{FO}^2(\mathscr{A})$] is NEXPTIME-complete if the atoms in $\mathscr{A}$ are themselves $\Sigma^1_1(\mathcal{FOC}^2)$-definable. This result can be used to show, e.g., that the satisfiability and finite satisfiability problems of the two-variable fragments of inclusion, exclusion, and independence logic are all NEXPTIME-complete.

The complexity of model checking of dependence logic and its variants has been studied in [20]. The paper introduces a general model-checking game for logics with team semantics, and by analyzing the corresponding games, shows several results on the complexity of model checking in the team semantics framework.

Theorem 15. *The model-checking problem for* $\mathcal{D}$ *is* NEXPTIME*-complete.*

In [20] it is also showed that NEXPTIME is an upper bound for the complexity of model checking for any variant $\mathcal{FO}(\mathscr{C})$ of $\mathcal{D}$ such that the atoms in $\mathscr{C}$ are PTIME-computable.

3 Propositional and Modal Dependence Logic

This section will be devoted to the study of team-based logics over Kripke structures, such as modal dependence logic and, as a special case thereof, propositional dependence logic. Again, we will survey results on expressiveness of the most important logics and on the complexity of the satisfiability and model-checking problems. We will first turn to modal logics and later mention briefly some results for the purely propositional case.

3.1 Preliminaries

We start by introducing team semantics for usual modal logic. The central semantic concept here is the *Kripke model*, which is a tuple $K = (W, R, \pi)$ where W is a nonempty set of worlds, $R \subseteq W \times W$, and $\pi: P \to 2^W$, where P is a set of propositional variables. The idea here is that π determines which variables hold (are set to *true*) in each world. So every world carries a propositional assignment. In analogy to first-order logic, we want to express dependencies among the values of certain variables; so we have to evaluate formulas relative to a set of assignments, hence in our case a set of worlds. A *team* of a model K is thus defined to be a set $T \subseteq W$. The central basic concept underlying Väänänen's modal dependence logic and all its variants, that modal formulas are evaluated not in a world but in a *team*, is made precise in the definitions to follow below.

Before that, we would like to point out that another formalism, *independence-friendly modal logic*, has also be considered. In analogy to Hintikka and Sandu's independence-friendly logic $\mathcal{IF}$ [34], the so-called slash modalities are introduced. Consider the example formula $\Box_1(\Diamond_2/\Box_1)p$. It is evaluated like $\Box\Diamond p$, but now the witness for $\Diamond$ has to be chosen independently of the witnesses for $\Box$; hence the formula states, when evaluated at w, that there is a world u accessible from all worlds v that are accessible from w, a kind of "confluence property". There are several competing formalisms for modal independence-friendly logic, cf., e.g., [54].

Väänänen [58] when introducing modal dependence logic $\mathcal{MDL}$ made a step analogous to the introduction of dependencies in first-order dependence logic by extending the logical language with atoms $=(p_1, \ldots, p_n)$ to express dependencies among propositional variables. *Caveat:* In modal dependencies we can express dependencies among variables, not among worlds, as one might first expect when thinking of the standard translation of modal logic into first-order logic.

In the following, we will define the syntax of the modal logics we consider by grammars in extended Backus-Naur form (EBNF).

Definition 4 (Syntax of $\mathcal{ML}$).

$$\varphi ::= p \mid \neg p \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid \Diamond\varphi \mid \Box\varphi,$$

where p is an atomic proposition.

Definition 5 (Semantics of $\mathcal{ML}$). Let $K = (W, R, \pi)$ be a Kripke model, let $T \subseteq W$ be a team, and let φ be an $\mathcal{ML}$-formula. We define when $K, T \models \varphi$ holds inductively:

- If $\varphi = p$, then $K, T \models \varphi$ if and only if $T \subseteq \pi(p)$.
- If $\varphi = \neg p$, then $K, T \models \varphi$ if and only if $T \cap \pi(p) = \emptyset$.
- If $\varphi = \psi \vee \chi$ for some formulas ψ and χ, then $K, T \models \varphi$ if and only if $T = T_1 \cup T_2$ with $K, T_1 \models \psi$ and $K, T_2 \models \chi$.
- If $\varphi = \psi \wedge \chi$ for some formulas ψ and χ, then $K, T \models \varphi$ if and only if $K, T \models \psi$ and $K, T \models \chi$.
- If $\varphi = \Diamond\psi$ for some formula ψ, then $K, T \models \varphi$ if and only if there is some team T' of K such that $K, T' \models \psi$, for each $w \in T$, there is some $w' \in T'$ with $(w, w') \in R$, and for each $w' \in T'$, there is some $w \in T$ with $(w, w') \in R$.
- If $\varphi = \Box\psi$ for some formula ψ, then $K, T \models \varphi$ if and only if $K, T' \models \psi$, where T' is the set $\{w' \in W \mid (w, w') \in R \text{ for some } w \in T\}$.

Team semantics for $\mathcal{ML}$ shares the so-called *flatness property*, see also Theorem 1:

Lemma 1 (Flatness of $\mathcal{ML}$). *For all K, T, φ, $K, T \models \varphi$ if and only if for all $w \in T$, we have $K, w \models \varphi$.*

So there is no essential semantic effect of team semantics compared to the usual semantics for modal logic. This changes when enriching the language with dependence atoms.

Definition 6 (Syntax of $\mathcal{MDL}$).

$$\varphi ::= p \mid \neg p \mid =(\{p, \} p) \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid \Diamond\varphi \mid \Box\varphi,$$

where p is an atomic proposition.

Before defining the semantics of dependence atoms, we introduce a useful shorthand notation.

Definition 7. Let $\mathbf{p} = (p_1, \ldots, p_n)$ be a sequence of atomic propositions and w, w' be worlds of a Kripke model $K = (W, R, \pi)$. Then w and w' are equivalent under π over $\mathbf{p}$, denoted by $w \equiv_{\pi,\mathbf{p}} w'$, if the following holds:

$$\pi(w) \cap \{p_1, \ldots, p_n\} = \pi(w') \cap \{p_1, \ldots, p_n\}.$$

Definition 8 (Semantics of $\mathcal{MDL}$). We extend Definition 5 by the following clause:

- $K, T \models =(\mathbf{p}, q)$ if and only if for all $w_1, w_2 \in T$: if $w_1 \equiv_{\pi,\mathbf{p}} w_2$, then $w_1 \equiv_{\pi,q} w_2$.

