

Complexity of Polyadic Boolean Modal Logics: Model Checking and Satisfiability

Reijo Jaakkola   

Tampere University, Finland

Abstract

We study the computational complexity of model checking and satisfiability problems of polyadic modal logics extended with permutations and Boolean operators on accessibility relations. First, we show that the combined complexity of the model checking problem for the resulting logic is PTIME-complete. Secondly, we show that the satisfiability problem of polyadic modal logic extended with negation on accessibility relations is EXPTIME-complete. Finally, we show that the satisfiability problem of polyadic modal logic with permutations and Boolean operators on accessibility relations is EXPTIME-complete, under the assumption that both the number of accessibility relations that can be used and their arities are bounded by a constant. If NEXPTIME is not contained in EXPTIME, then this assumption is necessary, since already the satisfiability problem of modal logic extended with Boolean operators on accessibility relations is NEXPTIME-hard.

2012 ACM Subject Classification Theory of computation → Modal and temporal logics

Keywords and phrases Polyadic modal logics, Boolean modal logics, Model checking, Satisfiability

Digital Object Identifier 10.4230/LIPIcs.CSL.2023.26

Acknowledgements I want to thank Antti Kuusisto for suggesting this topic to me and for discussing it with me on numerous occasions. I also want to thank the anonymous reviewers for their detailed comments which greatly improved the presentation of this paper.

1 Introduction

In recent years there has been increasing interest in generalizing complexity results for logics that only have access to relation symbols of arity at most two, to logics that have access to relations of arbitrary high arity, which we will call their *polyadic* extensions, see for example [2, 12, 13, 14, 15, 18, 19, 21]. In [12] the authors introduced the uniform one-dimensional logic U_1 , which is a polyadic extension of the two-variable logic FO^2 . It was proved in [18] that the complexity of U_1 satisfiability problem is NEXPTIME-complete, which is the same as for FO^2 [11]. In the very recent work [2] the forward guarded fragment FGF was introduced, which is a polyadic extension of the description logic \mathcal{ALC} with global diamond. In the same work it was established that its satisfiability and conjunctive query entailment problems have the same complexity as the corresponding problems in the case of \mathcal{ALC} , i.e., both are EXPTIME-complete. In the two aforementioned examples the complexities of the base logic and its polyadic extension coincided, but there are also examples where the polyadic extension has a higher complexity. For instance, in [17] it is proved that the satisfiability problem of guarded U_1 , which is a polyadic extension of guarded FO^2 , is NEXPTIME-complete, while the satisfiability problem of guarded FO^2 is EXPTIME-complete [8].

While lifting complexity results from base logics to their polyadic extensions is, at least to the author, already intrinsically interesting, it also has more applied motivations stemming from, say, database theory, since polyadic relations occur naturally in various contexts and having access to them can be advantageous. For instance, a simple relation such as “Alice received a message M from Bob” is best viewed as a ternary relation. One application area where the potential need for polyadic logics has been recognized is the very active field of *description logics* [1]. Indeed, given that several description logics used in knowledge



© Reijo Jaakkola;

licensed under Creative Commons License CC-BY 4.0

31st EACSL Annual Conference on Computer Science Logic (CSL 2023).

Editors: Bartek Klin and Elaine Pimentel; Article No. 26; pp. 26:1–26:18

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

representation can only access binary roles, it should not come as a surprise that several polyadic *description logics* have been already suggested in the literature [5, 26, 29]. Finding polyadic extensions of modal logics can be seen as attempts at finding natural polyadic description logics.

To make the research around polyadic extensions more systematic, in [14] the authors outlined a research program for systematically extending complexity results for description logics – and modal logics more generally – into their polyadic counterparts, following the ideas presented in [20]. The main idea is the following: since description logics can be seen as standard modal logic extended with *relation operators* – such as inverse roles and counting – and the standard modal logic has a canonical extension into a polyadic modal logic (namely the extension in which diamonds can bind multiple formulas), one can easily obtain quite canonical extensions of known description logics¹. To demonstrate their research program in action, the polyadic extension of \mathcal{ALCQI} , i.e., the extension of \mathcal{ALC} with inverse roles and counting, was studied in [14]. The authors proved that the concept satisfiability problem for this logic is PSPACE-complete, which is the same as in the case of \mathcal{ALCQI} .

The main purpose of the present work is to contribute to the above research program in the context of Boolean modal logics, which are modal logics extended with Boolean operators on accessibility relations [23]. In these logics one can for example write down formulas such as $\langle \neg R \rangle p$, which expresses that one can reach a world in which p is true via the complement of the accessibility relation R . While being very natural modal logics as such, they are also closely related to other interesting logics such as modal logics extended with the window operator [7, 23], the two-variable logic [27] and the uniform one-dimensional fragment [20]. They have also been considered in the context of description logics [25].

Boolean modal logics present novel and intriguing technical challenges, especially when studying the complexity of their satisfiability problems. Indeed, the most common explanation for the good algorithmic properties of modal logics is that they often enjoy some variant of the tree-model property [9, 28, 30], which Boolean modal logics lack. This is a consequence of the fact that Boolean modal logics have access to *complements* of accessibility relations. To clarify this important point, we point out that in Boolean modal logics one can write statements such as $[R][\neg R]\perp \wedge [\neg R][\neg R]\perp$, which enforce that R needs to be interpreted in the Kripke models of this sentence as the total binary relation. Clearly such sentences do not have a tree-model property.

Satisfiability problems of Boolean modal logics were first studied in [23], where it was proved that modal logic with negations of accessibility relations $\text{ML}(\neg)$ has EXPTIME-complete satisfiability problem, while the complexity of full Boolean modal logic $\text{ML}(\neg, \cap)$ is NEXPTIME-complete. As a follow-up to this work, in [27] authors studied the complexity of the satisfiability problem for Boolean modal logic extended with inverses of accessibility relations and equality $\text{ML}(I, s, \neg, \cap)$, and proved that if the number of accessibility relations is bounded by a constant, then its satisfiability problem is EXPTIME-complete. Since FO^2 has the same expressive power as $\text{ML}(I, s, \neg, \cap)$ [27], the aforementioned complexity result should be contrasted with the fact that the satisfiability problem of FO^2 is NEXPTIME-hard already over unary vocabularies.

In this work we partially extend these results to the polyadic case. First, we will prove that the satisfiability problem of polyadic $\text{ML}(\neg)$, which we denote by $\text{PML}(\neg)$, has EXPTIME-complete satisfiability problem. Secondly, we will partially generalize the complexity result

¹ Of course, some conceptual work is needed to figure out what are the canonical polyadic extensions of the relevant role operators.

of [27] by proving that polyadic Boolean modal logic extended with arbitrary permutations of accessibility relations (as opposed to just inverses of binary accessibility relations), which we denote by $\text{PML}(p, s, \neg, \cap)$, has EXPTIME -complete satisfiability problem, if the number of accessibility relations in the underlying vocabulary and *their arities* are bounded by a constant. We emphasize again that, if $\text{EXPTIME} \neq \text{NEXPTIME}$, then the aforementioned assumption is necessary, since in general the satisfiability problem of $\text{PML}(p, s, \neg, \cap)$ is NEXPTIME -complete (see Proposition 3).

An important technical contribution of the present work is that we prove these two results in a *unified manner*. Originally, the complexity of $\text{ML}(\neg)$ was established using automata theory [23], while the complexity of $\text{ML}(I, s, \neg, \cap)$ over vocabularies with at most a constant number of accessibility relations was established via a poly-time reduction to the satisfiability problem of modal logic with *difference operator* over a restricted class of Kripke frames. In contrast, we establish upper bounds on the satisfiability problems of $\text{PML}(\neg)$ and $\text{PML}(p, s, \neg, \cap)$ by simply using elementary (albeit in the second case quite technical) reductions to the satisfiability problem of polyadic ML extended with global diamond $\langle \text{E} \rangle$, which we denote by $\text{PML} + \langle \text{E} \rangle$. These reductions were inspired by the reduction used in [27] to establish the complexity of $\text{ML}(I, s, \neg, \cap)$ over vocabularies with at most a constant number of accessibility relations. The overall structure of these reductions is quite robust and hence we expect that similar reductions will yield in the future several extensions of the results presented in this paper.

In addition to studying satisfiability problems, we also study the combined complexity of the model checking problem of $\text{PML}(p, s, \neg, \cap)$, which is the variant where both the model and the formula itself are received as part of the input, and prove that it is PTIME -complete, the non-trivial part being the upper bound. It is well-known that model checking problems of various modal logics with tree-model property – even quite expressive ones such as the guarded fragment – are PTIME -complete [3, 10]. However, besides these modal logics and finite variable fragments of first-order logic [10], it seems that there are not that many natural logics known for which the complexity of the model checking problem lies in PTIME . Note that the formulas of $\text{PML}(p, s, \neg, \cap)$ are neither guarded nor are they expressible in any finite variable fragment of first-order logic. Thus, even though proving that the model checking problem of $\text{PML}(p, s, \neg, \cap)$ is in PTIME turns out to be quite straightforward, we still believe that the result is interesting since the logic $\text{PML}(p, s, \neg, \cap)$ is quite distinct from other logics known in the literature with PTIME -complete combined complexity. Indeed, it would be interesting to understand how much $\text{PML}(p, s, \neg, \cap)$ could be extended while keeping its combined complexity feasible.

The structure of this paper is as follows. First, in Section 2 we will formally define the logics that are studied in this paper. Then, in Sections 3 and 4 we prove that the satisfiability problems of $\text{PML}(\neg)$ and $\text{PML}(p, s, \neg, \cap)$ respectively are EXPTIME -complete, the latter under the assumption that the number of accessibility relations and their arities are bounded by a constant, which, as pointed out earlier, is necessary if $\text{EXPTIME} \neq \text{NEXPTIME}$. Finally, in Section 5 we prove that the combined complexity of $\text{PML}(p, s, \neg, \cap)$ is PTIME -complete.

