Logical Methods in Computer Science
Volume 15, Issue 2, 2019, pp. 4:1–4:24
https://lmcs.episciences.org/

Submitted Jan. 17, 2018
Published Apr. 26, 2019

VALIDITY AND ENTAILMENT IN MODAL AND PROPOSITIONAL DEPENDENCE LOGICS

MIIKA HANNULA

Department of Computer Science, University of Auckland, New Zealand

Department of Mathematics and Statistics, University of Helsinki, Finland
e-mail address: miika.hannula@helsinki.fi

Abstract. The computational properties of modal and propositional dependence logics have been extensively studied over the past few years, starting from a result by Sevenster showing **NEXPTIME**-completeness of the satisfiability problem for modal dependence logic. Thus far, however, the validity and entailment properties of these logics have remained mostly unaddressed. This paper provides a comprehensive classification of the complexity of validity and entailment in various modal and propositional dependence logics. The logics examined are obtained by extending the standard modal and propositional logics with notions of dependence, independence, and inclusion in the team semantics context. In particular, we address the question of the complexity of validity in modal dependence logic. By showing that it is **NEXPTIME**-complete we refute an earlier conjecture proposing a higher complexity for the problem.

1. Introduction

The notions of dependence and independence are pervasive in various fields of science. Usually these concepts manifest themselves in the presence of *multitudes* (e.g. events or experiments). Dependence logic [28] is a recent logical formalism which, in contrast to others, has exactly these multitudes as its underlying concept. In this article we study dependence logic in the propositional and modal logic context and present a complete classification of the computational complexity of their associated entailment and validity problems.

In first-order logic, the standard formal language behind all mathematics and computer science, dependencies between variables arise strictly from the order of their quantification. Consequently, more subtle forms of dependencies cannot be captured, a phenomenon exemplified by the fact that first-order logic lacks expressions for statements of the form

“for all x there is y, and for all u there is v, such that $R(x, y, u, v)$”

where y and v are to be chosen independently from one another. To overcome this barrier, branching quantifiers of Henkin [15] and independence-friendly logic of Hintikka and Sandu

Key words and phrases: dependence logic, inclusion logic, independenc logic, modal logic, propositional logic, complexity, validity, entailment.

This work was supported by the Marsden Fund grant UOA1628 and the Academy of Finland grant 308712.

DOI:10.23638/LMCS-15(2:4)2019

[16] suggested the use of quantifier manipulation. Dependence logic instead extends first-order logic at the atomic level with the introduction of new dependence atoms

$$\mathrm{dep}(x_1, \ldots, x_n) \tag{1.1}$$

which indicate that the value of x_n depends only on the values of $x_1, \ldots, x_{n-1}$. Dependence atoms are evaluated over *teams*, i.e., sets of assignments which form the basis of *team semantics*. The concept of team semantics was originally proposed by Hodges in refutation of the view of Hintikka that the logics of imperfect information, such as his independence-friendly logic, escape natural compositional semantics [17]. By the development of dependence logic it soon became evident that team semantics serves also as a connecting link between the aforenementioned logics and the relational database theory. In particular, team semantics enables the extensions of even weaker logics, such as modal and propositional logics, with various sophisticated dependency notions known from the database literature [5, 7, 18, 19]. In this article we consider modal and propositional dependence logics that extend modal and propositional logics with dependence atoms similar to (1.1), the only exception being that dependence atoms here declare dependencies between propositions. We establish a complete classification of the computational complexity of the associated entailment and validity problems, including a solution to an open problem regarding the complexity of validity in modal dependence logic.

Modal dependence logic was introduced by Väänänen in 2008 [29], and soon after it was shown to enjoy a **NEXPTIME**-complete satisfiability problem [27]. Since then the expressivity, complexity, and axiomatizability properties of modal dependence logic and its variants have been exhaustively studied. Especially the complexity of satisfiability and model checking for modal dependence logic and its variants has been already comprehensively classified [3, 4, 9, 10, 12, 14, 18, 19, 22, 24]. It is worth noting here that satisfiability and validity are not dual to each other in the dependence logic context. Dependence logic cannot express classical negation nor logical implication which renders its associated validity, satisfiability, and entailment problems genuinely different. Entailment and validity of modal and propositional dependence logics have been axiomatically characterized by Yang and Väänänen in [32, 31, 33] and also by Sano and Virtema in [26]. Nevertheless, the related complexity issues have remained almost totally unaddressed. The aim of this article is to address this shortage in research by presenting a comprehensive classification with regards to these questions.

A starting point for our endeavour is a recent result by Virtema which showed that the validity problem for propositional dependence logic is **NEXPTIME**-complete [30]. In that paper the complexity of validity for modal dependence logic remained unsettled, although it was conjectured to be harder than that for propositional dependence logic. This conjecture is refuted in this paper as the same exact **NEXPTIME** bound is shown to apply to modal dependence logic as well. Furthermore, we show that this result applies to the extension of propositional dependence logic with quantifiers as well as to the so-called extended modal logic which can express dependencies between arbitrary modal formulae (instead of simple propositions). These complexity bounds follow as corollaries from a more general result showing that the entailment problem for (extended) modal dependence and propositional dependence logics is complete for **co-NEXPTIME**$^{\mathbf{NP}}$. We also consider modal logic extended with so-called intuitionistic disjunction and show that the associated entailment, validity, and satisfiability problems are all **PSPACE**-complete, which is, in all the three categories the complexity of the standard modal logic.

The aforementioned results have interesting consequences. First, combining results from this paper and [27, 30] we observe that similarly to the standard modal logic case the complexity of validity and satisfiability coincide for (extended) modal dependence logic. Secondly, it was previously known that propositional and modal dependence logics deviate on the complexity of their satisfiability problem (**NP**-complete vs. **NEXPTIME**-complete [22, 27], resp.) and that the standard propositional and modal logics differ from one another on both satisfiability and validity (**NP**-complete/**co-NP**-complete vs. **PSPACE**-complete, resp. [2, 20, 21]). Based on this it is somewhat surprising to find out that modal and propositional dependence logics correspond to one another in terms of the complexity of both their validity and entailment problems.

We also establish exact complexity bounds for entailment and validity of quantified propositional independence and inclusion logics. These logics are extensions of propositional logic with quantifiers and either independence or inclusion atoms [8]. We obtain our results by investigating recent generalizations of the quantified Boolean formula problem. The validity and entailment problems for quantified propositional independence logic are both shown to be **co-NEXPTIME**$^{\mathbf{NP}}$-complete. For quantified propositional inclusion logic entailment is shown to be **co-NEXPTIME**-complete whereas validity is only **EXPTIME**-complete. Using standard reduction methods the examined quantified propositional logics can be interpreted as fragments of modal independence and inclusion logics. Our findings then imply that validity is harder for modal independence logic than it is for modal dependence logic (unless the exponential-time hierarchy collapses at a low level), although in terms of satisfiability both logics are **NEXPTIME**-complete [18]. We refer the reader to Table 1 for a summary of our results.

Organization. This article is organized as follows. In Section 2 we give a short introduction to modal dependence logics, followed by Section 3 which proves **co-NEXPTIME**$^{\mathbf{NP}}$-membership for modal dependence logic entailment. In Section 4 we define (quantified) propositional dependence logics, and in Section 5 we show **co-NEXPTIME**$^{\mathbf{NP}}$-hardness for entailment in this logic. In Section 6 these findings are drawn together to establish exact complexity bounds. In Sections 7 and 8 we shift focus to validity and entailment of modal and quantified propostitional logics defined in terms of independence and inclusion atoms, respectively. Finally, Section 9 is reserved for conclusions. Throughout the paper we assume that the reader is familiar with the basic concepts of propositional and modal logic, as well as those of computational complexity. All the hardness results in the paper are stated under polynomial-time reductions.

2. Modal Dependence Logics

In this section we introduce extensions of modal logic with dependencies and present some of their basic properties. Following the common convention in team semantics we restrict attention to formulae in negation normal form (NNF). Note that the concept of negation referred to here is not classical. In the team semantics context "$\neg$" is used to express that something is not true for all individual members of a team. Later in this section we also present an auxiliary relational variant of modal logic for pointed Kripke models to facilitate the upper bound proof of the next section. For this variant we employ negation classically and do not restrict its scope in formulae.

The syntax of *modal logic* (ML) is generated by the following grammar:

$$\phi ::= p \mid \neg p \mid (\phi \wedge \phi) \mid (\phi \vee \phi) \mid \Box\phi \mid \Diamond\phi. \tag{2.1}$$

Extensions of modal logic with different dependency notions are made possible via a generalization of the standard Kripke semantics by teams. A *Kripke model* over a set of variables V is a tuple $\mathcal{M} = (W, R, \pi)$ where W is a non-empty set of worlds, R is a binary relation over W, and $\pi\colon V \to \mathcal{P}(W)$ is a function that associates each variable with a set of worlds. A *team* T of a Kripke model $\mathcal{M} = (W, R, \pi)$ is a subset of W. For the team semantics of modal operators we define the set of successors of a team T as $R[T] := \{w \in W \mid \exists w' \in T : (w', w) \in R\}$ and the set of successor teams of a team T as $R\langle T\rangle := \{T' \subseteq R[T] \mid \forall w \in T\, \exists w' \in T' : (w, w') \in R\}$. The team semantics of modal logic is now defined as follows.

Definition 2.1 (Team Semantics of ML)**.** Let ϕ be an ML formula, let $\mathcal{M} = (W, R, \pi)$ be a Kripke model over $V \supseteq \mathsf{Var}(\phi)$, and let $T \subseteq W$. The satisfaction relation $\mathcal{M}, T \models \phi$ is defined as follows:

$$\begin{aligned}
\mathcal{M}, T \models p \quad &:\Leftrightarrow \quad T \subseteq \pi(p),\\
\mathcal{M}, T \models \neg p \quad &:\Leftrightarrow \quad T \cap \pi(p) = \emptyset,\\
\mathcal{M}, T \models \phi_1 \wedge \phi_2 \quad &:\Leftrightarrow \quad \mathcal{M}, T \models \phi_1 \text{ and } \mathcal{M}, T \models \phi_2,\\
\mathcal{M}, T \models \phi_1 \vee \phi_2 \quad &:\Leftrightarrow \quad \exists T_1, T_2 : T_1 \cup T_2 = T, \mathcal{M}, T_1 \models \phi_1, \text{ and } \mathcal{M}, T_2 \models \phi_2,\\
\mathcal{M}, T \models \Diamond\phi \quad &:\Leftrightarrow \quad \exists T' \in R\langle T\rangle : \mathcal{M}, T' \models \phi,\\
\mathcal{M}, T \models \Box\phi \quad &:\Leftrightarrow \quad \mathcal{M}, R[T] \models \phi.
\end{aligned}$$

We write $\phi \equiv \psi$ to denote that ϕ and ψ are *equivalent*, i.e., for all Kripke models $\mathcal{M}$ and teams T, $\mathcal{M}, T \models \phi$ iff $\mathcal{M}, T \models \phi'$. Let $\Sigma \cup \{\phi\}$ be a set of formulae. We write $\mathcal{M}, T \models \Sigma$ iff $\mathcal{M}, T \models \phi$ for all $\phi \in \Sigma$, and say that Σ *entails* ϕ if for all $\mathcal{M}$ and T, $\mathcal{M}, T \models \Sigma$ implies $\mathcal{M}, T \models \phi$. Let $\mathcal{L}$ be a logic in the team semantics setting. The *entailment problem* for $\mathcal{L}$ is to decide whether Σ entails ϕ (written $\Sigma \models \phi$) for a given finite set of formulae $\Sigma \cup \{\phi\}$ from $\mathcal{L}$. The *validity problem* for $\mathcal{L}$ is to decide whether a given formula $\phi \in \mathcal{L}$ is satisfied by all Kripke models and teams. The *satisfiability problem* for $\mathcal{L}$ is to decide whether a given formula $\phi \in \mathcal{L}$ is satisfied by some Kripke model and a non-empty team[1].

The following flatness property holds for all modal logic formulae. Notice that by $\models_{\mathsf{ML}}$ we refer to the usual satisfaction relation of modal logic.

