Modal Independence Logic

Kontinen, Juha

College Publications
2014


http://hdl.handle.net/10138/218294

Downloaded from Helda, University of Helsinki institutional repository.

This is an electronic reprint of the original article.

This reprint may differ from the original in pagination and typographic detail.

Please cite the original version.
Abstract
This paper introduces modal independence logic MIL, a modal logic that can explicitly talk about independence among propositional variables. Formulas of MIL are not evaluated in worlds but in sets of worlds, so called teams. In this vein, MIL can be seen as a variant of Väänänen’s modal dependence logic MDL. We show that MIL embeds MDL and is strictly more expressive. However, on singleton teams, MIL is shown to be not more expressive than usual modal logic, but MIL is exponentially more succinct. Making use of a new form of bisimulation, we extend these expressivity results to modal logics extended by various generalized dependence atoms. We demonstrate the expressive power of MIL by giving a specification of the anonymity requirement of the dining cryptographers protocol in MIL. We also study complexity issues of MIL and show that, though it is more expressive, its satisfiability and model checking problem have the same complexity as for MDL.

Keywords: dependence logic, team semantics, independence, expressivity over finite models, computational complexity.

1 Introduction
The concept of independence is ubiquitous in many scientific disciplines such as experimental physics, social choice theory, computer science, and cryptography. Dependence logic D, introduced by Jouko Väänänen in [14], is a new logical framework in which various notions of dependence and independence can be formalized and studied. It extends first-order logic by so called dependence atoms

$\neg (x_1, \ldots, x_{n-1}, x_n)$,

expressing that the value of the variable $x_n$ depends (only) on the values of $x_1, \ldots, x_{n-1}$, in other words, that $x_n$ is functionally dependent of $x_1, \ldots, x_{n-1}$. Of course, such a dependency does not make sense when talking about single
assignments; therefore dependence logic formulas are evaluated for so called
teams, i.e., sets of assignments. A team can for example be a relational
database table, a collection of plays of a game, or a set of agents with fea-
tures. It is this team semantics, together with dependence atoms, that gives
dependence logic its expressive power: it is known that $D$ is as expressive as $\Sigma_1^1$,
that is, the properties of finite structures that can be expressed in dependence
logic are exactly the NP-properties.

In a slightly later paper Väänänen [15] introduced dependence atoms into
(propositional) modal logic. Here, teams are sets of worlds, and a dependence
atom $=\langle p_1, \ldots, p_{n-1}, p_n \rangle$ holds in a team $T$ if there is a Boolean function that
determines the value of $p_n$ from those of $p_1, \ldots, p_{n-1}$ in all worlds in $T$. The
so obtained modal dependence logic $\text{MDL}$ was studied from the point of view
of expressivity and complexity in [13].

In this article we introduce a novel modal variant of dependence logic called
modal independence logic, $\text{MIL}$, extending the formulas of modal logic $\text{ML}$ by
so-called independence atoms

$$(p_1, \ldots, p_\ell) \perp_{\langle r_1, \ldots, r_m \rangle} (q_1, \ldots, q_n),$$

the meaning of which is that the propositional sequences $\vec{p}$ and $\vec{q}$ are indepen-
dent of each other for any fixed value of $\vec{r}$. Modal independence logic thus has
its roots in modal dependence logic $\text{MDL}$ [15] and first-order independence logic
[8]. In modal independence logic, dependencies between propositions can be
expressed, and thus, analogously to the first-order case, $\text{MDL}$ can be embedded
as a sublogic into $\text{MIL}$, and it is easy to see that $\text{MIL}$ is strictly more expressive
than $\text{MDL}$.

The aim of this paper is to initiate a study of the expressiveness and the
computational complexity of modal independence logic. For this end, we first
study the computational complexity of the satisfiability and the model checking
problem for $\text{MIL}$. We show that, though $\text{MIL}$ is more expressive than $\text{MDL}$, the
complexity of these decision problems stays the same, i.e., the satisfiability
problem is complete for nondeterministic exponential time (NEXP-complete,
[13]) and the model checking problem is NP-complete [6]. In order to settle
the complexity of satisfiability for $\text{MIL}$, we give a translation of $\text{MIL}$-formulas
to existential second-order logic formulas the first-order part of which is in the
Gödel–Kalmár–Schütte prefix class. Our result then follows from the classical
result that the satisfiability problem for this prefix class is NEXP-complete [3].
We will also show that the same upper bound on satisfiability can be obtained
for a whole range of variants of $\text{MIL}$ via the notion of a generalized (modal)
dependence atom (a notion introduced in the first-order framework in [11]).

The expressive power of $\text{MDL}$ was first studied by Sevenster [13], where he
showed that $\text{MDL}$ is equivalent to $\text{ML}$ on singleton teams. In this paper we
prove a general result showing that $\text{MIL}$, and in fact any variant of it whose
generalized dependence atoms are FO-definable, is bound to be equivalent to
$\text{ML}$ over singleton teams. Interestingly, it was recently shown in [5] that a so-
called extended modal dependence logic $\text{EMDL}$ is strictly more expressive than
To demonstrate the potential applications of MIL, we consider the dining cryptographers protocol [4], a classic example for anonymous broadcast which is used as a benchmark protocol in model checking of security protocols [1]. We show how the anonymity requirement of the protocol can be formalized in modal independence logic, where—unlike in the usual approaches using epistemic logic—we do not need to use the Kripke model’s accessibility relation to encode knowledge, but to express the “possible future” relation of branching-time models. In addition to demonstrating MIL’s expressivity, we also derive a succinctness result from our modeling of the dining cryptographers: While MIL and ML are equally expressive on singletons, MIL is exponentially more succinct.

2 Modal Independence Logic

Definition 2.1 The syntax of modal logic ML is inductively defined by the following grammar in extended Backus–Naur form:

\[ \phi ::= p | \neg \phi | \phi \land \phi | \phi \lor \phi | \Box \phi | \Diamond \phi. \]

The syntax of modal dependence logic MDL is defined by

\[ \phi ::= p | \neg \phi | \phi \land \phi | \phi \lor \phi | \Box \phi | \Diamond \phi, \]

where \( p \) is a propositional variable and \( \vec{q} \) a sequence of propositional variables.

The syntax of modal independence logic MIL is defined by

\[ \phi ::= p | \neg \phi | \phi \land \vec{p} \perp \vec{r} \vec{q} | \phi \lor \phi | \Box \phi | \Diamond \phi, \]

where \( p \) is a propositional variable and \( \vec{p}, \vec{r}, \vec{q} \) are sequences of propositional variables. The sequence \( \vec{r} \) may be empty.

It is worth noting that the negation of dependence logic, and that of MDL, is not the classical negation but a so-called "game theoretic" negation that still satisfies the usual De Morgan laws. As now customary with team semantics, we think of negation as a defined operation and restrict attention to formulas in which negation only appears in front of proposition symbols (formulas \( \neg=(\vec{q},p) \) are logically equivalent to falsum and hence can also be dispensed without loss of generality [15]). A more detailed account of the role of negation in dependence logic can be found in [10].