2 Preliminaries

In this paper we will only consider relation symbols of arity at least two. Given a relation symbol R , we will use $\text{ar}(R)$ to denote its arity. If τ is a set of relation symbols and Φ is a set of propositional symbols, then a *Kripke-model* over (τ, Φ) is a tuple $\mathfrak{M} = (W, (R^{\mathfrak{M}})_{R \in \tau}, V)$, where

26:4 Complexity of Polyadic Boolean Modal Logics

1. W is a non-empty set (the set of possible worlds),
2. for every $R \in \tau$ we have that $R^{\mathfrak{M}} \subseteq W^{ar(R)}$ and
3. $V : \Phi \rightarrow \mathcal{P}(W)$.

Members of τ are also called *accessibility relations*. In what follows we will never specify explicitly what the underlying set of propositional symbols is and we will only occasionally specify the set of accessibility relations.

In this paper we consider extensions of standard polyadic (multimodal) modal logic via *relation operators*, which are essentially mappings that map relational structures to relational structures and which are invariant under isomorphisms. Since we will focus our attention only on a specific set of relation operators, we will omit the formal definition of a relation operator here, which can be found in [16].

The main relation operators that we are going to consider are p, s, \neg, \cap , where p and s are called *cyclic permutation* and *swap permutation* respectively. Furthermore, for technical reasons we are going to need two additional relation operators \setminus, \cup . Let τ denote a set of relation symbols. Given $k \geq 2$, the set of *k-ary terms* $\text{GRA}_k(p, s, \neg, \cap, \setminus, \cup)[\tau]$ is generated by the following grammar

$$\mathcal{R} ::= R \mid p\mathcal{R} \mid s\mathcal{R} \mid \neg\mathcal{R} \mid \mathcal{R} \cap \mathcal{R} \mid \mathcal{R} \setminus \mathcal{R} \mid \mathcal{R} \cup \mathcal{R},$$

where $R \in \tau$ is a k -ary relation. We use $\text{GRA}(p, s, \neg, \cap, \setminus, \cup)[\tau]$ to denote

$$\bigcup_{k \geq 2} \text{GRA}_k(p, s, \neg, \cap, \setminus, \cup)[\tau],$$

which is the set of all terms. Arity of a term \mathcal{R} is denoted by $ar(\mathcal{R})$.

Let \mathfrak{M} be a Kripke model over τ and let $\mathcal{R} \in \text{GRA}(p, s, \neg, \cap, \setminus, \cup)$ be a k -ary term. We define the *interpretation* of \mathcal{R} over \mathfrak{M} recursively as follows.

1. If $\mathcal{R} = R \in \tau$, then we define $\llbracket \mathcal{R} \rrbracket_{\mathfrak{M}} = R^{\mathfrak{M}}$.
2. If $\mathcal{R} = p\mathcal{R}'$, then we define

$$\llbracket \mathcal{R} \rrbracket_{\mathfrak{M}} = \{(a_k, a_1, \dots, a_{k-1}) \in W^k \mid (a_1, \dots, a_k) \in \llbracket \mathcal{R}' \rrbracket_{\mathfrak{M}}\}$$

3. If $\mathcal{R} = s\mathcal{R}'$, then we define

$$\llbracket \mathcal{R} \rrbracket_{\mathfrak{M}} = \{(a_1, \dots, a_{k-2}, a_k, a_{k-1}) \in W^k \mid (a_1, \dots, a_k) \in \llbracket \mathcal{R}' \rrbracket_{\mathfrak{M}}\}$$

4. If $\mathcal{R} = \neg\mathcal{R}'$, then we define $\llbracket \mathcal{R} \rrbracket_{\mathfrak{M}} = W^k \setminus \llbracket \mathcal{R}' \rrbracket_{\mathfrak{M}}$
5. If $\mathcal{R} = \mathcal{R}' \cap \mathcal{R}''$, then we define $\llbracket \mathcal{R} \rrbracket_{\mathfrak{M}} = \llbracket \mathcal{R}' \rrbracket_{\mathfrak{M}} \cap \llbracket \mathcal{R}'' \rrbracket_{\mathfrak{M}}$
6. If $\mathcal{R} = \mathcal{R}' \setminus \mathcal{R}''$, then we define $\llbracket \mathcal{R} \rrbracket_{\mathfrak{M}} = \llbracket \mathcal{R}' \rrbracket_{\mathfrak{M}} \setminus \llbracket \mathcal{R}'' \rrbracket_{\mathfrak{M}}$
7. If $\mathcal{R} = \mathcal{R}' \cup \mathcal{R}''$, then we define $\llbracket \mathcal{R} \rrbracket_{\mathfrak{M}} = \llbracket \mathcal{R}' \rrbracket_{\mathfrak{M}} \cup \llbracket \mathcal{R}'' \rrbracket_{\mathfrak{M}}$

Given $k \geq 2$, we let S_k denote the set of all bijections $\{1, \dots, k\} \rightarrow \{1, \dots, k\}$. It is a well-known group theoretic fact that every permutation $\sigma \in S_k$ can be obtained by composing cyclic permutation with swap permutation. Given this, we will adopt the notational convention that we will use $\sigma \in S_k$ to denote some fixed (possibly empty) sequence consisting of operators p and s that generates it. The only explicit requirement that we impose on this sequence is that it should not be possible to rewrite it into a smaller one. Thus, for example, we use the identity permutation to denote the empty sequence.

The following lemma collects some elementary algebraic identities for terms.

► **Lemma 1.** Let $\mathcal{R}_1, \mathcal{R}_2 \in \text{GRA}(p, s, \neg, \cap, \setminus, \cup)[\tau]$ be k -ary terms and let $\sigma \in S_k$. Let \mathfrak{M} be a Kripke model over τ .

1. $\llbracket \neg \neg \mathcal{R}_1 \rrbracket_{\mathfrak{M}} = \llbracket \mathcal{R}_1 \rrbracket_{\mathfrak{M}}$
2. $\llbracket \sigma \neg \mathcal{R}_1 \rrbracket_{\mathfrak{M}} = \llbracket \neg \sigma \mathcal{R}_1 \rrbracket_{\mathfrak{M}}$
3. $\llbracket \sigma(\mathcal{R}_1 \cap \mathcal{R}_2) \rrbracket_{\mathfrak{M}} = \llbracket (\sigma \mathcal{R}_1 \cap \sigma \mathcal{R}_2) \rrbracket_{\mathfrak{M}}$
4. $\llbracket \sigma(\mathcal{R}_1 \cup \mathcal{R}_2) \rrbracket_{\mathfrak{M}} = \llbracket (\sigma \mathcal{R}_1 \cup \sigma \mathcal{R}_2) \rrbracket_{\mathfrak{M}}$
5. $\llbracket \neg(\mathcal{R}_1 \cap \mathcal{R}_2) \rrbracket_{\mathfrak{M}} = \llbracket (\neg \mathcal{R}_1 \cup \neg \mathcal{R}_2) \rrbracket_{\mathfrak{M}}$
6. $\llbracket \neg(\mathcal{R}_1 \cup \mathcal{R}_2) \rrbracket_{\mathfrak{M}} = \llbracket (\neg \mathcal{R}_1 \cap \neg \mathcal{R}_2) \rrbracket_{\mathfrak{M}}$
7. $\llbracket (\mathcal{R}_1 \cap \neg \mathcal{R}_2) \rrbracket_{\mathfrak{M}} = \llbracket (\mathcal{R}_1 \setminus \mathcal{R}_2) \rrbracket_{\mathfrak{M}}$
8. $\llbracket (\mathcal{R}_1 \cup \neg \mathcal{R}_2) \rrbracket_{\mathfrak{M}} = \llbracket \neg(\mathcal{R}_2 \setminus \mathcal{R}_1) \rrbracket_{\mathfrak{M}}$

Let $\mathcal{R} \in \text{GRA}(p, s, \neg)[\tau]$ be a k -ary term. We say that \mathcal{R} is a k -literal over τ , if it is either of the form $\neg \sigma R$ or σR , for a k -ary relation symbol $R \in \tau$. A maximally consistent set ρ of k -literals over σ is called a k -table over τ . We identify tables ρ with terms $\bigcap_{\alpha \in \rho} \alpha$.

Given a Kripke model \mathfrak{M} over τ and $(w_1, \dots, w_k) \in W^k$ we say that (w_1, \dots, w_k) realizes a k -table ρ , if for every k -literal α we have that

$$(w_1, \dots, w_k) \in \llbracket \alpha \rrbracket_{\mathfrak{M}} \Leftrightarrow \alpha \in \rho.$$

Note that since tables are maximally consistent, each tuple in a given Kripke model realizes a unique table. Conversely, the accessibility relations of a Kripke model can be described completely by specifying what tables different tuples realize.