Proposition 2.2 (Flatness [27])**.** *Let ϕ be a formula in ML, let $\mathcal{M} = (W, R, \pi)$ be a Kripke model over $V \supseteq \mathsf{Var}(\phi)$, and let $T \subseteq W$ be a team. Then:*

$$\mathcal{M}, T \models \phi \quad \Leftrightarrow \quad \forall w \in T : \mathcal{M}, w \models_{\mathsf{ML}} \phi.$$

Team semantics gives rise to different extensions of modal logic capable of expressing various dependency notions. In this article we consider dependence atoms that express functional dependence between propositions. To facilitate their associated semantic definitions, we first define for each world w of a Kripke model $\mathcal{M}$ a truth function $w_{\mathcal{M}}$ from ML formulae into $\{0, 1\}$ as follows:

$$w_{\mathcal{M}}(\phi) = \begin{cases} 1 & \text{if } \mathcal{M}, \{w\} \models \phi,\\ 0 & \text{otherwise.} \end{cases}$$

1) Dependence. *Modal dependence logic* (MDL) is obtained by extending ML with *dependence atoms*

$$\mathrm{dep}(\overline{p}, q) \tag{2.2}$$

[1] The empty team satisfies all formulae trivially.

where $\overline{p}$ is a sequence of propositional variables and q is a single propositional variable. Furthermore, we consider *extended dependence atoms* of the form

$$\mathrm{dep}\big(\overline{\phi}, \psi\big) \tag{2.3}$$

where $\overline{\phi}$ is a sequence of ML formulae and ψ is a single ML formula. The extension of ML with formulae of the form (2.3) is called *extended modal dependence logic* (EMDL). Formulae of the form (2.2) and (2.3) indicate that the (truth) value of the formula on the right-hand side is functionally determined by the (truth) values of the formulae listed on the left-hand side. The satisfaction relation for both (2.2) and (2.3) is defined accordingly as follows:

$$\mathcal{M}, T \models \mathrm{dep}\big(\overline{\phi}, \psi\big) \quad :\Leftrightarrow \quad \forall w, w' \in T : w_{\mathcal{M}}(\overline{\phi}) = w'_{\mathcal{M}}(\overline{\phi}) \text{ implies } w_{\mathcal{M}}(\psi) = w'_{\mathcal{M}}(\psi).$$

Here and below we use $w_{\mathcal{M}}(\overline{\theta})$ as a shorthand for $(w_{\mathcal{M}}(\theta_1), \ldots, w_{\mathcal{M}}(\theta_n))$ if $\overline{\theta}$ is a sequence of formulae $(\theta_1, \ldots, \theta_n)$.

We also examine so-called *intuituionistic disjunction* $\varovee$ defined as follows:

$$\mathcal{M}, T \models \phi_1 \varovee \phi_2 \quad :\Leftrightarrow \quad \mathcal{M}, T \models \phi_1 \text{ or } \mathcal{M}, T \models \phi_2. \tag{2.4}$$

We denote the extension of ML with intuitionistic disjunction $\varovee$ by $\mathsf{ML}(\varovee)$. Notice that the logics MDL and EMDL are expressively equivalent to $\mathsf{ML}(\varovee)$ but exponentially more succinct as the translation of (2.2) to $\mathsf{ML}(\varovee)$ involves a necessarily exponential blow-up [13]. These logics satisfy the following downward closure property which will be used in the upper bound result.

Proposition 2.3 (Downward Closure [3, 29, 31])**.** *Let ϕ be a formula in MDL, EMDL, or $\mathsf{ML}(\varovee)$, let $\mathcal{M} = (W, R, \pi)$ be a Kripke model over $V \supseteq \mathsf{Var}(\phi)$, and let $T \subseteq W$ be a team. Then:*

$$T' \subseteq T \text{ and } \mathcal{M}, T \models \phi \quad \Rightarrow \quad \mathcal{M}, T' \models \phi.$$

2) Independence. *Modal independence logic* (MLInd) extends ML with *independence atoms*

$$\overline{q} \perp_{\overline{p}} \overline{r} \tag{2.5}$$

where $\overline{p}, \overline{q}, \overline{r}$ are sequences of propositional variables. Intuitively, (2.5) expresses that the values of $\overline{q}$ and $\overline{r}$ are independent of one another, given any value of $\overline{r}$. The associated satisfaction relation is defined as follows:

$$\begin{aligned}\mathcal{M}, T \models \overline{q} \perp_{\overline{p}} \overline{r} \quad :\Leftrightarrow \quad & \forall w, w' \in T : w_{\mathcal{M}}(\overline{p}) = w'_{\mathcal{M}}(\overline{p}) \text{ implies} \\ & \exists w'' \in T : w_{\mathcal{M}}(\overline{pq}) = w''_{\mathcal{M}}(\overline{pq}) \text{ and} \\ & w'_{\mathcal{M}}(\overline{r}) = w''_{\mathcal{M}}(\overline{r}).\end{aligned}$$

The definition expresses that, fixing any values for $\overline{p}$, the values for $\overline{qr}$ form a cartesian product defined in terms of the values for $\overline{q}$ and $\overline{r}$. Furthermore, notice that MLInd subsumes MDL since (2.2) can be expressed by $q \perp_{\overline{p}} q$.

3) Inclusion. *Modal inclusion logic* (MLInc) extends ML with *inclusion atoms*

$$\overline{p} \subseteq \overline{q} \tag{2.6}$$

where $\overline{p}$ and $\overline{q}$ are sequences of propositional variables of the same length. This atom indicates that the values of $\overline{q}$ subsume all the values of $\overline{p}$. The satisfaction relation for (2.6) is defined as follows:

$$\mathcal{M}, T \models p \subseteq \overline{q} \quad :\Leftrightarrow \quad \forall w \in T \exists w' : w_{\mathcal{M}}(\overline{p}) = w'_{\mathcal{M}}(\overline{q}).$$

For the sake of our proof arguments, we also extend modal logic with predicates. The syntax of *relational modal logic* (RML) is given by the grammar:

$$\phi ::= p \mid \neg\phi \mid (\phi \wedge \phi) \mid \Box\phi \mid S(\phi_1,\dots,\phi_n). \tag{2.7}$$

The formulae of RML are evaluated over a world $w \in W$ in a *relational Kripke model* $\mathcal{M} = (W, R, \pi, S_1^{\mathcal{M}}, \dots, S_n^{\mathcal{M}})$ where each $S_i^{\mathcal{M}}$ is a set of binary sequences of length $\#S_i$, that is, the arity of the relation symbol S_i. We denote by $\mathcal{M}, w \models_{\mathsf{RML}} \phi$ the satisfaction relation obtained by extending the standard Kripke semantics of modal logic as follows:

$$\mathcal{M}, w \models_{\mathsf{RML}} S(\phi_1,\dots,\phi_n) :\Leftrightarrow (w_{\mathcal{M}}(\phi_1),\dots,w_{\mathcal{M}}(\phi_n)) \in S^{\mathcal{M}}.$$

In particular, this means that negation is treated classically:

$$\mathcal{M}, w \models_{\mathsf{RML}} \neg\phi :\Leftrightarrow \mathcal{M}, w \not\models_{\mathsf{RML}} \phi.$$

We also employ the usual shorthands $\phi \vee \psi := \neg(\neg\phi \wedge \neg\psi)$ and $\Diamond\phi := \neg\Box\neg\phi$ in RML.

We can now proceed to the upper bound result which states that the entailment problem for EMDL is in $\textbf{co-NEXPTIME}^{\textbf{NP}}$.

3. Upper Bound for EMDL Entailment

In this section we show that EMDL entailment is in $\textbf{co-NEXPTIME}^{\textbf{NP}}$. The idea is to represent dependence atoms using witnessing functions guessed universally on the left-hand side and existentially on the right-hand side of an entailment problem $\{\phi_1,\dots,\phi_{n-1}\} \models \phi_n$. This reduces the problem to validity of an RML formula of the form $\neg\phi_1^* \vee \dots \vee \neg\phi_{n-1}^* \vee \phi_n^*$ where ϕ_i^* is obtained by replacing in ϕ_i all dependence atoms with relational atoms whose interpretations are bound the guessed Boolean functions. We then extend an Algorithm by Ladner that shows a **PSPACE** upper bound for the validity problem of modal logic [20]. As a novel algorithmic feature we introduce recursive steps for relational atoms that query to the guessed functions. The $\textbf{co-NEXPTIME}^{\textbf{NP}}$ upper bound then follows by a straightforward running time analysis.

We start by showing how to represent dependence atoms using intuitionistic disjunctions defined over witnessing functions. We use $\phi^\perp$ to denote the NNF formula obtained from $\neg\phi$ by pushing the negation to the atomic level, and $\phi^\top$ to denote ϕ. Let $\overline{\alpha} = (\alpha_1,\dots,\alpha_n)$ be a sequence of ML formulae and let β be a single ML formula. Then we say that a function $f : \{\top,\bot\}^n \to \{\top,\bot\}$ is a *witness* of $d := \mathrm{dep}(\overline{\alpha},\beta)$, giving rise to a witnessing ML formula

$$D(f,d) := \bigvee_{a_1,\dots,a_n \in \{\top,\bot\}} \alpha_1^{a_1} \wedge \dots \alpha_n^{a_n} \wedge \beta^{f(a_1,\dots,a_n)}. \tag{3.1}$$

The equivalence

$$d \equiv \bigovee_{f:\,\{\top,\bot\}^n \to \{\top,\bot\}} D(f,d) \tag{3.2}$$

has been noticed in the contexts of MDL and EMDL respectively in [29, 3].

To avoid the exponential blow-up involved in both (3.1) and (3.2), we instead relate to RML by utilizing the following equivalence:

$$(W,R,\pi), w \models_{\mathsf{ML}} D(f,d) \Leftrightarrow (W,R,\pi,S^{\mathcal{M}}), w \models_{\mathsf{RML}} S(\overline{\alpha}\beta), \tag{3.3}$$

where $S^{\mathcal{M}} := \{(a_1,\dots,a_n,b) \in \{0,1\}^{n+1} \mid f(a_1,\dots,a_n) = b\}$. Before proceeding to the proof, we need the following simple proposition, based on [30, 31] where the statement has been proven for empty Σ.

Proposition 3.1. *Let Σ be a set of* ML *formulae, and let $\phi_0, \phi_1 \in \mathsf{ML}(\varovee)$. Then $\Sigma \models \phi_0 \varovee \phi_1$ iff $\Sigma \models \phi_0$ or $\Sigma \models \phi_1$.*

Proof. It suffices to show the only-if direction. Let $\mathcal{M}_0, T_0$ and $\mathcal{M}_1, T_1$ be counterexamples to $\Sigma \models \phi_0$ and $\Sigma \models \phi_1$, respectively. W.l.o.g. we may assume that $\mathcal{M}_0$ and $\mathcal{M}_1$ are disjoint. Since the truth value of an $\mathsf{ML}(\varovee)$ formula is preserved under taking disjoint unions of Kripke models (see Theorem 6.1.9 in [31], also Corollary 5.7 in [30]) we note that $\mathcal{M}, T_0$ and $\mathcal{M}, T_1$ are also counterexamples to $\Sigma \models \phi_0$ and $\Sigma \models \phi_1$, respectively. Let $T := T_0 \cup T_1$. By the flatness property of ML (Proposition 2.2) $\mathcal{M}, T \models \Sigma$, and by the downward closure property of $\mathsf{ML}(\varovee)$ (Proposition 2.3) $\mathcal{M}, T \not\models \phi_i$ for $i = 0, 1$. Consequently, $\Sigma \not\models \phi_0 \varovee \phi_1$ which concludes the proof. □

The proof now proceeds via Lemmata 3.2 and 3.3 of which the former constitutes the basis for our alternating exponential-time algorithm. Note that if ϕ is an EMDL formula with k dependence atom subformulae, listed (possibly with repetitions) in $d_1, \ldots d_k$, then we call $\overline{f} = (f_1, \ldots, f_k)$ a *witness sequence* of ϕ if each f_i is a witness of d_i. Furthermore, we denote by $\phi(\overline{f}/\overline{d})$ the ML formula obtained from ϕ by replacing each d_i with $D(f_i, d_i)$.

Lemma 3.2. *Let $\phi_1, \ldots, \phi_n$ be formulae in* EMDL*. Then $\{\phi_1, \ldots, \phi_{n-1}\} \models \phi_n$ iff for all witness sequences $\overline{f}_1, \ldots, \overline{f}_{n-1}$ of $\phi_1, \ldots, \phi_{n-1}$ there is a witness sequence $\overline{f}$ of ϕ_n such that*

$$\{\phi_1(\overline{f}_1/\overline{d}_1), \ldots, \phi_{n-1}(\overline{f}_{n-1}/\overline{d}_{n-1})\} \models \phi_n(\overline{f}_n/\overline{d}_n).$$

Proof. Assume first that ϕ is an arbitrary formula in EMDL, and let $d = \mathrm{dep}(\overline{\alpha}, \beta)$ be a subformula of ϕ. It is straightforward to show that ϕ is equivalent to

$$\bigvarovee_{f\colon \{\top,\bot\}^{|\overline{\alpha}|} \to \{\top,\bot\}} \phi(D(f, d)/d).$$

This follows from the fact that all $\vee, \wedge, \Diamond, \Box$ distribute over $\varovee$, note especially that $(\phi \varovee \psi) \vee \theta$ is equivalent to $(\phi \vee \theta) \varovee (\psi \vee \theta)$.

