Modal Logics with Hard Diamond-Free Fragments

Antonis Achilleos(✉)

The Graduate Center of The City University of New York, New York, USA
aachilleos@gradcenter.cuny.edu

Abstract. We investigate the complexity of modal satisfiability for certain combinations of modal logics. In particular we examine four examples of multimodal logics with dependencies and demonstrate that even if we restrict our inputs to diamond-free formulas (in negation normal form), these logics still have a high complexity. This result illustrates that having D as one or more of the combined logics, as well as the interdependencies among logics can be important sources of complexity even in the absence of diamonds and even when at the same time in our formulas we allow only one propositional variable. We then further investigate and characterize the complexity of the diamond-free, 1-variable fragments of multimodal logics in a general setting.

Keywords: Modal logic · Satisfiability · Computational complexity · Diamond-free fragments · Multi-modal · Lower bounds

1 Introduction

The complexity of the satisfiability problem for modal logic, and thus of its dual, modal provability/validity, has been extensively studied. Whether one is interested in areas of application of Modal Logic, or in the properties of Modal Logic itself, the complexity of modal satisfiability plays an important role. Ladner has established most of what are now considered classical results on the matter [17], determining that most of the usual modal logics are PSPACE-hard, while more for the most well-known logic with negative introspection, S5, satisfiability is NP-complete; Halpern and Moses [12] then demonstrated that KD45-satisfiability is NP-complete and that the multi-modal versions of these logics are PSPACE-complete. Therefore, it makes sense to try to find fragments of these logics that have an easier satisfiability problem by restricting the modal elements of a formula – or prove that satisfiability remains hard even in fragments that seem trivial (ex. [4,11]). In this paper we present mostly hardness results for this direction and for certain cases of multimodal logics with modalities that affect each other. Relevant syntactic restrictions and their effects on the complexity of various modal logics have been examined in [13,14]. For more on Modal Logic and its complexity, see [10,12,20].

A (uni)modal formula is a formula formed by using propositional variables and Boolean connectives, much like propositional calculus, but we also use two

© Springer International Publishing Switzerland 2016
S. Artemov and A. Nerode (Eds.): LFCS 2016, LNCS 9537, pp. 1–13, 2016.
DOI: 10.1007/978-3-319-27683-0_1

additional operators, $\Box$ (box) and $\Diamond$ (diamond): if ϕ is a formula, then $\Box\phi$ and $\Diamond\phi$ are formulas. Modal formulas are given truth values with respect to a Kripke model (W, R, V),[1] which can be seen as a directed graph (W, R) (with possibly an infinite number of vertices and allowing self-loops) together with a truth value assignment for the propositional variables for each world (vertex) in W, called V. We define $\Box\phi$ to be true in a world a if ϕ is true at every world b such that (a, b) is an edge, while $\Diamond$ is the dual operator: $\Diamond\phi$ is true at a if ϕ is true at some b such that (a, b) is an edge.

We are interested in the complexity of the satisfiability problem for modal formulas (in negation normal form, to be defined later) that have no diamonds – i.e. is there a model with a world at which our formula is true? When testing a modal formula for satisfiability (for example, trying to construct a model for the formula through a tableau procedure), a clear source of complexity are the diamonds in the formula. When we try to satisfy $\Diamond\phi$, we need to assume the existence of an extra world where ϕ is satisfied. When trying to satisfy $\Diamond p_1 \wedge \Diamond p_2 \wedge \Box\phi_n$, we require two new worlds where $p_1 \wedge \phi_n$ and $p_2 \wedge \phi_n$ are respectively satisfied; for example, for $\phi_0 = \top$ and $\phi_{n+1} = \Diamond p_1 \wedge \Diamond p_2 \wedge \Box\phi_n$, this causes an exponential explosion to the size of the constructed model (if the model we construct for ϕ_n has k states, then the model for ϕ_{n+1} has $2k+1$ states). There are several modal logics, but it is usually the case that in the process of satisfiability testing, as long as there are no diamonds in the formula, we are not required to add more than one world to the constructed model. Therefore, it is natural to identify the existence of diamonds as an important source of complexity. On the other hand, when the modal logic is D, its models are required to have a serial accessibility relation (no sinks in the graph). Thus, when we test $\Box\phi$ for D-satisfiability, we require a world where ϕ is satisfied. In such a unimodal setting and in the absence of diamonds, we avoid an exponential explosion in the number of worlds and we can consider models with only a polynomial number of worlds.

Several authors have examined the complexity of combinations of modal logic (ex. [9,15,18]). Very relevant to this paper work on the complexity of combinations of modal logic is by Spaan in [20] and Demri in [6]. In particular, Demri studied $L_1 \oplus_\subseteq L_2$, which is $L_1 \oplus L_2$ (see [20]) with the additional axiom $\Box_2\phi \rightarrow \Box_1\phi$ and where L_1, L_2 are among K, T, B, $\mathsf{S4}$, and $\mathsf{S5}$ – modality 1 comes from L_1 and 2 from L_2. For when L_1 is among K, T, B and L_2 among $\mathsf{S4}$, $\mathsf{S5}$, he establishes EXP-hardness for $L_1 \oplus_\subseteq L_2$-satisfiability. We consider $L_1 \oplus_\subseteq L_2$, where L_1 is a unimodal or bimodal logic (usually D, or $\mathsf{D4}$). When L_1 is bimodal, $L_1 \oplus_\subseteq L_2$ is $L_1 \oplus L_2$ with the extra axioms $\Box_3\phi \rightarrow \Box_1\phi$ and $\Box_3\phi \rightarrow \Box_2\phi$.

