

# Modal Separability of Fixpoint Formulae

Jean Christoph Jung<sup>1</sup>, Jędrzej Kołodziejcki<sup>1</sup>

<sup>1</sup>TU Dortmund University, Germany

## Abstract

We study *modal separability* for fixpoint formulae: given two mutually exclusive fixpoint formulae  $\varphi, \varphi'$ , decide whether there is a modal formula  $\psi$  that separates them, that is, that satisfies  $\varphi \models \psi \models \neg\varphi'$ . This problem has applications for finding simple reasons for inconsistency. Our main contributions are tight complexity bounds for deciding modal separability and optimal ways to compute a separator if it exists. More precisely, it is EXPTIME-complete in general and PSPACE-complete over words. Separators can be computed in doubly exponential time in general and in exponential time over words, and this is optimal as well. The results for general structures transfer to arbitrary, finitely branching, and finite trees. The word case results hold for finite, infinite, and arbitrary words.

## Keywords

Modal Logic, Fixpoint Logic, Separability, Interpolation

## 1. Introduction

For given logics  $\mathcal{L}, \mathcal{L}^+$ , the  $\mathcal{L}$ -separability problem for  $\mathcal{L}^+$  is to decide given two  $\mathcal{L}^+$ -formulae  $\varphi, \varphi'$  whether there is an  $\mathcal{L}$ -formula  $\psi$  that separates  $\varphi$  and  $\varphi'$  in the sense that  $\varphi \models \psi$  and  $\psi \models \neg\varphi'$ . Obviously, a separator can only exist when  $\varphi$  and  $\varphi'$  are mutually exclusive, and the problem is only meaningful when  $\mathcal{L}$  is less expressive than  $\mathcal{L}^+$ . Intuitively, a separator formulated in a “simpler” logic  $\mathcal{L}$  explains a given inconsistency in a “complicated” logic  $\mathcal{L}^+$ . Note that, for logics  $\mathcal{L}^+$  closed under negation,  $\mathcal{L}$ -separability generalizes the  $\mathcal{L}$ -definability problem for  $\mathcal{L}^+$ : decide whether a given  $\mathcal{L}^+$ -formula is equivalent to an  $\mathcal{L}$ -formula. Indeed,  $\varphi \in \mathcal{L}^+$  is equivalent to an  $\mathcal{L}$ -formula iff  $\varphi$  and  $\neg\varphi$  are  $\mathcal{L}$ -separable. Since separability is more general than definability, solving it requires an even better understanding of the expressive power of the logics under consideration.


**Example 1.** Consider  $\mathcal{L}$  being the modal logic ML, also known under the name  $\mathcal{ALC}$  in the context of description logics. Expressions of the logic (called formulae in ML terminology and concepts in description logic parlance) describe properties of colored, directed graphs with a distinguished point called the root. As the more expressive  $\mathcal{L}^+$  take PDL: the extension of ML with regular modalities (in DL terms: the extension  $\mathcal{ALC}_{reg}$  of  $\mathcal{ALC}$  with regular role expressions). Assume the graphs under consideration have edges labelled with colors  $A, B$  and  $C$  and consider properties:

$P$ : “There is a path from the root whose labeling belongs to  $A^+B$ .”

$P'$ : “The labeling of every (finite) path from the root belongs to  $C^*$ .”


These (contradictory) properties are expressed by PDL-formulae  $\varphi$  and  $\varphi'$  and it is easy to see that none of them can be expressed in the weaker ML. Nonetheless,  $\varphi$  and  $\varphi'$  are separated by a simple ML-formula  $\psi$  that says: “there is an  $A$ -labelled edge from the root”. Thus,  $\psi$  serves as an easy explanation of the inconsistency of  $\varphi$  and  $\varphi'$ .

Generalizing the example, in this paper we investigate ML-separability of formulae in the modal  $\mu$ -calculus  $\mu\text{ML}$  [1, 2], which extends PDL [3].  $\mu\text{ML}$  is a general framework capturing logics supporting

 DL 2024: 37th International Workshop on Description Logics, June 18–21, 2024, Bergen, Norway

 jean.jung@tu-dortmund.de (J. C. Jung); jedrzej.kolodziejcki@tu-dortmund.de (J. Kołodziejcki)

 <https://www.mimuw.edu.pl/~jkol/> (J. Kołodziejcki)

 0000-0002-4159-2255 (J. C. Jung); 0000-0001-5008-9224 (J. Kołodziejcki)



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

fixpoints that is relevant both for knowledge representation and reasoning and for verification. It describes all bisimulation-invariant properties definable in MSO [4, Theorem 11] and thus encompasses virtually all specification languages such as LTL and CTL [5].

Our results generalize the ML-definability problem for  $\mu\text{ML}$  which was shown decidable by Otto [6, Main Theorem]. The adaptation of the argument to the more general separability is relatively easy. However, Otto’s paper is focussed on deciding the existence of modal definitions. The problem of computing a definition when it exists is not discussed, and it seems that the formula which can be read off from the proof is at least tower-exponential big. This issue was addressed in [7]. Unfortunately, the approach in [7], although constructive, does not easily generalize from definability to separability. We fill the gap by providing a procedure which is *both* fully constructive and works for the more general separation case. Both the constructed formulae and the running time are optimal, as illustrated by suitable examples and reductions.