Theorem 16 (Downwards Closure). *Let ϕ be a $\mathcal{MDL}$-formula. Then for all Kripke model K and teams T, if $K, T \models \phi$ and $T' \subseteq T$, then $K, T' \models \phi$.*

As in the first-order case, also the independence atom has been introduced into modal logic, thus leading to modal independence logic [42].

Definition 9 (Syntax of $\mathcal{MIL}$).

$$\varphi ::= p \mid \neg p \mid \{p\}\, p \perp_{\{p\}} \{p\}\, p \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid \Diamond\varphi \mid \Box\varphi,$$

where p is an atomic proposition.

Definition 10 (Semantics of $\mathcal{MIL}$). We extend Definition 5 by the following clause, where $\mathbf{p}, \mathbf{q}, \mathbf{r}$ are sequences of atomic propositions:

- $K, T \models \mathbf{p} \perp_{\mathbf{r}} \mathbf{q}$ if and only if for all $w_1, w_2 \in T$ such that $w_1 \equiv_{\pi,\mathbf{r}} w_2$ there exists $w_3 \in T$ such that $w_1 \equiv_{\pi,\mathbf{r}} w_3$, $w_1 \equiv_{\pi,\mathbf{p}} w_3$, and $w_2 \equiv_{\pi,\mathbf{q}} w_3$.

We note that one motivation behind the introduction of $\mathcal{MIL}$ was that it can be used to express security of cryptographic protocols, see [42].

At this point, where we have extended basic modal logic by a *dependence* as well as an *independence* atom, we would like to point out that there is a general way to introduce such so-called *generalized dependence atoms*, expressing further $\mathcal{FO}$-definable *properties on teams*. To make this precise, consider an $\mathcal{FO}$-formula φ. We say that φ defines the atom D if

$$K, T \models D(p_1, \ldots, p_n) \iff \mathfrak{A} \models \varphi,$$

where the structure $\mathfrak{A}$ has universe T and unary relations $A_{p_i}^{\mathfrak{A}}$ with $w \in A_{p_i}^{\mathfrak{A}}$ iff $p_i \in \pi(w)$.

$\mathcal{FO}$-definitions of dependence, independence, and some further team properties can be found in Table 1. There, boldface symbols denote sequences of propositional variables.

A further possible extension of the so far considered modal logics is by introducing additional propositional connectives. In Section 2.1, different forms of implication and negation have been defined. Here we just define $\mathcal{MTL}$, *modal team logic*, to extend $\mathcal{ML}$ by a second type of negation, denoted by $\sim$ and interpreted just as classical negation.

Definition 11 (Syntax of $\mathcal{MTL}$).

$$\varphi ::= p \mid \neg p \mid {\sim}\varphi \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid \Diamond\varphi \mid \Box\varphi,$$

where p is a propositional variable.

Table 1 Definitions of some generalized dependence atoms

Notation	Atom	$\mathcal{FO}$-defining formula
$=(\mathbf{p}, q)$	Dependence	$\forall w \forall w' \Big(\bigwedge_{i=1}^{n} \big(A_{p_i}(w) \leftrightarrow A_{p_i}(w')\big) \rightarrow \big(A_q(w) \leftrightarrow A_q(w')\big)\Big)$
$\mathbf{p} \perp_{\mathbf{r}} \mathbf{q}$	Independence	$\forall w \forall w' \Big(\bigwedge_{i=1}^{n} \big(A_{r_i}(w) \leftrightarrow A_{r_i}(w')\big) \rightarrow \exists w'' \Big(\bigwedge_{i=1}^{n} \big(A_{r_i}(w'') \leftrightarrow A_{r_i}(w)\big) \wedge \bigwedge_{i=1}^{m} \big(A_{p_i}(w'') \leftrightarrow A_{p_i}(w)\big) \wedge \bigwedge_{i=1}^{s} \big(A_{q_i}(w'') \leftrightarrow A_{q_i}(w')\big)\Big)\Big)$
$\mathbf{p} \subseteq \mathbf{q}$	Inclusion	$\forall w \exists w' \bigwedge_{i=1}^{n} \big(A_{p_i}(w) \leftrightarrow A_{q_i}(w')\big)$
$\mathbf{p} \mid \mathbf{q}$	Exclusion	$\forall w \forall w' \bigvee_{i=1}^{n} \big(A_{p_i}(w) \leftrightarrow \neg A_{q_i}(w')\big)$
NE	Non-Emptiness	$\exists w \top$

Definition 12 (Semantics of $\mathcal{MTL}$). We extend Definition 5 by the following clause:

- If $\varphi = {\sim}\psi$ for some formula ψ, then $K, T \models \varphi$ if and only if $K, T \not\models \psi$.

We note that usually (see [47]), $\mathcal{MTL}$ also contains dependence atoms; however, since these atoms can be expressed in $\mathcal{MTL}$ we omit them in the syntax. To make clear what the power of classical negation in this context is, we discuss the definition of dependence as well as some additional propositional connectives in $\mathcal{MTL}$.

First, the classical disjunction $\veebar$ is readily expressed in $\mathcal{MTL}$: $\varphi \veebar \psi$ is logically equivalent to ${\sim}({\sim}\varphi \wedge {\sim}\psi)$. Next we note that, analogously to the first-order case [1], the atom $=(p_1, \ldots, p_n)$ is logically equivalent with

$$\Big(\bigwedge_{1 \le i \le n-1} =(p_i) \Big) \rightarrow =(p_n)\,,$$

where $\rightarrow$ is the modal version of the intuitionistic implication with the semantics

$$K, T \models \varphi \rightarrow \psi \text{ iff for all } T' \subseteq T\text{: if } K, T' \models \varphi \text{ then } K, T' \models \psi\,.$$

The connective $\rightarrow$ has a short logically equivalent definition in $\mathcal{MTL}$ (see [47]), hence so does the atom $=(p_1, \ldots, p_n)$. The intuitionistic implication has been studied in the modal team semantics context in [63].