A Kripke structure 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 \( \pi : W \rightarrow \mathcal{P}(V) \) is a labeling function for a set \( V \) of propositional variables. A team is a (possibly empty) set \( T \subseteq W \). As usual, in a Kripke structure \( \mathcal{M} \) the set of all successors of \( T \subseteq W \) is defined as \( R(T) = \{ s \in W \mid \exists s' \in T : (s', s) \in R \} \). Furthermore we define \( R(T) = \{ T' \subseteq R(T) \mid \forall s \in T : \exists s' \in T' : (s, s') \in R \} \), the set of legal successor teams.
Definition 2.2 Let \( \vec{p} = (p_1, \ldots, p_n) \) be a sequence of variables and \( w, w' \) be worlds of a Kripke model \( \mathcal{M} = (W, R, \pi) \). Then \( w \) and \( w' \) are equivalent under \( \pi \) over \( \vec{p} \), denoted by \( w \equiv_{\pi, \vec{p}} w' \), if the following holds:

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

Definition 2.3 (Semantics of ML, MDL, and MIL) Let \( \mathcal{M} = (W, R, \pi) \) be a Kripke structure, \( T \) be a team over \( \mathcal{M} \) and \( \phi \) be a formula. The semantic evaluation (denoted as \( \mathcal{M}, T \models \phi \)) is defined inductively as follows.

\[
\begin{align*}
\mathcal{M}, T \models p & \iff \forall w \in T: p \in \pi(w) \\
\mathcal{M}, T \models \vec{p} & \iff \forall w \in T: p \not\in \pi(w) \\
\mathcal{M}, T \models \phi_1 \land \phi_2 & \iff \mathcal{M}, T \models \phi_1 \text{ and } \mathcal{M}, T \models \phi_2 \\
\mathcal{M}, T \models \phi_1 \lor \phi_2 & \iff \exists T_1, T_2: T_1 \cup T_2 = T, \mathcal{M}, T_1 \models \phi_1 \text{ and } \mathcal{M}, T_2 \models \phi_2 \\
\mathcal{M}, T \models \lozenge \phi & \iff \exists T' \in R(T): \mathcal{M}, T' \models \phi \\
\mathcal{M}, T \models \Box \phi & \iff \mathcal{M}, R(T) \models \phi \\
\mathcal{M}, T \models =_{(\vec{q}, \vec{p})} & \iff \forall w, w' \in T: w \equiv_{\pi, \vec{q}} w' \text{ implies } w \equiv_{\pi, \vec{p}} w' \\
\mathcal{M}, T \models \vec{p}_1 \downarrow_{\vec{p}} \vec{p}_2 & \iff \forall w, w' \in T: w \equiv_{\pi, \vec{q}} w' \text{ implies } \exists w'' \in T: w'' \equiv_{\pi, \vec{p}_1} w \text{ and } w'' \equiv_{\pi, \vec{p}_2} w' \text{ and } w''' \equiv_{\pi, \vec{q}} w
\end{align*}
\]

Note that for modal logic formulas \( \phi \) we have \( \mathcal{M}, \{w\} \models \phi \) if and only if \( \mathcal{M}, w \models \phi \) (where in the latter case, \( \models \) is defined as in any textbook for usual modal logic).

In fact it is easy to see that without dependence or independence atom, our logic has the so called flatness property, stating that team semantics and usual semantics essentially do not make a difference:

Lemma 2.4 For every ML-formula \( \phi \) and all models \( \mathcal{M} \) and teams \( T, \mathcal{M}, T \models \phi \) iff \( \mathcal{M}, w \models \phi \) for all \( w \in T \).

Another simple observation is that the empty team satisfies all formulas:

Lemma 2.5 For every MIL-formula \( \phi \) and all models \( \mathcal{M}, \emptyset \models \phi \).

Team semantics and independence atoms together will lead to a richer expressive power, as we will prove in Section 6. However, we will also show that over teams \( T \) consisting of one world only, ML and MIL have the same expressive power.

Definition 2.6 Formulas \( \varphi \) and \( \varphi' \) are equivalent on singletons, if for every model \( \mathcal{M} \) and every \( w \in \mathcal{M} \), we have \( \mathcal{M}, \{w\} \models \varphi \) if and only if \( \mathcal{M}, \{w\} \models \varphi' \).

Note that on singleton teams, the independence atom trivially always evaluates to true. However, using the modal operators \( \Box \) and \( \lozenge \), a formula that is evaluated on singleton teams as a starting point clearly is able to talk about nontrivial teams as well. An example for this is the formula constructed in Section 5: The formula is evaluated on a singleton team—the starting point of the protocol—but specifies independence properties for much larger teams, namely subsets of all possible protocol outcomes.
3 Complexity Results

In this section we will study the computational complexity of the model checking and the satisfiability problem for MIL. In [8], it was observed that in first-order team semantics, $=(\vec{p}, q)$ is equivalent to $q \perp \vec{p} q$. This observation clearly carries over to MIL, and hence in particular shows that MIL is a generalization of MDL.

Lemma 3.1 Let $M$ be a model and $T$ a team over $M$, let $\vec{p}$ and $q$ be variables. Then $M, T \models = (\vec{p}, q)$ if and only if $M, T \models q \perp \vec{p} q$.

We now define the two decision problems whose complexity we wish to study, namely the model checking and the satisfiability problem for modal independence logic.

Problem: MIL-SAT
Input: MIL formula $\phi$
Question: Does there exists a Kripke model $M$ and a team $T \neq \emptyset$ with $M, T \models \phi$?

Problem: MIL-MC
Input: Kripke model $M$, team $T$ and MIL formula $\phi$
Question: $M, T \models \phi$?

The corresponding problems for modal dependence logic are denoted by MDL-SAT and MDL-MC.

It is easy to see that model checking for MIL is not more difficult than model checking for MDL, namely NP-complete.

Theorem 3.2 MIL-MC is NP-complete.

Proof. The lower bound follows immediately from Lemma 3.1 and NP-completeness of MDL-MC [6]. The upper bound follows from a simple extension of the well-known model checking algorithm for modal logic, see Algorithm 1. □

Next we will consider the complexity of the satisfiability problem MIL-SAT for modal independence logic. From Lemma 3.1 and the hardness of MDL-SAT for nondeterministic exponential time [13] we immediately obtain the following lower bound:

Lemma 3.3 MIL-SAT is NEXP-hard.

In order to show containment in NEXP, we need to recall the following classical result. Recall that the so-called Gödel–Kalmár–Schütte prefix class $[\exists^* \forall^2 \exists^*, all]$ contains sentences of FO, in a relational vocabulary without equality, which are in prefix normal form and have a quantifier prefix of the form $\exists^* \forall^2 \exists^*$.

Proposition 3.4 ([3]) Satisfiability of formulas in prefix class $[\exists^* \forall^2 \exists^*, all]$ can be decided in $\text{NTIME}(2^{\Theta(n/\log n)})$.

Next we will show that MIL-SAT $\in$ NEXP with the help of Proposition 3.4. We will first define a variant of the standard translation of ML into FO that
Algorithm 1 NP algorithm for MIL-MC

1: function milmc(M, T, φ)
2:  if φ = □ψ then
3:    return milmc(M, ψ, R(T))
4:  if φ = ◦ψ then
5:    existentially guess T' ∈ R(T)
6:    return milmc(M, ψ, T')
7:  else if φ = ψ_1 ∧ ψ_2 then
8:    return milmc(M, ψ_1, T) and milmc(M, ψ_2, T)
9:  else if φ = ψ_1 ∨ ψ_2 then
10: else existentially guess T_1 ⊔ T_2 = T
11: return milmc(M, ψ_1, T_1) and milmc(M, ψ_2, T_2)
12: else if φ = p then
13: for s ∈ T do
14:    if p ∉ π(s) then
15: return false
16: return true
17: else if φ = ¯p then
18: for s ∈ T do
19:    if p ∈ π(s) then
20: return false
21: return true
22: else if φ = p ⊥ q then
23: for s ∈ T do
24:    for s' ∈ T do
25:      if π(s') ∩ ¯p = π(s) ∩ ¯p then
26:        found ← false
27:       for s'' ∈ T do
28:         agreeP ← π(s'') ∩ p = π(s) ∩ p
29:         agreeQ ← π(s'') ∩ q = π(s') ∩ q
30:         agreeR ← π(s'') ∩ r = π(s) ∩ r
31:         if agreeP and agreeQ and agreeR then
32:           found ← true
33:           if not found then
34:             return false
35: return true

maps MIL-formulas to formulas of monadic existential second-order logic. For a Kripke structure (W, R, π), and a team T ⊆ W, we denote by (W, {A_i}_1, R, T) the first-order structure of vocabulary {R, T} ∪ {A_i}_1∈N encoding (W, R, π) in the obvious way.