Iterating these substitutions we obtain that $\{\phi_1, \ldots, \phi_{n-1}\} \models \phi_n$ iff

$$\{\bigvarovee_{\overline{f}_1} \phi_i(\overline{f}_1/\overline{d}_1), \ldots, \bigvarovee_{\overline{f}_{n-1}} \phi_i(\overline{f}_{n-1}/\overline{d}_{n-1})\} \models \bigvarovee_{\overline{f}_n} \phi_i(\overline{f}_n/\overline{d}_n), \tag{3.4}$$

where $\overline{f}_i$ ranges over the witness sequences of ϕ_i. Then (3.4) holds iff for all $\overline{f}_1, \ldots, \overline{f}_{n-1}$,

$$\{\phi_1(\overline{f}_1/\overline{d}_1), \ldots, \phi_{n-1}(\overline{f}_{n-1}/\overline{d}_{n-1})\} \models \bigvarovee_{\overline{f}_n} \phi(\overline{f}_n/\overline{d}_n). \tag{3.5}$$

Notice that each formula $\phi_i(\overline{f}_i/\overline{d}_i)$ belongs to ML. Hence, by Proposition 3.1 we conclude that (3.5) holds iff for all $\overline{f}_1, \ldots, \overline{f}_{n-1}$ there is $\overline{f}_n$ such that

$$\{\phi_1(\overline{f}_1/\overline{d}_1), \ldots, \phi_{n-1}(\overline{f}_{n-1}/\overline{d}_{n-1})\} \models \phi(\overline{f}_n/\overline{d}_n). \tag{3.6}$$

□

The next proof step is to reduce an entailment problem of the form (3.6) to a validity problem of an RML formula over relational Kripke models whose interpretations agree with the guessed functions. For the latter problem we then apply Algorithm 1 whose lines 1-14 and 19-26 constitute an algorithm of Ladner that shows the **PSPACE** upper bound for modal logic satisfiability [20]. Lines 15-18 consider those cases where the subformula is relational. Lemma 3.3 now shows that, given an oracle A, this extended algorithm yields a $\mathbf{PSPACE}^A$ decision procedure for satisfiability of RML formulae over relational Kripke models whose predicates agree with A. For an oracle set A of words from $\{0,1,\#\}^*$ and k-ary relation symbol R_i, we define $R_i^A := \{(b_1,\dots,b_k) \in \{0,1\}^k \mid \mathrm{bin}(i)^\frown\#b_1\dots b_k \in A\}$. Note that $a^\frown b$ denotes the concatenation of two strings a and b.

Input : $(\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D})$ where $\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D} \subseteq \mathsf{RML}$
Output: $\mathtt{Sat}(\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D})$

if $\mathcal{A}\cup\mathcal{B} \not\subseteq \mathsf{Prop}$ **then**
  choose $\phi \in ((\mathcal{A}\cup\mathcal{B}) \setminus \mathsf{Prop})$;
  if $\phi = \neg\psi$ *and* $\phi \in \mathcal{A}$ **then**
    return $\mathtt{Sat}(\mathcal{A}\setminus\{\phi\},\mathcal{B}\cup\{\psi\},\mathcal{C},\mathcal{D})$;
  else if $\phi = \neg\psi$ *and* $\phi \in \mathcal{B}$ **then**
    return $\mathtt{Sat}(\mathcal{A}\cup\{\psi\},\mathcal{B}\setminus\{\phi\},\mathcal{C},\mathcal{D})$;
  else if $\phi = \psi\wedge\theta$ *and* $\phi \in \mathcal{A}$ **then**
    return $\mathtt{Sat}((\mathcal{A}\cup\{\psi,\theta\})\setminus\{\phi\},\mathcal{B},\mathcal{C},\mathcal{D})$;
  else if $\phi = \psi\wedge\theta$ *and* $\phi \in \mathcal{B}$ **then**
    return $\mathtt{Sat}(\mathcal{A},(\mathcal{B}\cup\{\psi\})\setminus\{\phi\},\mathcal{C},\mathcal{D}) \vee \mathtt{Sat}(\mathcal{A},(\mathcal{B}\cup\{\theta\})\setminus\{\phi\},\mathcal{C},\mathcal{D})$;
  else if $\phi = \Box\psi$ *and* $\phi \in \mathcal{A}$ **then**
    return $\mathtt{Sat}(\mathcal{A}\setminus\{\phi\},\mathcal{B},\mathcal{C}\cup\{\psi\},\mathcal{D})$;
  else if $\phi = \Box\psi$ *and* $\phi \in \mathcal{B}$ **then**
    return $\mathtt{Sat}(\mathcal{A},\mathcal{B}\setminus\{\phi\},\mathcal{C},\mathcal{D}\cup\{\psi\})$;
  else if $\phi = S_i(\psi_1,\dots,\psi_k)$ *and* $\phi \in \mathcal{A}$ **then**
    return $\bigvee_{(b_1,\dots,b_k)\in S_i^A}\mathtt{Sat}((\mathcal{A}\cup\{\psi_j : b_j = 1\})\setminus\{\phi\},\mathcal{B}\cup\{\psi_j : b_j = 0\},\mathcal{C},\mathcal{D})$;
  else if $\phi = S_i(\psi_1,\dots,\psi_k)$ *and* $\phi \in \mathcal{B}$ **then**
    return $\bigvee_{(b_1,\dots,b_k)\notin S_i^A}\mathtt{Sat}(\mathcal{A}\cup\{\psi_j : b_j = 1\},(\mathcal{B}\cup\{\psi_j : b_j = 0\})\setminus\{\phi\},\mathcal{C},\mathcal{D})$;
  end
else if $(\mathcal{A}\cup\mathcal{B}) \subseteq \mathsf{Prop}$ **then**
  if $\mathcal{A}\cap\mathcal{B} \neq \emptyset$ **then**
    return false;
  else if $\mathcal{A}\cap\mathcal{B} = \emptyset$ *and* $\mathcal{C}\cap\mathcal{D} \neq \emptyset$ **then**
    return $\bigwedge_{D\in\mathcal{D}}\mathtt{Sat}(\mathcal{C},\{D\},\emptyset,\emptyset)$;
  else if $\mathcal{A}\cap\mathcal{B} = \emptyset$ *and* $\mathcal{C}\cap\mathcal{D} = \emptyset$ **then**
    return true;
  end
end

Algorithm 1: PSPACE^A algorithm for deciding validity in RML. Notice that queries to S_i^A range over $(b_1,\dots,b_k) \in \{0,1\}^k$.

Lemma 3.3. *Given an* RML *formula ϕ over a vocabulary $\{S_1,\dots,S_n\}$ and an oracle set of words A from $\{0,1,\#\}^*$, Algorithm 1 decides in* **PSPACE^A** *whether there is a relational Kripke structure $\mathcal{M} = (W,R,\pi,S_1^A,\dots,S_n^A)$ and a world $w \in W$ such that $\mathcal{M},w \models_{\mathsf{RML}} \phi$.*

Proof. We leave it to the reader to show (by a straightforward structural induction) that, given an input $(\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D})$ where $\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D} \subseteq \mathsf{RML}$, Algorithm 1 returns $\mathtt{Sat}(\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D})$ true iff there is a relational Kripke model $\mathcal{M} = (W,R,\pi,S_1^A,\dots,S_n^A)$ and a world w such that

- $\mathcal{M}, w \models_{\mathsf{RML}} \phi$ if $\phi \in \mathcal{A}$;
- $\mathcal{M}, w \not\models_{\mathsf{RML}} \phi$ if $\phi \in \mathcal{B}$;
- $\mathcal{M}, w \models_{\mathsf{RML}} \Box\phi$ if $\phi \in \mathcal{C}$; and
- $\mathcal{M}, w \not\models_{\mathsf{RML}} \Box\phi$ if $\phi \in \mathcal{D}$.

Hence, $\texttt{Sat}(\{\psi\}, \emptyset, \emptyset, \emptyset)$ returns true iff ψ is satisfiable by $\mathcal{M}, w$ with relations $S_i^{\mathcal{M}}$ obtained from the oracle. Note that the selection of subformulae ϕ from $\mathcal{A} \cup \mathcal{B}$ can be made deterministically by defining an ordering for the subformulae. Furthermore, we note, following [20], that this algorithm runs in $\mathbf{PSPACE}^A$ as it employs $\mathcal{O}(n)$ recursive steps that each take space $\mathcal{O}(n)$.

Size of each recursive step. At each recursive step $\texttt{Sat}(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$ is stored onto the work tape by listing all subformulae in $\mathcal{A} \cup \mathcal{B} \cup \mathcal{C} \cup \mathcal{D}$ in such a way that each subformula ψ has its major connective (or relation/proposition symbol for atomic formulae) replaced with a special marker which also points to the position of the subset where ψ is located. In addition we store at each disjunctive/conjunctive recursive step the subformula or binary number that points to the disjunct/conjunct under consideration. Each recursive step takes now space $\mathcal{O}(n)$.

Number of recursive steps. Given a set of formulae $\mathcal{A}$, we write $|\mathcal{A}|$ for $\Sigma_{\phi \in \mathcal{A}}|\phi|$ where $|\phi|$ is the length of ϕ. We show by induction on $n = |\mathcal{A} \cup \mathcal{B} \cup \mathcal{C} \cup \mathcal{D}|$ that $\texttt{Sat}(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$ has $2n + 1$ levels of recursion. Assume that the claim holds for all natural numbers less than n, and assume that $\texttt{Sat}(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$ calls $\texttt{Sat}(\mathcal{A}', \mathcal{B}', \mathcal{C}', \mathcal{D}')$. Then $|\mathcal{A}' \cup \mathcal{B}' \cup \mathcal{C}' \cup \mathcal{D}'| < n$ except for the case where $\mathcal{A} \cap \mathcal{B}$ is empty and $\mathcal{C} \cap \mathcal{D}$ is not. In that case it takes at most one extra recursive step to reduce to a length $< n$. Hence, by the induction assumption the claim follows. We conclude that the space requirement for Algorithm 1 on $\texttt{Sat}(\{\phi\}, \emptyset, \emptyset, \emptyset)$ is $\mathcal{O}(n^2)$. □

Using Lemmata 3.2 and 3.3 we can now show the **co-NEXPTIME**$^{\mathbf{NP}}$ upper bound. In the proof we utilize the following connection between alternating Turing machines and the exponential time hierarchy at the level **co-NEXPTIME**$^{\mathbf{NP}} = \Pi_2^{\mathrm{EXP}}$.

Theorem 3.4 [1, 23]**.** Σ_k^{EXP} *(or* Π_k^{EXP}*) is the class of problems recognizable in exponential time by an alternating Turing machine which starts in an existential (universal) state and alternates at most* $k - 1$ *many times.*

Theorem 3.5. *The entailment problem for* EMDL *is in* **co-NEXPTIME**$^{\mathbf{NP}}$*.*

Proof. Assuming an input $\phi_1, \ldots, \phi_n$ of EMDL formulae, we show how to decide in Π_2^{EXP} whether $\{\phi_1, \ldots, \phi_{n-1}\} \models \phi_n$. By Theorem 3.4 it suffices to construct an alternating exponential-time algorithm that switches once from an universal to an existential state. By Lemma 3.2, $\{\phi_1, \ldots, \phi_{n-1}\} \models \phi_n$ iff for all $\overline{f}_1, \ldots, \overline{f}_{n-1}$ there is $\overline{f}_n$ such that

$$\{\phi_1(\overline{f}_1/\overline{d}_1), \ldots, \phi_{n-1}(\overline{f}_{n-1}/\overline{d}_{n-1})\} \models \phi(\overline{f}_n/\overline{d}_n). \tag{3.7}$$

Recall from the proof of Lemma 3.2 that all the formulae in (3.7) belong to ML. Hence by the flatness property (Proposition 2.2) $\models$ is interchangeable with $\models_{\mathsf{ML}}$ in (3.7). It follows that (3.7) holds iff

$$\phi := \phi_1(\overline{f}_1/\overline{d}_1) \wedge \ldots \wedge \phi_{n-1}(\overline{f}_{n-1}/\overline{d}_{n-1}) \wedge \neg\phi(\overline{f}_n/\overline{d}_n) \tag{3.8}$$

is not satisfiable with respect to the standard Kripke semantics of modal logic. By the equivalence in (3.3) we notice that (3.8) is not satisfiable with respect to $\models_{\mathsf{ML}}$ iff ϕ^* is not satisfiable over the selected functions with respect to $\models_{\mathsf{RML}}$, where ϕ^* is obtained from ϕ by replacing each $D(f, \overline{\alpha}, \beta)$ of the form (3.1) with the predicate $f(\overline{\alpha}) = \beta$. The crucial point here is that ϕ^* is only of length $\mathcal{O}(n \log n)$ in the input.