We want to mention one final extension of modal dependence logic in this section, the so-called extended modal dependence logic, $\mathcal{EMDL}$. It allows $\mathcal{ML}$-formulas instead of atoms inside the dependence atom [10].

Definition 13 (Syntax of $\mathcal{EMDL}$).

$$\varphi ::= p \mid \neg p \mid =(\{\psi,\}\,\psi) \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid \Diamond\varphi \mid \Box\varphi,$$

where p is an atomic proposition and ψ is an $\mathcal{ML}$-formula.

Definition 14 (Semantics of $\mathcal{EMDL}$). We extend Definition 5 by the following clause:

- $K, T \models =(\psi_1, \ldots, \psi_n)$ if for all $w_1, w_2 \in T$, if $K, \{w_1\} \models \psi_i \Leftrightarrow K, \{w_2\} \models \psi_i$ for $1 \leq i \leq n-1$, then $K, \{w_1\} \models \psi_n \Leftrightarrow K, \{w_2\} \models \psi_n$.

The interest in $\mathcal{EMDL}$ stems mainly from the fact, that it allows us to formulate some basic temporal dependencies. We give only one very simple example: The formula

$$=\!\left(\Diamond p, \Diamond^2 p, \ldots, \Diamond^n p, p\right)$$

expresses that "truth of p at this moment only depends on the truth of p in the previous n time steps" (on frame classes where the relation R denotes a backwards-oriented time-relation). So in a sense, $\mathcal{EMDL}$ can be seen as a *basic temporal dependence logic*.

We would like to point out that $\mathcal{EMDL}$ shares the *downwards closure property* of $\mathcal{MDL}$, but analogously to the first-order case, neither $\mathcal{MIL}$ nor $\mathcal{MTL}$ has this property.

3.2 Expressivity

The first results on expressive power of modal team-based logics are due to Sevenster [53]:

Theorem 17. *1. $\mathcal{MDL}$ is strictly more expressive than $\mathcal{ML}$.*
2. On singleton teams of evaluation, $\mathcal{MDL}$ is as expressive as $\mathcal{ML}$.

While the first result simply follows from the fact that $\mathcal{ML}$ is closed under union (of teams) but $\mathcal{MDL}$ is not, the second result requires an interesting proof that we would like to sketch. Given an $\mathcal{MDL}$-formula φ, we first use existentially quantified (Boolean) Skolem functions to replace dependence atoms. Next, we replace the existential quantifier by a big classical disjunction $\Vee$ over all possibilities for such functions. The result now follows since over singleton teams, the interpretations of the connectives $\Vee$ and $\vee$ agree.

This proof sketch points out the importance of classical disjunction. If we add $\Vee$ to $\mathcal{ML}$ or $\mathcal{MDL}$ we obtain the same expressive power, namely that of $\mathcal{EMDL}$ [10, 29]:

Theorem 18. $\mathcal{ML} < \mathcal{MDL} < \mathcal{ML}(\textcircled{=}) \equiv \mathcal{MDL}(\textcircled{=}) \equiv \mathcal{EMDL}$.

Concerning modal independence logic, the following is known [42]:

Theorem 19. *1.* $\mathcal{ML} < \mathcal{MDL} < \mathcal{MIL}$.
2. On singleton teams of evaluation, $\mathcal{MIL}$ *is as expressive as* $\mathcal{ML}$.

Again, the first result follows from simple closure properties. We will sketch the proof of the second result, since it will lead us to an important topic. Given an $\mathcal{MIL}$-formula φ, it is clear that on singleton teams it captures a property of Kripke models that

- is invariant under modal bisimulation (more on that in the next paragraph),
- only depends on the worlds that can be reached in a number of steps bounded by the modal depth of the formula.

These observations allow us to construct an $\mathcal{ML}$-formula that describes (by a big disjunction) all possibilities for satisfying Kripke models.

The just given proof can be extended to show that on singleton teams, $\mathcal{ML}$ extended by any $\mathcal{FO}$-definable dependence atoms (see Table 1) is as expressive as $\mathcal{ML}$.

The ideas used in the just sketched proof go back to a fundamental result by Johan van Benthem [59], characterizing exactly the properties of Kripke structures that modal logic $\mathcal{ML}$ can define in terms of the so-called bisimilarity. His results can be generalized to modal team logic, as we describe next.

Definition 15 (k-bisimulation). Let $K_1 = (W_1, R_1, \pi_1)$ and $K_2 = (W_2, R_2, \pi_2)$ be Kripke models. We define inductively what it means for worlds $w_1 \in W_1$ and $w_2 \in W_2$ to be k-bisimilar, for some $k \in \mathbb{N}$, written as $(K_1, w_1) \rightleftharpoons_k (K_2, w_2)$.

- $(K_1, w_1) \rightleftharpoons_0 (K_2, w_2)$ holds if for each propositional variable p, we have that $K_1, w_1 \models p$ if and only if $K_2, w_2 \models p$.
- $(K_1, w_1) \rightleftharpoons_{k+1} (K_2, w_2)$ holds if the following three conditions are satisfied:
  1. $(K_1, w_1) \rightleftharpoons_0 (K_2, w_2)$,
  2. for each successor w_1' of w_1 in K_1, there is a successor w_2' of w_2 in K_2 such that $(K_1, w_1') \rightleftharpoons_k (K_2, w_2')$ (*forward* condition),
  3. for each successor w_2' of w_2 in K_2, there is a successor w_1' of w_1 in K_1 such that $(K_1, w_1') \rightleftharpoons_k (K_2, w_2')$ (*backward* condition).