We consider both general models and “word models” which are Kripke structures in which each point has at most one successor. The latter are relevant from a verification perspective and for temporal reasoning. In order to obtain our results we first prove model-theoretic characterizations in terms of bisimulations. We then exploit the close connection of  $\mu\text{ML}$  to nondeterministic parity tree automata to give (1) optimal procedures for the separability problem and (2) upper bounds on the modal depth of a separator, if it exists. In (1) we show  $\text{EXPTIME}$ -completeness of modal separability in general and  $\text{PSPACE}$ -completeness over words. The lower bounds are essentially inherited from satisfiability. The upper bounds derived in (2) are then used together with the automata to compute so-called *n-uniform consequences*, that is, modal formulae that have exactly the same modal consequences as a given  $\mu\text{ML}$ -formula, up to modal depth  $n$ . These  $n$ -uniform consequences are then used as separators. Also here, our procedures are optimal: they compute separators of at most double exponential size, and we show that there are  $\mu\text{ML}$ -formulae that are expressible in ML but any equivalent ML-formulae must have doubly exponential size. This means that there is a double exponential succinctness gap between  $\mu\text{ML}$  and ML. In the word case, our procedures compute exponentially sized separators and there is only an exponential succinctness gap. All lower bounds (both computational and succinctness) already hold for PDL ( $\mathcal{ALC}_{reg}$ ) in place of  $\mu\text{ML}$ , and for definability in place of separability.

It is interesting to note that our results hold over classes of models definable by  $\mu\text{ML}$ -formulae. This observation allows us to cover the more general notion of separation in presence of an *ontology* (i.e. a background theory imposing some conditions on models). As long as the ontology is expressible in  $\mu\text{ML}$ , separability and computation of separators reduce to the ontology-free setting. Without much effort the same observation lets us transfer our results to finite words, infinite words, and finite trees.

All the missing proofs and constructions can be found in the full version [8].

**Related Work.** Separability has been intensively studied in formal language theory. A seminal result in this area is that separability of regular word languages by a first-order language is decidable in  $\text{EXPTIME}$  [9]. Since  $\mu\text{ML}$  over words defines precisely the regular languages and first-order logic captures ML, this is particularly related to our results over words.

In logic, a recent work investigates the complexity of separating between formulae supporting counting quantifiers by formulae that do not support them [10]. The used techniques exploit compactness, which makes them inapplicable to our case and inherently non-constructive.

Another related problem is the question of interpolant existence. An *interpolant* of two  $\mathcal{L}$ -formulae  $\varphi$  and  $\varphi'$  is an  $\mathcal{L}$ -formula  $\psi$  with  $\varphi \models \psi \models \varphi'$  and such that the signature of  $\psi$  is contained in the signatures of both  $\varphi$  and  $\varphi'$ . Thus, the problem resembles separability but the restriction on  $\psi$  is in terms of the signature instead of in terms of the logic. Sometimes this question reduces to entailment, as many logics enjoy the *Craig interpolation property*: an interpolant of  $\varphi$  and  $\varphi'$  exists whenever  $\varphi \models \varphi'$ . Interpolant existence for logics that lack Craig interpolation has recently been studied in [11, 12]. The used tools, however, are similar in nature to the ones from [10] and therefore inapplicable to our problem.

Finally, a related problem is *separability of data examples*. There, the task is to separate sets of pointed

structures instead of formulae (see [13, 14] and the references therein). Separability of data examples can be cast as an instance of (our logical notion of) separability if  $\mathcal{L}^+$  is expressive enough to describe the data examples. Conversely,  $\mathcal{L}$ -separability of formulae  $\varphi$  and  $\varphi'$  is the same as data separability of the (possibly infinite) sets of their models by an  $\mathcal{L}$ -formula.

## 2. Preliminaries

Assuming familiarity of the reader with modal logic and the modal  $\mu$ -calculus, we recall here only the main notions and refer to [15] for more details.

**Syntax.** We consider modal logic ML and its fixpoint extension  $\mu$ ML over a *modal signature* consisting of two finite sets: *actions* Act and *propositions* Prop. The syntax of ML is given as:

$$\varphi ::= \top \mid \perp \mid \tau \mid \neg\tau \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \langle a \rangle \varphi \mid [a]\varphi$$

with  $\tau \in \text{Prop}$  and  $a \in \text{Act}$ . If  $\text{Act} = \{a\}$  is a singleton, we use  $\diamond\varphi$  and  $\square\varphi$  in place of  $\langle a \rangle \varphi$  and  $[a]\varphi$ . The syntax of  $\mu$ ML is obtained by extending the above with additional clauses:

$$\varphi ::= x \mid \mu x. \varphi \mid \nu x. \varphi$$

where  $x$  belongs to a fixed set Var of variables. The restriction to a fixed finite signature is only for the sake of readability. All results in the paper remain true with arbitrary signature.