The family of logics we consider in this paper can be considered part of the much more general family of *regular grammar logics (with converse)*. Demri and De Nivelle have shown in [8] through a translation into a fragment of first-order logic that the satisfiability problem for the whole family is in EXP (see also [7]). Then, Nguyen and Szałas in [19] gave a tableau procedure for the general satisfiability

[1] There are numerous semantics for modal logic, but in this paper we only use Kripke semantics.

problem (where the logic itself is given as input in the form of a finite automaton) and determined that it is also in EXP.

In this paper, we examine the effect on the complexity of modal satisfiability testing of restricting our input to diamond-free formulas under the requirement of seriality and in a multimodal setting with connected modalities. In particular, we initially examine four examples: $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K}$, $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$, $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$, and $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$.[2] For these logics we look at their diamond-free fragment and establish that they are PSPACE-hard and in the case of $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$, EXP-hard. Furthermore, $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K}$, $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$, and $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$ are PSPACE-hard and $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$ is EXP-hard even for their 1-variable fragments. Of course these results can be naturally extended to more modal logics, but we treat what we consider simple characteristic cases. For example, it is not hard to see that nothing changes when in the above multimodal logics we replace K by D, or K4 by D4, as the extra axiom $\Box_3\phi \rightarrow \Diamond_3\phi$ ($\Box_2\phi \rightarrow \Diamond_2\phi$ for $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$) is a derived one. It is also the case that in these logics we can replace K4 by other logics with positive introspection (ex. S4, S5) without changing much in our reasoning.

Then, we examine a general setting of a multimodal logic (we consider combinations of modal logics K, D, T, D4, S4, KD45, S5) where we include axioms $\Box_i\phi \rightarrow \Box_j\phi$ for some pairs i, j. For this setting we determine exactly the complexity of satisfiability for the diamond-free (and 1-variable) fragment of the logic and we are able to make some interesting observations. The study of this general setting is of interest, because determining exactly when the complexity drops to tractable levels for the diamond-free fragments illuminates possibly appropriate candidates for parameterization: if the complexity of the diamond-free, 1-variable fragment of a logic drops to P, then we may be able to develop algorithms for the satisfiability problem of the logic that are efficient for formulas of few diamonds and propositional variables; if the complexity of that fragment does not drop, then the development of such algorithms seems unlikely (we may be able to parameterize with respect to some other parameter, though). Another argument for the interest of these fragments results from the hardness results of this paper. The fact that the complexity of the diamond-free, 1-variable fragment of a logic remains high means that this logic is likely a very expressive one, even when deprived of a significant part of its syntax.

A very relevant approach is presented in [13,14]. In [13], Hemaspaandra determines the complexity of Modal Logic when we restrict the syntax of the formulas to use only a certain set of operators. In [14], Hemaspaandra et al. consider multimodal logics and all Boolean functions. In fact, some of the cases we consider have already been studied in [14]. Unlike [14], we focus on multimodal logics where the modalities are not completely independent – they affect each other through axioms of the form $\Box_i\phi \rightarrow \Box_j\phi$. Furthermore in this setting we only consider diamond-free formulas, while at the same time we examine the cases where we allow only one propositional variable. As far as our results are concerned, it is interesting to note that in [13,14] when we consider frames with

[2] In general, in $A \oplus_{\subseteq} B$, if A a bimodal (resp. unimodal) logic, the modalities 1 and 2 (resp. modality 1) come(s) from A and 3 (resp. 2) comes from logic B.

serial accessibility relations, the complexity of the logics under study tends to drop, while in this paper we see that serial accessibility relations (in contrast to arbitrary, and sometimes reflexive, accessibility relations) contribute substantially to the complexity of satisfiability.

Another motivation we have is the relation between the diamond-free fragments of Modal Logic with Justification Logic. Justification Logic can be considered an explicit counterpart of Modal Logic. It introduces justifications to the modal language, replacing boxes ($\Box$) by constructs called justification terms. When we examine a justification formula with respect to its satisfiability, the process is similar to examining the satisfiability of a modal formula without any diamonds (with some extra nontrivial parts to account for the justification terms). Therefore, as we are interested in the complexity of systems of Multimodal and Multijustification Logics, we are also interested in these diamond-free fragments. For more on Justification Logic and its complexity, the reader can see [3,16]; for more on the complexity of Multi-agent Justification Logic and how this paper is connected to it, the reader can see [2].

It may seem strange that we restrict ourselves to formulas without diamonds but then we implicitly reintroduce diamonds to our formulas by considering serial modal logics – still, this is not the same situation as allowing the formula to have any number of diamonds, as seriality is only responsible for introducing at most one accessible world (for every serial modality) from any other. This is a nontrivial restriction, though, as we can see from this paper's results. Furthermore it corresponds well with the way justification formulas behave when tested for satisfiability.