Full bisimulation is defined analogously as follows:

Definition 16 (full bisimulation). Let $K_1 = (W_1, R_1, \pi_1), K_2 = (W_2, R_2, \pi_2)$ be Kripke models and let w_1 and w_2 be worlds of K_1 and K_2. Then (K_1, w_1) and (K_2, w_2) are *bisimilar*, written as $(K_1, w_1) \rightleftharpoons (K_2, w_2)$, if there is a relation $Z \subseteq W_1 \times W_2$ such that $(w_1, w_2) \in Z$, and Z fulfils the following closure property:

- $(K_1, w_1) \rightleftharpoons_0 (K_2, w_2)$, for all $(w_1, w_2) \in Z$,
- for each successor w_1' of w_1 in K_1, there is a successor w_2' of w_2 in K_2 with $(w_1', w_2') \in Z$ (forward condition),

- for each successor w_2' of w_2 in K_2, there is a successor w_1' of w_1 in K_1 with $(w_1', w_2') \in Z$ (backward condition).

The famous theorem by van Benthem characterizing the expressive power of modal logics can now be stated as follows:

A *property of pointed models* is a class of pairs (K, w), where K is a Kripke model and w a world of K. For a formula φ we say that φ *expresses* the property $\{(K, w) \mid K, w \models \varphi\}$ (under the usual Kripke semantics). A team property is bisimulation-invariant if it is closed under bisimulation.

Theorem 20 (van Benthem's Theorem [59]). *Let P be a property of pointed Kripke structures. There is an $\mathcal{ML}$-formula which expresses P if and only if there is a first-order formula which expresses P and P is bisimulation-invariant.*

The result of van Benthem has been transferred into the field of modal logics with team semantics, as we want to explain next. First, the notion of bisimulation can very naturally be lifted to teams.

Definition 17 (team bisimulation). Let $K_1 = (W_1, R_1, \pi_1)$, $K_2 = (W_2, R_2, \pi_2)$ be Kripke models, let T_1 and T_2 be teams of K_1 and K_2. Then (K_1, T_1) and (K_2, T_2) are k-bisimilar, written as $K_1, T_1 \rightleftharpoons_k K_2, T_2$ if the following holds:

- for each $w_1 \in T_1$, there is some $w_2 \in T_2$ such that $(K_1, w_1) \rightleftharpoons_k (K_2, w_2)$,
- for each $w_2 \in T_2$, there is some $w_1 \in T_1$ such that $(K_1, w_1) \rightleftharpoons_k (K_2, w_2)$.

Analogously, we say that (K_1, T_1) and (K_2, T_2) are (fully) bisimilar, written as $K_1, T_1 \rightleftharpoons K_2, T_2$ if the following holds:

- for each $w_1 \in T_1$, there is some $w_2 \in T_2$ such that $(K_1, w_1) \rightleftharpoons (K_2, w_2)$,
- for each $w_2 \in T_2$, there is some $w_1 \in T_1$ such that $(K_1, w_1) \rightleftharpoons (K_2, w_2)$.

Now we can characterize the expressive power of some of our modal logics. The expressive power of a logic here is defined to be the set of properties expressible in it. More precisely, a *team property* is a class of pairs (K, T), where K is a Kripke model and T a team of K. For a formula φ we say that φ *expresses* the property $\{(K, T) \mid K, T \models \varphi\}$. A team property is bisimulation-invariant if it is closed under bisimulation.

The following characterizations of the expressive power of $\mathcal{EMDL}$ and $\mathcal{MTL}$ were obtained in [29, 43]:

Theorem 21. *Let P be a team property.*

1. *There is an $\mathcal{EMDL}$-formula which expresses P if and only if P is invariant under k-bisimulation for some k and downwards-closed.*
2. *There is an $\mathcal{MTL}$-formula which expresses P if and only if P is invariant unter k-bisimulation for some k.*

While the just given result already completely settles the question of expressivity of $\mathcal{EMDL}$ and $\mathcal{MTL}$ in terms of bisimulation, the following complete analogue of van Benthem's Theorem was finally obtained in [43]:

Theorem 22. *Let P be a team property. There is an $\mathcal{MTL}$-formula which expresses P if and only if there is a first-order formula which expresses P and P is bisimulation-invariant.*

It is worth to note that all our extensions of modal logic by further connectives or generalized dependence atoms define only team properties that are bisimulation-invariant. Hence the just given theorem implies that modal team logic $\mathcal{MTL}$ gives an upper bound with respect to expressivity for all these logics. In particular, modal logic with team semantics and classical negation is sufficient to express all FO-definable generalized dependence atoms. This observation can be strengthened as follows [43]:

Let $\mathsf{ML}^{\mathsf{FO}}$ denote the extension of $\mathcal{ML}$ by all generalized dependence atoms D that are $\mathcal{FO}$-definable without identity, in the extended setting, i.e., dependence atoms are applied not only to propositions but $\mathcal{ML}$-formulae.

Theorem 23. $\mathsf{ML}^{\mathsf{FO}}$ *is equally expressive as* $\mathcal{MTL}$.

3.3 Complexity

The starting point for complexity studies of modal logic is a paper by Ladner from 1977 [45] in which he proved the following theorem:

Theorem 24. *Satisfiability for modal logic* $\mathcal{ML}$ *is* PSPACE*-complete.*

His result easily carries over from the usual modal semantics to team semantics. The upper bound follows from Ladner's so-called witness algorithm, which is not so important for us here. The lower bound is given by a reduction from the standard PSPACE-complete problem QBF, the evaluation problem for quantified Boolean formulas, where alternations of modalities are used to force a satisfying Kripke model to imitate the evaluation tree of the given formula [2].

In a very clever way this was extended by Sevenster [53] as follows:

Theorem 25. *Satisfiability for* $\mathcal{MDL}$ *is* NEXPTIME*-complete.*

To prove the upper bound, one has to express the dependencies by Boolean Skolem functions, similar as in the proof of Theorem 17. Then we can use nondeterminism to guess those functions in exponential time, and check satisfiability. For the lower bound, Sevenster presents a reduction from Dependence-QBF, a variant of the above-mentioned QBF extended by dependencies among variables [49], a problem which is NEXPTIME-complete. This reduction is essentially Ladner's reduction extended by dependence atoms.