Given a k -table ρ and $\sigma \in S_k$ we define

$$\sigma[\rho] := \bigcap_{\sigma' R \in \rho} (\sigma \circ \sigma') R \cap \bigcap_{\neg \sigma' R \in \rho} \neg(\sigma \circ \sigma') R.$$

Note that for every k -table ρ and $\sigma_1, \sigma_2 \in S_k$ we have that $\sigma_1[\sigma_2[\rho]] = (\sigma_1 \circ \sigma_2)[\rho]$. Furthermore, we note that it can be the case that $\sigma[\rho] = \rho$, even if σ is not the identity permutation.²

Let Φ be a set of propositional symbols, τ a set of relation symbols and $\mathcal{F} \subseteq \{p, s, \neg, \cap, \setminus, \cup\}$. The set of formulas $\text{PML}(\mathcal{F})[\tau, \Phi]$ is generated by the following grammar

$$\varphi ::= p \mid \neg \varphi \mid (\varphi \wedge \varphi) \mid \langle \mathcal{R} \rangle \underbrace{(\varphi, \dots, \varphi)}_{k\text{-times}}$$

where $p \in \Phi$ and $\mathcal{R} \in \text{GRA}(\mathcal{F})[\tau]$ is a $(k+1)$ -ary term. We will use $\text{PML}(\mathcal{F})[\tau, \Phi] + \langle \text{E} \rangle$ to denote the set of formulas generated by the grammar of $\text{PML}(\mathcal{F})[\tau, \Phi]$ extended with the rule

$$\varphi ::= \langle \text{E} \rangle \varphi$$

If τ contains only binary relation symbols, then we emphasize this by writing $\text{ML}(\mathcal{F})[\tau, \Phi]$.

Given a natural number c a vocabulary τ is called c -bounded, if $|\tau| \leq c$ and for every $R \in \tau$ we have that $\text{ar}(R) \leq c$. For example, the empty vocabulary is 0-bounded and a binary vocabulary with five relation symbols is 5-bounded. If we are only considering c -bounded vocabularies, then we emphasize this by writing PML_c . Thus $\text{PML}_c[\tau, \Phi]$ entails that τ is c -bounded. We emphasize that c should be thought of as a fixed constant.

² Consider for example the permutation s and the 2-table $\{R, sR\}$ over $\{R\}$, where R is a binary relation.

26:6 Complexity of Polyadic Boolean Modal Logics

As in the case of Kripke models, for the rest of this paper we will never explicitly mention the underlying set of propositional symbols. We will use standard shorthand notations: $\varphi \vee \psi := \neg(\neg\varphi \wedge \neg\psi)$, $[\mathcal{R}] := \neg\langle\mathcal{R}\rangle\neg$ and $\langle A \rangle = \neg\langle E \rangle\neg$. Given a formula φ we let $\text{Subf}(\varphi)$ denote the set of subformulas of φ .

The semantics of our logics are standard and we will only recall here the semantic clauses of $\langle\mathcal{R}\rangle(\psi_1, \dots, \psi_k)$ and $\langle E \rangle\psi$. Given a Kripke model \mathfrak{M} over τ and $w \in W$ we define

$$\mathfrak{M}, w \Vdash \langle\mathcal{R}\rangle(\psi_1, \dots, \psi_k) \Leftrightarrow$$

there exists $(w, w_1, \dots, w_k) \in \llbracket\mathcal{R}\rrbracket_{\mathfrak{M}}$ such that $\mathfrak{M}, w_\ell \Vdash \psi_\ell$, for every $1 \leq \ell \leq k$

and

$$\mathfrak{M}, w \Vdash \langle E \rangle\psi \Leftrightarrow \text{there exists some } w' \in W \text{ such that } \mathfrak{M}, w' \Vdash \psi.$$

We will conclude this section by stating the complexity of the satisfiability problem of $\text{PML} + \langle E \rangle$.

► **Proposition 2.** *The satisfiability problem of $\text{PML} + \langle E \rangle$ is EXPTIME-complete.*

Proof. Lower bound follows from the well-known fact that the satisfiability problem of $\text{ML} + \langle E \rangle$ is EXPTIME-complete [4, Exercise 6.8.1]. Upper bound follows, say, from the recent result that the satisfiability problem of the so-called *guarded forward fragment* FGF is EXPTIME-complete [2]. Indeed, by using variables carefully, one can guarantee that the standard translation of $\text{PML} + \langle E \rangle$ into FO produces sentences of FGF. ◀

As mentioned in the introduction, existing results in the literature imply that the satisfiability problem of $\text{PML}(p, s, \neg, \cap)$ is NEXPTIME-complete.

► **Proposition 3.** *The satisfiability problem of $\text{PML}(p, s, \neg, \cap)$ is NEXPTIME-complete.*

Proof. Lower bound follows from the fact that the satisfiability problem of $\text{ML}(\neg, \cap)$ is already NEXPTIME-hard [24]. For upper bound we note that $\text{PML}(p, s, \neg, \cap)$ is contained in U_1 [20, Theorem 7], for which the satisfiability problem is NEXPTIME-complete [18]. ◀

3 Satisfiability problem of polyadic modal logic with negation

In this section we will show how the satisfiability problem of $\text{PML}(\neg)$ can be reduced in polynomial time to that of $\text{PML} + \langle E \rangle$, which will yield the following result.

► **Theorem 4.** *The satisfiability problem of $\text{PML}(\neg)$ is EXPTIME-complete.*

It seems most probable that the above complexity result continues to hold for $\text{PML}(p, s, \neg)$, but we have not yet been able to show this.

Before presenting the reduction from $\text{PML}(\neg)$ to $\text{PML} + \langle E \rangle$, we first describe one brief application of Theorem 4, which reflects one original motivation for the study of Boolean modal logics [24]. Namely, we show that the logic PML extended with *polyadic window operator*, which we denote by $\text{PML} + \nabla$, has an EXPTIME-complete satisfiability problem. Consider a vocabulary τ . The grammar for generating the formulas of $\text{PML}[\tau] + \nabla$ is the grammar of $\text{PML}[\tau]$ extended with the rule

$$\varphi ::= \nabla_R(\underbrace{\varphi, \dots, \varphi}_{k\text{-times}}),$$

where R is a $(k+1)$ -ary accessibility relation, for every $R \in \tau$. The semantics of formulas of the form $\nabla_R(\psi_1, \dots, \psi_k)$ is defined as follows:

$\mathfrak{M}, w \Vdash \nabla_R(\psi_1, \dots, \psi_k) \Leftrightarrow$ For every $(w_1, \dots, w_k) \in W^k$ we have that if

$\mathfrak{M}, w_\ell \Vdash \psi_\ell$, for every $1 \leq \ell \leq k$, then $(w, w_1, \dots, w_k) \in R^{\mathfrak{M}}$.

It is easy to see that $\mathfrak{M}, w \Vdash \nabla_R(\psi_1, \dots, \psi_k)$ is equivalent with

$\mathfrak{M}, w \Vdash [\neg R](\neg\psi_1, \dots, \neg\psi_k)$

and hence Theorem 4 immediately implies the following complexity results.

► **Corollary 5.** *The satisfiability problem of $\text{PML} + \nabla$ is EXPTIME -complete.*

Now we will present the reduction. Fix a formula $\varphi \in \text{PML}(\neg)$ and let τ denote the set of accessibility relations occurring in φ . For every symbol $R \in \tau$, we will introduce two fresh symbols of the same arity, R_1 and R_2 . Given $\psi \in \text{Subf}(\varphi)$, we let $t(\psi)$ denote the formula obtained from ψ by replacing each $\langle R \rangle$ with $\langle R_1 \rangle$ and each $\langle \neg R \rangle$ with $\langle R_2 \rangle$. Consider then the following formula $\theta := t(\varphi) \wedge \eta$, where

$$\begin{aligned} \eta := & \bigwedge_{\substack{\langle R_1 \rangle(\psi_1, \dots, \psi_k), \\ \langle R_2 \rangle(\chi_1, \dots, \chi_k) \\ \in \text{Subf}(t(\varphi))}} \left(\langle E \rangle(\neg\langle R_1 \rangle(\psi_1, \dots, \psi_k) \wedge \neg\langle R_2 \rangle(\chi_1, \dots, \chi_k)) \right) \\ \rightarrow & \bigvee_{1 \leq \ell \leq k} \langle A \rangle(\neg\psi_\ell \vee \neg\chi_\ell). \end{aligned}$$

Intuitively speaking, in every model of η we can extend the interpretations of R_1 and R_2 , for $R \in \tau$, in such a way that they cover $W^{\text{ar}(R)}$, i.e., every tuple of length $\text{ar}(R)$ belongs either to the interpretation of R_1 or to the interpretation of R_2 , while maintaining that the resulting model is a model of $t(\varphi)$, if the original model was.

Since the big conjunction in η ranges over only those formulas that occur as subformulas in φ , the size of η is $O(|\varphi|^2)$, i.e., polynomial with respect to $|\varphi|$. The rest of this section is devoted to proving that the above reduction is correct, i.e., φ is satisfiable iff θ is. We will start with the left to right direction.

► **Lemma 6.** *If φ is satisfiable, then so is θ .*

Proof. Let $\mathfrak{M} = (W, (R)_{R \in \tau}, V)$ be a Kripke model and let $w \in W$ be a world so that $\mathfrak{M}, w \Vdash \varphi$. We then define the Kripke model $\mathfrak{N} = (W, (R_1)_{R \in \tau}, (R_2)_{R \in \tau}, V)$ by setting that for every $R \in \tau$, $R_1^{\mathfrak{N}} = R^{\mathfrak{M}}$ and $R_2^{\mathfrak{N}} = W^{\text{ar}(R)} \setminus R^{\mathfrak{M}}$. Clearly $\mathfrak{N}, w \Vdash t(\varphi)$. To verify that \mathfrak{N} satisfies η , suppose that there exists $\langle R \rangle(\psi, \dots, \psi_k), \langle \neg R \rangle(\chi_1, \dots, \chi_k) \in \text{Subf}(t(\varphi))$ and $w_0 \in W$ so that

$\mathfrak{N}, w_0 \Vdash \neg\langle R_1 \rangle(\psi_1, \dots, \psi_k) \wedge \neg\langle R_2 \rangle(\chi_1, \dots, \chi_k)$

but

$\mathfrak{N}, w \not\Vdash \bigvee_{1 \leq \ell \leq k} \langle A \rangle(\neg\psi_\ell \vee \neg\chi_\ell)$.