The algorithm now proceeds as follows. The first step is to universally guess functions listed in $\overline{f}_1 \ldots \overline{f}_{n-1}$, followed by an existential guess over functions listed in $\overline{f}_n$. The next step is to transform the input to the described RML formula ϕ^*. The last step is to run Algorithm 1 on $\mathtt{Sat}(\phi^*, \emptyset, \emptyset, \emptyset)$ replacing queries to the oracle with investigations on the guessed functions, and return true iff the algorithm returns false. By Lemma 3.3, Algorithm 1 returns false iff (3.7) holds over the selected functions. Hence, by Lemma 3.2 we conclude that the overall algorithm returns true iff $\{\phi_1, \ldots, \phi_{n-1}\} \models \phi_n$.

Note that this procedure involves polynomially many guesses, each of at most exponential length. Also, Algorithm 1 runs in exponential time and thus each of its implementations has at most exponentially many oracle queries. Hence, we conclude that the given procedure decides EMDL-entailment in **co-NEXPTIME$^{\mathbf{NP}}$**. □

Notice that the decision procedure for $\models \phi$ does not involve any universal guessing. Therefore, we obtain immediately a **NEXPTIME** upper bound for the validity problem of EMDL.

Corollary 3.6. *The validity problem for* EMDL *is in* **NEXPTIME**.

4. Propositional Dependence Logics

Before showing that **co-NEXPTIME$^{\mathbf{NP}}$** is also the lower bound for the entailment problem of the propositional fragment of MDL, we need to formally define this fragment. We also need to present other propositional variants that will be examined later in this article. Beside extending our investigations to propositional independence and inclusion logics, we will also study the extensions of these logics with additional universal and existential quantifiers. This section is divided into two subsections. Sect 4.1 introduces different variants of propositional dependence logic. Sect 4.2 shows that decision problems for quantified propositional dependence logics can be reduced to the same problems over modal dependence logics.

4.1. Introduction to Propositional Dependence Logics.

The syntax of *propositional logic* (PL) is generated by the following grammar:

$$\phi ::= p \mid \neg p \mid (\phi \wedge \phi) \mid (\phi \vee \phi) \tag{4.1}$$

The syntaxes of propositional dependence, independence, and inclusion logics (PDL, PLInd, PLInc, resp.) are obtained by extending the syntax of PL with dependence atoms of the form (2.2), independence atoms of the form (2.5), and inclusion atoms of the form (2.6), respectively. Furthermore, the syntax of PL($\ovee$) extends (4.1) with the grammar rule $\phi ::= \phi \ovee \phi$.

The formulae of these logics are evaluated against propositional teams. Let V be a set of variables. We say that a function $s\colon V \to \{0,1\}$ is a *(propositional) assignment* over V, and a *(propositional) team* X over V is a set of propositional assignments over V. A team X over V induces a Kripke model $\mathcal{M}_X = (T_X, \emptyset, \pi)$ where $T_X = \{w_s \mid s \in X\}$ and $w_s \in \pi(p) \Leftrightarrow s(p) = 1$ for $s \in X$ and $p \in V$. The team semantics for propositional formulae is now defined as follows:

$$X \models \phi :\Leftrightarrow \mathcal{M}_X, T_X \models \phi,$$

where $\mathcal{M}_X, T_X \models \phi$ refers to the team semantics of modal formulae (see Sect. 2). If ϕ^* is a formula obtained from ϕ by replacing all propositional atoms p (except those inside a dependence atom) with predicates $A(p)$, then we can alternatively describe that $X \models \phi$ iff $\mathcal{M} = (\{0,1\}, A := \{1\})$ and X satisfy ϕ^* under the lax team semantics of first-order dependence logics [5].

Quantified propositional logic (QPL) is obtained by extending that of PL with universal and existential quantification over propositional variables. Their semantics is given in terms of so-called duplication and supplementation teams. Let p be a propositional variable and s an assignment over V. We denote by $s(a/p)$ the assignment over $V \cup \{p\}$ that agrees with s everywhere, except that it maps p to a. Universal quantification of a propositional variable p is defined in terms of *duplication teams* $X[\{0,1\}/p] := \{s(a/p) \mid s \in X, a \in \{0,1\}\}$ that extend teams X with all possible valuations for p. Existential quantification is defined in terms of *supplementation teams* $X[F/p] := \{s(a/p) \mid s \in X, a \in F(s)\}$ where F is a mapping from X into $\{\{0\},\{1\},\{0,1\}\}$. The supplementation team $X[F/p]$ extends each assignment of X with a non-empty set of values for p. The satisfaction relations $X \models \exists p\phi$ and $X \models \forall p\phi$ are now given as follows:

$$\begin{aligned} X \models \exists p\phi \quad &:\Leftrightarrow \quad \exists F \in {}^X\{\{0\},\{1\},\{0,1\}\} : X[F/p] \models \phi, \\ X \models \forall p\phi \quad &:\Leftrightarrow \quad X[\{0,1\}/p] \models \phi. \end{aligned}$$

We denote by QPDL the extension of PDL with quantifiers and define QPLInd, QPLInc, and $\mathsf{QPL}(\mathbin{\rotatebox[origin=c]{180}{$\otimes$}})$ analogously. Observe that the flatness and downward closure properties of modal formulae (Propositions 2.2 and 2.3, resp.) apply now analogously to propositional formulae. We also have that QPLInc is closed under taking unions of teams. Note that $\models_{\mathsf{PL}}$ here refers to the standard semantics of propositional logic. Furthermore, $\mathsf{Var}(\phi)$ refers to the set of variables appearing in ϕ, and $\mathsf{Fr}(\phi)$ to the set of free variables appearing in a formula ϕ, both defined in the standard way. Sometimes we write $\phi(p_1,\dots,p_n)$ instead of ϕ to emphasize that $\mathsf{Fr}(\phi) = \{p_1,\dots,p_n\}$.

Proposition 4.1 (Flatness [28])**.** *Let ϕ be a formula in QPL, and let X be a team over $V \supseteq \mathsf{Fr}(\phi)$. Then:*

$$X \models \phi \quad \Leftrightarrow \quad \forall s \in X : s \models_{\mathsf{PL}} \phi.$$

Proposition 4.2 (Downward Closure [28])**.** *Let ϕ be a formula in QPDL or $\mathsf{QPL}(\mathbin{\rotatebox[origin=c]{180}{$\otimes$}})$, and let X be a team over a set $V \supseteq \mathsf{Fr}(\phi)$ of propositional variables. Then:*

$$Y \subseteq X \text{ and } X \models \phi \quad \Rightarrow \quad Y \models \phi.$$

Proposition 4.3 (Union Closure [5])**.** *Let ϕ be a formula in QPLInc, and let X and Y be teams over a set $V \supseteq \mathsf{Fr}(\phi)$ of propositional variables. Then:*

$$X \models \phi \text{ and } Y \models \phi \quad \Rightarrow \quad X \cup Y \models \phi.$$

We denote the *restriction* of an assignment s to variables in V by $s \upharpoonright V$, and define the restriction of a team X to V, written $X \upharpoonright V$, as $\{s \upharpoonright V \mid s \in X\}$. We conclude this section by noting that, similar to the first-order case, quantified propositional dependence logic satisfies the following locality property.

Proposition 4.4 (Locality [28])**.** *Let ϕ be a formula in $\mathcal{L}$ where $\mathcal{L} \in \{\mathsf{QPDL}, \mathsf{QPLInd}, \mathsf{QPLInc}, \mathsf{QPL}(\mathbin{\rotatebox[origin=c]{180}{$\otimes$}})\}$, let X be a team over $V \supseteq \mathsf{Fr}(\phi)$, and let V' be such that $\mathsf{Fr}(\phi) \subseteq V' \subseteq V$. Then:*

$$X \models \phi \quad \Leftrightarrow \quad X \upharpoonright V' \models \phi.$$

4.2. **Reductions from Quantified Propositional to Modal Logics.** In this section we show how to generate simple polynomial-time reductions from quantified propositional dependence logics to modal dependence logics with respect to their entailment and validity problem. First we present Lemma 4.5 which is a direct consequence of [6, Lemma 14] that presents prenex normal form translations in the first-order dependence logic setting over structures with universe size at least 2. The result follows by the obvious first-order interpretation of quantified propositional formulae: satisfaction of a quantified propositional formula ϕ by a binary team X can be replaced with satisfaction of ϕ^* by $\mathcal{M} := (\{0,1\}, P^{\mathcal{M}} := \{1\})$ and X, where ϕ^* is a formula obtained from ϕ by replacing atomic propositional formulae p and $\neg p$ respectively with $P(p)$ and $\neg P(p)$.

Lemma 4.5 [6]**.** *Any formula ϕ in L, where $\mathsf{L} \in \{\mathsf{QPDL}, \mathsf{QPLInc}, \mathsf{QPLInd}\}$, is logically equivalent to a polynomial size formula $Q_1 p_1 \dots Q_n p_n \psi$ in L where ψ is quantifier-free and $Q_i \in \{\exists, \forall\}$ for $i = 1, \dots, n$.*

Next we show how to describe in modal terms a quantifier block $Q_1 p_1 \dots Q_n p_n$. Using the standard method in modal logic we construct a formula $\mathsf{tree}(V, p_1, \dots, p_n)$ that enforces the complete binary assignment tree over $p_1, \dots, p_n$ for a team over V where $\{p_1, \dots, p_n\}$ and V are disjoint [20]. The formulation of $\mathsf{tree}(V, p_1, \dots, p_n)$ follows the presentation in [8]. We define $\mathsf{store}_n(p) := (p \wedge \Box^n p) \vee (\neg p \wedge \Box^n \neg p)$, where $\Box^n$ is a shorthand for $\overbrace{\Box \cdots \Box}^{n \text{ many}}$, to impose the existing values for p to successors in the tree. We also define $\mathsf{branch}_n(p) := \Diamond p \wedge \Diamond \neg p \wedge \Box \mathsf{store}_n(p)$ to indicate that there are ≥ 2 successor states which disagree on the variable p and that all successor states preserve their values up to branches of length n. Then we let

$$\mathsf{tree}(V, p_1, \dots, p_n) := \bigwedge_{q \in V} \bigwedge_{i=1}^{n} \mathsf{store}_i(q) \wedge \bigwedge_{i=0}^{n-1} \Box^i \mathsf{branch}_{n-(i+1)}(p_{i+1}).$$

Note that $\mathsf{tree}(V, p_1, \dots, p_n)$ is an ML formula and hence has the flatness property by Proposition 2.2.

Theorem 4.6. *The satisfiability, validity, and entailment problems for QPDL, QPLInd, or QPLInc are polynomial-time reducible to the satisfiability, validity, and entailment problems for MDL, MLInd, or MLInc, respectively.*

Proof. Consider first the entailment problem, and assume that $\Sigma \cup \{\phi\}$ is a finite set of formulae in either QPDL, QPLInd, or QPLInc. By Lemma 4.5 each formula in $\theta \in \Sigma \cup \{\phi\}$ can be transformed in polynomial time to the form $\theta_0 = Q_1 p_1 \dots Q_n p_n \psi$ where ψ is quantifier-free. We may assume that the variable sequences $p_1, \dots, p_n$ corresponding to these quantifier blocks are initial segments of a shared infinite list $p_1, p_2, p_3, \dots$ of variables. Assume m is the maximal length of the quantifier blocks that appear in any of the translations, and let V be the set of variables that appear free in some of them. W.l.o.g. we may assume that $\{p_1, \dots, p_m\}$ and V are disjoint. We let θ_1 be obtained from θ_0 by replacing quantifiers $\exists$ and $\forall$ respectively with $\Diamond$ and $\Box$. It follows that $\Sigma \models \phi$ iff $\{\theta_1 \mid \theta \in \Sigma\} \cup \{\mathsf{tree}(V, p_1, \dots, p_n)\} \models \phi_1$.[2] For the validity problem, we observe that $\models \phi$ iff $\models \mathsf{tree}(V, p_1, \dots, p_n)^{\perp} \vee (\mathsf{tree}(V, p_1, \dots, p_n) \wedge \phi_1)$.