Lohmann and Vollmer [46] extended Sevenster's result by determining the complexity of every fragment of $\mathcal{MDL}$, given by any subset of the modalities $\Box$ and $\Diamond$, by restricting the allowed propositional connectives to any subset of $\{\wedge, \vee, \neg, \top, \bot, \mathbin{|}\}$ and by considering fragments with and without dependence atoms. Their results can be summarized as in Table 2. An entry "+" in the table

Table 2 Complexity of satisfiability for fragments of modal dependence logic

$\Box$	$\Diamond$	$\wedge$	$\vee$	$\neg$	$\top$	$\bot$	$=()$	$\mathbb{V}$	Complexity
+	+	+	*	+	*	*	+	*	NEXPTIME
+	+	+	+	+	*	*	−	*	PSPACE
+	+	+	+	−	*	+	*	*	PSPACE
+	+	+	−	+	*	*	−	+	Σ_2^p
+	+	+	−	−	*	+	*	+	Σ_2^p
+	+	+	−	+	*	*	−	−	coNP
+	+	+	−	−	*	+	*	−	coNP
+	−	+	+	+	*	*	*	*	NP
−	+	+	+	+	*	*	*	*	NP
+	−	+	−	+	*	*	*	+	NP
−	+	+	−	+	*	*	*	+	NP
+	−	+	−	+	*	*	*	−	PTIME
−	+	+	−	+	*	*	*	−	PTIME
+	−	+	*	−	*	*	*	*	PTIME
−	+	+	*	−	*	*	*	*	PTIME
*	*	−	*	*	*	*	*	*	PTIME
*	*	*	*	−	*	−	*	*	trivial
−	−	+	+	+	*	*	*	*	NP
−	−	+	*	+	*	*	*	+	NP
−	−	*	−	*	*	*	*	−	PTIME
−	−	*	*	−	*	*	*	*	PTIME

means that the syntactic element is allowed, "−" means it is forbidden, and "∗" means that the complexity does not depend on whether the element is present or not. All rows in the table denote completeness results for the respective complexity class under polynomial-time many-one reductions, except those for PTIME.

In a similar way, the complexity of the model-checking problem for fragments of modal dependence logic has been classified in [9]. For some fragments, model checking is NP-complete, for others it is solvable in polynomial time.

Theorem 26. *Model checking for $\mathcal{MDL}$ is* NP*-complete.*

We want to turn to the explanation of one special case in Table 2, the complexity of Poor Man's Logic. In the context of modal logic, *poor man's formulas* are just formulas that do not contain $\vee$. Hemaspaandra [31, 32] showed that poor man's modal logic is PSPACE-complete over the class of Kripke structures in which every world has at most two successors. The proof is again by a reduction from QBF, where we express the QBF-tree by alternations of modalities. Important now is that without disjunction, we cannot express the tree-structure. In fact, satisfiability for poor man's modal logic over K (the class of all Kripke structures) is only coNP-complete. The requirement that in every model each world has at most two successors is essential for the complexity. Lohmann and Vollmer [46] showed:

Theorem 27. *Poor man's modal dependence logic, i.e., the fragment of modal dependence logic allowing only the propositional connectives $\wedge$ and $\neg$ (besides the two modalites and dependence atoms), is* NEXPTIME*-complete.*

The proof relies as before on a reduction from Dependence-QBF. We express the QBF-tree by alternations of modalities. As in the case of Hemaspaandra's proof we cannot enforce tree-structure of the Kripke model without disjunction. In this case, however, we can use dependence atoms to ensure that everything in the model that does not belong to the tree is essentially nothing else than a copy of a subtree; hence, in difference to Hemaspaandra, we do not need the requirement about the number of successors of a world.

Next, we turn to modal independence logic. The following has been proven in [42]:

Theorem 28. *Satisfiability for $\mathcal{MIL}$ is* NEXPTIME*-complete.*

While the lower bound follows trivially from the complexity result for $\mathcal{MDL}$, the upper bound is proven by an embedding of $\mathcal{MIL}$ into the Gödel-Kalmár-Schütte fragment of all $\mathcal{FO}$-sentences with prefix $\exists^*\forall^2\exists^*$ (without function symbols, without equality) in a satisfiability preserving way. This fragment is decidable in NEXPTIME [3]. This proof thus is different from the one given by Sevenster (and it is not clear how to extend his ideas to include the independence atom), but it yields Sevenster's result as a corollary.

The embedding into the $\mathcal{FO}$-fragment is even possible in a much more general context [42]:

Theorem 29. *Satisfiability for $\mathcal{ML}$ extended by dependence atoms that can be defined in $\exists^*\forall^2\exists^*$ is in* NEXPTIME.

For a large class of generalized dependence atoms, satisfiability can thus be placed into the class NEXPTIME, while for a few particular atoms, we have completeness—above we already mentioned this for dependence and independence, and in [30], satisfiability for modal inclusion logic $\mathcal{MINCL}$, i.e., $\mathcal{ML}$ extended by the inclusion atom (see Table 1), was shown to be NEXPTIME-complete as well.

Also the model-checking problem for modal logic extended by generalized dependence atoms was studied in [42].

Theorem 30. *Model checking for $\mathcal{ML}$ extended by $\mathcal{FO}$-definable dependence atoms is in* NP.

Recall that by Theorem 26 model checking for $\mathcal{MDL}$ (and also for $\mathcal{MIL}$) is NP-complete.

3.4 Propositional Logic

Propositional logic with team semantics is nothing else than modal logic with team semantics but without connections between the team members (worlds). The syntax is the following:

Definition 18 (Syntax of $\mathcal{PL}$).

$$\varphi ::= p \mid \neg p \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi), \qquad \text{where } p \in \Phi.$$

The semantics is exactly the same as for the modal case. Propositional logic can now be extended by different generalized dependence atoms (such as those given in Table 1) as well as different propositional connectives (intuitionistic implication, classical disjunction, etc.) The expressive power of some of these logics (i.e. the classes of teams that are definable) has been studied by Yang [63]; in particular, propositional dependence logic ($\mathcal{PL}$ plus dependence atoms) can define all nonempty downwards-closed team properties, and propositional team logic ($\mathcal{PL}$ plus classical negation) can define all team properties.