For an extended version with omitted proofs the reader can see [1].

2 Modal Logics and Satisfiability

For the purposes of this paper it is convenient to consider modal formulas in negation normal form (NNF) – negations are pushed to the atomic level (to the propositional variables) and we have no implications. Note that for all logics we consider, every formula can be converted easily to its NNF form, so the NNF fragment of each logic we consider has exactly the same complexity as the full logic. We discuss modal logics with one, two, and three modalities, so we have three modal languages, $L_1 \subseteq L_2 \subseteq L_3$. They all include propositional variables, usually called $p_1, p_2, \ldots$ (but this may vary based on convenience) and $\bot$. If p is a propositional variable, then p and $\neg p$ are called literals and are also included in the language and so is $\neg\bot$, usually called $\top$. If ϕ, ψ are in one of these languages, so are $\phi \vee \psi$ and $\phi \wedge \psi$. Finally, if ϕ is in L_3, then so are $\Box_1\phi, \Box_2\phi, \Diamond_1\phi, \Diamond_2\phi, \Box_3\phi, \Diamond_3\phi$. L_2 includes all formulas in L_3 that have no $\Box_3, \Diamond_3$ and L_1 includes all formulas in L_2 that have no $\Box_2, \Diamond_2$. In short, L_n is defined in the following way, where $1 \leq i \leq n$: $\phi ::= p \mid \neg p \mid \bot \mid \neg\bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \Diamond_i \phi \mid \Box_i \phi$. If we consider formulas in L_1, $\Box_1$ may just be called $\Box$.[3]

A Kripke model for a trimodal logic (a logic based on language L_3) is a tuple $\mathcal{M} = (W, R_1, R_2, R_3, V)$, where $R_1, R_2, R_3 \subseteq W \times W$ and for every propositional variable p, $V(p) \subseteq W$. Then, (W, R_1, V) (resp. (W, R_1, R_2, V)) is a Kripke model for a unimodal (resp. bimodal) logic. Then, $(W, R_1), (W, R_1, R_2)$, and (W, R_1, R_2, R_3) are called frames and R_1, R_2, R_3 are called accessibility relations. We define the truth relation $\models$ between models, worlds (elements of W, also called states) and formulas in the following recursive way:

$\mathcal{M}, a \not\models \bot$;
$\mathcal{M}, a \models p$ iff $a \in V(p)$ and $\mathcal{M}, a \models \neg p$ iff $a \notin V(p)$;
$\mathcal{M}, a \models \phi \wedge \psi$ iff both $\mathcal{M}, a \models \phi$ and $\mathcal{M}, a \models \psi$;
$\mathcal{M}, a \models \phi \vee \psi$ iff $\mathcal{M}, a \models \phi$ or $\mathcal{M}, a \models \psi$;
$\mathcal{M}, a \models \Diamond_i \phi$ iff there is some $b \in W$ such that aR_ib and $\mathcal{M}, b \models \phi$;
$\mathcal{M}, a \models \Box_i \phi$ iff for all $b \in W$ such that aR_ib it is the case that $\mathcal{M}, b \models \phi$.

In this paper we deal with five logics: K, $\mathsf{D}_2 \oplus_\subseteq \mathsf{K}$, $\mathsf{D}_2 \oplus_\subseteq \mathsf{K4}$, $\mathsf{D} \oplus_\subseteq \mathsf{K4}$, and $\mathsf{D4}_2 \oplus_\subseteq \mathsf{K4}$. All except for K and $\mathsf{D} \oplus_\subseteq \mathsf{K4}$ are trimodal logics, based on language L_3, K is a unimodal logic (the simplest normal modal logic) based on L_1, and $\mathsf{D} \oplus_\subseteq \mathsf{K4}$ is a bimodal logic based on L_2. Each modal logic M is associated with a class of frames C. A formula ϕ is then called M-satisfiable iff there is a frame $\mathcal{F} \in C$, where C the class of frames associated to M, a model $\mathcal{M} = (\mathcal{F}, V)$, and a state a of $\mathcal{M}$ such that $\mathcal{M}, a \models \phi$. We say that $\mathcal{M}$ satisfies ϕ, or a satisfies ϕ in $\mathcal{M}$, or $\mathcal{M}$ models ϕ, or that ϕ is true at a.

K is the logic associated with the class of all frames;
$\mathsf{D}_2 \oplus_\subseteq \mathsf{K}$ is the logic associated with the class of frames (W, R_1, R_2, R_3) for which R_1, R_2 are serial (for every a there are b, c such that aR_1b, aR_2c) and $R_1 \cup R_2 \subseteq R_3$;
$\mathsf{D}_2 \oplus_\subseteq \mathsf{K4}$ is the logic associated with the class of frames $\mathcal{F} = (W, R_1, R_2, R_3)$ for which R_1, R_2 are serial, $R_1 \cup R_2 \subseteq R_3$, and R_3 is transitive;
$\mathsf{D} \oplus_\subseteq \mathsf{K4}$ is the logic associated with the class of frames $\mathcal{F} = (W, R_1, R_2)$ for which R_1 is serial, $R_1 \subseteq R_2$, and R_2 is transitive;
$\mathsf{D4}_2 \oplus_\subseteq \mathsf{K4}$ is the logic associated with the class of frames $\mathcal{F} = (W, R_1, R_2, R_3)$ for which R_1, R_2 are serial, $R_1 \cup R_2 \subseteq R_3$ and R_1, R_2, R_3 are transitive.