**Semantics.** The models we consider are pointed Kripke structures. That is, a model  $\mathcal{M}$  consists of a set  $M$  (called its *universe*) with a distinguished point  $v_I \in M$  called the *root*, an interpretation  $\xrightarrow{a} \subseteq M \times M$  for every  $a \in \text{Act}$  and a valuation  $\text{val} : M \rightarrow \mathcal{P}(\text{Prop})$ . We call the set  $\mathcal{P}(\text{Prop})$  *colors* and denote it by  $\Sigma$ . Both ML and  $\mu$ ML are interpreted in points of models in a standard way. Since models are by definition pointed we write  $\mathcal{M} \models \varphi$  meaning that the root of  $\mathcal{M}$  satisfies  $\varphi$ . The same symbol denotes entailment:  $\varphi \models \psi$  means that every model of  $\varphi$  is a model of  $\psi$ . In the case only models from some fixed class  $\mathcal{C}$  are considered we talk about satisfiability and entailment *over*  $\mathcal{C}$  and in the latter case write  $\varphi \models_{\mathcal{C}} \psi$ .

A particularly relevant class of models are trees. A model  $\mathcal{M}$  is a *tree* if the underlying directed graph  $(M, \bigcup\{\xrightarrow{a} \mid a \in \text{Act}\})$  is a tree with  $v_I$  as its root. The *branching* or *outdegree* of a point is the number of its children in this underlying graph. The class of all trees is denoted by Trees. We identify *words* (both finite and infinite) over alphabet  $\Sigma$  with trees over a single action of outdegree at most one. Points of such models are interpreted as positions in the word, the unique accessibility relation represents the successor relation, and the valuation determines the letter at each position. A *prefix* of a tree is a subset of its universe closed under taking ancestors. When no confusion arises we identify a prefix  $N \subseteq M$  with the induced subtree  $\mathcal{N}$  of  $\mathcal{M}$  that has  $N$  as its universe. The *depth* of a point is the distance from the root. The prefix of depth  $n$  (or just *n-prefix*) is the set of all points at depth at most  $n$  and is denoted by  $M|_n$  (and the corresponding subtree by  $\mathcal{M}|_n$ ).

We define bisimulations and bisimilarity in a standard way except that in the case of trees for convenience we assume that bisimulations only link points at the same depth. An *n-step bisimulation* (or just *n-bisimulation*) between trees  $\mathcal{M}$  and  $\mathcal{N}$  is a bisimulation between their *n-prefixes*. We denote *n-bisimilarity* by  $\Leftrightarrow^n$ .

**Size of formulae.** The *size* of a formula  $\varphi$ , denoted  $|\varphi|$ , is the number of nodes in its syntax tree. Similarly, its *depth* is the maximal length of paths in the syntax tree. The depth of a formula should not be confused with its *modal depth* which is the maximal nesting of modal operators; all formulae of modal depth at most  $n$  are denoted  $\text{ML}^n$ .

When we specify formulae in the paper, we use syntactic sugar  $\bigvee \Phi$ ,  $\bigwedge \Phi$ , and *nabla*  $\nabla \Phi$ , for finite sets of formulae  $\Phi$ . The first two are self-explanatory and allow for higher branching in the syntax

tree. The last one,  $\nabla\Phi$ , intuitively means that “every formula in  $\Phi$  is true in some child and every child satisfies some formula from  $\Phi$ ” and is an abbreviation for

$$\nabla\Phi = \bigwedge_{\varphi \in \Phi} \diamond\varphi \wedge \bigvee_{\varphi \in \Phi} \square\varphi. \quad (1)$$

It is well-known that  $\bigvee\Phi$  and  $\bigwedge\Phi$  can be rewritten into basic syntax under polynomial cost. We also include the colors  $\Sigma$  directly in the syntax:  $c \in \Sigma$  is a shorthand for the formula  $\bigwedge\{\tau, \neg\tau' \mid \tau \in c, \tau' \notin c\}$ . Rewriting colors increases the size only by a factor linear in  $|\text{Prop}|$ .

**Automata.** Throughout the paper we use automata over tree models of both bounded and arbitrary outdegree. A *nondeterministic parity tree automaton (NPTA)* is a tuple  $\mathcal{A} = (Q, \Sigma, q_I, \delta, \text{rank})$  where  $Q$  is a finite set of states,  $q_I \in Q$  is the initial state,  $\Sigma$  is the alphabet fixed above, and  $\text{rank}$  assigns each state a priority. The transition function  $\delta$  is of type:

$$\delta : Q \times \Sigma \rightarrow \mathcal{P}(\mathcal{P}(Q)).$$

Intuitively,  $\delta(q, c) = \{S_1, \dots, S_l\}$  means that in the state  $q$  upon reading color  $c$  the automaton (i) chooses a transition  $S_i$  and (ii) labels all the children of the current point with states from  $S_i$  so that every  $p \in S_i$  is assigned to some child. A run of  $\mathcal{A}$  on a tree  $\mathcal{M}$  is an assignment  $\rho : M \rightarrow Q$  consistent with  $\delta$  in such sense and sending the root of the tree to  $q_I$ . The run is accepting if for every infinite path  $v_0, v_1 \dots$  in  $\mathcal{M}$  the sequence  $\text{rank}(\rho(v_0)), \text{rank}(\rho(v_1)), \dots$  satisfies the parity condition. We write  $\mathcal{M} \models \mathcal{A}$  in case  $\mathcal{A}$  has an accepting run on  $\mathcal{M}$ . An automaton that is identical to  $\mathcal{A}$  except that the original initial state is replaced with  $q$  is denoted  $\mathcal{A}[q_I \leftarrow q]$ . We refer with NPWA to an NPTA working over words.