Lemma 3.5 For any formula φ ∈ MIL there is a sentence φ^* of monadic existential second-order logic of the form

\[ \exists Y_1 \ldots \exists Y_m \forall x \forall y \exists z_1 \ldots \exists z_k \theta, \] (1)
where θ is quantifier-free, and such that for all (W, R, π) and T it holds

\[(W, R, π), T \models φ ↔ (W, \{A_i\}_{i \in \mathbb{N}}, R, T) \models φ^∗.\]

**Proof.** We first define an auxiliary translation \(φ \mapsto φ'\) for which correctness is obvious and then indicate how to go from \(φ'\) to \(φ^∗\).

(i) Suppose \(φ\) is \(p_i\). Then \(φ'\) is defined as

\[φ' := \forall x(T(x) \rightarrow A_{p_i}(x)).\]

(ii) Suppose \(φ\) is \(\neg p_i\). Then \(φ'\) is defined as

\[φ' := \forall x(T(x) \rightarrow \neg A_{p_i}(x)).\]

(iii) Suppose \(φ\) is \(ψ_1 \lor ψ_2\). Then \(φ'\) is defined as

\[φ' := \exists Y_1 \exists Y_2(∀x(T(x) \leftrightarrow (Y_1(x) \lor Y_2(x))) \land ψ'_1(T/Y_1) \land ψ'_2(T/Y_2)).\]

(iv) Suppose \(φ\) is \(ψ_1 \land ψ_2\). Then \(φ'\) is defined as

\[φ' := ψ'_1 \land ψ'_2.\]

(v) Suppose \(φ\) is \(ψ\). Then \(φ'\) is defined as

\[φ' := \exists Y(∀x(T(x) \rightarrow \exists z(Y(z) \land E(x, z))) \land \exists u(T(u) \land E(u, x))) \land ψ'(T/Y)).\]

(vi) Suppose \(φ\) is \(\Box ψ\). Then \(φ'\) is defined as

\[φ' := \exists Y(∀x∀y(((T(x) \land E(x, y)) \rightarrow Y(y)) \land (Y(x) \rightarrow \exists z(T(z) \land E(z, x))) \land ψ'(T/Y))).\]

(vii) Suppose \(φ\) is \(\vec{p}_1 \perp \vec{p}_2 \vec{p}_3\). Then \(φ'\) is defined as

\[φ' := \forall x∀y(((T(x) \land T(y)) \land EQ_{\vec{p}_1}(x, y)) \rightarrow \exists z(T(z) \land EQ_{\vec{p}_2}(x, z) \land EQ_{\vec{p}_3}(x, z) \land EQ_{\vec{p}_3}(y, z))),\]

where \(EQ_{\vec{p}_i}(v, w)\) is a shorthand for the formula

\[\bigwedge_{p \in \vec{p}_i} A_p(v) \leftrightarrow A_p(w).\]

It remains to define the translation \(φ \mapsto φ^∗\). This translation is defined by modifying the above clauses by essentially moving all quantifiers to the left of the formula, and by possibly renaming some of the bound variables. We will indicate these modifications by considering the case of disjunction. The other cases are analogous. Assume that \(ψ^*_1\) and \(ψ^*_2\) are defined already:

\[ψ^*_1 = \exists Y∀x∀y∃z θ_i,\]
where $\theta_i$ is quantifier free, and $\psi^*_i \equiv \psi'_i$. By renaming of bound variables, we may assume that $Y_2 = Y_3 \ldots Y_k$, and $Y_1 = Y_{k+1} \ldots Y_m$, and that $z_1$ and $z_2$ do not have any common variables either. Then $(\psi_1 \lor \psi_2)^*$ is defined by replacing $\psi'{_i}$ by $\psi^*_{i}$ in the definition of $(\psi_1 \lor \psi_2)'$ (see clause iii), and by extending the scopes of the quantifiers:

$$(\psi_1 \lor \psi_2)^* := \exists Y_1 \ldots \exists Y_m \forall x \forall y \exists z_2 \exists z_1 ((T(x) \leftrightarrow (Y_1(x) \lor Y_2(x))) \land \theta_1(T/Y_1) \land \theta_2(T/Y_2)).$$

**Theorem 3.6** MIL-SAT is in NEXP.

**Proof.** Let $\phi \in \text{MIL}$. Then $\phi$ is satisfiable by a Kripke model $\mathcal{M}$ and a team $T \neq \emptyset$ if and only if $\phi^* \land \exists w T(w)$ is satisfiable. This follows from the previous lemma and the fact that there is a 1-1 correspondence with Kripke structures $(W, R, \pi)$, and teams $T$ for $\phi$ and $\{R, T\} \cup \{A_i\}_{1 \leq i \leq n}$-structures $(W, \{A_i\}_{1 \leq i \leq n}, R, T)$, where $n$ is large enough such that all $p_i$ appearing in $\phi$ satisfy $i \leq n$.

Recall now that $\phi^*$ has the form

$$\exists Y_1 \ldots \exists Y_m \forall x \forall y \exists z_1 \ldots \exists z_k \theta,$$

hence $\phi^* \land \exists w T(w)$ is logically equivalent to

$$\exists Y_1 \ldots \exists Y_m \forall x \forall y \exists z_1 \ldots \exists z_k \exists w (\theta \land T(w)),$$

which is satisfiable if and only if the first-order sentence

$$\forall x \forall y \exists z_1 \ldots \exists z_k \exists w (\theta \land T(w))$$

of vocabulary $\{Y_1, \ldots, Y_m\} \cup \{R, T\} \cup \{A_i\}_{1 \leq i \leq n}$ is satisfiable. The sentence (2) is contained in prefix class $[\exists^\ast \forall^2 \exists^*, \text{all}]$, hence the satisfiability of it, and also of $\phi^* \land \exists w T(w)$, can be decided in time $\text{NTIME}(2^{O(|\phi^*|)})$. The claim now follows from the fact the mapping $\phi \mapsto \phi^*$ can be computed in time polynomial in $|\phi|$. $\square$

**Corollary 3.7** MIL-SAT is NEXP-complete.

It is interesting to note that Theorem 3.6 and Lemma 3.1 directly imply the result of Sevenster [13] that MDL-SAT is contained in NEXP. On the other hand, it seems that the original argument of Sevenster does not immediately generalize to MIL.

**Corollary 3.8** MDL-SAT is NEXP-complete.

### 4 Generalized Dependency Notions

MIL can be seen as an extension of modal logic with team semantics by the independence atom—let us denote such an extension by $\text{ML}(\perp)$. Similarly, we can extend modal logic with other atoms, so-called *generalized dependence atoms*, which we define now.
Definition 4.1 Let $\mathcal{M} = (W, R, \pi)$ be a Kripke model and $T = (w_1, \ldots, w_m)$ be a team over $\mathcal{M}$. Then for any propositional variable $p$, $T(p)$ is defined as the tuple $(s_1, \ldots, s_m)$, where $s_i$ for $1 \leq i \leq m$ is defined as:

$$s_i = \begin{cases} 1 & w_i \in \pi(p) \\ 0, & \text{otherwise.} \end{cases}$$

For a set of propositions $\vec{q} = (q_1, \ldots, q_k)$, we define $T(\vec{q})$ analogously as $(T(q_1), \ldots, T(q_k))$.