Thus for every $1 \leq \ell \leq k$ there exists $w_\ell \in W$ so that $\mathfrak{N}, w_\ell \Vdash \psi_\ell \wedge \chi_\ell$. By construction, we must either have that $(w_0, w_1, \dots, w_k) \in R_1^{\mathfrak{N}}$ or $(w_0, w_1, \dots, w_k) \in R_2^{\mathfrak{N}}$, but clearly both of these cases lead to a contradiction. ◀

26:8 Complexity of Polyadic Boolean Modal Logics

Suppose then that θ is satisfiable. \mathfrak{M} be a Kripke model and let $w \in W$ be a world so that $\mathfrak{M}, w \Vdash \theta$.

► **Lemma 7.** *For every $(w_0, w_1, \dots, w_k) \in W^{k+1}$ and $R \in \tau$ there exists $i \in \{1, 2\}$ so that for every $\langle R_i \rangle(\psi_1, \dots, \psi_k) \in \text{Subf}(t(\varphi))$ we have that if $\mathfrak{M}, w_\ell \Vdash \psi_\ell$, for every $1 \leq \ell \leq k$, then $\mathfrak{M}, w_0 \Vdash \langle R_i \rangle(\psi_1, \dots, \psi_k)$.*

Proof. Suppose that this is not the case. Thus there exists

$$\langle R_1 \rangle(\psi_1, \dots, \psi_k), \langle R_2 \rangle(\chi_1, \dots, \chi_k) \in \text{Subf}(t(\varphi))$$

so that $\mathfrak{M}, w_\ell \Vdash \psi_\ell \wedge \chi_\ell$, for every $1 \leq \ell \leq k$, but

$$\mathfrak{M}, w_0 \Vdash \neg \langle R_1 \rangle(\psi_1, \dots, \psi_k) \wedge \neg \langle R_2 \rangle(\chi_1, \dots, \chi_k).$$

Since $\mathfrak{M}, w \Vdash \eta$, we have that

$$\mathfrak{M}, w \Vdash \bigvee_{1 \leq \ell \leq k} \langle A \rangle(\neg \psi_\ell \vee \neg \chi_\ell),$$

which is a clear contradiction. ◀

Using Lemma 7, we can extend the model \mathfrak{M} as follows. For every $(w_1, \dots, w_k) \notin R_1^{\mathfrak{M}} \cup R_2^{\mathfrak{M}}$, we choose $i \in \{1, 2\}$ with the properties described in Lemma 7, and add (w_1, \dots, w_k) to $R_i^{\mathfrak{M}}$. We still use \mathfrak{M} to denote the resulting model. We emphasize that \mathfrak{M} has now the property that for every $(w_1, \dots, w_k) \in W^k$ and $R \in \tau$, either $(w_1, \dots, w_k) \in R_1^{\mathfrak{M}}$ or $(w_1, \dots, w_k) \in R_2^{\mathfrak{M}}$.

► **Lemma 8.** $\mathfrak{M}, w \Vdash t(\varphi)$

Proof. A routine induction. ◀

We now define a Kripke model $\mathfrak{N} = (W^*, (R^{\mathfrak{N}})_{R \in \tau}, V^*)$ over τ as follows. First, we specify that $W^* := W \times \{0, 1\}$ and that for every $(w, i) \in W^*$ we have that $(w, i) \in V^*(p)$ iff $w \in V(p)$. Next we need to define interpretations of relation symbols $R \in \tau$. Fix such a relation symbol R . For every $(w_0, i) \in W^*$ we define that if $(w_0, w_1, \dots, w_k) \in R_1^{\mathfrak{M}}$, then

$$((w_0, i), (w_1, i + 1 \bmod 2), \dots, (w_k, i + 1 \bmod 2)) \in R^{\mathfrak{N}}.$$

Then, for every $(w_0, w_1, \dots, w_k) \in R_2^{\mathfrak{M}}$ we define that

$$((w_0, i), (w_1, i), \dots, (w_k, i)) \notin R^{\mathfrak{N}}.$$

Finally, for every $((w_0, i_0), \dots, (w_k, i_k))$ for which we have not specified whether they belong to $R^{\mathfrak{N}}$, we define that if $(w_0, \dots, w_k) \notin R_2^{\mathfrak{M}}$ then $((w_0, i_0), \dots, (w_k, i_k)) \in R^{\mathfrak{N}}$.

► **Lemma 9.** *For every $\psi \in \text{Subf}(\varphi)$ and $w_0 \in W$ we have that*

$$\mathfrak{N}, (w_0, i_0) \Vdash \psi \Leftrightarrow \mathfrak{M}, w_0 \Vdash t(\psi).$$

Proof. A routine induction. ◀

In particular \mathfrak{N} is a model of φ , since $\mathfrak{N}, (w, 0) \Vdash \varphi$, and hence φ is satisfiable. Thus we can conclude that φ is satisfiable iff θ is.

4 Satisfiability problem of polyadic Boolean modal logic with permutations over bounded vocabularies

Recall that $\text{PML}_c(p, s, \neg, \cap)$ denotes the restriction of $\text{PML}(p, s, \neg, \cap)$ where we consider only c -bounded vocabularies, c being a natural number which should be thought of as a fixed constant. In this section we will give a polynomial time reduction from the satisfiability problem of $\text{PML}_c(p, s, \neg, \cap)$ to that of $\text{PML} + \langle E \rangle$, which will yield the following result.

► **Theorem 10.** *The satisfiability problem of $\text{PML}_c(p, s, \neg, \cap)$ is EXPTIME -complete.*

The reduction used in the proof of Theorem 10 is very similar to the reduction that was used in the previous section to prove Theorem 4. The reader is encouraged to keep this in mind when parsing the reduction, since even though the underlying ideas are again elementary, the resulting reduction is quite technical.

An important property of $\text{PML}(\neg)$ is that it can not speak about intersections of accessibility relations. In the case of $\text{PML}(p, s, \neg, \cap)$ this is obviously no longer the case, but it is still possible to convert each sentence of $\text{PML}(p, s, \neg, \cap)$ into an equi-satisfiable sentence with an analogous property. If the number of underlying accessibility relations and their arities are bounded by some constant, then this translation can also be carried out in polynomial time.

► **Lemma 11.** *Let $\varphi \in \text{PML}_c(p, s, \neg, \cap)[\tau]$ be a formula. Then we can transform φ in polynomial time to a formula $\varphi^* \in \text{PML}_c(p, s, \neg, \cap)[\tau]$, which has the following properties.*

1. φ is satisfiable if and only if φ^* is.
2. For every $\langle \mathcal{R} \rangle(\psi_1, \dots, \psi_k) \in \text{Subf}(\varphi^*)$ the term \mathcal{R} is a k -table over τ .

Proof. Note that since τ is c -bounded, the number of tables over τ is bounded by a constant. By applying repeatedly Lemma 1, we can assume that in every formula $\langle \mathcal{R} \rangle(\psi_1, \dots, \psi_k) \in \text{Subf}(\varphi)$ the term \mathcal{R} is a boolean combination of $(k + 1)$ -literals over τ . Pick an innermost such subformula of φ . The term \mathcal{R} is clearly equivalent with the term

$$\bigcup_{\rho \models \mathcal{R}} \rho,$$

where each ρ is a $(k + 1)$ -table over τ . Let $p_{\psi_1}, \dots, p_{\psi_k}$ denote fresh propositional symbols. In φ we replace $\langle \mathcal{R} \rangle(\psi_1, \dots, \psi_k)$ with the following formula

$$\bigvee_{\rho \models \mathcal{R}} \langle \rho \rangle(p_{\psi_1}, \dots, p_{\psi_k}).$$

Note the above formula is essentially of constant size, since the number of tables over τ is bounded by a constant. Now, let φ' denote the resulting formula. Without loss of generality we will assume that τ contains at least one binary relation symbol. With this technical assumption it is clear that φ is equisatisfiable with the formula

$$\varphi' \wedge \bigwedge_{1 \leq \ell \leq k} \bigwedge_{\text{2-table } \rho} [\rho](p_{\psi_\ell} \leftrightarrow \psi_\ell).$$

Since the size of τ is bounded by a constant, the number of 2-tables over τ is also bounded by a constant. Thus the size of the above formula is polynomial with respect to the size of the original formula φ . It should be clear that by repeating the above procedure sufficiently many times, we will eventually reach the desired formula. ◀

26:10 Complexity of Polyadic Boolean Modal Logics

► **Remark 12.** In Lemma 11 it is not enough to assume that there is a constant bound on the *arities* of relations in τ to guarantee that the translation is efficient, i.e., polynomial time computable, since already in the case of binary vocabularies there are exponentially many tables. Likewise, it is not enough to assume that only the size of τ is bounded by a constant, since the number of tables over a vocabulary which consists of a single relation R of arity n is bounded from below by 2^n .

For the rest of this section we will assume that φ satisfies property (ii) of Lemma 11. Now, for $2 \leq k \leq m$, where $m = \max\{ar(R) \mid R \in \tau\}$, let \mathfrak{T}_k denote the set of all tables over τ . Each k -table $\rho \in \mathfrak{T}_k$ can be seen as a k -ary accessibility relation and hence we will consider the vocabulary $\tau_{\mathfrak{T}} := \bigcup_{2 \leq k \leq m} \mathfrak{T}_k$. We let $t(\varphi)$ denote the sentence in $\text{PML}_c[\tau_{\mathfrak{T}}]$ which is obtained from φ by replacing each table $\rho \in \text{GRA}(p, s, \neg, \cap)[\tau]$ with the corresponding relation symbol $\rho \in \tau_{\mathfrak{T}}$.