Concerning complexity questions, the following results are known:

Theorem 31. *1. The satisfiability and the model-checking problems for propositional dependence logic and propositional independence logic are* NP*-complete [9, 27, 46].*

2. *The satisfiability problem for propositional inclusion logic is* EXPTIME*-complete [30], while its model-checking problem is in* PTIME *(Lauri Hella, personal communication).*
3. *The model-checking problem for propositional dependence logic extended by classical negation is* PSPACE*-complete [47].*
4. *The satisfiability problem for propositional inclusion logic extended by classical negation as well as for propositional independence logic extended by classical negation is complete for the class* AEXPTIME(poly) *of all problems solvable by alternating Turing machines in exponential time making only a polynomial number of alternations; the model-checking problem for both logics is* PSPACE*-complete [30].*

Besides the above-summarized results, also the validity problem for propositional logic and some of its extensions have been studied. It is worth noting that, unlike for classical logics closed under the classical negation, the satisfiability problem and the validity problem for most of the logics discussed in this article are not dual to each other. Partial results for axiomatizability have also been obtained [27, 52, 61, 64].

4 Conclusion

1. A number of complexity questions for fragments of first-order dependence logic or variants remain unsettled. The complexity of the validity problem for $\mathcal{D}^2$ is one example. More generally, one might ask what decidability results for first-order logic, e.g., for formula classes defined by restricted quantifier prefixes, transfer into the context of team-based logics.
2. While many expressivity results of modal and propositional logic have been stated in this survey and this issue is more or less settled in the first-order case, it has to be mentioned that some very basic cases in modal logic still remain unsettled. In particular, what is the expressive power of propositional or modal independence logic? It is worth noting that very recently a version of Theorem 21 for (extended) modal inclusion logic has been shown in [28].
3. In Section 3.2 we stated equal expressivity for many dialects of modal logics. However, we did not touch the topic of succinctness. As an example, while on singleton teams of evaluation, $\mathcal{MIL}$ equally expressive than $\mathcal{ML}$, it can be proven that it is exponentially more succinct [42]. Also, in Theorem 18 we stated that $\mathcal{EMDL}$ and $\mathcal{ML}$ plus classical disjunction have the same expressive power. In [29] it was shown that any translation from $\mathcal{EMDL}$ to $\mathcal{ML}$ with disjunction necessarily leads to an exponential blow-up in formula size. In most other cases, the question of succinctness remains unsettled so far.
4. The question of axiomatizability has not been covered in detail in this survey. We mention that axiomatizability of some sublogics of $\mathcal{MTL}$ has been studied, e.g., in [63] and [52], but remains open for many logics. Related to this is the complexity of the tautology problem. In particular it remains open if we can axiomatize $\mathcal{MTL}$.
5. While we mentioned a number of complexity results on modal dependence logic and some of its extensions, this issue remains unsettled for full $\mathcal{MTL}$. In particular, what is the complexity of satisfiability and validity of $\mathcal{MTL}$?

Acknowledgements The authors thank the anonymous referee for corrections and valuable comments. The second author was supported by grants 292767, 275241, and 264917 of the Academy of Finland.

References

1. Abramsky, S., Väänänen, J.: From IF to BI: a tale of dependence and separation. Synthese **167**(2, Knowledge, Rationality & Action), 207–230 (2009). doi:10.1007/s11229-008-9415-6. http://dx.doi.org/10.1007/s11229-008-9415-6
2. Blackburn, P., de Rijke, M., Venema, Y.: Modal logics. In: Cambridge Tracts in Theoretical Computer Science, vol. 53. Cambridge University Press, Cambridge (2001)
3. Börger, E., Grädel, E., Gurevich, Y.: The Classical Decision Problem. Perspectives in Mathematical Logic. Springer, Berlin (1997)
4. Church, A.: A note on the Entscheidungsproblem. J. Symb. Log. **1**(1), 40–41 (1936)

5. Cook, S.A.: A hierarchy for nondeterministic time complexity. In: Conference Record, Fourth Annual ACM Symposium on Theory of Computing, pp. 187–192. ACM, New York (1972)
6. Durand, A., Kontinen, J.: Hierarchies in dependence logic. ACM Trans. Comput. Log. **13**(4), 1–21 (2012)
7. Durand, A., Ebbing, J., Kontinen, J., Vollmer, H.: Dependence logic with a majority quantifier. In: Chakraborty, S., Kumar, A. (eds.) FSTTCS, LIPIcs, vol. 13, pp. 252–263. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany (2011)
8. Durand, A., Ebbing, J., Kontinen, J., Vollmer, H.: Dependence logic with a majority quantifier. J. Log. Lang. Inf. **24**(3), 289–305 (2015). doi:10.1007/s10849-015-9218-3. http://dx.doi.org/10.1007/s10849-015-9218-3
9. Ebbing, J., Lohmann, P.: Complexity of model checking for modal dependence logic. In: SOFSEM 2012: Theory and Practice of Computer Science. Lecture Notes in Computer Science, vol. 7147, pp. 226–237. Springer, Berlin, Heidelberg (2012)
10. Ebbing, J., Hella, L., Meier, A., Müller, J.S., Virtema, J., Vollmer, H.: Extended modal dependence logic. In: WoLLIC. Lecture Notes in Computer Science, vol. 8071, pp. 126–137. Springer, Berlin, Heidelberg (2013)
11. Ebbing, J., Kontinen, J., Müller, J., Vollmer, H.: A fragment of dependence logic capturing polynomial time. Log. Methods Comput. Sci. **10**(3) (2014). doi:10.2168/LMCS-10(3:3)2014. http://dx.doi.org/10.2168/LMCS-10(3:3)2014
12. Engström, F.: Generalized quantifiers in dependence logic. J. Log. Lang. Inf. **21**, 299–324 (2012). http://dx.doi.org/10.1007/s10849-012-9162-4. 10.1007/s10849-012-9162-4
13. Engström, F., Kontinen, J.: Characterizing quantifier extensions of dependence logic. J. Symb. Log. **78**, 0–9 (2013)
14. Fagin, R.: Generalized first-order spectra and polynomial-time recognizable sets. In: Complexity of Computation. Proceedings of SIAM-AMS Symposium in Applied Mathematics, New York, 1973, pp. 43–73. SIAM–AMS Proceedings, vol. VII. American Mathematical Society, Providence, RI (1974)
15. Fagin, R.: Finite-model theory - a personal perspective. Theor. Comput. Sci. **116**(1&2), 3–31 (1993)
16. Galliani, P.: Inclusion and exclusion dependencies in team semantics - on some logics of imperfect information. Ann. Pure Appl. Log. **163**(1), 68–84 (2012)
17. Galliani, P., Hella, L.: Inclusion logic and fixed point logic. In: Rocca, S.R.D. (ed.) Computer Science Logic 2013 (CSL 2013). Leibniz International Proceedings in Informatics (LIPIcs), vol. 23, pp. 281–295. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2013). doi:http://dx.doi.org/10.4230/LIPIcs.CSL.2013.281. http://drops.dagstuhl.de/opus/volltexte/2013/4203
18. Galliani, P., Hannula, M., Kontinen, J.: Hierarchies in independence logic. In: Rocca, S.R.D. (ed.) Computer Science Logic 2013 (CSL 2013). Leibniz International Proceedings in Informatics (LIPIcs), vol. 23, pp. 263–280. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2013). doi:http://dx.doi.org/10.4230/LIPIcs.CSL.2013.263. http://drops.dagstuhl.de/opus/volltexte/2013/4202
19. Grädel, E.: Capturing complexity classes by fragments of second-order logic. Theor. Comput. Sci. **101**(1), 35–57 (1992)
20. Grädel, E.: Model-checking games for logics of imperfect information. Theor. Comput. Sci. **493**, 2–14 (2013). doi:10.1016/j.tcs.2012.10.033. http://dx.doi.org/10.1016/j.tcs.2012.10.033
21. Grädel, E., Väänänen, J.A.: Dependence and independence. Stud. Logica **101**(2), 399–410 (2013)
22. Grandjean, E., Olive, F.: Graph properties checkable in linear time in the number of vertices. J. Comput. Syst. Sci. **68**(3), 546–597 (2004)
23. Hannula, M.: Axiomatizing first-order consequences in independence logic. Ann. Pure Appl. Logic **166**(1), 61–91 (2015). doi:10.1016/j.apal.2014.09.002. http://dx.doi.org/10.1016/j.apal.2014.09.002