Similar to Kuusisto’s [11] definition of generalized first-order dependence atoms we give a definition of generalized modal dependence atoms. In the following, a set of matrices $D$ is invariant under permutations of rows, if for every matrix $M \in D$, if $M'$ is obtained from $M$ by permuting $M$’s rows, then $M'$ is an element of $D$ as well.

Definition 4.2 Let $D$ be a set of Boolean $n$-column matrices that is invariant under permutation of rows. The semantics of the generalized dependence atom defined by $D$ is given as follows:

Let $\mathcal{M}$ be a Kripke model, $T$ be a team over $\mathcal{M}$ and $p_1, \ldots, p_n$ atomic propositions. Then

$$\mathcal{M}, T \models D(p_1, \ldots, p_n) \iff (T(p_1), \ldots, T(p_n)) \in D.$$ 

The width of $D$ is defined to be $n$.

Note that for simplicity we do not distinguish in notation between the logical atom $D$ and the set $D$ of Boolean matrices.

The Boolean matrix $(T(p_1), \ldots, T(p_n))$ contains one column for each of the variables $p_1, \ldots, p_n$; each row of the matrix corresponds to one world from $T$. The entry for variable $p_i$ and world $w \in T$ is 1 if and only if the variable $p_i$ is satisfied in the world $w$. We require that $D$ is invariant under permutation of rows in order to ensure that whether $\mathcal{M}, T \models D(p_1, \ldots, p_n)$ holds does not depend on the ordering of the worlds in $T$ that is used in computing the tuple $T(p)$.

In the following we will mainly be interested in generalized dependence atom definable by first-order formulae. For this purpose let $D$ be an atom of width $n$ as above, and $\phi$ be a first-order sentence over signature $\langle A_1, \ldots, A_n \rangle$. Then $\phi$ defines $D$ if for all Kripke models $\mathcal{M} = (W, R, \pi)$ and teams $T$ over $\mathcal{M}$,

$$\mathcal{M}, T \models D(p_1, \ldots, p_n) \iff \mathcal{A} \models \phi,$$

where $\mathcal{A}$ is the first-order structure with universe $T$ and relations $A_i^\mathcal{A}$ for $1 \leq i \leq n$, where for all $w \in T$, $w \in A_i^\mathcal{A} \iff p_i \in \pi(w)$.

We say that a generalized dependence atom $D$ is FO-definable if there exists a FO-formula $\phi$ defining $D$ as above. Strictly speaking, the dependence atoms considered in the literature are families of dependence atoms for different width, e.g., the simple dependence $=(p_1, \ldots, p_n)$ is defined for arbitrary values of $n$. 
Let us say that such a family is \((P\text{-uniformly})\) \(\text{FO}\)-definable if there exists a family of defining first-order formulae \(\phi_n\) such that \(\phi_n\) defines the atom of width \(n\) and the mapping \(1^n \mapsto \langle \phi_n \rangle\) is computable in polynomial time; that is, an encoding of formula \(\phi_n\) is computable in time polynomial in \(n\). Note that in particular this implies that \(|\phi_n| = p(n)\) for some polynomial \(p\).

As examples let us show how to define some well-studied generalized dependence atoms as follows.

\[
\begin{align*}
\vec{p} \equiv \vec{q} &\iff \forall w \forall w' ((\bigwedge_{1 \leq i \leq n}(A_{p_i}(w) \leftrightarrow A_{p_i}(w'))) \rightarrow (A_q(w) \leftrightarrow A_q(w'))) \\
\vec{p} \subseteq \vec{q} &\iff \forall w \exists w' ((\bigwedge_{1 \leq i \leq n}(A_{p_i}(w) \leftrightarrow A_{q_i}(w')))) \\
\vec{p} \mid \vec{q} &\iff \forall w \forall w' ((\bigvee_{1 \leq i \leq n}(A_{p_i}(w) \leftrightarrow \neg A_{q_i}(w')))
\end{align*}
\]

The latter two so-called inclusion and exclusion atoms were introduced by Galliani in [7]. In particular all above atoms are \(\text{FO}\)-definable. The independence atom \(\vec{p}_1 \perp \vec{q}_2\) is also \(\text{FO}\)-definable in the obvious way.\(^1\)

We use \(\text{ML}(D)\) to denote the extension of \(\text{ML}\) by a generalized dependence atom \(D\). We will next show that our complexity upper bounds from Section 3 can be generalized to cover \(\text{ML}(D)\) for certain \(\text{FO}\)-definable dependence atoms \(D\). For model checking, we simply use the fact that first-order formulas can be verified in polynomial time and obtain the following corollary:

**Corollary 4.3** Let \(D\) be a \(P\text{-uniformly} \text{FO}\)-definable generalized dependence atom. Then \(\text{ML}(D)\)-MC is in \(\text{NP}\).

For the satisfiability problem, we generalize the proof of Theorem 3.5 in case (vii) to dependence atoms which are definable by a \([\exists^* \forall^2 \exists^*]\) formula.

**Corollary 4.4** Let \(D\) be a generalized dependence atom that is \(P\text{-uniformly} \text{FO}\)-definable by a (family of) first-order formula(e) in the prefix class \([\exists^* \forall^2 \exists^*]\). Then \(\text{ML}(D)\)-SAT is in \(\text{NEXP}\).

5 Example: The Dining Cryptographers

The dining cryptographers [4], a standard example for anonymous broadcast, is the following problem: A group of cryptographers \(\{c_0, \ldots, c_{n-1}\}\) with \(n \geq 3\) sit in a restaurant, where \(c_i\) sits between \(c_{i-1}\) and \(c_{i+1}\). (Indices of the cryptographers are always modulo \(n\), and \(i\) always ranges over \(0, \ldots, n - 1\).)

After dinner, it turns out that someone already paid. There are only two possibilities: Either one of the cryptographers secretly paid, or the NSA did. Naturally, they want to know which of these is the case, but without revealing the paying cryptographer if one of them paid. They use the following protocol (in the protocol, \(\oplus\) defines the exclusive-or of two bits, where \(b_1 \oplus b_2 = b_1 + b_2 \mod 2\)):

\[\text{Since the FO-formula } \phi \text{ may only depend on the width, we restrict ourselves to occurrences of } \vec{p}_1 \perp \vec{q}_2 \text{ where } |\vec{p}_1| = |\vec{q}_2| = |\vec{q}|, \text{ if these sets are nonempty, which we can always assume without loss of generality by repeating variable occurrences, the case that one of these sets is empty can then be encoded into widths that are not multiples of 3 in a straightforward manner.}\]
• For each $i$, let $p_i$ be 1 iff $c_i$ paid. Each $c_i$ knows the value of $p_i$, but not of $p_j$ for $j \neq i$. There is at most one $i$ with $p_i = 1$. The protocol computes the value $p_0 \oplus p_1 \oplus \cdots \oplus p_{n-1}$, which is the same as $p_0 \lor p_1 \lor \cdots \lor p_{n-1}$.

• Each adjacent pair $\{c_i, c_{i+1}\}$ computes a random bit $\text{bit}_{\{i,i+1\}}$.

• Each $c_i$ publicly announces the value $\text{announce}_i = p_i \oplus \text{bit}_{\{i,i-1\}} \oplus \text{bit}_{\{i,i+1\}}$.

• Then, $p_0 \oplus p_1 \oplus \cdots \oplus p_{n-1} = \text{announce}_0 \oplus \text{announce}_1 \oplus \cdots \oplus \text{announce}_{n-1}$.