We next describe sentences of $\text{PML}[\tau_{\mathfrak{T}}] + \langle \text{E} \rangle$ which together play the same role that η did in the previous section. We start with the following sentence, which we denoted by ξ_1 .

$$\begin{aligned} \xi_1 := \langle \text{A} \rangle & \bigwedge_{1 \leq k < m} \bigwedge_{\substack{g: \mathfrak{T}_{k+1} \rightarrow S_{k+1} \\ \sigma_\rho := g(\rho)}} \bigwedge_{\substack{h: \mathfrak{T}_{k+1} \rightarrow (\text{Subf}(t(\varphi)))^k \\ h(\rho) = (\psi_1^{\sigma_\rho}, \dots, \psi_k^{\sigma_\rho})}} \left(\left(\bigwedge_{\rho \in X_0} \neg \langle \sigma_\rho[\rho] \rangle (\psi_1^{\sigma_\rho}, \dots, \psi_k^{\sigma_\rho}) \right) \right. \\ & \left. \wedge \bigwedge_{1 \leq \ell \leq k} \langle \text{E} \rangle \left(\bigwedge_{\rho \in X_\ell} \neg \langle \sigma_\rho[\rho] \rangle (\psi_1^{\sigma_\rho}, \dots, \psi_k^{\sigma_\rho}) \wedge \bigwedge_{\rho \notin X_\ell} \psi_{\sigma_\rho^{-1}(\ell)}^{\sigma_\rho} \right) \right) \rightarrow \bigvee_{\rho \notin X_0} \neg \psi_{\sigma_\rho^{-1}(0)}^{\sigma_\rho} \end{aligned}$$

In the above sentence we use X_ℓ to denote the set $\{\rho \mid \sigma_\rho(0) = \ell\}$. Note that since τ is c -bounded, ξ_1 is only of size at most polynomial with respect to the size of φ .

In addition to ξ_1 , we will need the following sentences, which will be denoted by ξ_2 and ξ_3 respectively.

$$\begin{aligned} \xi_2 := \langle \text{A} \rangle & \bigwedge_{1 \leq k < m} \bigwedge_{\rho \in \mathfrak{T}_{k+1}} \bigwedge_{\substack{\sigma \in S_{k+1} \\ \sigma(0) \neq 0}} \bigwedge_{\psi_1, \dots, \psi_k \in \text{Subf}(t(\varphi))} \\ & \left(\neg \psi_{\sigma^{-1}(0)} \vee \neg \langle \rho \rangle (\neg \psi_{\sigma^{-1}(1)}, \dots, \underbrace{\langle \sigma[\rho] \rangle (\psi_1, \dots, \psi_k)}_{\sigma(0): \text{th formula}}, \dots, \neg \psi_{\sigma^{-1}(k)}) \right) \\ \xi_3 := \langle \text{A} \rangle & \bigwedge_{1 \leq k < m} \bigwedge_{\rho \in \mathfrak{T}_{k+1}} \bigwedge_{\substack{\sigma \in S_{k+1} \\ \sigma(0) = 0}} \bigwedge_{\psi_1, \dots, \psi_k \in \text{Subf}(t(\varphi))} \\ & \left(\langle \rho \rangle (\psi_{\sigma^{-1}(1)}, \dots, \psi_{\sigma^{-1}(k)}) \rightarrow \langle \sigma[\rho] \rangle (\psi_1, \dots, \psi_k) \right) \end{aligned}$$

Again, since τ is c -bounded, ξ_1 and ξ_2 are only of size at most polynomial with respect to the size of φ . We let $\Theta := t(\varphi) \wedge \xi_2 \wedge \xi_3$.

The sentences ξ_1, ξ_2, ξ_3 might look rather complicated, but we emphasize again that they are simply playing essentially the same role that η did in the previous section. Namely, they axiomatize enough properties of tables so that in any model of Θ we can enlarge the interpretations of the accessibility relations ρ in such a way that they cover all the tuples of relevant length, while maintaining that the resulting model is still a model of $t(\varphi)$.

The rest of this section is devoted towards proving that φ is satisfiable iff Θ . We start with the easy direction.

► **Lemma 13.** *If φ is satisfiable, then so is Θ .*

Proof. Let $\mathfrak{M} = (W, (R^{\mathfrak{M}})_{R \in \tau}, V)$ be a Kripke model and let $w \in W$ be a world so that $\mathfrak{M}, w \Vdash \varphi$. We then define the Kripke model $\mathfrak{N} = (W, (\rho^{\mathfrak{N}})_{\rho \in \tau_{\mathfrak{N}}}, V)$ by setting that for every $\rho \in \tau_{\mathfrak{N}}$, $\rho^{\mathfrak{N}} = \llbracket \rho \rrbracket_{\mathfrak{M}}$. Clearly $\mathfrak{N}, w \Vdash t(\varphi)$.

Let us then verify that $\mathfrak{N}, w_0 \Vdash \xi_1$, for any $w_0 \in W$. Fix k, g and h . Suppose that

$$\mathfrak{N}, w_0 \Vdash \bigwedge_{\rho \in X_0} \neg \langle \sigma_{\rho}[\rho] \rangle (\psi_1^{\sigma_{\rho}}, \dots, \psi_k^{\sigma_{\rho}})$$

and that for every $1 \leq \ell \leq k$ there exists w_{ℓ} so that

$$\mathfrak{N}, w_{\ell} \Vdash \bigwedge_{\rho \in X_{\ell}} \neg \langle \sigma_{\rho}[\rho] \rangle (\psi_1^{\sigma_{\rho}}, \dots, \psi_k^{\sigma_{\rho}}) \wedge \bigwedge_{\rho \notin X_{\ell}} \psi_{\sigma_{\rho}^{-1}(\ell)}^{\sigma_{\rho}}.$$

Our goal is to show that

$$\mathfrak{N}, w_0 \Vdash \bigvee_{\rho \notin X_0} \neg \psi_{\sigma_{\rho}^{-1}(0)}^{\sigma_{\rho}}.$$

Aiming for a contradiction, suppose that this is not the case. Since every tuple realizes a table in \mathfrak{M} , there exists $\rho \in \tau_{\mathfrak{N}}$ so that $(w_0, w_1, \dots, w_k) \in \rho^{\mathfrak{N}}$. Recall that the function g associates a permutation σ_{ρ} with ρ . Now $(w_{\sigma_{\rho}(0)}, \dots, w_{\sigma_{\rho}(k)}) \in \llbracket \sigma_{\rho} \rho \rrbracket_{\mathfrak{M}}$. Let $\ell_0 := \sigma_{\rho}(0)$. Then, for every $\ell \neq \ell_0$ we have that

$$\mathfrak{N}, w_{\ell} \Vdash \psi_{\sigma_{\rho}^{-1}(\ell)}^{\sigma_{\rho}},$$

since $\rho \notin X_{\ell}$. This implies that

$$\mathfrak{N}, w_{\sigma_{\rho}(\ell)} \Vdash \psi_{\ell}^{\sigma_{\rho}},$$

for every $1 \leq \ell \leq k$. On the other hand

$$\mathfrak{N}, w_{\sigma_{\rho}(0)} \Vdash \neg \langle \sigma_{\rho}[\rho] \rangle (\psi_1^{\sigma_{\rho}}, \dots, \psi_k^{\sigma_{\rho}}),$$

which is a contradiction, since $\sigma_{\rho}[\rho]^{\mathfrak{N}} = \llbracket \sigma_{\rho} \rho \rrbracket_{\mathfrak{M}}$.

Next, we will verify that $\mathfrak{N}, w_0 \Vdash \xi_2$, for any $w_0 \in W$. Fix k, σ and $\psi_1, \dots, \psi_k \in \text{Subf}(t(\varphi))$. Aiming for a contradiction, suppose that

$$\mathfrak{N}, w_0 \Vdash \left(\psi_{\sigma^{-1}(0)} \wedge \langle \rho \rangle (\psi_{\sigma^{-1}(1)}, \dots, \neg \langle \sigma[\rho] \rangle (\psi_1, \dots, \psi_k), \dots, \psi_{\sigma^{-1}(k)}) \right)$$

Thus there exists $(w_0, w_1, \dots, w_k) \in \rho^{\mathfrak{N}}$ so that $\mathfrak{M}, w_{\ell} \Vdash \psi_{\sigma^{-1}(\ell)}$, for every $\ell \neq \sigma(0)$ (including $\ell = 0$), and

$$\mathfrak{M}, w_{\sigma(0)} \Vdash \neg \langle \sigma[\rho] \rangle (\psi_1, \dots, \psi_k).$$

Now $(w_{\sigma(0)}, \dots, w_{\sigma(k)}) \in (\sigma[\rho])^{\mathfrak{M}}$ and furthermore $\mathfrak{M}, w_{\sigma(\ell)} \Vdash \psi_{\ell}$, for every $1 \leq \ell \leq k$, which is a contradiction.