Tableau. A way to test for satisfiability is by using a tableau procedure. A good source on tableaux is [5]. We present tableau rules for K and for the diamond-free fragments of $\mathsf{D}_2 \oplus_\subseteq \mathsf{K}$ and then for the remaining three logics. The main reason we present these rules is because they are useful for later proofs and because they help to give intuition regarding the way we can test for satisfiability. The ones for K are classical and follow right away. Formulas used in the tableau are

[3] It may seem strange that we introduce languages with diamonds and then only consider their diamond-free fragments. When we discuss K, we consider the full language, so we introduce diamonds for L_1, L_2, L_3 for uniformity.

Table 1. Tableau rules for K.

$\dfrac{\sigma\ \phi \vee \psi}{\sigma\ \phi \mid \sigma\ \psi}$	$\dfrac{\sigma\ \phi \wedge \psi}{\begin{array}{c}\sigma\ \phi\\ \sigma\ \psi\end{array}}$	$\dfrac{\sigma\ \Box\phi}{\sigma.i\ \phi}$ where $\sigma.i$ has already appeared in the branch.	$\dfrac{\sigma\ \Diamond\phi}{\sigma.i\ \phi}$ where $\sigma.i$ has not yet appeared in the branch.

given a prefix, which intuitively corresponds to a state in a model we attempt to construct and is a string of natural numbers, with . representing concatenation. The tableau procedure for a formula ϕ starts from $0\ \phi$ and applies the rules it can to produce new formulas and add them to the set of formulas we construct, called a branch. A rule of the form $\frac{a}{b \mid c}$ means that the procedure nondeterministically chooses between b and c to produce, i.e. a branch is closed under that application of that rule as long as it includes b or c. If the branch has $\sigma\ \bot$, or both $\sigma\ p$ and $\sigma\ \neg p$, then it is called propositionally closed and the procedure rejects its input. Otherwise, if the branch contains $0\ \phi$, is closed under the rules, and is not propositionally closed, it is an accepting branch for ϕ; the procedure accepts ϕ exactly when there is an accepting branch for ϕ. The rules for K are in Table 1.

For the remaining logics, we are only concerned with their diamond-free fragments, so we only present rules for those to make things simpler. As we mention in the Introduction, all the logics we consider can be seen as regular grammar logics with converse ([8]), for which the satisfiability problem is in EXP. This already gives an upper bound for the satisfiability of $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$ (and for the general case of $(N, \subset, F)$ from Sect. 4). We present the tableau rules anyway (without proof), since it helps to visually give an intuition of each logic's behavior, while it helps us reason about how some logics reduce to others.

To give some intuition on the tableau rules, the main differences from the rules for K are that in a frame for these logics we have two or three different accessibility relations (lets assume for the moment that they are R_1, R_3, and possibly R_2), that one of them (R_3) is the (transitive closure of the) union of the others, and that we can assume that due to the lack of diamonds and seriality, R_1 and R_2 are total functions on the states. To establish this, notice that the truth of diamond-free formulas in NNF is preserved in submodels; when R_1, R_2 are not transitive, we can simply keep removing pairs from R_1, R_2 in a model as long as they remain serial. As for the tableau for $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$, notice that for $i = 1, 2$, R_i can map each state a to some c such that for every $\Box_i\psi$, subformula of ϕ, $c \models \Box_i\psi \rightarrow \psi$. If a is such a c, we map a to a; otherwise we can find such a c in the following way. Consider a sequence $bR_ic_1R_ic_2R_i\cdots$; if some $c_j \not\models \Box_i\psi \rightarrow \psi$, then $c_j \models \Box_i\psi$, so for every $j' > j$, $c_{j'} \models \Box_i\psi \rightarrow \psi$. Since the subformulas of ϕ are finite in number, we can find some large enough $j \in \mathbb{N}$ and set $c = c_j$. Notice that using this construction on c, R_i maps c to c, is transitive and serial.

The rules for $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K}$ are in Table 2. To come up with tableau rules for the other three logics, we can modify the above rules. The first two rules that cover

Table 2. The rules for $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K}$

$$\frac{\sigma\ \phi \vee \psi}{\sigma\ \phi \mid \sigma\ \psi} \qquad \frac{\sigma\ \phi \wedge \psi}{\begin{array}{c}\sigma\ \phi\\ \sigma\ \psi\end{array}} \qquad \frac{\sigma\ \Box_1\phi}{\sigma.1\ \phi} \qquad \frac{\sigma\ \Box_2\phi}{\sigma.2\ \phi} \qquad \frac{\sigma\ \Box_3\phi}{\begin{array}{c}\sigma.1\ \phi\\ \sigma.2\ \phi\end{array}}$$

the propositional cases are always the same, so we give the remaining rules for each case. In the following, notice that the resulting branch may be infinite. However we can simulate such an infinite branch by a finite one: we can limit the size of the prefixes, as after a certain size (up to $2^{|\phi|}$, where ϕ the tested formula) it is guaranteed that there will be two prefixes that prefix the exact same set of formulas. Thus, we can either assume the procedure terminates or that it generates a full branch, depending on our needs. In that latter case, to ensure a full branch is generated, we can give lowest priority to a rule when it generates a new prefix.