In NPTAs over trees of bounded outdegree  $k$  it might be more common to use a transition function of type  $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q^k)$ , but the difference is not essential: our NPTAs can be represented in this way and conversely, all relevant constructions for such NPTAs can be adapted to our setting. Most importantly, we rely on the following classical result (see for example the discussion in [16] and the well-presented Dealternation Theorem 5.7 in [17]):

**Theorem 1.** *For every  $\mu\text{ML}$ -formula  $\varphi$ , we can construct an equivalent NPTA  $\mathcal{A}$ , that is,  $\mathcal{M} \models \varphi$  iff  $\mathcal{M} \models \mathcal{A}$ , for every tree  $\mathcal{M}$ , with number of states at most exponential in  $|\varphi|$ . If we consider models of bounded outdegree  $k$  then  $\mathcal{A}$  is computed in exponential time, otherwise in doubly exponential time.*

### 3. Foundations of Separability

We start with recalling the notion of separability and discuss some of its basic properties.

**Definition 1.** *Given  $\varphi, \varphi' \in \mu\text{ML}$ , a modal separator of  $\varphi, \varphi'$  is  $\psi \in \text{ML}$  with  $\varphi \models \psi$  and  $\psi \models \neg\varphi'$ . It is a modal separator over a class  $\mathcal{C}$  if  $\varphi \models_{\mathcal{C}} \psi$  and  $\psi \models_{\mathcal{C}} \neg\varphi'$ .*

The notion induces the problem of *modal separability*: given two  $\mu\text{ML}$ -formulae  $\varphi, \varphi'$ , decide whether a modal separator exists. Clearly, ML-definability of  $\varphi$  or  $\varphi'$  is a sufficient condition for the existence of a modal separator between  $\varphi, \varphi'$ . However, Example 1 shows that it is not a necessary one: neither  $\varphi$  nor  $\varphi'$  are ML-definable, yet a separator exists. We make some foundational observations.

Inspired by the notion of Craig interpolation, one could also consider the notion of a *Craig modal separator*, which is a modal separator  $\psi$  of  $\varphi, \varphi'$  which only uses symbols occurring in both  $\varphi$  and  $\varphi'$ . However, based on the fact that ML enjoys Craig interpolation, we show in Theorem 2 (proof in the full version) that Craig modal separability and modal separability coincide. Since ML enjoys Craig interpolation over many classes of models [18, Theorem 1], Theorem 2 remains true over all classes of models considered below. We thus focus on modal separability.

**Theorem 2.**  *$\varphi, \varphi' \in \mu\text{ML}$  admit a modal separator iff they admit a Craig modal separator.*

Inspired by the notion of *uniform interpolation* [19, 20], it is natural to ask whether every  $\varphi \in \mu\text{ML}$  admits a *uniform modal separator*, that is, a formula  $\psi \in \text{ML}$  that is a modal separator of  $\varphi, \varphi'$  for every  $\varphi' \in \mu\text{ML}$  with  $\varphi \models \neg\varphi'$ . However, substituting  $\neg\varphi$  for  $\varphi'$  we get that the uniform modal separator  $\psi$  for  $\varphi$  is actually equivalent to  $\varphi$ . Consequently, a  $\mu\text{ML}$ -formula has a uniform modal separator iff it is modally definable. This is contrast with the fact that both  $\text{ML}$  [19] and  $\mu\text{ML}$  [20] enjoy uniform interpolation.

Since  $\mu\text{ML}$  has both the finite model property and the (finitely branching) tree model property, the notions of a modal separator over finite models, arbitrary tree models, and finitely branching tree models all coincide with modal separator (over arbitrary models). Unsurprisingly, this does not apply to the class of all finite trees.

**Example 2.** Consider a  $\mu\text{ML}$ -formula  $\varphi_\infty = \nu x. \diamond x$  expressing that there exists an infinite path originating in the root. It is satisfiable, but unsatisfiable over finite trees. Thus  $\perp$  is an  $\text{ML}$ -definition of  $\varphi_\infty$  over finite trees, but  $\varphi_\infty$  is not  $\text{ML}$ -definable (over arbitrary models).

We deal with separability over finite trees as follows. Call a class  $\mathcal{C}$  of models  $\mu\text{ML}$ -definable in  $\mathcal{D}$  if there is a  $\mu\text{ML}$ -formula  $\theta$  such that  $\mathcal{M} \in \mathcal{C}$  iff  $\mathcal{M} \models \theta$ , for all models  $\mathcal{M} \in \mathcal{D}$ .