Finally, we will verify that $\mathfrak{N}, w_0 \Vdash \xi_3$, for any $w_0 \in W$. Fix k, σ and $\psi_1, \dots, \psi_k \in \text{Subf}(t(\varphi))$. Aiming for a contradiction, suppose that

$$\mathfrak{M}, w_0 \Vdash \langle \rho \rangle (\psi_{\sigma^{-1}(1)}, \dots, \psi_{\sigma^{-1}(k)}),$$

but

$$\mathfrak{M}, w_0 \Vdash \neg \langle \sigma[\rho] \rangle (\psi_1, \dots, \psi_k).$$

Now there exists $(w_0, w_1, \dots, w_k) \in \rho^{\mathfrak{N}}$ so that $\mathfrak{M}, w_{\ell} \Vdash \psi_{\sigma^{-1}(\ell)}$. Hence $(w_0, w_{\sigma(1)}, \dots, w_{\sigma(k)}) \in \llbracket \sigma \rho \rrbracket_{\mathfrak{M}} = (\sigma[\rho])^{\mathfrak{M}}$, which is a contradiction, since $\mathfrak{M}, w_{\sigma(\ell)} \Vdash \psi_{\ell}$, for every $1 \leq \ell \leq k$. ◀

26:12 Complexity of Polyadic Boolean Modal Logics

Suppose now that \mathfrak{M} is a Kripke model and that there is a $w \in W$ so that $\mathfrak{M}, w \Vdash \Theta$. To obtain a model for φ , we start by extending \mathfrak{M} as follows.

1. For every $2 \leq k \leq m$, $\rho \in \mathfrak{T}_k$, $\sigma \in S_k$ and $(w_1, \dots, w_k) \in \rho^{\mathfrak{M}}$, we will add $(w_{\sigma(1)}, \dots, w_{\sigma(k)})$ to $(\sigma[\rho])^{\mathfrak{M}}$.
2. For every $2 \leq k \leq m$ and $(w_1, \dots, w_k) \in W^k$, for which there does not exist $\rho \in \mathfrak{T}_k$ so that $(w_1, \dots, w_k) \in \rho^{\mathfrak{M}}$, we let ρ denote the relation promised by Lemma 14 (see below) and we will add the tuple $(w_{\sigma(1)}, \dots, w_{\sigma(k)})$ to $(\sigma[\rho])^{\mathfrak{M}}$, for every $\sigma \in S_k$.

Letting \mathfrak{M}^* denote the resulting model, we need to show that $\mathfrak{M}^*, w \Vdash t(\varphi)$. We start with the following lemma which guarantees that we can extend the interpretations of k -ary accessibility relations in \mathfrak{M} in such a way that every tuple will belong to the interpretation of at least one such relation.

► **Lemma 14.** *For every $1 \leq k < m$ and $(w_0, w_1, \dots, w_k) \in W^{k+1}$ there exists $\rho \in \mathfrak{T}_{k+1}$ so that for all $\sigma \in S_{k+1}$ and $\psi_1, \dots, \psi_k \in \text{Subf}(t(\varphi))$ we have that if $\mathfrak{M}, w_{\sigma(\ell)} \Vdash \psi_\ell$, for every $1 \leq \ell \leq k$, then $\mathfrak{M}, w_{\sigma(0)} \Vdash \langle \sigma[\rho] \rangle(\psi_1, \dots, \psi_k)$.*

Proof. Fix $(w_0, w_1, \dots, w_k) \in W^{k+1}$. Aiming for a contradiction, suppose that for every $\rho \in \mathfrak{T}_{k+1}$ there exists $\sigma_\rho \in S_{k+1}$ and $\psi_1^{\sigma_\rho}, \dots, \psi_k^{\sigma_\rho} \in \text{Subf}(t(\varphi))$ so that

$$\mathfrak{M}, w_{\sigma_\rho(\ell)} \Vdash \psi_\ell^{\sigma_\rho},$$

for every $1 \leq \ell \leq k$, but

$$\mathfrak{M}, w_{\sigma_\rho(0)} \Vdash \neg \langle \sigma_\rho[\rho] \rangle(\psi_1^{\sigma_\rho}, \dots, \psi_k^{\sigma_\rho}).$$

It is simple to verify that this entails that

$$\begin{aligned} \mathfrak{M}, w_0 \Vdash & \bigwedge_{\rho \in X_0} \neg \langle \sigma_\rho[\rho] \rangle(\psi_1^{\sigma_\rho}, \dots, \psi_k^{\sigma_\rho}) \\ & \wedge \bigwedge_{1 \leq \ell \leq k} \langle E \rangle \left(\bigwedge_{\rho \in X_\ell} \neg \langle \sigma_\rho[\rho] \rangle(\psi_1^{\sigma_\rho}, \dots, \psi_k^{\sigma_\rho}) \wedge \bigwedge_{\rho \notin X_\ell} \psi_{\sigma_\rho^{-1}(\ell)}^{\sigma_\rho} \right) \end{aligned}$$

and since $\mathfrak{M}, w_0 \Vdash \xi_1$, we have that

$$\mathfrak{M}, w_0 \Vdash \bigvee_{\rho \notin X_0} \neg \psi_{\sigma_\rho^{-1}(0)}^{\sigma_\rho},$$

which is a contradiction, since by assumption $\mathfrak{M}, w_0 \Vdash \bigwedge_{\rho \notin X_0} \psi_{\sigma_\rho^{-1}(0)}^{\sigma_\rho}$. ◀

Secondly, we will need a lemma which guarantees that we can close the interpretations of accessibility relations under permutations.

► **Lemma 15.** *For every $1 \leq k < m$, $\rho \in \mathfrak{T}_{k+1}$, $\sigma \in S_{k+1}$ we have that if $(w_0, w_1, \dots, w_k) \in \rho^{\mathfrak{M}}$, then for every $\psi_1, \dots, \psi_k \in \text{Subf}(t(\varphi))$ we have that if $\mathfrak{M}, w_{\sigma(\ell)} \Vdash \psi_\ell$, for every $1 \leq \ell \leq k$, then $\mathfrak{M}, w_{\sigma(0)} \Vdash \langle \sigma[\rho] \rangle(\psi_1, \dots, \psi_k)$.*

Proof. Fix k, ρ, σ and $(w_0, w_1, \dots, w_k) \in \rho^{\mathfrak{M}}$. Aiming for a contradiction, suppose that there exists $\psi_1, \dots, \psi_k \in \text{Subf}(t(\varphi))$ so that

$$\mathfrak{M}, w_{\sigma(\ell)} \Vdash \psi_\ell,$$

for every $1 \leq \ell \leq k$, but

$$\mathfrak{M}, w_{\sigma(0)} \Vdash \neg \langle \sigma[\rho] \rangle(\psi_1, \dots, \psi_k).$$

We have now two cases based on whether or not $\sigma(0) = 0$. First, if $\sigma(0) = 0$, then since $\mathfrak{M}, w_0 \Vdash \xi_3$, we have that

$$\mathfrak{M}, w_0 \Vdash \langle \rho \rangle (\psi_{\sigma^{-1}(1)}, \dots, \psi_{\sigma^{-1}(k)}) \rightarrow \langle \sigma[\rho] \rangle (\psi_1, \dots, \psi_k).$$

By assumption, $(w_0, w_1, \dots, w_k) \in \rho^{\mathfrak{M}}$ and $\mathfrak{M}, w_{\sigma(\sigma^{-1}(\ell))} \Vdash \psi_{\sigma^{-1}(\ell)}$, for every $1 \leq \ell \leq k$, and hence

$$\mathfrak{M}, w_0 \Vdash \langle \rho \rangle (\psi_{\sigma^{-1}(1)}, \dots, \psi_{\sigma^{-1}(k)}),$$

which implies that

$$\mathfrak{M}, w_0 \Vdash \langle \sigma[\rho] \rangle (\psi_1, \dots, \psi_k),$$

a contradiction.

Consider then the case $\sigma(0) \neq 0$. Since $\mathfrak{M}, w_0 \Vdash \xi_2$, we have that

$$\mathfrak{M}, w_0 \Vdash \left(\neg \psi_{\sigma^{-1}(0)} \vee \neg \langle \rho \rangle (\neg \psi_{\sigma^{-1}(1)}, \dots, \underbrace{\langle \sigma[\rho] \rangle (\psi_1, \dots, \psi_k)}_{\sigma(0):\text{th formula}}, \dots, \neg \psi_{\sigma^{-1}(k)}) \right)$$

By assumption $\mathfrak{M}, w_{\sigma(\sigma^{-1}(0))} \Vdash \psi_{\sigma^{-1}(0)}$. Furthermore, since $(w_0, w_1, \dots, w_k) \in \rho^{\mathfrak{M}}$ and $\mathfrak{M}, w_{\sigma(\sigma^{-1}(\ell))} \Vdash \psi_{\sigma^{-1}(\ell)}$, for every $\ell \neq \sigma(0)$, we must have that

$$\mathfrak{M}, w_{\sigma(0)} \Vdash \langle \sigma[\rho] \rangle (\psi_1, \dots, \psi_k),$$

which is a clear contradiction. ◀

A routine induction, which uses the previous two lemmas in the case of formulas of the form $\langle \rho \rangle (\psi_1, \dots, \psi_k)$, can be used to establish that $\mathfrak{M}^*, w \Vdash t(\varphi)$. For the rest of this section we will use \mathfrak{M} to denote \mathfrak{M}^* .

Now we are ready to use \mathfrak{M} to construct a model for φ . We define a Kripke model $\mathfrak{N} = (W^*, (R^\tau)_{R \in \tau}, V^*)$ over τ as follows. First, we define that

$$W^* := W \times \{2, \dots, m\} \times \mathbb{N},$$

where \mathbb{N} is the set of natural numbers. In what follows we will adopt the convention that we will associate to every k -table ρ a unique index from the set \mathbb{N} , which we simply denote by ρ . We start our model construction by specifying that for every $(w, \ell, r) \in W^*$ we have that $(w, \ell, r) \in V^*(p)$ iff $w \in V(p)$. Next we will assign tables to tuples. We first define that for every $(w_0, \ell, r) \in W^*$ and $(w_0, w_1, \dots, w_k) \in \rho^{\mathfrak{M}}$ the tuple

$$((w_0, \ell, r), (w_1, 2, r + \rho), \dots, (w_k, k + 1, r + \rho))$$

realizes the type ρ . Observe that each such tuple consists of $k + 1$ distinct elements. Indeed, the first element is distinct from the remaining elements because $r \neq r + \rho$, while the remaining elements in the tuple are pairwise distinct because they differ with respect to their second coordinate.