The rules for the diamond-free fragment of $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$ are in Table 3; the rules for the diamond-free fragment of $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$ in Table 4; and the rules for the diamond-free fragment of $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$ are in Table 5.

Proposition 1. *The satisfiability problem for the diamond-free fragments of* $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K}$*, of* $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$*, and of* $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$ *is in* PSPACE*; satisfiability for the diamond-free fragment of* $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$ *is in* EXP*.*

The cases of $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$ and $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$ are especially interesting. In [6], Demri established that $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$-satisfiability (and because of the following section's results also $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$-satisfiability) is EXP-complete. In this paper, though, we establish that the complexity of these two logics' diamond-free (and one-variable) fragments are PSPACE-complete (in this section we establish the PSPACE upper bounds, while in the next one the lower bounds), which is a drop in complexity

Table 3. Tableau rules for the diamond-free fragment of $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$

$$\frac{\sigma\ \Box_1\phi}{\sigma.1\ \phi} \qquad \frac{\sigma\ \Box_2\phi}{\sigma.2\ \phi} \qquad \frac{\sigma\ \Box_3\phi}{\begin{array}{c}\sigma.1\ \phi\\ \sigma.2\ \phi\\ \sigma.1\ \Box_3\phi\\ \sigma.2\ \Box_3\phi\end{array}}$$

Table 4. Tableau rules for the diamond-free fragment of $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$

$$\frac{\sigma\ \Box_1\phi}{\sigma.1\ \phi} \qquad \frac{\sigma\ \Box_2\phi}{\begin{array}{c}\sigma.1\ \phi\\ \sigma.1\ \Box_2\phi\end{array}}$$

Table 5. Tableau rules for the diamond-free fragment of $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$

$$\frac{\sigma\ \Box_1\phi}{n_1(\sigma)\ \phi} \qquad \frac{\sigma\ \Box_2\phi}{n_2(\sigma)\ \phi} \qquad \frac{\sigma\ \Box_3\phi}{\begin{array}{c} n_1(\sigma)\ \phi \\ n_2(\sigma)\ \phi \\ n_1(\sigma)\ \Box_3\phi \\ n_2(\sigma)\ \Box_3\phi \end{array}}$$

where $n_i(\sigma) = \sigma$ if $\sigma = \sigma'.i$ for some σ' and $n_i(\sigma) = \sigma.i$ otherwise.

(assuming PSPACE $\neq$ EXP), but not one that makes the problem tractable (assuming P $\neq$ PSPACE).

3 Lower Complexity Bounds

In this section we give hardness results for the logics of the previous section – except for K. In [4], the authors prove that the variable-free fragment of K remains PSPACE-hard. We make use of that result here and prove the same for the diamond-free, 1-variable fragment of $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K}$. Then we prove EXP-hardness for the diamond-free fragment of $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$ and PSPACE-hardness for the diamond-free fragments of $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$ and of $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$, which we later improve to the same result for the diamond-free, 1-variable fragments of these logics.

Proposition 2. *The diamond-free, 1-variable fragment of* $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K}$ *is* PSPACE-*complete.*

For the remaining logics we first present a lower complexity bound for their diamond-free fragments and then we can use translations to their 1-variable fragments to transfer the lower bounds to these fragments. We first treat the case of $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$.

Lemma 1. *The diamond-free fragment of* $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$ *is* EXP-*complete, while the diamond-free fragments of* $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$ *and of* $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$ *are* PSPACE-*complete.*

From Lemma 1, with some extra work, we can prove the following.

Proposition 3. *The 1-variable, diamond-free fragment of* $\mathsf{D}_2 \oplus_{\subseteq} \mathsf{K4}$ *is* EXP-*complete; the 1-variable, diamond-free fragments of* $\mathsf{D} \oplus_{\subseteq} \mathsf{K4}$ *and of* $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$ *are* PSPACE-*complete.*

One may wonder whether we can say the same for the variable-free fragment of these logics. The answer however is that we cannot. The models for these logics have accessibility relations that are all serial. This means that any two models are bisimilar when we do not use any propositional variables, thus any satisfiable formula is satisfied everywhere in any model, thus we only need one prefix for our tableau and we can solve satisfiability recursively on ϕ in polynomial time.