[2]Notice that the direction from left to right does not hold under the so-called strict team semantics where $\exists$ and $\Diamond$ range over individuals. These two logics are not downwards closed and the modal translation does not prevent the complete binary tree of having two distinct roots that agree on the variables in V. The same proviso applies to the translations of validity and satisfiability.

Intuitively the formula on the right states that excluding those worlds which do not encode a complete binary tree that preserves V, the resulting worlds form a team that satisfies the encoding ϕ_1 of ϕ. Furthermore, for the satisfiability problem we have that ϕ is satisfiable iff $\mathsf{tree}(V, p_1, \ldots, p_n) \wedge \phi_1$ is. Since the reductions are clearly polynomial, this concludes the proof. □

5. Lower Bound for PDL Entailment

In this section we prove that the entailment problem for PDL is **co-NEXPTIME**$^{\mathbf{NP}}$-hard. This result is obtained by reducing from a variant of the dependency quantified Boolean formula problem, which is, a complete problem for **NEXPTIME** extending the standard quantified Boolean formula problem by constraints. The variant we present was introduced in [8], and it gives complete problems for different levels of the exponential hierarchy. The following presentation follows mainly [8], deviating only in notation.

Definition 5.1. A *dependency quantified Boolean formula* (DQBF) is a pair $(\phi, \overline{C})$ where $\overline{C} = (\overline{c}_1, \ldots, \overline{c}_m)$ is a list of sequences of propositional variables from $\{p_1, \ldots, p_n\}$, and ϕ is a formula of the form

$$\exists f_1 \ldots \exists f_m \forall p_1 \ldots \forall p_n \theta$$

where $p_1, \ldots, p_n$ are pairwise disjoint, $f_1, \ldots, f_m$ are pairwise disjoint functional variables, and θ is a quantifier-free propositional formula in which only the quantified variables p_i and functional variables f_i with arguments $\overline{c}_i$ may appear. The DQBF instance $(\phi, \overline{C})$ is true if there are functions $f_i\colon \{0,1\}^{|\overline{c}_i|} \to \{0,1\}$ such that θ is true for each assignment $s\colon \{p_1, \ldots, p_n\} \to \{0,1\}$.

The true quantified Boolean formula problem, TRUE(DQBF), is now the problem of determining the truth value of a given DQBF instance.

Theorem 5.2 [25]**.** *TRUE(DQBF) is* **NEXPTIME**-*complete.*

We now generalize TRUE(DQBF) by introducing alternation to the quantifier block for functions.[3]

Definition 5.3 [8]**.** A Σ_k-*alternating dependency quantified Boolean formula* (Σ_k-ADQBF) is a pair $(\phi, \mathcal{C})$ where ϕ is an expression of the form

$$\phi := (\exists f_1^1 \ldots \exists f_{j_1}^1)\,(\forall f_1^2 \ldots \forall f_{j_2}^2)\,(\exists f_1^3 \ldots \exists f_{j_3}^3) \quad \ldots \quad (\,Q f_1^k \ldots Q f_{j_k}^k)\,\forall p_1 \ldots \forall p_n\,\theta,$$

where $Q \in \{\exists, \forall\}$, $\mathcal{C} = (\overline{c}_1^1, \ldots, \overline{c}_{j_k}^k)$ is a list of sequences of propositional variables from $\{p_1, \ldots, p_n\}$, and θ is a quantifier-free propositional formula in which only the quantified variables p_i and functional variables f_j^i with arguments $\overline{c}_j^i$ may appear. Analogously, a Π_k-*alternating dependency quantified Boolean formula* (Π_k-ADQBF) is a pair $(\phi, \mathcal{C})$ where ϕ is an expression of the form

$$\phi := (\forall f_1^1 \ldots \forall f_{j_1}^1)\,(\exists f_1^2 \ldots \exists f_{j_2}^2)\,(\forall f_1^3 \ldots \forall f_{j_3}^3) \quad \ldots \quad (\,Q f_1^k \ldots Q f_{j_k}^k)\,\forall p_1 \ldots \forall p_n\,\theta.$$

The sequence $\mathcal{C}$ is called the *constraint* of ϕ.

[3] Apart from notational differences, the following definition is from [8]. Here we use explicitly functional variables instead of proposition variables for quantification of functions. Furthermore, to improve the correspondence between syntax and semantics, the quantifier block $\forall p_1 \ldots \forall p_n$ for propositional variables is here written after the quantifier block for functional variables.

The truth condition of a Σ_k-ADQBF or a Π_k-ADQBF instance is defined by a generalization of that in Definition 5.1 such that each Qf^i_j where $Q \in \{\exists, \forall\}$ is interpreted as existential/universal quantification over functions $f^i_j\colon \{0,1\}^{|\overline{c}^i_j|} \to \{0,1\}$. Let us now denote the associated decision problems by TRUE(Σ_k-ADQBF) and TRUE(Π_k-ADQBF). These problems characterize levels of the exponential hierarchy in the following way.

Theorem 5.4 [8]**.** *Let $k \geq 1$. For odd k the problem* TRUE(Σ_k-ADQBF) *is Σ^{EXP}_k-complete. For even k the problem* TRUE(Π_k-ADQBF) *is Π^{EXP}_k-complete.*

Since TRUE(Π_2-ADQBF) is **co-NEXPTIME$^{\mathbf{NP}}$**-complete, we can show the lower bound via an reduction from it. Notice that regarding the validity problem of PDL, we already have the following lower bound.

Theorem 5.5 [30]**.** *The validity problem for* PDL *is* **NEXPTIME**-*complete, and for* MDL *and* EMDL *it is* **NEXPTIME**-*hard.*

This result was shown by a reduction from the dependency quantified Boolean formula problem (i. e. TRUE(Σ_1-ADQBF)) to the validity problem of PDL. We use essentially the same technique to reduce from TRUE(Π_2-ADQBF) to the entailment problem of PDL.

Theorem 5.6. *The entailment problem for* PDL *is* **co-NEXPTIME$^{\mathbf{NP}}$**-*hard.*

Proof. By Theorem 5.4 it suffices to show a reduction from TRUE(Π_2-ADQBF). Let $(\phi, \mathcal{C})$ be an instance of Π_2-ADQBF in which case ϕ is of the form

$$\forall f_1 \dots \forall f_m \exists f_{m+1} \dots \exists f_{m+m'} \forall p_1 \dots \forall p_n \theta$$

and $\mathcal{C}$ lists tuples $\overline{c}_i \subseteq \{p_1, \dots, p_n\}$, for $i = 1, \dots, m+m'$. Let q_i be a fresh propositional variable for each functional variable f_i. We define $\Sigma := \{\mathrm{dep}(\overline{c}_i, q_i) \mid i = 1, \dots, m\}$ and

$$\psi := \theta' \vee \bigvee_{i=m+1}^{m+m'} \mathrm{dep}(\overline{c}_i, q_i)\,,$$

where θ' is obtained from θ by replacing occurrences of $f_i(\overline{c}_i)$ with q_i. Clearly, Σ and ψ can be constructed from $(\phi, \mathcal{C})$ in polynomial time. It remains to show that $\Sigma \models \psi$ iff ϕ is true.

Assume first that $\Sigma \models \psi$ and let $f_i\colon \{0,1\}^{|\overline{c}_i|} \to \{0,1\}$ be arbitrary for $i = 1, \dots, m$. Construct a team X that consists of all assignments s that map $p_1, \dots, p_n, q_{m+1}, \dots, q_{m+m'}$ into $\{0,1\}$ and $q_1, \dots, q_m$ respectively to $f_1(s(\overline{c}_1)), \dots, f_m(s(\overline{c}_m))$. Since $X \models \Sigma$ we find $Z, Y_1, \dots, Y_{m'} \subseteq X$ such that $Z \cup Y_1 \cup \ldots \cup Y_{m'} = X$, $Z \models \theta'$, and $Y_i \models \mathrm{dep}(\overline{c}_{m+i}, q_{m+i})$ for $i = 1, \dots, m'$. We may assume that each Y_i is a maximal subset satisfying $\mathrm{dep}(\overline{c}_{m+i}, q_{m+i})$, i.e., for all $s \in X \setminus Y_i$, $Y_i \cup \{s\} \not\models \mathrm{dep}(\overline{c}_{m+i}, q_{m+i})$. Since X takes only Boolean values, the complement $X \setminus Y_i$ is a maximal subset satisfying $\mathrm{dep}(\overline{c}_{m+i}, q_{m+i})$, too. By downward closure (Proposition 4.2) we may assume that Z does not intersect any of the subsets $Y_1, \dots, Y_{m'}$. Consequently, Z is a maximal subset that satisfies all $\mathrm{dep}(\overline{c}_{m+i}, q_{m+i})$ for $i = 1, \dots, m'$. It follows that there are functions $f_i\colon \{0,1\}^{|\overline{c}_i|} \to \{0,1\}$, for $i = m+1, \dots, m+m'$, such that

$$Z = \{s\big(f_{m+1}(s(\overline{c}_{m+1}))/q_{m+1}, \dots, f_{m+m'}(s(\overline{c}_{m+m'}))/q_{m+m'}\big) \mid s \in X\}.$$

Notice that Z is maximal with respect to $p_1, \dots, p_n$, i.e., $Z \upharpoonright \{p_1, \dots, p_n\} = {}^{\{p_1,\dots,p_n\}}\{0,1\}$. Hence, by the flatness property (Proposition 4.1), and since $Z \models \theta'$, it follows that θ' holds for all values of $p_1, \dots, p_n$ and for the values of $q_1, \dots, q_{m+m'}$ chosen respectively according to $f_1, \dots, f_{m+m'}$. Therefore, ϕ is true which shows the direction from left to right.

Assume then that ϕ is true, and let X be a team satisfying Σ. Then there are functions $f_i\colon \{0,1\}^{|\overline{c}_i|} \to \{0,1\}$ such that $f(s(\overline{c}_i)) = s(q_i)$ for $s \in X$ and $i = 1, \dots, m$. Since ϕ is true we find functions $f_i\colon \{0,1\}^{|\overline{c}_i|} \to \{0,1\}$, for $i = m+1, \dots, m+m'$, such that for all $s \in X$:

$$s\big(f_{m+1}(s(\overline{c}_{m+1}))/q_{m+1}, \dots, f_{m+m'}(s(\overline{c}_{m+m'}))/q_{m+m'}\big) \models \theta'. \tag{5.1}$$

Again, since X is Boolean, it follows that $Y_i := \{s \in X \mid s(q_i) \neq f(s(\overline{c}_i))\}$ satisfies $\mathrm{dep}(\overline{c}_i, q_i)$ for $i = m+1, \dots, m+m'$. Then it follows by (5.1) and flatness (Proposition 4.1) that $X \setminus (Y_{m+1} \cup \dots \cup Y_{m+m'})$ satisfies θ'. Therefore, $\Sigma \models \psi$ which concludes the direction from right to left. □

6. Validity and Entailment in Modal and Propositional Dependence Logics

We may now draw together the main results of Sections 3 and 5. There it was shown that in terms of the entailment problem **co-NEXPTIME**$^{\mathbf{NP}}$ is both an upper bound for EMDL and an lower bound for PDL. Therefore, we obtain in Theorem 6.1 that for all the logics inbetween it is also the exact complexity bound. Furthermore, Theorem 4.6 implies that we can count QPDL in this set of logics.

Theorem 6.1. *The entailment problem for* EMDL, MDL, QPDL, *and* PDL *is* **co-NEXPTIME**$^{\mathbf{NP}}$*-complete.*

Proof. The upper bound for EMDL and MDL was shown in Theorem 3.5, and by Theorem 4.6 the same upper bound applies to QPDL and PDL. The lower bound for all of the logics comes from Theorem 5.6. □

We also obtain that all the logics inbetween PDL and EMDL are **NEXPTIME**-complete in terms of their validity problem. The proof arises analogously from Corollary 3.6 and Theorem 5.5.

Theorem 6.2. *The validity problem for* EMDL, MDL, QPDL, *and* PDL *is* **NEXPTIME**-*complete.*

Recall that this close correspondence between propositional and modal dependence logics only holds with respect to their entailment and validity problems. Satisfiability of propositional dependence logic is only **NP**-complete whereas it is **NEXPTIME**-complete for its modal variant. It is also worth noting that the proof of Theorem 3.5 gives rise to an alternative proof for the **NEXPTIME** upper bound for MDL (and EMDL) satisfiability, originally proved in [27]. Moreover, the technique can be succesfully applied to $\mathsf{ML}(\varovee)$. The following theorem entails that $\mathsf{ML}(\varovee)$ is no more complex than the ordinary modal logic.