**Lemma 1.** Let  $\mathcal{C}$  be  $\mu\text{ML}$ -definable in  $\mathcal{D}$  by  $\theta$  and let  $\psi \in \text{ML}$ . Then  $\psi$  is a modal separator of  $\varphi, \varphi' \in \mu\text{ML}$  over  $\mathcal{C}$  iff  $\psi$  is a modal separator of  $\theta \wedge \varphi$  and  $\theta \wedge \varphi'$  over  $\mathcal{D}$ .

Intuitively, Lemma 1 provides us with a reduction of modal separability over  $\mathcal{C}$  to modal separability over (the larger)  $\mathcal{D}$ . It has a number of interesting consequences. First, observe that the formula  $\neg\varphi_\infty$  from Example 2 defines the class of finite trees in the class of all finitely branching trees. Hence  $\neg\varphi_\infty$  provides a reduction of modal separability over finite trees to modal separability over finitely branching trees, and thus to modal separability. Similarly, and again using  $\varphi_\infty$ , Lemma 1 reduces modal separability over finite words and over infinite words to modal separability over (arbitrary) words. Finally, the lemma can be used to reduce modal separability relative to background knowledge to modal separability. Call  $\psi$  a modal separator of  $\varphi, \varphi' \in \mu\text{ML}$  relative to  $\theta_0 \in \mu\text{ML}$  if it is a modal separator of  $\varphi, \varphi'$  over the class of models satisfying  $\theta_0$  in every point. This setting is most relevant for the DL community since  $\theta_0$  plays the role of an ontology. In particular, the question whether two  $\mathcal{ALC}_{\text{reg}}$ -concepts  $\varphi, \varphi'$  are separable by an  $\mathcal{ALC}$ -concept relative to an  $\mathcal{ALC}_{\text{reg}}$ -ontology is an instance of that problem (recall that every  $\mathcal{ALC}_{\text{reg}}$ -concept can be expressed as a  $\mu\text{ML}$ -formula). Let  $\theta$  be the  $\mu\text{ML}$ -formula expressing that  $\theta_0$  is satisfied in every point reachable via the accessibility relations. Using Lemma 1 and bisimulation invariance of  $\mu\text{ML}$ , it is routine to verify that  $\psi$  is a modal separator of  $\varphi, \varphi'$  relative to  $\theta_0$  iff  $\psi$  is a modal separator of  $\theta \wedge \varphi$  and  $\theta \wedge \varphi'$ .

In view of what was said so far, we will from now on concentrate on deciding modal separability over general and word models and computing separators if they exist. A main ingredient for both tasks is to show that if there is a modal separator for  $\mu\text{ML}$ -formula  $\varphi, \varphi'$ , then there is one of modal depth  $n$  at most exponential in the size of  $\varphi$  and  $\varphi'$ . As a necessary tool for showing this exponential bound on  $n$ , and for efficiently deciding if a given  $n$  suffices, we establish the following model-theoretic characterization. Fix  $\varphi, \varphi' \in \mu\text{ML}$  for the rest of the paper and denote their size by  $k = |\varphi| + |\varphi'|$ .

**Proposition 1.** Let  $n \in \mathbb{N}$ . The following are equivalent:

- (i) There is  $\psi \in \text{ML}$  of modal depth  $n$  separating  $\varphi$  and  $\varphi'$ ;
- (ii) For all models  $\mathcal{M}$  and  $\mathcal{M}'$  bisimilar up to depth  $n$ :  $\mathcal{M} \models \varphi$  implies  $\mathcal{M}' \not\models \varphi'$ ;
- (iii) For all trees  $\mathcal{M}$  and  $\mathcal{M}'$  identical up to depth  $n$ :  $\mathcal{M} \models \varphi$  implies  $\mathcal{M}' \not\models \varphi'$ ;
- (iv) For all trees  $\mathcal{M}$  and  $\mathcal{M}'$  identical up to depth  $n$  and whose branching is bounded by  $k$ :  $\mathcal{M} \models \varphi$  implies  $\mathcal{M}' \not\models \varphi'$ .

The proof can be found in the full version. Based on Proposition 1, we show that  $\text{ML}$ -separability of  $\mu\text{ML}$ -formulae is  $\text{EXPTIME}$ -complete and thus not harder than  $\text{ML}$ -definability.

**Theorem 3.** *Modal separability of  $\mu\text{ML}$ -formulae is  $\text{EXPTIME}$ -complete over arbitrary models.*

$\text{EXPTIME}$ -hardness already holds for  $\text{ML}$ -definability and is proved by an immediate reduction from  $\mu\text{ML}$ -satisfiability, which is  $\text{EXPTIME}$ -complete already for its fragment  $\text{PDL}$  [3, Section 4]. It is not hard to modify the original hardness proof for  $\text{PDL}$ -satisfiability to work over finite trees, so Theorem 3 remains valid over finite trees as well. For the upper bound, we mostly follow the technical development in [6]. Thanks to Proposition 1 separability is equivalent to the existence of  $n \in \mathbb{N}$  for which condition (iv) holds. This can be expressed as an  $\text{MSO}$  statement about the full  $k$ -ary tree, and thus decided. However, for optimal complexity and to extract bounds that we use later we apply a lower-level automata-theoretic analysis.