The protocol clearly computes the correct answer, the interesting aspect is the anonymity requirement: No cryptographer $c_i$ should learn anything about the values $p_j$ for $j \neq i$ except for what follows from the values $p_i$ or the result (if $c_i$ or the NSA paid then $c_j$ did not). The protocol models anonymous broadcast, since the message “1” is, if sent, received by all cryptographers, but the sender remains anonymous. We formalize this using modal independence logic. We start by capturing the protocol in the following Kripke model:

```
q0
pNSA
p0
p1
\cdots
pn-1
```

The protocol starts in $q_0$, the model then branches into states $p_{\text{NSA}}$, $p_0$, $\ldots$, $p_{n-1}$, depending on whether the NSA or some $c_i$ paid. Each of these states has $2^n$ successor states, for the $2^n$ possible random bit values, these states are final. The relation $R$ is as indicated. We use the following variables:

• $p_{\text{NSA}}$ and $p_i$ are true if the NSA, resp. cryptographer $c_i$ paid, i.e., in the states denoted with the same name as the variable and in their successors.

• each of the $n$ variables $\text{bit}_{\{i,i+1\}}$ is true in the states where the bit shared between $c_i$ and $c_{i+1}$ is 1.

• each $\text{announce}_i$ is true in all final states which satisfy $p_i \oplus \text{bit}_{\{i,i-1\}} \oplus \text{bit}_{\{i,i+1\}}$ (this encodes that the cryptographers follow the protocol).

For each $c_i$, we define the set $k_i$ of the variables whose values $c_i$ knows after the protocol run as $k_i := \{p_i, \text{bit}_{\{i,i-1\}}, \text{bit}_{\{i,i+1\}}\} \cup \{\text{announce}_j \mid j \neq i\}$. Clearly, $c_i$ also knows the value $\text{announce}_i$, but since this can be computed from $p_i$, $\text{bit}_{\{i,i-1\}}$ and $\text{bit}_{\{i,i+1\}}$, we omit it from $k_i$.

The formula expressing the anonymity requirement consists of several parts, one global part and then, for each combination of cryptographers, a local part. We start with the global part, which merely expresses that none of the individual bits that some cryptographer knows determines the value of any $p_i$ on its
own, with the exception that if \( p_i = 1 \), then of course cryptographer \( c_i \) knows that \( p_j = 0 \) for all \( j \neq i \). The global part \( \varphi_g \) is as follows:

\[
\varphi_g = \bigwedge_{v \in \{ \text{bit}_{(i,i+1)}, \text{announce} \}, \ k \in \{0, \ldots, n-1\}} \big( \Diamond \Diamond (v \land p_k) \land \Diamond \Diamond (\overline{v} \land \overline{p_k}) \land \Diamond \Diamond (\overline{\overline{v}} \land \overline{p_k}) \big)
\]

\( \land \bigwedge_{i \neq j} \big( p_i \land \overline{p_j} \big) \land \big( \Diamond \Diamond (\overline{p_i} \land \overline{p_j}) \big) \land \big( \Diamond \Diamond (p_i \land p_j) \big) \land \big( \Diamond \Diamond (\overline{p_i} \land \overline{p_j}) \big) \)

The first line of the formula requires that, for every variable \( v \) of the \( \text{bit}_{(i,i+1)} \) or \( \text{announce} \)-variables, and every cryptographer \( c_k \), every combination of truth values of \( v \) and \( p_k \) appears. This encodes that the value of a single variable \( v \) does not give away any information about the value of any \( p_k \).

The second line is a similar requirement for the value \( p_i \). If \( c_i \) paid, then she knows that \( c_j \) did not pay, for \( i \neq j \). However, the combination \( "p_i \land p_j" \) for \( i \neq j \) should be the only one not appearing. Hence the formula requires that all other combinations appear in some final state. The global part \( \varphi_g \) hence ensures that each individual bit that \( c_i \) knows does not tell him whether \( c_j \) paid, unless of course \( i = j \) or \( p_i = 1 \).

The more interesting part is to encode that even the combination of the above bits does not lead to additional knowledge; this is where the independence atom is crucial. We introduce some notation to enumerate the variables in \( \vec{k} \):

- for each \( i \), let \( \vec{k}_i = \{ v_i^1, \ldots, v_i^{n+2} \} \),
- for \( j \leq k \), let \( V_{j \rightarrow k}^i = \{ v_j^1, \ldots, v_k^1 \} \),
- let \( V_{j}^i = V_{j \rightarrow j}^i \).

We now use modal independence logic to express that if each single variable from \( \vec{k}_i \) does not tell \( c_i \) anything about the value of \( p_k \), then their combination does not, either. This is achieved with the following formula:

\[
\varphi^{i,k} = \Box \Box \left( (V_{1 \rightarrow j}^1 p_k V_{j}^1) \land (V_{1 \rightarrow 2}^1 p_k V_{2}^1) \land \cdots \land (V_{1 \rightarrow n+1}^1 p_k V_{n+2}^1) \right)
\]

This formula requires that for each \( j \), each pair of variable assignments \( I_1 \) to \( V_{1 \rightarrow j-1}^j \) and \( I_2 \) to \( V_j^j \) that is “locally compatible” with some truth value \( P(p_k) \)—in other words, neither of these assignments by itself implies that the actual value of \( p_k \) is not \( P(p_k) \)—is also compatible with that value for the combination of \( I_1 \) and \( I_2 \), i.e., there is some state satisfying \( I_1 \cup I_2 \cup P \) (where the notion of a state satisfying a propositional assignment is defined as expected and the union of these assignments is well-defined since their domains are disjoint).

As a consequence, the formula requires that for each \( I: \vec{k} \to \{0, 1\} \) and each \( P: \{ p_k \} \to \{0, 1\} \), if for each \( v \in \vec{k} \), there is a world \( w \) such that \( w \models I|_{\{v\}} \) and \( w \models P \), then there is a world \( w \) such that \( w \models I \cup P \).

The following proposition formally states that our above-developed formulas indeed express the anonymity property of the protocol as intended. From a single cryptographer \( c_1 \)'s point of view, it says that every observation \( I \) which
can arise when $c_i$ follows the protocol, as long as some cryptographer different from $c_i$ paid for the dinner, then for every $k$ different from $i$, both possibilities—$c_k$ paid for the dinner, or $c_k$ did not pay—cannot be ruled out by the observation $I$. We say that an assignment $I: \overline{k}_i \cup \{\text{announce}_i\} \rightarrow \{0,1\}$ is consistent if $i$ follows the protocol, i.e., if $I(\text{announce}_i) = I(p_i) \oplus I(\text{bit}_{(i,i-1)}) \oplus I(\text{bit}_{(i,i+1)})$. Note that in the models we are interested in, only consistent assignments appear in final states.

**Proposition 5.1** If a Kripke model $M = (W,R,\pi)$ satisfies the formula $\varphi_q \land \bigwedge_{i,k \in \{0,\ldots,n-1\},i \neq k} \varphi_i^{i,k}$ at the world $q_0$, then the team $T = T(R(\{q_0\}))$ satisfies the following condition: For each $i \neq k \in \{0,\ldots,n-1\}$ and each consistent $I: \overline{k}_i \cup \{\text{announce}_i\} \rightarrow \{0,1\}$ with $I(p_i) = 0$ and $\oplus_{j=0}^{n-1} I(\text{announce}_j) = 1$, there are worlds $w'_1, w'_2 \in T$ with $w'_1 \models I, p_k$ and $w'_2 \models I, \overline{p_k}$.

We omit the easy proof; the proposition immediately follows from the semantics of the independence atom.

Our discussion only treats the anonymity property of the protocol. For a complete treatment, one also has to address other aspects as e.g., correctness, we omit this discussion here.

Note that in comparison to express the anonymity requirement using epistemic logic (see, e.g., [1,12]), we do not use the relation of the Kripke model to represent knowledge, but to express branching time. In particular, our approach only uses a single modality.