24. Hannula, M.: Hierarchies in inclusion logic with lax semantics. In: Banerjee, M., Krishna, S.N. (eds.) Proceedings of Logic and Its Applications - 6th Indian Conference, ICLA 2015, Mumbai, India, January 8–10, 2015. Lecture Notes in Computer Science, vol. 8923, pp. 100–118. Springer, Berlin (2015). doi:10.1007/978-3-662-45824-2_7
25. Hannula, M.: On variants of dependence logic: axiomatizability and expressiveness. Ph.D. thesis, University of Helsinki (2015)
26. Hannula, M., Kontinen, J.: Hierarchies in independence and inclusion logic with strict semantics. J. Log. Comput. **25**(3), 879–897 (2015). doi:10.1093/logcom/exu057. http://dx.doi.org/10.1093/logcom/exu057
27. Hannula, M., Kontinen, J., Virtema, J., Vollmer, H.: Complexity of propositional independence and inclusion logic. In: Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9234, pp. 269–280. Springer, Berlin (2015)
28. Hella, L., Stumpf, J.: The expressive power of modal logic with inclusion atoms. In: Esparza, J., Tronci, E. (eds.) Proceedings Sixth International Symposium on Games, Automata, Logics and Formal Verification, GandALF 2015, Genoa, Italy, 21–22nd September 2015, EPTCS, vol. 193, pp. 129–143 (2015). doi:10.4204/EPTCS.193.10
29. Hella, L., Luosto, K., Sano, K., Virtema, J.: The expressive power of modal dependence logic. In: Advances in Modal Logic 10, Invited and Contributed Papers from the Tenth Conference on "Advances in Modal Logic," Held in Groningen, The Netherlands, August 5–8, 2014, pp. 294–312. College Publications, London (2014). http://www.aiml.net/volumes/volume10/Hella-Luosto-Sano-Virtema.pdf
30. Hella, L., Kuusisto, A., Meier, A., Vollmer, H.: Modal inclusion logic: being lax is simpler than being strict. In: Italiano, G.F., Pighizzini, G., Sannella, D. (eds.) Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24–28, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9234, pp. 281–292. Springer, Berlin (2015)
31. Hemaspaandra, E.: The complexity of poor man's logic. J. Log. Comput. **11**(4), 609–622 (2001). Corrected version: [32]
32. Hemaspaandra, E.: The complexity of poor man's logic. CoRR **cs.LO/9911014v2** (2005). http://arxiv.org/abs/cs/9911014v2
33. Henkin, L.: Logical Systems Containing Only a Finite Number of Symbols. Presses De l'Université De Montréal, Montreal (1967)
34. Hintikka, J., Sandu, G.: Informational independence as a semantical phenomenon. In: Logic, Methodology and Philosophy of Science, VIII (Moscow, 1987). Studies in Logic and the Foundations of Mathematics, vol. 126, pp. 571–589. North-Holland, Amsterdam (1989). doi:10.1016/S0049-237X(08)70066-1. http://dx.doi.org/10.1016/S0049-237X(08)70066-1
35. Hodges, W.: Compositional semantics for a language of imperfect information. Log. J. IGPL **5**, 539–563 (1997)
36. Immerman, N.: Relational queries computable in polynomial time. Inf. Control. **68** (1-3), 86–104 (1986). doi:10.1016/S0019-9958(86)80029-8. http://dx.doi.org/10.1016/S0019-9958(86)80029-8
37. Kontinen, J.: Coherence and computational complexity of quantifier-free dependence logic formulas. Stud. Logica **101**(2), 267–291 (2013)
38. Kontinen, J., Nurmi, V.: Team logic and second-order logic. Fundamenta Informaticae **106** (2–4), 259–272 (2011)
39. Kontinen, J., Väänänen, J.: On definability in dependence logic. J. Log. Lang. Inf. **18**(3), 317–332 (2009)
40. Kontinen, J., Kuusisto, A., Lohmann, P., Virtema, J.: Complexity of two-variable dependence logic and if-logic. Inf. Comput. **239**, 237–253 (2014). doi:10.1016/j.ic.2014.08.004. http://dx.doi.org/10.1016/j.ic.2014.08.004
41. Kontinen, J., Kuusisto, A., Virtema, J.: Decidable fragments of logics based on team semantics. CoRR **abs/1410.5037** (2014). http://arxiv.org/abs/1410.5037