Over words, we essentially follow the same approach. Since the tree automata used in the proof of Theorem 3 can be replaced by word automata, the complexity drops to  $\text{PSPACE}$ . A matching lower bound can be derived as above by a reduction from satisfiability in  $\text{LTL}$  over words [21, Theorem 4.1] (which, in fact, can be rephrased in terms of  $\text{PDL}$ ).

**Theorem 4.** *Modal separability of  $\mu\text{ML}$ -formulae is  $\text{PSPACE}$ -complete over words.*

The proofs for both cases, the case of arbitrary models and the case of words, can be found in the full version. As announced, an important step in the proofs of the upper bounds is the following proposition which we will also use later.

**Proposition 2.** *If  $\varphi, \varphi' \in \mu\text{ML}$  are separable then they are separable by a formula of modal depth  $l$  exponential in their size  $k$ . The same is true over words.*

In the remainder of the paper we will deal with computing separators based on Proposition 2. Before we proceed, let us note that our approach differs from the treatment of modal definability from [7]. There, the authors rewrite given  $\varphi$  into modal  $\psi$  in such a way that if the initial  $\varphi$  is modally definable then  $\varphi$  and  $\psi$  are equivalent. In the case when  $\varphi$  is not modally definable, however, the output  $\psi$  is rather random. For example,  $\psi$  obtained from the formula  $\varphi_\infty$  from Example 2 is equivalent to  $\perp$  which is not even a consequence of  $\varphi_\infty$ . Thus, a different construction is needed to obtain separators. We will actually compute something slightly stronger that might be of independent interest.

**Definition 2.** *Given  $\varphi \in \mu\text{ML}$  and  $n \in \mathbb{N}$ , a formula  $\psi \in \text{ML}^n$  is an  $n$ -uniform consequence of  $\varphi$  if, for all  $\theta \in \text{ML}^n$ :*

$$\varphi \models \theta \quad \iff \quad \psi \models \theta$$

*An analogous notion relative to a fixed class  $\mathcal{C}$  of models is obtained by replacing  $\models$  with  $\models_{\mathcal{C}}$ .*

In words:  $\psi$  is an  $n$ -uniform consequence of  $\varphi$  if it has modal depth  $n$ , is a consequence of  $\varphi$ , and entails every other consequence of  $\varphi$  of modal depth  $n$ . In particular, if  $\varphi$  and  $\varphi'$  are separable by some modal formula of modal depth  $n$  and  $\psi$  is an  $n$ -uniform consequence of  $\varphi$ , then this  $\psi$  separates  $\varphi$  from  $\varphi'$  as well. Observe that  $n$ -uniform consequences exist for every  $\varphi \in \mu\text{ML}$  and  $n \in \mathbb{N}$ . Indeed, given  $\varphi$  and  $n$  we can obtain an  $n$ -uniform consequence  $\psi$  of  $\varphi$  by taking the disjunction of all  $\text{ML}^n$ -types consistent with  $\varphi$ . Here, by an  $\text{ML}^n$ -type we mean a maximal consistent subset of  $\text{ML}^n$ . Since up to equivalence there are only finitely many formulae in  $\text{ML}^n$ , each  $\text{ML}^n$ -type can be represented as a single  $\text{ML}^n$ -formula and the mentioned disjunction  $\psi$  is well-defined.

In view of Proposition 2, it thus suffices to compute  $n$ -uniform consequences of  $\varphi$ . Unfortunately, the naive construction given above is nonelementary in the size of the separated formulae  $\varphi$  and  $\varphi'$ . In the next sections we give better constructions.

## 4. Optimal Separators: Arbitrary Models

We construct doubly exponentially sized separators and provide matching lower bounds.

## 4.1. Construction

**Theorem 5.** *If  $\varphi$  and  $\varphi'$  are modally separable then a separator  $\varphi$  of size doubly exponential in  $k = |\varphi| + |\varphi'|$  exists and can be computed in doubly exponential time.*

The above is a consequence of the following lemma.

**Lemma 2.** *For every  $\varphi \in \mu\text{ML}$  and  $n \in \mathbb{N}$ , one can construct an  $n$ -uniform consequence  $\psi_n \in \text{ML}^n$  of  $\varphi$  with branching doubly exponential in  $|\varphi|$  and depth linear in  $n$ .*

We show how Theorem 5 follows from Lemma 2. Proposition 2 guarantees that if a modal separator for  $\varphi$  and  $\varphi'$  exists then there is one with modal depth  $l$  exponential in  $k$ . Since  $\psi_l$  entails this separator it follows that  $\psi_l$  is a separator itself.