### 6 Expressiveness

We now compare the expressiveness of MIL and classical modal logic, which we abbreviate with ML. We show that MIL is strictly more expressive than ML on teams (simply because MIL is not downwards closed, i.e., from $M,T \models \varphi$ and $T' \subseteq T$, it does not follow that $M,T' \models \varphi$), but that their expressiveness coincides on singleton teams. However, on singletons, MIL is exponentially more succinct than ML. We then study the expressiveness of MIL with a generalized dependence atom as introduced in Section 4 instead of the independence atom.

#### 6.1 Expressiveness of MIL and ML

Clearly, since MIL is not downward-closed, we obtain the following:

**Proposition 6.1** There is an MIL-formula $\varphi_{\text{MIL}}$ such that there is no ML-formula $\varphi_{\text{ML}}$ with the property that $M,T \models \varphi_{\text{ML}}$ if and only if $M,T \models \varphi_{\text{MIL}}$ for all models $M$ and all teams $T$.

**Proof.** This is true for every formula $\varphi_{\text{MIL}}$ that is not downwards closed: In this case we have teams $T' \subseteq T$ of the same model $M$ with $M,T \models \varphi_{\text{MIL}}$ and $M,T' \not\models \varphi_{\text{MIL}}$. However, for any modal formula $\varphi_{\text{ML}}$, clearly if $M,w \models \varphi_{\text{ML}}$ for all $w \in T$, then the same is true for all $w \in T'$ as $T' \subseteq T$. An easy example for a formula that is not downwards closed is $x \bot y$. This formula is satisfied on a team $T$ in which every combination of truth values of $x$ and $y$ is realized in some world, but not on its subset $T'$ containing only worlds $w$ and $w'$ with assignments $x \land y$ and $\pi \land \overline{\pi}$, respectively. \qed
The proposition remains true for classical modal logic extended with a
global modality, or for MDL, since these logics remain downward-closed. In [13],
it was shown that MDL is as expressive as classical modal logic on singletons.
Therefore, a natural question to ask is whether on singletons, MIL is still more
expressive than ML. We show that this is not the case, but we will also see that
MDL is exponentially more succinct than ML, even on singletons. For our proof,
we use bisimulations, which are a well-established tool to compare expressiveness
of different concepts. We recall the classical definition of bisimulation for
modal logic:

**Definition 6.2** Let \( M = (W, R, \pi) \) and \( M' = (W', R', \pi') \) be Kripke models.
A relation \( Z \subseteq W \times W' \) is a modal bisimulation if for every \((w, w') \in Z\), the following holds:

- \( \pi(w) = \pi'(w') \), i.e., \( w \) and \( w' \) satisfy the same propositional variables,
- if \( u \) is an \( R \)-successor of \( w \), then there is an \( R' \)-successor \( u' \) of \( w' \) such that
  \((u, u') \in Z \) (forward condition),
- if \( u' \) is an \( R' \)-successor of \( w' \), then there is an \( R \)-successor \( u \) of \( w \) such that
  \((u, u') \in Z \) (backward condition).

It is well-known and easy to see that modal logic is invariant under bisimulation,
i.e., if \( Z \) is a bisimulation and \((w, w') \in Z\), then \( w \) and \( w' \) satisfy the same modal formulas. We now “lift” this property to modal independence logic
by considering a bisimulation \( Z \) as above on the team level:

**Definition 6.3** Let \( M = (W, R, \pi) \) and \( M' = (W', R', \pi') \) be models, let \( T \subseteq W \) and \( T' \subseteq W' \) be teams. Let \( Z \subseteq W \times W' \) be a modal bisimulation. Then
\( T \) and \( T' \) are \( Z \)-bisimilar if the following is true:

- for each \( w \in T \), there is a \( w' \in T' \) such that \((w, w') \in Z \),
- for each \( w' \in T' \), there is a \( w \in T \) such that \((w, w') \in Z \).

We now show that on the team level, bisimulation for modal independence
logic plays the same role as it does on the world level for modal logic: Simply
stated, bisimilar teams satisfy the same formulas. Due to Lemma 3.1, the result
also applies to modal dependence logic. This lemma may be of independent
interest (for example, it implies a “family-of-trees”-like model property), we
use it to compare the expressiveness of MIL and ML.

**Lemma 6.4** Let \( M = (W, R, \pi) \) and \( M' = (W', R', \pi') \) be Kripke models, let
\( T \subseteq W \) and \( T' \subseteq W' \) be teams that are \( Z \)-bisimilar for a modal bisimulation \( Z \). Then for any MIL-formula \( \phi \), we have that \( M, T \models \phi \) if and only if \( M', T' \models \phi \).

**Proof.** We show the lemma by induction on \( \phi \). Clearly it suffices to show that
if \( M, T \models_{\text{MIL}} \phi \), then \( M', T' \models_{\text{MIL}} \phi \). Hence assume \( M, T \models_{\text{MIL}} \phi \).

- Let \( \phi = x \) for some propositional variable \( x \), and let \( w' \in T' \). Since \( T \) and \( T' \)
are \( Z \)-bisimilar, there is a world \( w \in T \) with \((w, w') \in Z \). Since \( M, T \models_{\text{MIL}} x \), the variable \( x \) is true at \( w \) in \( M \). Since \( Z \) is a modal bisimulation, it
follows that \( x \) is true at \( w' \) in \( M' \), and hence every world \( w' \in T' \) satisfies \( x \).
Therefore, it follows that \( M', T' \models \varphi \).

- If \( \varphi = \neg x \), the proof is the same as above.
- Let \( \varphi = \varphi_1 \land \varphi_2 \). This case trivially follows inductively.

- Let \( \varphi = \varphi_1 \lor \varphi_2 \). Since \( M, T \models \varphi \), it follows that \( T = T_1 \cup T_2 \) for teams \( T_1 \) and \( T_2 \) with \( M, T_1 \models \varphi_1 \) and \( M, T_2 \models \varphi_2 \). We define teams \( T'_1 \) and \( T'_2 \) as follows:
  \[ T'_1 = \{ w' \in T' \mid (w, w') \in Z \text{ for some } w \in T_1 \}, \]
  \[ T'_2 = \{ w' \in T' \mid (w, w') \in Z \text{ for some } w \in T_2 \}. \]

We prove the following:
(i) \( T' = T'_1 \cup T'_2 \)
(ii) \( T_1 \) and \( T'_1 \) are \( Z \)-bisimilar,
(iii) \( T_2 \) and \( T'_2 \) are \( Z \)-bisimilar.

By induction, it then follows that \( M', T'_1 \models \varphi_1 \) and \( M', T'_2 \models \varphi_2 \), which, since \( T' = T'_1 \cup T'_2 \) implies that \( M', T' \models \varphi \). We prove these points:
(i) By construction, \( T'_1 \cup T'_2 \subseteq T' \). Hence let \( w' \in T' \). Since \( T \) and \( T' \) are \( Z \)-bisimilar, there is some \( w \in T \) such that \( (w, w') \in Z \). Since \( T = T_1 \cup T_2 \), we can, without loss of generality, assume that \( w \in T_1 \). By definition of \( T'_1 \), it follows that \( w' \in T'_1 \).
(ii) First let \( w \in T_1 \subseteq T \). Since \( T \) and \( T' \) are \( Z \)-bisimilar, there is some \( w' \in T' \) such that \( (w, w') \in Z \). Due to the definition of \( T'_1 \), it follows that \( w' \in T'_1 \).

For the converse, assume that \( w' \in T'_1 \). By definition, there is some \( w \in T_1 \) such that \( (w, w') \in Z \).
(iii) This follows with the same proof as for \( T_1 \) and \( T'_1 \).

Hence \( M', T' \models \varphi \) as required.