Then what about $\mathsf{D4} \oplus_{\subseteq} \mathsf{K4}$? Maybe we could attain similar hardness results for this logic as for $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$. Again, the answer is no. As frames for D4 come with a serial and transitive accessibility relation, frames for $\mathsf{D4} \oplus_{\subseteq} \mathsf{K4}$ are of the form (W, R_1, R_2), where $R_1 \subseteq R_2$, R_1, R_2 are serial, and R_1 is transitive. It is not hard to come up with the following tableau rule(s) for the diamond-free fragment, by adjusting the ones we gave for $\mathsf{D4}_2 \oplus_{\subseteq} \mathsf{K4}$ to simply produce $0.1\ \phi$ from every $\sigma\ \Box_i\phi$. This drops the complexity of satisfiability for the diamond-free fragment of $\mathsf{D4} \oplus_{\subseteq} \mathsf{K4}$ to NP (and of the diamond-free, 1-variable fragment to P), as we can only generate two prefixes during the tableau procedure. The following section explores when we can produce hardness results like the ones we gave in this section.

4 A General Characterization

In this section we examine a more general setting and we conclude by establishing tight conditions that determine the complexity of satisfiability of the diamond-free (and 1-variable) fragments of such multimodal logics.

A general framework would be to describe each logic with a triple $(N, \subset, F)$, where $N = \{1, 2, \ldots, |N|\} \neq \emptyset$, $\subset$ a binary relation on N, and for every $i \in N$, $F(i)$ is a modal logic; a frame for $(N, \subset, F)$ would be $(W, (R_i)_{i \in N})$, where for every $i \in N$, (W, R_i) a frame for $F(i)$ and for every $i \subset j$, $R_i \subset R_j$. It is reasonable to assume that $(N, \subset)$ has no cycles – otherwise we can collapse all modalities in the cycle to just one – and that $\subset$ is transitive. Furthermore, we also assume that all $F(i)$'s have frames with serial accessibility relations – otherwise there is either some $j \subseteq i$ for which $F(j)$'s frames have serial accessibility relations and $R(i)$ would inherit seriality from R_j, or when testing for satisfiability, $\Box_i\psi$ can always be assumed true by default (the lack of diamonds means that we do not need to consider any accessible worlds for modality i), which allows us to simply ignore all such modalities, making the situation not very interesting from an algorithmic point of view. Thus, we assume that $F(i) \in \{\mathsf{D}, \mathsf{T}, \mathsf{D4}, \mathsf{S5}\}$.[4,5] The cases for which $\subset = \emptyset$ have already had the complexity of their diamond-free (and other) fragments determined in [14]. For the general case, we already have an EXP upper bound from [8].

The reader can verify that $(N, \subset, F)$ is, indeed, a (fragment of a) regular grammar modal logic with converse. For example, $\mathsf{D}_2 \oplus_{\subset} \mathsf{D4}$ can easily be reduced to $\mathsf{K}_2 \oplus_{\subset} \mathsf{K4}$ by mapping ϕ to $\Diamond_1\top \wedge \Diamond_2\top \wedge \Box_3(\Diamond_1\top \wedge \Diamond_2\top) \wedge \phi$ to impose seriality, for which the corresponding regular languages would be $\Box_1$, $\Box_2$, and $(\Box_1 + \Box_2 + \Box_3)^*$ (see [8] for more on regular grammar modal logics with converse and their complexity and the extended version of this paper, [1], for more details on why $(N, \subset, F)$ belongs in that category).

[4] We can consider more logics as well, but these ones are enough to make the points we need. Besides, it is not hard to extend the reasoning of this section to other logics (ex. B, $\mathsf{S4}$, $\mathsf{KD45}$ and due to the observation above, also K, $\mathsf{K4}$), especially since the absence of diamonds makes the situation simpler.

[5] Frames for D have serial accessibility relations; frames for T have reflexive accessibility relations; frames for $\mathsf{D4}$ have serial and transitive accessibility relations; frames for $\mathsf{S5}$ have accessibility relations that are equivalence relations (reflexive, symmetric, transitive).

Table 6. Tableau rules for the diamond-free fragment of $(N, \subset, F)$

$\dfrac{\sigma\ \Box_i\phi}{\sigma\ \Box_j\phi}$	$\dfrac{\sigma\ \Box_i\phi}{n_i(\sigma)\ \phi}$	$\dfrac{\sigma\ \Box_i\phi}{\sigma\ \phi}$	$\dfrac{\sigma\ \Box_i\phi}{n_j(\sigma)\ \Box_i\phi}$
where $j \subset i$	where $i \in \min(N)$	where the frames of $F(i)$ have reflexive acc. relations	where $j \in \min(i)$ and $F(i)$'s frames have transitive acc. relations

For every $i \in N$, let

$$\min(i) = \{j \in N \mid j \subset i \text{ or } j = i, \text{ and } \nexists j' \subset j\}$$

and $\min(N) = \bigcup_{i \in N} \min(i)$. We can now give tableau rules for $(N, \subset, F)$. Let

- $n_i(\sigma) = \sigma$, if either
  - the accessibility relations of the frames for $F(i)$ are reflexive, or
  - $\sigma = \sigma'.i$ for some σ' and the accessibility relations of the frames for $F(i)$ are transitive;
- $n_i(\sigma) = \sigma.i$, otherwise.

The tableau rules appear in Table 6.