The branching  $m$  of  $\psi_l$  is at most doubly exponential in  $|\varphi|$  and thus also in  $k$ :  $m \leq 2^{2^{k^x}}$  for some constant  $x$ . The depth  $d$  of  $\psi_l$  is linear in  $l$  and therefore  $d \leq 2^{k^y}$  for some  $y$ . Altogether this means that the size of  $\psi_l$ :

$$|\psi_l| \leq m^d \leq (2^{2^{k^x}})^{2^{k^y}}$$

is at most doubly exponential in  $k$ . It remains to prove Lemma 2.

*Proof.* Let  $\mathcal{A} = (Q, \Sigma, q_I, \delta, \text{rank})$  be the NPTA equivalent to  $\varphi$  with exponentially many states, which exists due to Theorem 1. For each  $n \in \mathbb{N}$  and  $q \in Q$  we construct  $\psi_{n,q} \in \text{ML}^n$  of branching  $2^{2^{|Q|}}$  such that:

$$\mathcal{M} \models \psi_{n,q} \iff \text{there exists } \mathcal{N} \models \mathcal{A}[q_I \leftrightarrow q] \text{ with } \mathcal{M} \cong^n \mathcal{N} \quad (2)$$

for every structure  $\mathcal{M}$ . Then,  $\psi_{n,q_I}$  is our desired  $n$ -uniform consequence  $\psi_n$  of  $\varphi$ .

We proceed by induction on  $n \in \mathbb{N}$ . For the base case we put:

$$\psi_{0,q} = \bigvee \{c \in \Sigma \mid \text{there is } \mathcal{N} \models \mathcal{A}[q_I \leftrightarrow q] \text{ with } \mathcal{N} \models c\}$$

which clearly satisfies the induction goal (2). For the induction step define:

$$\psi_{n+1,q} = \bigvee_{c \in \Sigma} \bigvee_{S \in \delta(q,c)} c \wedge \nabla \{\psi_{n,p} \mid p \in S\}.$$

The proof that the above construction preserves (2) is not hard and can be found in the full version.  $\square$

Let us remark that Lemma 2 can be easily adapted to deal with vocabulary restrictions. That is, given  $P \subseteq \text{Prop}$  we could construct  $\psi_n^P$  similar to  $\psi_n$  but only using atomic propositions from  $P$  and only entailing  $\text{ML}^n$ -consequences of  $\varphi$  whose vocabulary is contained in  $P$ . To that end, it suffices to project-out atomic propositions not in  $P$  from the automaton  $\mathcal{A}$  and only then proceed with our construction. Such  $(P, n)$ -uniform consequence  $\psi_n^P$  of  $\varphi$  can then be taken as a Craig modal separator, in the same way as  $\psi_n$  serves as a modal separator.

## 4.2. Lower bounds

For the lower bounds, we show that over arbitrary structures (in fact, already binary trees)  $\mu\text{ML}$  is doubly exponentially more succinct than  $\text{ML}$ . The example is essentially taken from [22, Section 3.1]. There the authors use game-theoretic tools which are later applied to more complicated cases. Since we are only interested in this example, we provide a straightforward self-contained argument.

**Proposition 3.** *There is a sequence  $(\varphi_n)_{n \in \mathbb{N}}$  of  $\mu\text{ML}$ -formulae of size polynomial in  $n$  such that each  $\varphi_n$  is equivalent to a  $\text{ML}$ -formula but every  $\psi \in \text{ML}$  equivalent to  $\varphi_n$  has size at least  $2^{2^n}$ .*

*Proof.* We only give a sketch, the details are found in the full version. We assume two different actions  $a$  and  $b$ . For each  $n \in \mathbb{N}$  consider the property:

$B_n$ : “No path (over all actions) longer than  $2^n$  starts in the root.”

This can be enforced by encoding an  $n$ -bit binary counter into the structure of the model, and requiring that on every path the counter values are strictly increasing. Let  $C_n$  be this (technically stronger) property expressing the behavior of the encoded counter. Assuming that the encoding is reasonably efficient,  $C_n$  can be easily expressed by a  $\mu$ ML-formula  $\varphi_n$  of size polynomial in  $n$  (in fact, a weak fragment of PDL is already sufficient). Since the lengths of paths are bounded,  $C_n$  can be also expressed in ML.

However, every  $\psi \in \text{ML}$  equivalent to  $\varphi_n$  has size at least  $2^{2^n}$ . The reason is that for every sequence of actions  $a$  and  $b$  of length  $2^n$ , the syntax tree of  $\psi$  must contain a descending sequence of subformulae of length  $2^n$  such that the  $i$ -th subformula begins with a modal operator corresponding to the  $i$ -th action. This allows to embed a binary tree of height  $2^n$  into the syntax tree of  $\psi$ .  $\square$

Note that the presence of two different actions  $a$  and  $b$  is essential for the argument. We conjecture that  $\mu$ ML is doubly exponentially more succinct than ML already in the monomodal setting. Consider the following Property  $P_n$ , parameterized by  $n \in \mathbb{N}$ :

$P_n$ : “ $C_n$  and there exists a maximal path on which the number of points satisfying  $\tau$  is even.”

where  $C_n$  is the same as in Proposition 3. It is not difficult to come up with small, that is, of size polynomial in  $n$ ,  $\mu$ ML-formulae  $\varphi'_n$  expressing  $P_n$ . Unfortunately, proving that no small ML formula can be equivalent to  $\varphi'_n$  seems difficult. For instance, consider models where every non-leaf point has a child satisfying  $\tau$  and a child satisfying  $\neg\tau$ . Then a trick similar to the famous example of Potthoff (showing, roughly, that the language of all binary trees of even depth is first-order definable) [23, Example 1] can be exploited to get a modal formula equivalent to  $\varphi'_n$  (over such models), but of size only single exponential in  $n$ . Moreover, the results in the next Section 5 show that looking at words only is not sufficient either.