42. Kontinen, J., Müller, J., Schnoor, H., Vollmer, H.: Modal independence logic. In: Advances in Modal Logic 10, Invited and Contributed Papers from the Tenth Conference on "Advances in Modal Logic," Held in Groningen, The Netherlands, August 5–8, 2014, pp. 353–372. College Publications, London (2014). http://www.aiml.net/volumes/volume10/Kontinen-Mueller-Schnoor-Vollmer.pdf
43. Kontinen, J., Müller, J.S., Schnoor, H., Vollmer, H.: A Van Benthem Theorem for Modal Team Semantics. In: 24th EACSL Annual Conference on Computer Science Logic (CSL 2015). Leibniz International Proceedings in Informatics (LIPIcs), vol. 41, pp. 277–291. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2015)
44. Kuusisto, A.: A double team semantics for generalized quantifiers. J. Log. Lang. Inf. **24**(2), 149–191 (2015). doi:10.1007/s10849-015-9217-4. http://dx.doi.org/10.1007/s10849-015-9217-4
45. Ladner, R.E.: The computational complexity of provability in systems of modal propositional logic. SIAM J. Comput. **6**(3), 467–480 (1977)
46. Lohmann, P., Vollmer, H.: Complexity results for modal dependence logic. Stud. Logica **101**(2), 343–366 (2013)
47. Müller, J.S.: Satisfiability and model checking in team based logics. Ph.D. thesis, Leibniz Universität Hannover (2014)
48. Nurmi, V.: Dependence logic: Investigations into higher-order semantics defined on teams. Ph.D. thesis, University of Helsinki (2009)
49. Peterson, G., Reif, J., Azhar, S.: Lower bounds for multiplayer noncooperative games of incomplete information. Comput. Math. Appl. **41**(7-8), 957–992 (2001). doi:10.1016/S0898-1221(00)00333-3. http://www.sciencedirect.com/science/article/B6TYJ-43P387P-19/2/cd72ba7ccb5f3c2a2b65eb3c45aa2ca7
50. Pratt-Hartmann, I.: Complexity of the two-variable fragment with counting quantifiers. J. Log. Lang. Inf. **14**, 369–395 (2005). doi:10.1007/s10849-005-5791-1. http://portal.acm.org/citation.cfm?id=1080942.1080949
51. Rönnholm, R.: Capturing k-ary existential second order logic with k-ary inclusion-exclusion logic. CoRR **abs/1502.05632** (2015). http://arxiv.org/abs/1502.05632
52. Sano, K., Virtema, J.: Axiomatizing propositional dependence logics. In: Kreutzer, S. (ed.) 24th EACSL Annual Conference on Computer Science Logic, CSL 2015, September 7–10, 2015, Berlin, Germany. LIPIcs, vol. 41, pp. 292–307. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2015). doi:10.4230/LIPIcs.CSL.2015.292
53. Sevenster, M.: Model-theoretic and computational properties of modal dependence logic. J. Log. Comput. **19**(6), 1157–1173 (2009). doi:10.1093/logcom/exn102. http://logcom.oxfordjournals.org/cgi/content/abstract/exn102v1
54. Tulenheimo, T., Sevenster, M.: On modal logic, IF logic, and IF modal logic. In: Advances in Modal Logic 6, Papers from the Sixth Conference on "Advances in Modal Logic," Held in Noosa, Queensland, Australia, on 25–28 September 2006, pp. 481–501. College Publications, London (2006). http://www.aiml.net/volumes/volume6/Tulenheimo-Sevenster.ps
55. Turing, A.: On computable numbers, with an application to the entscheidungsproblem. Proc. Lond. Math. Soc. Ser. 2 **42**, 230–265 (1936)
56. Väänänen, J.: Dependence Logic. London Mathematical Society Student Texts, vol. 70. Cambridge University Press, Cambridge (2007). doi:10.1017/CBO9780511611193. http://dx.doi.org/10.1017/CBO9780511611193
57. Väänänen, J.: Team Logic. In: Johan van Benthem Benedikt Löowe, D.G. (ed.) Interactive Logic. Texts in Logic and Games, vol. 1, pp. 281–302. Amsterdam University Press, Amsterdam (2007)
58. Väänänen, J.: Modal dependence logic. In: Apt, K.R., van Rooij, R. (eds.) New Perspectives on Games and Interaction. Texts in Logic and Games, vol. 4, pp. 237–254. Amsterdam University Press, Amsterdam (2008)
59. van Benthem, J.: Modal Logic and Classical Logic. Bibliopolis, Berkeley, CA (1985)

60. Vardi, M.Y.: The complexity of relational query languages (extended abstract). In: Lewis, H.R., Simons, B.B., Burkhard, W.A., Landweber, L.H. (eds.) Proceedings of the 14th Annual ACM Symposium on Theory of Computing, May 5–7, 1982, San Francisco, California, USA, pp. 137–146. ACM, New York (1982). doi:10.1145/800070.802186
61. Virtema, J.: Complexity of validity for propositional dependence logics. In: Proceedings Fifth International Symposium on Games, Automata, Logics and Formal Verification, GandALF 2014, Verona, Italy, September 10–12, 2014. Electronic Proceedings in Theoretical Computer Science, vol. 161, pp. 18–31 (2014). doi:10.4204/EPTCS.161.5. http://dx.doi.org/10.4204/EPTCS.161.5
62. Yang, F.: Expressing second-order sentences in intuitionistic dependence logic. Stud. Logica **101**(2), 323–342 (2013)
63. Yang, F.: On extensions and variants of dependence logic – a study of intuitionistic connectives in the team semantics setting. Ph.D. thesis, University of Helsinki (2014)
64. Yang, F., Väänänen, J.: Propositional logics of dependence. Annals of Pure and Applied Logic 167, 557–589 (2016)