- Let \( \varphi = \Box \psi \). Since \( M, T \models \varphi \), there exists a team \( U \subseteq R(T) \) such that for each \( w \in T \), the set \( u(w) := R(\{w\}) \cap T \) is not empty, and \( M, U \models \psi \). We define a corresponding team \( U' \) of \( M' \) as follows: Start with \( U' = \emptyset \) and then for each \( (w, w') \in (T \times T') \cap Z \), do the following:
  - For each \( R \)-successor \( v \) of \( w \) that is an element of \( U \), since \( Z \) is a modal bisimulation and \( (w, w') \in Z \), there is at least one \( R' \)-successor \( v' \) of \( w' \) with \( (v, v') \in Z \). Add all such \( v' \) to the set \( U' \).

By construction, \( U' \) only contains worlds that are \( R' \)-successors of worlds in \( T' \). Hence to show that \( M', T' \models \Box \psi \), it remains to show that
(i) for each \( w' \in T' \), the team \( U' \) contains a world \( v' \) that is an \( R' \)-successor of \( w' \),
(ii) the teams \( U \) and \( U' \) are \( Z \)-bisimilar.

The claim then follows by induction, since \( M, U \models \psi \). We now show the above two points:
(i) Let \( w' \in T' \). Since \( T \) and \( T' \) are \( Z \)-bisimilar, there is some \( w \in T \) with \( (w, w') \in Z \). Due to the choice of \( U \), there is some \( v \in U \) which is an \( R \)-successor of \( w \). By construction of the set \( U' \), a world \( v' \) that is an \( R' \)-successor of \( w' \) has been added to \( U' \).
(ii) By construction, for every \( R \)-successor \( v \) of some \( w \) in \( T \), at least one \( v' \)
has been added to $U'$ with $(v, v') \in Z$. For the converse, by construction of $U'$, for every $v'$ added to $U'$ there is a $v \in U$ with $(v, v') \in Z$. Hence $M', T' \models \varphi \psi$ as required.

- Now assume that $\varphi = \Box \psi$, and let $U$ be the set of all $R$-successors of worlds in $T$, let $U'$ be the set of all $R'$-successors of worlds in $T'$. By induction, it suffices to show that $U$ and $U'$ are $Z$-bisimilar. Hence let $v \in U$ be the $R$-successor of some $w \in T$. Since $T$ and $T'$ are $Z$-bisimilar, there is some $w' \in T'$ such that $(w, w') \in Z$. Since $v$ is an $R$-successor of $w$, and since $Z$ is a modal bisimulation, there is some $v'$ which is an $R'$-successor of $w'$ such that $(v, v') \in Z$. Since $v'$ is an $R'$-successor of $w'$, it follows that $v' \in U'$.

The converse direction follows analogously.

- Let $\varphi = \vec{p}_1 \perp_q \vec{p}_2$. To show that $M', T' \models \varphi$, let $u, u' \in T'$ with the same truth values of the variables in $q$. Since $T$ and $T'$ are bisimilar, there are worlds $w, w' \in T$ such that $(w, u) \in Z$ and $(w', u') \in Z$. Since $Z$ is a modal bisimulation, the $Z$-related worlds have the same propositional truth assignment. In particular, $w$ and $w'$ agree on the values for the variables in $q$. Since $M, T \models \varphi$, there is some world $w'' \in T$ such that
  \[
  \cdot w'' \equiv_q w' \equiv_q w, \text{ and since } Z \text{ is a modal bisimulation it follows that } w'' \text{ and both } u \text{ and } u' \text{ have the same } \vec{q}\text{-assignment,}
  \]
  \[
  \cdot \begin{array}{l}
  \cdot u'' \equiv_{\vec{p}_1} w, \text{ and hence } w'' \text{ and } u \text{ have the same } \vec{p}_1\text{-assignment,} \\
  \cdot u'' \equiv_{\vec{p}_2} w', \text{ and hence } w'' \text{ and } u' \text{ have the same } \vec{p}_2\text{-assignment.}
  \end{array}
  \]

Since $T$ and $T'$ are $Z$-bisimilar, there is a world $u'' \in T'$ such that $(w'', u'') \in Z$. Since $Z$ is a modal bisimulation, $w''$ and $u''$ satisfy the same propositional variables, and hence for $u''$ we have that
  \[
  \cdot u'' \equiv_q u, u'' \equiv_q u' \]
  \[
  \cdot u'' \equiv_{\vec{p}_1} u \\
  \cdot u'' \equiv_{\vec{p}_2} u'.
  \]

Therefore, $M', T' \models \varphi$ as required.

\[\square\]

With Lemma 6.4 and an application of van Benthem’s Theorem [2], it follows directly that MIL and ML are in fact equivalent in expressiveness on singletons:

\textbf{Theorem 6.5} For each MIL-formula $\varphi_{\text{MIL}}$, there is an ML-formula $\varphi_{\text{ML}}$ such that $\varphi_{\text{MIL}}$ and $\varphi_{\text{ML}}$ are equivalent on singletons.

\textbf{Proof.} Due to Lemma 6.4, we know that MIL is invariant under bisimulation of teams. Since for singleton teams, bisimulation on teams and bisimulation on worlds coincide, it follows that MIL on singletons is invariant under modal bisimulation. Clearly, when evaluating an MIL-formula $\varphi_{\text{MIL}}$ on a singleton team $\{w\}$, all worlds in the model that have a distance from $w$ which exceeds the modal depth (i.e., maximal nesting degree of modal operators) of $\varphi$, are irrelevant for the question whether $M, \{w\} \models \varphi$. Therefore, $\varphi_{\text{MIL}}$, evaluated on singletons, captures a property of Kripke models that is invariant under modal bisimulation and only depends on the worlds that can be reached in
at most $md(\varphi_{\text{MIL}})$ steps. Due to [2], such a property can be encoded by a standard modal logic formula $\varphi_{\text{ML}}$. (The formula $\varphi_{\text{ML}}$ can be obtained, for example, as the disjunction of formulas $\varphi_{M,w}$ which, for each model $M$ and world $w$ with $M, \{w\} \models_{\text{ML}} \varphi_{\text{MIL}}$, encodes the finite tree unfolding of $M, w$ up to depth $md(\varphi_{\text{MIL}})$, up to bisimulation, and only taking into account the variables appearing in $\varphi_{\text{MIL}}$. This unfolding is finite and hence first-order definable, therefore we can apply the result from [2].)

Since the application of van Benthem’s Theorem yields a potentially very large formula, the above result does not give a “efficient” translation from MIL to ML. It turns out that one cannot do much better: MIL is exponentially more succinct than ML.

**Theorem 6.6** There is a family of MIL-formulas $(\varphi_i)_{i \in \mathbb{N}}$ such that the length of $\varphi_i$ is quadratic in $i$, and for any family of ML-formulas $(\psi_i)_{i \in \mathbb{N}}$ such that for all $i$, $\varphi_i$ and $\psi_i$ are equivalent on singletons, the length of $\psi_i$ grows exponentially in $i$.

**Proof.** Let $\varphi_i$ be the formula describing the security property of the dining cryptographers protocol in Section 5, with every sequence $\diamond \diamond$ replaced with $\diamond$. As argued in that section, if $\varphi_i$ is satisfied at $M, \{w\}$, then the number of propositional assignments appearing in the set of worlds that can be reached from $w$ in one step (two steps for the original formula) is exponential in $i$. Now let $(\psi_i)_{i \in \mathbb{N}}$ a family of ML-formulas such that $\varphi_i$ and $\psi_i$ are locally equivalent for each $i$. Since $\psi_i$ is locally equivalent to $\varphi_i$, we can without loss of generality assume that $md(\psi_i) = 1$ for all $i$ (if $\psi_i$ contains deeper nestings of modal operators, the formula can be simplified since the truth value of $\psi_i$ cannot depend on worlds reachable in 2 or more steps). Therefore, modal operators do not appear nested in $\psi_i$. Without loss of generality, we can assume that only $\diamond$ appears in $\psi$. It is clear that if $M, w \models \psi_i$, then there is a submodel of $M$ which contains $w$, and in which the number of successors of $w$ is bounded by the number of $\diamond$-operators appearing in $\psi_i$. Therefore, $\psi_i$ must have an exponential number of $\diamond$-operators, which proves the theorem.