Theorem 6.3. *The satisfiability, validity, and entailment problems for* $\mathsf{ML}(\varovee)$ *are* **PSPACE**-*complete.*

Proof. The lower bound follows from the Flatness property of ML (Proposition 2.2) and the **PSPACE**-hardness of satisfiability and validity problems for ML [20]. For the upper bound, it suffices to consider the entailment problem. The other cases are analogous. Ananlogously to the proof of Lemma 3.2 (see also Theorem 5.2 in [30]) we reduce $\mathsf{ML}(\varovee)$ formulae to large disjunctions with the help of appropriate witness sequences. Let θ be an $\mathsf{ML}(\varovee)$ formula that has m $\varovee$-disjunctions. Given a sequence $\overline{s} = (s_1, \dots, s_m) \in \{0,1\}^m$ we determine top-down recursively an ML formula $\eta^{\overline{s}}$ for each subformula η of θ in the following way. We let

$(\eta_0 \owedge \eta_1)^{\overline{s}} := \eta^{\overline{s}}_{s_i}$ where $(\eta_0 \owedge \eta_1)$ is the ith $\owedge$-disjunction of θ in some underlying ordering. Otherwise, we let $(\Box\eta)^{\overline{s}} := \Box\eta^{\overline{s}}$, $(\Diamond\eta)^{\overline{s}} := \Diamond\eta^{\overline{s}}$, $(\eta_0 \wedge \eta_1)^{\overline{s}} := \eta_0^{\overline{s}} \wedge \eta_1^{\overline{s}}$, $(\eta_0 \vee \eta_1)^{\overline{s}} := \eta_0^{\overline{s}} \vee \eta_1^{\overline{s}}$, $p^{\overline{s}} := p$, and $(\neg p)^{\overline{s}} := \neg p$. It is straightforward to show that θ is equivalent to $\bigowedge_{\overline{s}} \theta^{\overline{s}}$.

Let now $\phi_1, \dots, \phi_n$ be a sequence of $\mathsf{ML}(\owedge)$ formulae. Analogously to the proof of Lemma 3.2 we can show using Proposition 3.1 that $\{\phi_1, \dots, \phi_{n-1}\} \models \phi_n$ iff for all appropriate sequences $\overline{s}_1, \dots, \overline{s}_{n-1}$ there is $\overline{s}_n$ such that $\{\phi_1^{\overline{s}_1}, \dots, \phi_{n-1}^{\overline{s}_{n-1}}\} \models \phi_n^{\overline{s}_n}$. Notice that the number of $\owedge$-disjunctions appearing in $\phi_1, \dots, \phi_n$ is polynomial, and hence the sequence $\overline{s}_1 \dots \overline{s}_n$ is polynomial. It follows that the decision procedure presented in the proof of Theorem 3.5 can be now implemented in polynomial space. We immediately obtain the **PSPACE** upper bound for validity. For satisfiability of an $\mathsf{ML}(\owedge)$ formula ϕ, notice that the equivalent disjunction $\bigowedge_{\overline{s}} \phi^{\overline{s}}$ is satisfiable iff $\phi^{\overline{s}}$ is satisfiable for some sequence $\overline{s}$. Since satisfiability of an ML formula $\phi^{\overline{s}}$ can be determined in **PSPACE**, it follows that satisfiability of ϕ can be determined in **PSPACE**, too. This concludes the proof. $\square$

Combining the proofs of Theorem 3.5 and Theorem 6.3 we also notice that satifiability, validity, and entailment can be decided in **PSPACE** for EMDL formulae whose dependence atoms are of *logarithmic length*.

7. Validity and Entailment in Modal and Quantified Propositional Independence Logics

Next we turn to quantified propositional logic extended with either independence or inclusion atoms. We start in this section by considering the logic QPLInd and show that the complexity of its validity and entailment are both $\textbf{co-NEXPTIME}^{\textbf{NP}}$-complete. The upper bound is a simple adaptation of the standard model checking algorithm for team-based logics. The lower bound is shown by reducing from TRUE(Π_2-ADQBF) to the validity problem of quantified propositional logic extended with both dependence and inclusion atoms. Then using Galliani's translation [5] from dependence and inclusion logic to independence logic the result follows. Let us start with the upper bound result.

Lemma 7.1. *The entailment problem for* QPLInd *is in* $\textbf{co-NEXPTIME}^{\textbf{NP}}$.

Proof. By Theorem 3.4 it suffices to describe an exponential-time alternating algorithm that recognizes the entailment problem for QPLInd and switches once from an universal to an existential state. This algorithm utilizes the standard model checking algorithm for team-based logics (see, e.g., [4]), adapted for QPLInd in Algorithm 2. The input for Algorithm 2 is a propositional team X and a QPLInd formula ϕ, and the algorithm returns 1 for $\texttt{MC}(X, \phi)$ iff X satisfies ϕ. Note that the running time for $\texttt{MC}(X, \phi)$ is not bounded by a polynomial due to possible quantification in ϕ. Instead, the procedure takes time $f(|X|)g(|\phi|)$ for some polynomial function f and an exponential function g, and hence $\texttt{MC}(X, \phi)$ can be determined in **NEXPTIME**.

Let us denote by $\texttt{co-MC}(X, \phi)$ the outcome of the non-deterministic algorithm obtained from Algorithm 2 by replacing existential selections with universal ones. This algorithm returns 0 for $\texttt{co-MC}(X, \phi)$ iff X does not satisfy ϕ, and hence decides the complement of model checking in **co-NEXPTIME**. Given a sequence $\phi_1, \dots, \phi_n$ of QPLInd formulae, we can now determine whether $\{\phi_1, \dots, \phi_{n-1}\}$ entails ϕ_n in the following way. First universally guess a team X over variables that occur free in some $\phi_1, \dots, \phi_n$. Then for $i = 1, \dots, n-1$, if $\texttt{co-MC}(X,\phi_i)$ is 0, then return true. If $\texttt{co-MC}(X,\phi_i)$ is 1 and $i<n-1$, then move to $i+1$. Otherwise, if $\texttt{co-MC}(X,\phi_i)$ is 1 and $i=n-1$, then switch to existential state and return true iff $\texttt{MC}(X,\phi_n)$ is 1. It is straightforward to check that the described algorithm returns true iff $\{\phi_1,\dots,\phi_{n-1}\}\models\phi_n$. Since the algorithm alternates once from universal to existential state and runs in in exponential time, it follows that the procedure is in Π_2^{EXP}. □

Input : (X,ϕ) where X is a propositional team over $\mathsf{Fr}(\phi)$ and $\phi\in\mathsf{QPLInd}$
Output: $\texttt{MC}(X,\phi)$

if $\phi=\exists p\psi$ **then**
  existentially choose $F: X\to\{\{0,1\},\{0\},\{1\}\}$ and **return** $\texttt{MC}(X[F/p],\psi)$;
else if $\phi=\forall p\psi$ **then**
  return $\texttt{MC}(X[\{0,1\}/p],\psi)$;
else if $\phi=\psi\vee\theta$ **then**
  existentially choose $Y,Z\subseteq X$ such that $Y\cup Z=X$ and **return** $\texttt{MC}(Y,\psi)\wedge\texttt{MC}(Z,\theta)$;
else if $\phi=\psi\wedge\theta$ **then**
  return $\texttt{MC}(X,\psi)\wedge\texttt{MC}(X,\theta)$;
else if ϕ *is an atom* **then**
  return 1 iff $X\models\phi$;

Algorithm 2: A non-deterministic model checking algorithm for QPLInd

For the lower bound, we apply the fact that dependence atoms as well as inclusion atoms can be defined in independence logic. A translation for inclusion atoms can be given as follows.

Theorem 7.2 [5]**.** *The inclusion atom $\overline{p}\subseteq\overline{q}$ is equivalent to*

$$\begin{aligned}\phi:=&\forall v_1\forall v_2\forall\overline{r}\big((\overline{r}\neq\overline{p}\wedge\overline{r}\neq\overline{q})\vee(v_1\neq v_2\wedge\overline{r}\neq\overline{q})\vee\\&((v_1=v_2\vee\overline{r}=\overline{q})\wedge\overline{r}\perp_{\emptyset}v_1v_2)\big).\end{aligned}$$

The above theorem was shown in the first-order inclusion and independence logic setting but can be applied to the quantified propositional setting too since ϕ and $\overline{p}\subseteq\overline{q}$ are satisfied by a binary team X in the quantified propositional setting iff they are satisfied by X and the structure $\{0,1\}$ in the first-order setting. The lower bound can be now shown by a reduction from TRUE(Π_2-ADQBF) to validity of quantified propositional logic extended with dependence and inclusion atoms.

Lemma 7.3. *The validity problem for* QPLInd *is* **co-NEXPTIME$^{\mathbf{NP}}$***-hard.*

Proof. We reduce from TRUE(Π_2-ADQBF) which is **co-NEXPTIME$^{\mathbf{NP}}$**-hard by Theorem 5.4. Let $(\phi,\mathcal{C})$ be an instance of Π_2-ADQBF. Then ϕ is of the form

$$\forall f_1\dots\forall f_m\exists f_{m+1}\dots\exists f_{m+m'}\forall p_1\dots\forall p_n\theta$$

and $\mathcal{C}$ lists tuples $\overline{c}_i$ of elements from $\{p_1,\dots,p_n\}$, for $i=1,\dots,m+m'$. We show how to construct in polynomial time from ϕ a QPLInd formula ψ such that ψ is valid iff ϕ is true. The free variables of ψ consist of p_i, and fresh variables q_i that replace $f_i(\overline{c}_i)$. We construct ψ such that $X\models\psi$ iff one can select from X a maximal subteam $Y\subseteq X$ that satisfies

$\mathrm{dep}(\overline{c}_i, q_i)$, for $i = 1, \ldots, m$, and furthermore from Y a subteam $Z \subseteq Y$ that satisfies θ and $\mathrm{dep}(\overline{c}_i, q_i)$, for $i = m+1, \ldots, m+m'$.

First, let us construct top-down recursively formulae ψ_i, for $i = 1, \ldots, m$, as follows:

$$\psi_i := \exists r_i\big(\mathrm{dep}(\overline{c}_i q_i, r_i) \wedge \mathrm{dep}(\overline{c}_i r_i, q_i) \wedge \\ \forall r_i'(\neg r_i' \vee (r_i' \wedge \overline{c}_i r_i' \subseteq \overline{c}_i r_i)) \wedge (\neg r_i \vee (r_i \wedge \psi_{i+1}))\big). \tag{7.1}$$

We use $\mathrm{dep}(\overline{p}, q)$ as a shorthand for $q \perp_{\overline{p}} q$. For a team X (7.1) expresses that some maximal subteam $Y \subseteq X$ satisfying $\mathrm{dep}(\overline{c}_i, q_i)$ also satisfies ψ_{i+1}. The first three conjuncts of (7.1) entail that the subteam where r_i takes value 1 is a maximal subteam satisfying $\mathrm{dep}(\overline{c}_i, q_i)$. The last conjunct indicates that this subteam satisfies ψ_{i+1}. Consequently, if X already satisfies $\mathrm{dep}(\overline{c}_i, q_i)$, then (7.1) entails that X satisfies ψ_{i+1}. Define then

$$\psi_{m+1} := \theta' \vee \bigvee_{i=m+1}^{m+m'} \mathrm{dep}(\overline{c}_i, q_i)\,,$$

where θ' is obtained from θ by replacing occurrences of $f_i(\overline{c}_i)$ with q_i. This disjunction amounts to the existential selection of the functions $\overline{c}_i \mapsto q_i$ for $i = m+1, \ldots, m+m'$.