From these tableau rules we can reestablish EXP-upper bounds for all of these cases (see the previous sections). To establish correctness, we only show how to construct a model from an accepting branch for ϕ, as the opposite direction is easier. Let W be the set of all the prefixes that have appeared in the branch. The accessibility relations are defined in the following (recursive) way: if $i \in \min(N)$, then $R_i = \{(\sigma, n_i(\sigma)) \in W^2\} \cup \{(\sigma, \sigma) \in W^2 \mid n_i(\sigma) \notin W$ or $F(i)$ has reflexive frames$\}$; if $i \notin \min(N)$ and the frames of $F(i)$ do not have transitive or reflexive accessibility relations, then $R_i = \bigcup_{j \subseteq i} R_j$; if $i \notin \min(N)$ and the frames of $F(i)$ do have transitive (resp. reflexive, resp. transitive and reflexive) accessibility relations, then R_i is the transitive (resp. reflexive, resp. transitive and reflexive) closure of $\bigcup_{j \subseteq i} R_j$. Finally, (as usual) $V(p) = \{w \in W \mid w\ p$ appears in the branch$\}$. Again, to show that the constructed model satisfies ϕ, we use a straightforward induction.

By taking a careful look at the tableau rules above, we can already make some simple observations about the complexity of the diamond-free fragments of these logics. Modalities in $\min(N)$ have an important role when determining the complexity of a diamond-free fragment. In fact, the prefixes that can be produced by the tableau depend directly on $\min(N)$.

Lemma 2. *If for every $i \in \min(N)$, $F(i)$ has frames with reflexive accessibility relations ($F(i) \in \{\mathsf{T}, \mathsf{S5}\}$), then the satisfiability problem for the diamond-free fragment of $(N, \subset, F)$ is* NP*-complete and the satisfiability problem for the diamond-free, 1-variable fragment of $(N, \subset, F)$ is in* P.

Corollary 1. *If* $\min(N) \subseteq \{i\} \cup A$ *and* $F(i)$ *has frames with transitive accessibility relations* ($F(i) \in \{\mathsf{D4}, \mathsf{S5}\}$) *and for every* $j \in A$, $F(j)$ *has frames with reflexive accessibility relations, then the satisfiability problem for the diamond-free fragment of* $(N, \subset, F)$ *is* NP*-complete and the satisfiability problem for the diamond-free, 1-variable fragment of* $(N, \subset, F)$ *is in* P.

In [6], Demri shows that satisfiability for $\mathsf{L}_1 \oplus_\subseteq \mathsf{L}_2 \oplus_\subseteq \cdots \oplus_\subseteq \mathsf{L}_n$ is EXP-complete, as long as there are $i < j \leq n$ for which $\mathsf{L}_i \oplus_\subseteq \mathsf{L}_j$ is EXP-hard. On the other hand, Corollary 1 shows that for all these logics, their diamond-free fragment is in NP, as long as L_1 has frames with transitive (or reflexive) accessibility relations.

Finally, we can establish general results about the complexity of the diamond-free fragments of these logics. For this, we introduce some terminology. We call a set $A \subset N$ *pure* if for every $i \in A$, $F(i)$'s frames do not have the condition that their accessibility relation is reflexive (given our assumptions, $F[A] \cap \{\mathsf{T}, \mathsf{S5}\} = \emptyset$). We call a set $A \subset N$ *simple* if for some $i \in A$, $F(i)$'s frames do not have the condition that their accessibility relation is transitive (given our assumptions, $F[A] \cap \{\mathsf{D}, \mathsf{T}\} \neq \emptyset$). An agent $i \in N$ is called pure (resp. simple) if $\{i\}$ is pure (resp. simple).

Theorem 1. *1. If there is some* $i \in N$ *and some pure* $A \subseteq \min(i)$ *for which* $F(i)$ *has frames with transitive accessibility relations* ($F(i) \in \{\mathsf{D4}, \mathsf{S5}\}$) *and either*
- $|A| = 2$ *and* A *is simple, or*
- $|A| = 3$,

then the satisfiability problem for the diamond-free, 1-variable fragment of $(N, \subset, F)$ *is* EXP*-complete;*

2. otherwise, if there is some $i \in N$ *and some pure* $A \subseteq \min(i)$ *for which either*
- $|A| = 2$ *and there is some pure and simple* $j \in \min(N)$, *or*
- $|A| = 3$,

then the satisfiability problem for the diamond-free, 1-variable fragment of $(N, \subset, F)$ *is* PSPACE*-complete;*

3. otherwise, if there is some $i \in N$ *and some pure* $A \subseteq \min(i)$ *for which* $F(i)$ *has frames with transitive accessibility relations* ($F(i) \in \{\mathsf{D4}, \mathsf{S5}\}$) *and either*
- $|A| = 1$ *and* A *is simple or*
- $|A| = 2$,

then the satisfiability problem for the diamond-free (1-variable) fragment of $(N, \subset, F)$ *is* PSPACE*-complete;*

4. otherwise the satisfiability problem for the diamond-free (resp. and 1-variable) fragment of $(N, \subset, F)$ *is* NP*-complete (resp. in* P*).*

5 Final Remarks

We examined the complexity of satisfiability for the diamond-free fragments and the diamond-free, 1-variable fragments of multimodal logics equipped with an inclusion relation $\subset$ on the modalities, such that if $i \subset j$, then in every frame $(W, R_1, \ldots, R_n)$ of the logic, $R_i \subseteq R_j$ (equivalently, $\Box_j \to \Box_i$ is an axiom).