Notice that if we force a tuple to realize a table ρ , then for every $\sigma \in S_{k+1}$ the permutation of this tuple under σ realizes the type $\sigma[\rho]$. Hence, it is not obvious that the above procedure does not assign different tables to some tuples.

▷ **Claim 16.** In the above procedure, no tuple is assigned a table more than once.

26:14 Complexity of Polyadic Boolean Modal Logics

Proof. Suppose that we have assigned tables ρ_1 and ρ_2 to a tuple (w_0, w_1, \dots, w_k) . We want to show that $\rho_1 = \rho_2$. By construction, our assumption implies that there are tuples

$$((w'_0, \ell', r'), (w'_1, 2, r' + \rho'), \dots, (w'_k, k + 1, r' + \rho'))$$

and

$$((w''_0, \ell'', r''), (w''_1, 2, r'' + \rho''), \dots, (w''_k, k + 1, r'' + \rho''))$$

and permutations σ_1 and σ_2 so that σ_1 (respectively σ_2) applied to the first (respectively the second) tuple gives (w_0, w_1, \dots, w_k) , and furthermore that $\sigma_1[\rho'] = \rho_1$ and $\sigma_2[\rho''] = \rho_2$. Since for every $2 \leq \ell \leq k + 1$ there exists a unique element in the tuple (w_0, w_1, \dots, w_k) which has ℓ as its second coordinate, we must have that the two above tuples are in fact the same tuples, since they are both permutations of (w_0, w_1, \dots, w_k) . In particular, $\rho' = \rho''$, since $r' = r''$. Finally, because this single tuple consists of $k + 1$ distinct elements and permutating it with either σ_1 or σ_2 gives the same result – namely (w_0, w_1, \dots, w_k) – we must have that $\sigma_1 = \sigma_2$, and hence $\rho_1 = \rho_2$. \triangleleft

Finally, for every tuple $((w_0, \ell_0, r_0), \dots, (w_k, \ell_k, r_k))$ for which we have not yet assigned a table, we will pick some $\rho \in \mathfrak{T}_{k+1}$ for which $(w_0, \dots, w_k) \in \rho^{\text{m}}$ and assign the corresponding table to our tuple. This completes the definition of \mathfrak{N} .

► **Lemma 17.** *For every $\psi \in \text{Subf}(\varphi)$ and $w_0 \in W$ we have that*

$$\mathfrak{N}, (w_0, \ell, r) \Vdash \psi \Leftrightarrow \mathfrak{M}, w_0 \Vdash t(\psi).$$

Proof. A routine induction. \blacktriangleleft

In particular \mathfrak{N} is a model of φ and hence φ is satisfiable. Thus we can conclude that φ is satisfiable iff Θ is.

5 Model checking problem of polyadic Boolean modal logic with permutations

In this section we prove that the combined complexity of $\text{PML}(p, s, \neg, \cap)$ is PTIME-complete. Note that the corresponding lower bound follows already from the fact that the combined complexity of standard modal logic is PTIME-complete [10]. Throughout this section we will assume that the domains of the models are equipped with some (arbitrary) linear order.

We start by defining precisely how we will encode Kripke models. In fact, we will describe how we will encode arbitrary relational models, since it avoids some notational clutter. Given two strings x and y , we will use $x\#y$ to denote their concatenation. The *database encoding* of \mathfrak{A} is the sequence

$$1^{|A|} \triangleright \text{lenc}(R_1) \triangleright \dots \triangleright \text{lenc}(R_m)$$

where \triangleright is a separator character (the use of which could be of course avoided) and each $\text{lenc}(R_i)$ is a sequence

$$r_1 \# r_2 \# \dots \# r_{|R_i|},$$

where each r_j is a sequence consisting of $ar(R_i)$ -many binary strings of length $\log_2(|A|)$ which encodes the j th tuple in $R_i^{\mathfrak{A}}$. We note that the length of the database encoding of a model \mathfrak{A} , denoted by $|\mathfrak{A}|$, is

$$O\left(|A| + \sum_{1 \leq i \leq m} |R_i| ar(R_i) \log_2(|A|)\right).$$

► **Remark 18.** The encoding of models that we have presented here is not the only encoding of relational structures one encounters in the literature. Another standard choice of encoding can be found in [22], where the encoding essentially describes the “adjacency” matrix of each relation, i.e., for every relation R and every tuple of length $ar(R)$ there is a single bit which indicates whether that tuple belongs to the interpretation of R . Note that if the arities of relation symbols are bounded by a constant, then this encoding of models, which we call the *matrix encoding*, is essentially equivalent with the database encoding of models.

If matrix encoding of models is used, then the PTIME upper bound on the model checking problem of $\text{PML}(p, s, \neg, \cap)$ becomes somewhat trivial. Indeed, one can then compute complements of relations in linear time, which allows one to easily reduce the model checking problem of $\text{PML}(p, s, \neg, \cap)$ to the model checking problem of, say, the guarded fragment, which is known to be PTIME-complete [3].

Next we will present our model checking algorithm. As an important preliminary step, the following lemma will allow us to restrict our attention to formulas that contain only terms which use negation in a very restricted way.

► **Lemma 19.** *Suppose that $\mathcal{R} \in \text{GRA}(p, s, \neg, \cap)[\tau]$. We can compute in polynomial time a term $\mathcal{R}' \in \text{GRA}(p, s, \neg, \setminus, \cap, \cup)$ so that \mathcal{R} is equivalent with \mathcal{R}' and \mathcal{R}' is either of the form \mathcal{R}'' or of the form $\neg\mathcal{R}''$, for some $\mathcal{R}'' \in \text{GRA}(p, s, \setminus, \cap, \cup)[\tau]$.*

Proof. Using Lemma 1, we can bring all the negations occurring in the input term \mathcal{R} to the start of the term, which – after eliminating consecutive negations – results in a term which is either of the form \mathcal{R}' or $\neg\mathcal{R}'$, where $\mathcal{R}' \in \text{GRA}(p, s, \setminus, \cap, \cup)[\tau]$. ◀

Suppose now that $\varphi \in \text{PML}(p, s, \neg, \cap)[\tau]$ and $\mathfrak{M} = (W, (R^m)_{R \in \tau}, V)$. Our goal is to compute the set of worlds in \mathfrak{M} where φ is true. By applying Lemma 19, we can assume that in each subformula $\langle \mathcal{R} \rangle(\psi_1, \dots, \psi_k)$ the term \mathcal{R} is either of the form \mathcal{R}' or $\neg\mathcal{R}'$, where $\mathcal{R}' \in \text{GRA}(p, s, \setminus, \cap, \cup)[\tau]$. Using induction, one can show that the size of $\llbracket \mathcal{R}' \rrbracket_{\mathfrak{M}}$ is only polynomial with respect to $|\mathfrak{M}|$.

Now we will describe the model checking algorithm, which extends the standard labeling algorithm that is often used in the context of modal logics [30]. The algorithm will use some enumeration of the subformulas $\varphi_1, \dots, \varphi_n$ of φ which satisfies the requirement that if φ_j is a proper subformula of φ_i , then $j < i$.

$\nu = \emptyset$

for $i = 1$ **through** n **do:**

if $\varphi_i = p$ **then** $\nu := \nu \cup \{(p, V(p))\}$

if $\varphi_i = \neg\varphi_j$ **then** $\nu := \nu \cup \{(\varphi_i, W \setminus \nu(\varphi_j))\}$

if $\varphi_i = \varphi_j \wedge \varphi_k$ **then** $\nu := \nu \cup \{(\varphi_i, \nu(\varphi_j) \cap \nu(\varphi_k))\}$

if $\varphi_i = \langle \mathcal{R} \rangle(\varphi_{i_1}, \dots, \varphi_{i_k})$ **then**

$$\nu := \nu \cup \left(\varphi_i, \left\{ w \in W \mid \left(w \times \nu(\varphi_{i_1}) \times \dots \times \nu(\varphi_{i_k}) \right) \cap \llbracket \mathcal{R} \rrbracket_{\mathfrak{M}} \neq \emptyset \right\} \right)$$

if $\varphi_i = \langle \neg\mathcal{R} \rangle(\varphi_{i_1}, \dots, \varphi_{i_k})$ **then**

$U := \emptyset$

for $w \in W$ **do:**

for $(w_1, \dots, w_k) \in \nu(\varphi_{i_1}) \times \dots \times \nu(\varphi_{i_k})$ **do:**

if $(w, w_1, \dots, w_k) \notin \llbracket \mathcal{R} \rrbracket_{\mathfrak{M}}$ **then** $U := U \cup \{w\}$ **and break**

$\nu := \nu \cup \{(\varphi_i, U)\}$

return $\nu(\varphi)$

It is straightforward to check that the above algorithm is correct. We note that in certain places the description of the algorithm is, for ease of exposition, somewhat informal. In particular, we have not specified how in the case of $\langle \neg \mathcal{R} \rangle(\varphi_{i_1}, \dots, \varphi_{i_k})$ the for-loop going through the set $\nu(\varphi_{i_1}) \times \dots \times \nu(\varphi_{i_k})$ should be implemented, which is in fact a somewhat important detail, since we can not always construct this set explicitly as in the worst case its size is proportional to $|W|^k$ (which is exponential in the size of the input, since k is not bounded by a constant). However, this explicit construction can be avoided by maintaining k pointers to elements of the sets $\nu(\varphi_{i_\ell})$. More precisely, we can initialize k pointers which at the beginning point at the first element of A and which we will then increment in a lexicographical manner.