We now claim that $\psi := \psi_1$ is valid iff ϕ is true. Assume first that ψ is valid, and let f_i be any function from $\{0,1\}^{|\overline{c}_i|} \to \{0,1\}$ for $i = 1, \ldots, m$. Let X be the team that consists of all assignments s that map $p_1, \ldots, p_n, q_{m+1}, \ldots, q_{m+m'}$ into $\{0,1\}$ and $q_1, \ldots, q_m$ to $f(s(\overline{c}_1)), \ldots, f(s(\overline{c}_m))$. By the assumption $X \models \psi$. Hence, we find $F_1 \colon X \to \mathcal{P}(X) \setminus \{\emptyset\}$ such that

$$X[F_1/r_1] \models \mathrm{dep}(\overline{c}_1 q_1, r_1) \wedge \mathrm{dep}(\overline{c}_1 r_1, q_1) \wedge \\ \forall r_1'(\neg r_1' \vee (r_1' \wedge \overline{c}_1 r_1' \subseteq \overline{c}_1 r_1)) \wedge (\neg r_1 \vee (r_1 \wedge \psi_2)). \tag{7.2}$$

Let $X' := X[F_1/r_1][1/r_1']$. Then $X' \models \mathrm{dep}(\overline{c}_1, q_1)$ by the construction and $X' \models \mathrm{dep}(\overline{c}_1 q_1, r_1)$ by (7.2), and hence $X' \models \mathrm{dep}(\overline{c}_1, r_1)$. Also by the third conjunct of (7.2) $X' \models \overline{c}_1 r_1' \subseteq \overline{c}_1 r_1$. Therefore, it cannot be the case that that $s(r_1) = 0$ for some $s \in X'$, and hence by the last conjunct of (7.2) and Proposition (4.4), $X[1/r_1] \models \psi_2$. After n iterations we obtain that $X[1/r_1] \ldots [1/r_n] \models \psi_{m+1}$ which implies by Proposition (4.4) that $X \models \psi_{m+1}$. Hence, there are $Z, Y_1, \ldots, Y_{m'} \subseteq X$ such that $Z \cup Y_1 \cup \ldots \cup Y_{m'} = X$, $Z \models \theta'$, and $Y_i \models \mathrm{dep}(\overline{c}_{m+i}, q_{m+i})$ for $i = 1, \ldots, m'$. Notice that we are now at the same position as in the proof of Theorem 5.6. Hence, we obtain that ϕ is true and that the direction from left to right holds.

Assume then that ϕ is true, and let X be any team whose domain contains variables $p_1, \ldots, p_n, q_1, \ldots, q_{m+m'}$. Considering ψ_i, we may choose a mapping F_i so that for each value $\overline{b} \in X(\overline{c}_i)$, we select $l \in \{0,1\}$ such that $\overline{b}l \in X(\overline{c}_i q_i)$, and set $F_i(s) = 1$ iff $s(\overline{c}_i q_i) = \overline{b}l$. This amounts to a selection of a maximal subteam satisfying $\mathrm{dep}(\overline{c}_i, q_i)$. It follows that $X[F_i/r_i]$ satisfies the first three conjuncts of (7.1) and that $\{s \in X[F_i/r_i] \mid F_i(s) = 0\}$ satisfies $\neg r_i$. It remains to show that $\{s \in X[F_i/r_i] \mid F_i(s) = 1\}$ satisfies ψ_{i+1}. Iterating this procedure we need only to show that $X' := \{s \in X[F_1/q_1] \ldots [F_n/q_n] : F_1(s) = \ldots = F_n(s_n) = 1\}$ satisfies ψ_{m+1}. By the selection of functions $F_1, \ldots, F_n$ we notice that X' satisfies $\mathrm{dep}(\overline{c}_i, q_i)$ for $i = 1, \ldots, n$. Again, following the proof of Theorem 5.6 we obtain that X' satisfies $\theta' \vee \bigvee_{i=m+1}^{m+m'} \mathrm{dep}(\overline{c}_i, q_i)$. This shows the direction from right to left. Since the reduction from ϕ to ψ can be done in polynomial time, this concludes the proof. $\square$

The exact **co-NEXPTIME$^{\mathbf{NP}}$** bound for QPLInd entaiment and validity follows now by Lemmata 7.1 and 7.3. Theorems 7.4 and 4.6 then imply the same lower bound for MLInd.

This means that validity in modal independence logic is at least as hard as entailment in modal dependence logic. We leave determining the exact complexity of MLInd entailment and validity as an open question.

Theorem 7.4. *The entailment and the validity problems for* QPLInd *are* **co-NEXPTIME$^{\mathbf{NP}}$**-*complete.*

Corollary 7.5. *The entailment and the validity problems for* MLInd *are* **co-NEXPTIME$^{\mathbf{NP}}$**-*hard.*

8. Validity and Entailment in Modal and Quantified Propositional Inclusion Logics

Next we consider quantified propositional inclusion logic and show that its validity and entailment problems are complete for **EXPTIME** and **co-NEXPTIME**, respectively. The result regarding validity is a simple observation.

Theorem 8.1. *The validity problem for* QPLInc *is* **EXPTIME**-*complete.*

Proof. For the lower bound, note that the satisfiability problem for PLInc has been shown to be **EXPTIME**-complete in [12]. Also, note that $\phi(\overline{p})$ is satisfiable iff $\exists\overline{p}\phi(\overline{p})$ is valid. Consequently, QPLInc validity is **EXPTIME**-hard.

For the upper bound, we notice by union closure (Proposition 4.3) that a formula $\phi(\overline{p}) \in$ QPLInc is valid iff it is valid over all singleton teams. Let us denote by ϕ^* the formula obtained from ϕ by replacing all inclusion atoms $\overline{q} \subseteq \overline{r}$ with $\overline{pq} \subseteq \overline{pr}$. We observe that $\phi(\overline{p})$ is valid over all singletons iff $\{\emptyset\} \models \forall\overline{p}\phi^*(\overline{p})$. The direction from left to right follows by union closure and since $\overline{q} \subseteq \overline{r}$ entails $\overline{pq} \subseteq \overline{pr}$. For the direction from right to left it can be shown by induction on ψ that $X \models \psi^*$ entails $\{s \in X \mid s(\overline{p}) = \overline{a}\} \models \psi$ for all teams X and sequences of Boolean values $\overline{a}$. By locality (Proposition 4.4) $\{\emptyset\} \models \forall\overline{p}\phi^*(\overline{p})$ iff $\forall\overline{p}\phi^*(\overline{p})$ is satisfiable. Since MLInc satisfiability is in **EXPTIME** [11], we conclude by Theorem 4.6 that the **EXPTIME** upper bound holds. □

Let us now prove the exact **co-NEXPTIME** complexity bound for QPLInc entailment. For the proof of the lower bound, we apply TRUE(Σ_1-ADQBF).

Lemma 8.2. *The entailment problem for* QPLInc *is* **co-NEXPTIME**-*hard.*

Proof. Theorem 5.4 states that TRUE(Σ_1-ADQBF) is **NEXPTIME**-hard. We reduce from its complement problem. Let $(\phi, \mathcal{C})$ be an instance of Σ_1-ADQBF. Then ϕ is of the form $\exists g_1 \ldots \exists g_m \forall p_1 \ldots \forall p_n \theta$ and $\mathcal{C}$ lists tuples $\overline{c}_i$ of elements from $\{p_1, \ldots, p_n\}$, for $i = 1, \ldots, n$. We show how to construct in polynomial time from $(\phi, \mathcal{C})$ two QPLInc formulae ψ and ψ' such that $(\phi, \mathcal{C})$ is false iff $\psi \models \psi'$. These formulae use fresh variables q_i for encoding $g_i(\overline{c}_i)$ and variables t and f for encoding true and false. Denote by $\overline{p}$ and $\overline{q}$ sequences $p_1 \ldots p_n$ and $q_1 \ldots q_m$, and by $\overline{p}'$ and $\overline{q}'$ their distinct copies, respectively. Let $\overline{d}_i$ list the variables of $\{p_1, \ldots, p_n\}$ that do not occur in $\overline{c}_i$. Moreover, let $\overline{v}_i$ be lists of fresh variables of length $|\overline{c}_i|$. The idea is to describe that whenever a team X is complete with respect to the values of $\overline{p}$, then either one of the constraints in $\mathcal{C}$ is falsified or one of the assignments of X falsifies θ. We let

$$\psi := t \wedge \neg f \wedge \bigwedge_{i=1}^{n} (p_1 \ldots p_{i-1} t \subseteq p_1 \ldots p_{i-1} p_i \wedge p_1 \ldots p_{i-1} f \subseteq p_1 \ldots p_{i-1} p_i), \tag{8.1}$$

and define

$$\psi' := \exists \overline{p}'\overline{q}'(\theta_0^{\perp}(\overline{p}'\overline{q}'/\overline{pq}) \wedge \overline{p}'\overline{q}' \subseteq \overline{pq}) \vee \bigvee_{i=1}^{m} \exists \overline{v}_i(\overline{v}_i t \subseteq \overline{c}_i q_i \wedge \overline{v}_i f \subseteq \overline{c}_i q_i), \tag{8.2}$$

where θ_0 is obtained from θ by replacing occurrences of $g_i(\overline{c}_i)$ with q_i. Above, (8.1) exrpresses that a team X is complete with respect to $\overline{p}$. In (8.2) the first disjunct entails that θ_0 is false for some assignment in X, and the second disjunct indicates that some constraint in $\mathcal{C}$ does not hold in X. Furthermore, all disjunctions in (8.2) are essentially Boolean; if a disjunct is satisfied by a non-empty subteam of X, it is likewise satisfied by the full team X. We leave it to the reader to verify that $(\phi, \mathcal{C})$ is false iff $\psi \models \psi'$. ☐

For the upper bound we refer to Algorithm 3 that was first presented in [11] in the modal logic context. Given a team X and a formula $\phi \in \mathsf{QPLInd}$, this algorithm computes deterministically the maximal subset of X that satisfies ϕ. Note that the existence of such a team is guaranteed by the union closure property of QPLInc (Proposition 4.3). Given an instance $\Sigma \cup \phi$ of the entailment problem, the proof idea is now to first universally guess a team X (possibly of exponential size), and then check using Algorithm 3 whether X is a witness of $\Sigma \not\models \phi$. Since the last part can be executed deterministically in exponential time, the **co-NEXPTIME** upper bound follows.

Lemma 8.3. *The entailment problem for* QPLInc *is in* **co-NEXPTIME**.

Proof. Consider the computation of $\texttt{MaxSub}(X, \phi)$ in Algorithm 3. We leave it to the reader to show, by straightforward induction on the complexity of ϕ, that for all X, Y over a shared domain $V \supseteq \mathsf{Fr}(\phi)$ the following two claims holds:

(1) $\texttt{MaxSub}(X, \phi) \models \phi$, and
(2) $Y \subseteq X$ and $Y \models \phi \Rightarrow Y \subseteq \texttt{MaxSub}(X, \phi)$.

Note that $\texttt{MaxSub}(X, \phi)$ is the unique maximal subteam of X satisfying ϕ. The idea is that each subset of X satisfying ϕ survives each iteration step. Since we have $\texttt{MaxSub}(X, \phi) \subseteq X$, it now follows directly from (1) and (2) that $\texttt{MaxSub}(X, \phi) = X$ iff $X \models \phi$.

Let us now present the universal exponential-time algorithm for deciding entailment for QPL. Assuming an input sequence $\phi_1, \ldots, \phi_n$ from QPLInc, the question is to decide whether $\{\phi_1, \ldots, \phi_{n-1}\} \models \phi_n$. The algorithm first universally guesses a team X over $\bigcup_{i=1}^{n} \mathsf{Fr}(\phi_i)$ and then using Algorithm 3 deterministically tests whether X is a counterexample for $\{\phi_1, \ldots, \phi_{n-1}\} \models \phi_n$, returning true iff this is not the case. Note that X is a counterexample for $\{\phi_1, \ldots, \phi_{n-1}\} \models \phi_n$ iff $\texttt{MaxSub}(X, \phi_n) \neq X$ and $\texttt{MaxSub}(X, \phi_i) = X$ for $i = 1, \ldots, n-1$. By the locality principle of QPL (Proposition 4.4) this suffices, i. e., each universal branch returns true iff $\{\phi_1, \ldots, \phi_{n-1}\} \models \phi_n$.