We gave a complete characterization of these cases (Theorem 1), determining that, depending on $\subset$, every logic falls into one of the following three complexity classes: NP (P for the 1-variable fragments), PSPACE, and EXP – Theorem 1 actually distinguishes four possibilities, depending on the way we prove each bound. We argued that to have nontrivial complexity bounds we need to consider logics based on frames with at least serial accessibility relations, which is a notable difference in flavor from the results in [13,14].

One direction to take from here is to consider further syntactic restrictions and Boolean functions in the spirit of [14]. Another would be to consider different classes of frames. Perhaps it would also make sense to consider different types of natural relations on the modalities and see how these results transfer in a different setting. From a Parameterized Complexity perspective there is a lot to be done, such as limiting the modal depth/width, which are parameters that can remain unaffected from our ban on diamonds. For the cases where the complexity of the diamond-free, 1-variable fragments becomes tractable, a natural next step would be to examine whether we can indeed use the number of diamonds as a parameter for an FPT algorithm to solve satisfiability.

Another direction which interests us is to examine what happens with more/different kinds of relations on the modalities. An example would be to introduce the axiom $\Box_i\phi \rightarrow \Box_j\Box_i\phi$, a generalization of Positive Introspection. This would be of interest in the case of the diamond-free fragments of these systems, as it brings us back to our motivation in studying the complexity of Justification Logic, where such systems exist. Hardness results like the ones we proved in this paper are not hard to transfer in this case, but it seems nontrivial to immediately characterize the complexity of the whole family.

Acknowledgments. The author is grateful to anonymous reviewers, whose input has greatly enhanced the quality of this paper.

References

1. Achilleos, A.: Modal Logics with Hard Diamond-free Fragments. CoRR abs/1401.5846 (2014)
2. Achilleos, A.: Interactions and Complexity in Multi-Agent Justification Logic. Ph.D. thesis, The City University of New York (2015)
3. Artemov, S.: The logic of justification. Rev. Symbolic Logic **1**(4), 477–513 (2008)
4. Chagrov, A.V., Rybakov, M.N.: How many variables does one need to prove PSPACE-hardness of modal logics. In: Advances in Modal Logic, pp. 71–82 (2002)
5. Agostino, M., Gabbay, D.M., Hähnle, R., Posegga, J. (eds.): Handbook of Tableau Methods. Springer, Heidelberg (1999)
6. Demri, S.: Complexity of simple dependent bimodal logics. In: Dyckhoff, R. (ed.) TABLEAUX 2000. LNCS, vol. 1847, pp. 190–204. Springer, Heidelberg (2000)
7. Demri, S.: The complexity of regularity in grammar logics and related modal logics. J. Logic Comput. **11**(6), 933–960 (2001). http://logcom.oxfordjournals.org/content/11/6/933.abstract

8. Demri, S., De Nivelle, H.: Deciding regular grammar logics with converse through first-order logic. J. Logic Lang. Inform. **14**(3), 289–329 (2005). http://dx.doi.org/10.1007/s10849-005-5788-9
9. Gabbay, D.M., Kurucz, A., Wolter, F., Zakharyaschev, M. (eds.): Many-Dimensional Modal Logics Theory and Applications. Studies in Logic and the Foundations of Mathematics, vol. 148. Elsevier, North Holland (2003)
10. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.Y.: Reasoning About Knowledge. The MIT Press, Cambridge (1995)
11. Halpern, J.Y.: The effect of bounding the number of primitive propositions and the depth of nesting on the complexity of modal logic. Artif. Intell. **75**, 361–372 (1995)
12. Halpern, J.Y., Moses, Y.: A guide to completeness and complexity for modal logics of knowledge and belief. Artif. Intell. **54**(3), 319–379 (1992)
13. Hemaspaandra, E.: The complexity of poor Man's logic. J. Logic Comput. **11**(4), 609–622 (2001)
14. Hemaspaandra, E., Schnoor, H., Schnoor, I.: Generalized modal satisfiability. J. Comput. Syst. Sci. **76**(7), 561–578 (2010). http://www.sciencedirect.com/science/article/pii/S0022000009001007
15. Kurucz, A.: Combining modal logics. Stud. Logic Pract. Reasoning **3**, 869–924 (2007)
16. Kuznets, R.: Complexity Issues in Justification Logic. Ph.D. thesis, CUNY Graduate Center, May 2008
17. Ladner, R.E.: The computational complexity of provability in systems of modal propositional logic. SIAM J. Comput. **6**(3), 467–480 (1977). http://link.aip.org/link/?SMJ/6/467/1
18. Marx, M., Venema, Y.: Multi-dimensional Modal Logic. Springer, Heidelberg (1997)
19. Nguyen, L.A., Szałas, A.: EXPTIME tableau decision procedures for regular grammar logics with converse. Stud. Logic. **98**(3), 387–428 (2011)
20. Spaan, E.: Complexity of Modal Logics. Ph.D. thesis, University of Amsterdam (1993)