We are now left with the easy task of proving that our algorithm runs in polynomial time.

► **Lemma 20.** *The above algorithm runs in time polynomial with respect to*

$$|\varphi| \times \|\mathfrak{M}\|.$$

Proof. The outermost for-loop is executed $|\text{Subf}(\varphi)| \leq |\varphi|$ times, so it suffices to argue that each case within the loop can be done in time polynomial with respect to $|\varphi| \times \|\mathfrak{M}\|$. The most non-trivial case is the case of $\langle \neg \mathcal{R} \rangle(\varphi_{i_1}, \dots, \varphi_{i_k})$, where we can make the simple observation that the running time of the for-loop going through $\nu(\varphi_{i_1}) \times \dots \times \nu(\varphi_{i_k})$ is bounded above by the size of $\llbracket \mathcal{R} \rrbracket_{\mathfrak{M}} \leq \|\mathfrak{M}\|$, since it stops after we have encountered a tuple which does not belong to $\llbracket \mathcal{R} \rrbracket_{\mathfrak{M}}$ (or after we have went through all the relevant k -tuples). ◀

Since the model checking problem of standard modal logic is PTIME-complete, we have the desired result.

► **Theorem 21.** *The model checking problem of $\text{PML}(p, s, \neg, \cap)$ is PTIME-complete.*

6 Conclusions

We have studied the computational complexity of model checking and satisfiability problems of polyadic modal logics extended with permutations and Boolean operators on accessibility relations. Concerning satisfiability problems, we have proved that the satisfiability problems of both polyadic modal logic extended with negations of accessibility relations $\text{PML}(\neg)$ and full polyadic Boolean modal logic extended with permutations over $\text{PML}(p, s, \neg, \cap)$ are EXPTIME-complete, the latter under the assumption that the underlying set of accessibility relations and their arities are bounded by a constant, which is necessary if $\text{EXPTIME} \neq \text{NEXPTIME}$, since the satisfiability problem of $\text{PML}(p, s, \neg, \cap)$ is in general NEXPTIME-complete. We have also established that the model checking problem for full polyadic Boolean modal logic extended with permutations $\text{PML}(p, s, \neg, \cap)$ is PTIME-complete. Our results contribute to the research program outlined in [14] and extend the results of [24, 27] to polyadic context.

Concerning future research directions, the reductions that we used in establishing complexity bounds on satisfiability problems seem to be quite robust, and hence we expect that in the future they can be used to extend the results presented here. For instance, one can most likely show that $\text{PML}(p, s, \neg)$ has an EXPTIME-complete satisfiability problem, which we have not yet been able to do. In this direction a natural intermediate problem would be to establish that $\text{PML}(p, s) + \langle \text{E} \rangle$ has an EXPTIME-complete satisfiability problem, since then one might be able to adapt the techniques used in this paper to reduce the satisfiability problem of $\text{PML}(p, s, \neg)$ to that of $\text{PML}(p, s) + \langle \text{E} \rangle$. Following [14], we are also quite confident that $\text{PML}(p, s, \neg)$ could be extended with counting without affecting its EXPTIME-completeness.

Another obvious direction would be to show that if the number of accessibility relations and their arities are bounded by a constant, then the satisfiability problem of $\text{PML}(p, s, \neg, \cap)$ extended with an equality operator I is EXPTIME -complete. Indeed, such a result would fully generalize the main result of [27] to polyadic context, which states the satisfiability problem of $\text{ML}(I, s, \neg, \cap)$ is EXPTIME -complete, when the number of accessibility relations is bounded by a constant. Here the main technical difficulty is that it seems that one needs to reduce the satisfiability problem of $\text{PML}(I, p, s, \neg, \cap)$ to that of PML extended with the *difference modality* $\langle d \rangle$ [6] (as opposed to just reducing it to $\text{PML} + \langle E \rangle$). $\langle d \rangle \psi$ states that there exists a world which is *different* from the current world and in which ψ is true. Roughly speaking, the need for $\langle d \rangle$ arises from the fact that one needs to encode basic properties of equality I in the reduction.

References

- 1 Franz Baader, Ian Horrocks, Carsten Lutz, and Ulrike Sattler. *An Introduction to Description Logic*. Cambridge University Press, 2017.
- 2 Bartosz Bednarczyk. Exploiting forwardness: Satisfiability and query-entailment in forward guarded fragment. In *Logics in Artificial Intelligence*, pages 179–193. Springer, 2021.
- 3 Dietmar Berwanger and Erich Grädel. Games and model checking for guarded logics. In *Proceedings of the Artificial Intelligence on Logic for Programming*, pages 70–84. Springer-Verlag, 2001.
- 4 Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*, volume 53 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 2001. doi:10.1017/CB09781107050884.
- 5 Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. On the decidability of query containment under constraints. In Alberto O. Mendelzon and Jan Paredaens, editors, *Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 149–158, 1998. doi:10.1145/275487.275504.
- 6 Maarten de Rijke. The modal logic of inequality. *Journal of Symbolic Logic*, 57(2):566–584, 1992. doi:10.2307/2275293.
- 7 G Gargov, S Passy, and T Tinchev. Modal environment for boolean speculations. In *Mathematical logic and its applications*, pages 253–263, 1987.
- 8 Erich Grädel. On the restraining power of guards. *J. Symb. Log.*, 64(4):1719–1742, 1999. doi:10.2307/2586808.
- 9 Erich Grädel. *Why Are Modal Logics so Robustly Decidable?*, pages 393–408. World Scientific Publishing Co., Inc., 2001.
- 10 Erich Grädel. Finite model theory and descriptive complexity. In *Finite Model Theory and Its Applications*, pages 125–230. Springer, 2007.
- 11 Erich Grädel, Phokion Kolaitis, and Moshe Vardi. On the decision problem for two-variable first-order logic. *Bulletin of Symbolic Logic*, 3(1):53–69, 1997.
- 12 Lauri Hella and Antti Kuusisto. One-dimensional fragment of first-order logic. In *Proceedings of Advances in Modal Logic*, pages 274–293, 2014.
- 13 Jonne Iso-Tuisku and Antti Kuusisto. Uniform one-dimensional fragment over ordered structures. *arXiv*, abs/1812.08732, 2018. arXiv:1812.08732.
- 14 Jonne Iso-Tuisku and Antti Kuusisto. Description logics as polyadic modal logics. *arXiv*, abs/2108.08838, 2021. arXiv:2108.08838.
- 15 Reijo Jaakkola. Ordered fragments of first-order logic. In *46th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, pages 62:1–62:14, 2021.
- 16 Reijo Jaakkola and Antti Kuusisto. Algebraic classifications for fragments of first-order logic and beyond. *arXiv*, abs/2005.01184, 2020. arXiv:2005.01184.

- 17 Emanuel Kieronski. One-dimensional guarded fragments. In *44th International Symposium on Mathematical Foundations of Computer Science, MFCS*, pages 16:1–16:14, 2019. doi:10.4230/LIPIcs.MFCS.2019.16.
- 18 Emanuel Kieronski and Antti Kuusisto. Complexity and expressivity of uniform one-dimensional fragment with equality. In *39nd International Symposium on Mathematical Foundations of Computer Science (MFCS)*, pages 365–376, 2014.
- 19 Emanuel Kieronski and Antti Kuusisto. One-dimensional fragment over words and trees. *Journal of Logic and Computation*, February 2022.
- 20 Antti Kuusisto. On the uniform one-dimensional fragment. In *International Workshop on Description Logics (DL)*, 2016.
- 21 Antti Kuusisto and Carsten Lutz. Weighted model counting beyond two-variable logic. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 619–628, 2018.
- 22 Leonid Libkin. *Elements of Finite Model Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2004. doi:10.1007/978-3-662-07003-1.
- 23 Carsten Lutz and Ulrike Sattler. The complexity of reasoning with boolean modal logics. In *Proceedings of Advances in Modal Logic*, pages 329–348, 2000.
- 24 Carsten Lutz and Ulrike Sattler. The complexity of reasoning with boolean modal logics. In *Advances in Modal Logic*, 2000.
- 25 Carsten Lutz and Ulrike Sattler. Mary likes all cats. In *Proceedings of the 2000 International Workshop on Description Logics (DL2000)*, pages 213–226, 2000.
- 26 Carsten Lutz, Ulrike Sattler, and Stephan Tobies. A suggestion for an n-ary description logic. In Patrick Lambrix, Alexander Borgida, Maurizio Lenzerini, Ralf Möller, and Peter F. Patel-Schneider, editors, *Proceedings of the 1999 International Workshop on Description Logics (DL'99)*, 1999.
- 27 Carsten Lutz, Ulrike Sattler, and Frank Wolter. Modal logic and the two-variable fragment. In *Proceedings of the 15th International Workshop on Computer Science Logic (CSL)*, pages 247–261, 2001.
- 28 Maarten Marx and Yde Venema. Local variations on a loose theme: Modal logic and decidability. In *Finite Model Theory and Its Applications*, pages 371–429. Springer, 2007.
- 29 James G. Schmolze. Terminological knowledge representation systems supporting n-ary terms. In *Proceedings of the 1st International Conference on Principles of Knowledge Representation and Reasoning (KR'89)*, 1989.
- 30 Moshe Y. Vardi. Why is modal logic so robustly decidable? In *Descriptive Complexity and Finite Models*, pages 149–183. American Mathematical Society, 1996.