It remains to show that the procedure runs in exponential time. Consider first the running time of Algorithm 3 over an input (X, ϕ). First note that one can find an exponential function g such that at each recursive step $\texttt{MaxSub}(Y, \psi)$ the size of the team Y is bounded by $|X|g(|\phi|)$. The possible exponential blow-up comes from nested quantification in ϕ. Furthermore, each base step $\texttt{MaxSub}(Y, \psi)$ can be computed in polynomial time in the size of Y. Also, each recursive step $\texttt{MaxSub}(Y, \psi)$ involves at most $|Y|$ iterations consisting of either computations of $\texttt{MaxSub}(Z, \theta)$ for $Z \subseteq Y$ and a subformula θ of ψ, or removals of assignments from Y. It follows by induction that there exists a polynomial f and an exponential h such that the running time of $\texttt{MaxSub}(Y, \psi)$ is bounded by $f(|X|)h(|\phi|)$. The overall algorithm now guesses first a team X whose size is possibly exponential in the input. By the previous

Input : (X, ϕ) where X is a propositional team over $\mathsf{Fr}(\phi)$ and $\phi \in \mathsf{QPLInc}$
Output : $\mathtt{MaxSub}(X, \phi)$

if $\phi = \exists p \psi$ **then**
  return $\{s \in X \mid s(0/p) \in \mathtt{MaxSub}(X[\{0,1\}/p], \psi)$ or $s(1/p) \in \mathtt{MaxSub}(X[\{0,1\}/p], \psi)\}$;
else if $\phi = \forall p \psi$ **then**
  $Y \leftarrow X[\{0,1\}/p]$;
  while $Y \neq \{s \in Y \mid \{s(0/p), s(1/p)\} \subseteq \mathtt{MaxSub}(Y, \psi)\}$ **do**
    $Y \leftarrow \{s \in Y \mid \{s(0/p), s(1/p)\} \subseteq \mathtt{MaxSub}(Y, \psi)\}$;
  end
  return $\{s \in X \mid \{s(0/p), s(1/p)\} \subseteq Y\}$
else if $\phi = \psi \vee \theta$ **then**
  return $\mathtt{MaxSub}(X, \psi) \cup \mathtt{MaxSub}(X, \theta)$;
else if $\phi = \psi \wedge \theta$ **then**
  $Y \leftarrow X$;
  while $Y \neq \mathtt{MaxSub}(\mathtt{MaxSub}(Y, \psi), \theta)$ **do**
    $Y \leftarrow \mathtt{MaxSub}(\mathtt{MaxSub}(Y, \psi), \theta)$;
  end
  return Y;
else if $\phi = p$ **then**
  return $\{s \in X \mid s(p) = 1\}$;
else if $\phi = \neg p$ **then**
  return $\{s \in X \mid s(p) = 0\}$;
else if $\overline{p} \subseteq \overline{q}$ **then**
  $Y \leftarrow X$;
  while $Y \neq \{s \in Y \mid \exists s' \in Y : s(\overline{p}) = s'(\overline{q})\}$ **do**
    $Y \leftarrow \{s \in Y \mid \exists s' \in Y : s(\overline{p}) = s'(\overline{q})\}$;
  end
  return Y
end

Algorithm 3: A deterministic model checking algorithm for QPLInc

reasoning, the running time of $\mathtt{MaxSub}(X, \phi_i)$ remains exponential for each ϕ_i. This shows the claim. □

Lemmata 8.3 and 8.2 now show the **co-NEXPTIME**-completeness of the QPLInc entailment problem. The same lower bound have been shown in [11] to apply already to QPLInc validity. The exact complexity of MLInc validity and entailment however remains an open problem.

Theorem 8.4. *The entailment problem for* QPLInc *is* **co-NEXPTIME**-*complete.*

Theorem 8.5 [11]**.** *The entailment and the validity problems for* MLInc *are* **co-NEXPTIME**-*hard.*

	satisfiability	validity	entailment
PL	**NP** [2, 21]	**co-NP** [2, 21]	**co-NP** [2, 21]
ML	**PSPACE** [20]	**PSPACE** [20]	**PSPACE** [20]
ML($\varovee$)	**PSPACE** [27]	**PSPACE** [Thm. 6.3]	**PSPACE** [Thm. 6.3]
PDL	**NP** [22]	**NEXPTIME** [30]	**co-NEXPTIME**$^{\mathbf{NP}}$ [Thm. 6.1]
QPDL, MDL, EMDL	**NEXPTIME** [25, 27], [Thm. 4.6]	**NEXPTIME** [Thm. 6.2]	**co-NEXPTIME**$^{\mathbf{NP}}$ [Thm. 6.1]
QPLInd	**NEXPTIME** [25, 18], [Thm. 4.6]	**co-NEXPTIME**$^{\mathbf{NP}}$ [Thm. 7.4]	**co-NEXPTIME**$^{\mathbf{NP}}$ [Thm. 7.4]
MLInd	**NEXPTIME** [18]	$\geq$ **co-NEXPTIME**$^{\mathbf{NP}}$ [Cor. 7.5]	$\geq$ **co-NEXPTIME**$^{\mathbf{NP}}$ [Cor. 7.5]
QPLInc	**EXPTIME** [12], [Thm. 4.6]	**EXPTIME** [Thm. 8.1]	**co-NEXPTIME** [Thm. 8.4]
MLInc	**EXPTIME** [11]	$\geq$ **co-NEXPTIME** [11]	$\geq$ **co-NEXPTIME** [11]

Table 1: Summary of results. The stated complexity classes refer to completeness results, except that the prefix "$\geq$" refers to hardness results.

9. Conclusion

We have examined the validity and entailment problem for various modal and propositional dependence logics (see Table 1). We showed that the entailment problem for (extended) modal and (quantified) propositional dependence logic is **co-NEXPTIME**$^{\mathbf{NP}}$-complete, and that the corresponding validity problems are **NEXPTIME**-complete. We also showed that modal logic extended with intuitionistic disjunction is **PSPACE**-complete with respect to its satisfiability, validity, and entailment problems, therefore being not more complex than the standard modal logic. Furthermore, we examined extensions of propositional and modal logics with independence and inclusion atoms. Quantified propositional independence logic was proven to be **co-NEXPTIME**$^{\mathbf{NP}}$-complete both in terms of its validity and entailment problem. For quantified propositional inclusion logic the validity and entailment problems were shown to be **EXPTIME**-complete and **co-NEXPTIME**-complete, respectively. Using standard reduction methods we established the same lower bounds for modal independence and inclusion logic, although for validity of modal inclusion logic a higher lower bound of **co-NEXPTIME** is known to apply. However, we leave determining the exact complexities of validity/entailment of MLInd and MLInc as an open problem. It is plausible that solving these questions will open up possibilities for novel axiomatic characterizations.

References

[1] Ashok K. Chandra, Dexter Kozen, and Larry J. Stockmeyer. Alternation. *J. ACM*, 28(1):114–133, 1981.

[2] Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, STOC '71, pages 151–158, New York, NY, USA, 1971. ACM.

[3] Johannes Ebbing, Lauri Hella, Arne Meier, Julian-Steffen Müller, Jonni Virtema, and Heribert Vollmer. Extended modal dependence logic. In *Logic, Language, Information, and Computation - 20th International Workshop, WoLLIC 2013, Darmstadt, Germany, August 20-23, 2013. Proceedings*, pages 126–137, 2013. `doi:10.1007/978-3-642-39992-3_13`.

[4] Johannes Ebbing and Peter Lohmann. Complexity of model checking for modal dependence logic. In Mária Bieliková, Gerhard Friedrich, Georg Gottlob, Stefan Katzenbeisser, and György Turán, editors, *SOFSEM 2012: Theory and Practice of Computer Science*, volume 7147 of *Lecture Notes in Computer Science*, pages 226–237. 2012.

[5] Pietro Galliani. Inclusion and exclusion dependencies in team semantics: On some logics of imperfect information. *Annals of Pure and Applied Logic*, 163(1):68 – 84, 2012.

[6] Pietro Galliani, Miika Hannula, and Juha Kontinen. Hierarchies in independence logic. In *Computer Science Logic 2013 (CSL 2013)*, volume 23, pages 263–280, 2013.

[7] Erich Grädel and Jouko Väänänen. Dependence and independence. *Studia Logica*, 101(2):399–410, 2013.

[8] Miika Hannula, Juha Kontinen, Martin Lück, and Jonni Virtema. On quantified propositional logics and the exponential time hierarchy. In *Proceedings of the Seventh International Symposium on Games, Automata, Logics and Formal Verification, GandALF 2016, Catania, Italy, 14-16 September 2016.*, pages 198–212, 2016.

[9] Miika Hannula, Juha Kontinen, Jonni Virtema, and Heribert Vollmer. Complexity of propositional independence and inclusion logic. In *Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24-28, 2015, Proceedings, Part I*, pages 269–280, 2015.

[10] Miika Hannula, Juha Kontinen, Jonni Virtema, and Heribert Vollmer. Complexity of propositional logics in team semantic. *ACM Trans. Comput. Log.*, 19(1):2:1–2:14, 2018. URL: `http://doi.acm.org/10.1145/3157054`, `doi:10.1145/3157054`.

[11] Lauri Hella, Antti Kuusisto, Arne Meier, and Jonni Virtema. Model checking and validity in propositional and modal inclusion logics. In *42nd International Symposium on Mathematical Foundations of Computer Science, MFCS 2017, August 21-25, 2017 - Aalborg, Denmark*, pages 32:1–32:14, 2017. URL: `https://doi.org/10.4230/LIPIcs.MFCS.2017.32`, `doi:10.4230/LIPIcs.MFCS.2017.32`.

[12] Lauri Hella, Antti Kuusisto, Arne Meier, and Heribert Vollmer. Modal inclusion logic: Being lax is simpler than being strict. In *Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24-28, 2015, Proceedings, Part I*, pages 281–292, 2015.

[13] Lauri Hella, Kerkko Luosto, Katsuhiko Sano, and Jonni Virtema. 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*, pages 294–312, 2014.

[14] Lauri Hella and Johanna Stumpf. The expressive power of modal logic with inclusion atoms. In *Proceedings Sixth International Symposium on Games, Automata, Logics and Formal Verification, GandALF 2015, Genoa, Italy, 21-22nd September 2015.*, pages 129–143, 2015.

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

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

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

[18] Juha Kontinen, Julian-Steffen Müller, Henning Schnoor, and Heribert Vollmer. 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*, pages 353–372, 2014.

[19] Andreas Krebs, Arne Meier, and Jonni Virtema. A team based variant of CTL. In *22nd International Symposium on Temporal Representation and Reasoning, TIME 2015, Kassel, Germany, September 23-25, 2015*, pages 140–149, 2015.

[20] Richard E. Ladner. The computational complexity of provability in systems of modal propositional logic. *SIAM J. Comput.*, 6(3):467–480, 1977.

[21] Leonid A Levin. Universal sequential search problems. *Problemy Peredachi Informatsii*, 9(3):115–116, 1973.

[22] Peter Lohmann and Heribert Vollmer. Complexity results for modal dependence logic. *Studia Logica*, 101(2):343–366, 2013.

[23] Martin Lück. Complete problems of propositional logic for the exponential hierarchy. *CoRR*, abs/1602.03050, 2016. URL: `http://arxiv.org/abs/1602.03050`.

[24] Julian-Steffen Müller and Heribert Vollmer. Model checking for modal dependence logic: An approach through post's lattice. In *Logic, Language, Information, and Computation - 20th International Workshop, WoLLIC 2013, Darmstadt, Germany, August 20-23, 2013. Proceedings*, pages 238–250, 2013.

[25] Gary L. Peterson, John H. Reif, and Salman Azhar. Lower bounds for multiplayer noncooperative games of incomplete information. *Computers & Mathematics with Applications*, 41(7-8):957–992, April 2001.

[26] Katsuhiko Sano and Jonni Virtema. Axiomatizing propositional dependence logics. In *24th EACSL Annual Conference on Computer Science Logic, CSL 2015, September 7-10, 2015, Berlin, Germany*, pages 292–307, 2015.

[27] Merlijn Sevenster. Model-theoretic and computational properties of modal dependence logic. *Journal of Logic and Computation*, 19(6):1157–1173, 2009.

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

[29] Jouko Väänänen. Modal Dependence Logic. In Krzysztof R. Apt and Robert van Rooij, editors, *New Perspectives on Games and Interaction*. Amsterdam University Press, Amsterdam, 2008.

[30] Jonni Virtema. Complexity of validity for propositional dependence logics. *Inf. Comput.*, 253:224–236, 2017. URL: `https://doi.org/10.1016/j.ic.2016.07.008`, `doi:10.1016/j.ic.2016.07.008`.

[31] Fan Yang. On extensions and variants of dependence logic. PhD thesis, University of Helsinki, 2014.

[32] Fan Yang. Modal dependence logics: axiomatizations and model-theoretic properties. *Logic Journal of the IGPL*, 25(5):773–805, 2017. URL: `+http://dx.doi.org/10.1093/jigpal/jzx023`, `arXiv:/oup/backfile/content_public/journal/jigpal/25/5/10.1093_jigpal_jzx023/1/jzx023.pdf`, `doi:10.1093/jigpal/jzx023`.

[33] Fan Yang and Jouko Väänänen. Propositional logics of dependence. *Annals of Pure and Applied Logic*, 167(7):557 – 589, 2016.