As far as we know, the analogue of Theorem 6.6 for MDL has not yet been showed. Sevenster [13] has shown the analogue of Theorem 6.5 for MDL, but he did not show that going from MDL to ML (over singletons) inevitably leads to an exponential blow-up in the formula size. On the other hand, a closely related result showing that any formula of ML($\diamond$), where $\diamond$ is the classical disjunction) that is logically equivalent to

$$=(p_1, \ldots, p_n, q),$$

has to have length exponential in $n$ was obtained in [9].

### 6.2 Expressiveness of ML with Generalized Dependence Atoms

Theorem 6.5 applies not only to MIL, but (with the same proof) to all extensions of MIL with a dependence operator that, evaluated on a team $T$, depends only
on the set of propositional assignments that occur in some world of the team. This is because the set of assignments is clearly invariant under bisimulation. As examples, operators like those from exclusion logic or inclusion logic can be added to MIL without increasing its expressiveness. Hence, in light of Section 4, a natural question to ask is the following: For which generalized dependence atoms $D$ is the logic $ML(D)$ as expressive as $ML$ on singletons, and which atoms do in fact add expressiveness?

–It is easy to see that there are generalized dependence atoms $D$ that, even on singletons, add expressiveness beyond classical modal logic (and hence, beyond MIL). This is because with no restriction on the dependence operator $D$, one can express properties that depend on the number of worlds in a team, which clearly cannot be done in ML. As an example, consider the following:

\[ D \]

Example 6.7 Let $D$ be the relation $\{(0)\}$, and consider the models $M$ and $M'$ above, where the single variable $x$ is false in every world of both models. It is easy to see that $M, \{w\} \models \Box D(x)$, while $M', \{w'\} \not\models \Box D(x)$: For the model $M$, the set of successors of $w$ is the team $T = \{u\}$, since $x$ is false in $w$, it follows that $T(x) = (0)$, hence $T \models D(x)$. On the other hand, the set of successors of $w'$ in $M'$ is $T' = \{u_1, u_2\}$, hence $T'(x) = (0, 0)$ and $T' \not\models D(x)$.

Clearly, no ML-formula can distinguish $M, w$ and $M', w'$, since the relation $Z = \{(w, w'), (u, u_1), (u, u_2)\}$ is a bisimulation.

However, as mentioned earlier, the proof of Theorem 6.5 can be generalized to handle the generalized dependence atoms discussed earlier. In general, we obtain the following result: For a $D$ which is definable using first-order formulas without equality, the expressiveness of $ML(D)$ coincides with that on ML for singletons. For the proof, we show that $ML(D)$ remains invariant under bisimulation, and then apply the proof of Theorem 6.5 again. On the other hand, clearly if $D$ is not FO-definable, then $D$ cannot be expressed in modal logic, as modal logic can be translated to first-order logic with the standard translation, and if $D$ cannot be expressed without equality, then $D$ cannot be invariant under bisimulation. Hence we obtain the following theorem:

Theorem 6.8 Let $D$ be a generalized dependence atom. Then the following statements are equivalent:

(i) $D$ can be expressed in first-order logic without equality,

(ii) $ML(D)$ and $ML$ are equally expressive over singletons, i.e., for each $ML(D)$-formula $\varphi_{ML}$, there is an ML-formula $\varphi_{ML}$ such that $\varphi_{ML}$ and $\varphi_{ML}$ are equivalent on singletons.
Proof. We first assume that $D$ is FO-definable. As mentioned above, it suffices to adapt the proof of Lemma 6.4 to $\text{ML}(D)$. Clearly, in the induction, we only need to cover the case that $\varphi = D(p_1, \ldots, p_n)$ for propositional variables $p_1, \ldots, p_n$. Hence assume that $M, T \models D(p_1, \ldots, p_n)$, we show that $M', T' \models D(p_1, \ldots, p_n)$, where $T$ and $T'$ are $\mathcal{Z}$-bisimilar for a modal bisimulation $Z$.

Let $\phi$ be the first-order formula defining $D$. We prove the claim inductively over the formula, where we only cover the key cases explicitly. Let the free variables of $\phi$ be $\omega_1, \ldots, \omega_t$. We show that if $w_1, \ldots, w_k \in T$ and $w'_1, \ldots, w'_k \in T'$ such that $(w_i, w'_i) \in Z$ for all $i$, then $\phi(w_1, \ldots, w_k)$ evaluates to true if and only if $\phi(w'_1, \ldots, w'_k)$ does. If $\phi$ is quantifier-free, the claim is clear: Since $Z$ is a modal bisimulation, $w_i$ and $w'_i$ satisfy the same propositional variables, and due to the choice of $\phi$, the truth value of $\phi$ only depends on the propositional assignments of the worlds instantiating the variables $\omega_1, \ldots, \omega_t$. The second relevant case is when $\phi = \exists \omega \psi(\omega, \omega_1, \ldots, \omega_t)$. If $M, T \models \phi$, then there is a world $w \in T$ such that $\psi(w, w_1, \ldots, w_k)$ is true. Since $T$ and $T'$ are $\mathcal{Z}$-bisimilar, it follows that there is a world $w' \in T'$ such that $(w, w') \in Z$. Due to induction, it follows that if $\psi(w, w_1, \ldots, w_k)$ is true, then so is $\psi(w', w_1, \ldots, w_k)$. This completes the proof that $\text{ML}(D)$ and $\text{ML}$ are equally expressive over singletons.

For the converse, assume that $\text{ML}(D)$ is as expressive as $\text{ML}$ over singletons. In particular, then for every sequence $x_1, \ldots, x_n$ of variables, there is a modal formula $\varphi$ such that for every model $M$ and every world $w \in M$, we have that $M, w \models \varphi$ if and only if $M, \{w\} \models \Box D(x_1, \ldots, x_n)$. By the standard translation from modal logic to first-order logic, this implies that $D(x_1, \ldots, x_n)$ can be expressed as a FO-formula $\phi_D$.

Since whether $M, T \models D(x_1, \ldots, x_n)$ is invariant under bisimulation (this can be seen by adding a single world $w_0$, connected to all $w' \in T$ and evaluating the formula $\Box D(x_1, \ldots, x_n)$ at the singleton team $\{w_0\}$—since $\text{ML}(D)$ on singletons is equivalent to $\text{ML}$, which is invariant under bisimulation, it follows that $\text{ML}(D)$ on singletons is invariant under bisimulation as well), it follows that whether $M, T \models D(x_1, \ldots, x_n)$ is in particular invariant under adding/removing identical copies of worlds in $T$. Therefore, $\phi_D$ can be rewritten into a formula without equality.

\[\square\]

7 Conclusion and Open questions

In this paper we introduced modal independence logic $\text{MIL}$ and settled the computational complexity of its satisfiability and model checking problem. Furthermore we compared the expressivity of $\text{MIL}$ with that of classical modal logic. It turned out that most of our results can be generalized to modal logic extended with various so called generalized dependence atoms.

We end this paper by the following interesting open questions:

(i) Are there classes of frames definable in $\text{MIL}$ that cannot be defined with $\text{ML}$?

(ii) Is it possible to formulate and prove a version of van Benthem’s Theorem for our generalization of bisimilarity from the world level to the team level?
Acknowledgement
The authors thank the anonymous referees for very useful comments. The first author was supported by the Academy of Finland grants 264917 and 275241.

References