## 5. Optimal Separators: Word Case

In this section we show that optimal modal separators (over words) can be computed exponentially faster and are exponentially smaller compared to the case with arbitrary models.

**Theorem 6.** *If  $\varphi$  and  $\varphi'$  are modally separable over words, then a separator of size exponential in  $|\varphi| + |\varphi'|$  exists and can be computed in exponential time.*

As with arbitrary models, Proposition 2 gives an upper bound on the modal depth of a separator and so it suffices to construct  $n$ -uniform consequences of  $\varphi$  of small size.

We illustrate the idea first. Consider the classes  $\text{EVEN}_n$  and  $\text{ODD}_n$ ,  $n \in \mathbb{N}$  of all word structures of length  $n$  in which proposition  $a$  is satisfied in an even and odd, respectively, number of points. Constructing modal formulae  $\varphi_n$  and  $\varphi'_n$  defining  $\text{EVEN}_n$  and  $\text{ODD}_n$  in the following, naive way leads to exponential formulae since  $\varphi_{i+1}$  contains both  $\varphi_i$  and  $\varphi'_i$ :

$$\begin{aligned} \varphi_0 &= \neg a \wedge \Box \perp & \varphi_{i+1} &= \Diamond \top \wedge ((a \wedge \varphi'_i) \vee (\neg a \wedge \varphi_i)) \\ \varphi'_0 &= a \wedge \Box \perp & \varphi'_{i+1} &= \Diamond \top \wedge ((a \wedge \varphi_i) \vee (\neg a \wedge \varphi'_i)) \end{aligned}$$

This exponential blow-up can be avoided, however, using “divide-and-conquer” as follows:

$$\begin{aligned} \varphi_{2n} &= (\varphi_n \wedge \Diamond^n \varphi_n) \vee (\varphi'_n \wedge \Diamond^n \varphi'_n) \\ \varphi'_{2n} &= (\varphi_n \wedge \Diamond^n \varphi'_n) \vee (\varphi'_n \wedge \Diamond^n \varphi_n) \end{aligned}$$

Although several copies of formulae of smaller index are used as well, but since the index is halved, we end up with formulae of roughly quadratic size. The proof of the following analogue of Lemma 2 relies on this idea.



**Lemma 3.** *For every  $n \in \mathbb{N}$  and every NPWA  $\mathcal{A}$  with states  $Q$ , one can construct a formula  $\psi_n \in \text{ML}^n$  which is  $\mathcal{A}$ 's  $n$ -uniform consequence over words and has size polynomial in  $n$  and  $|Q|$ . The construction requires polynomial time.*

To see that Lemma 3 implies Theorem 6, let  $\varphi$  and  $\varphi'$  admit a modal separator over words. Let  $\mathcal{A}$  be an NPWA that is equivalent to  $\varphi$ . By Theorem 1,  $\mathcal{A}$  has exponentially many states and can be computed in exponential time. Proposition 2 implies that there is a modal separator of modal depth  $l$  at most exponential in  $k = |\varphi| + |\varphi'|$ . As with arbitrary models,  $\mathcal{A}$ 's  $l$ -uniform consequence  $\psi_l$  from Lemma 3 is the sought separator. We now prove the lemma.

*Proof.* Let  $\mathcal{A} = (Q, \Sigma, \delta, q_I, \text{rank})$  be an NPWA. The main idea is to construct, for every  $p, q \in Q$  and  $m \in \mathbb{N}$ , a formula  $\psi_{p,q}^m$  such that for every input word  $\mathcal{M}$ :

$$\mathcal{M} \models \psi_{p,q}^m \iff \text{there is a run from } p \text{ to } q \text{ over the } m\text{-prefix of } \mathcal{M},$$

The key step is the recursive splitting similar to the definitions of  $\text{EVEN}_n$  and  $\text{ODD}_n$  above. Intuitively,  $\psi_{p,q}^{2m}$  is the disjunction over all  $s \in Q$  of the conditions “there is a run from  $p$  in the initial position to  $s$  in position  $m$ , and a run from  $s$  in position  $m$  to  $q$  in position  $2m$ .” The latter conditions are recursively expressed using  $\psi_{p,s}^m$  and  $\psi_{s,q}^m$ . The constructed formulas  $\psi_{p,q}^m$ ,  $m \leq n$  are then used to describe all possible  $n$ -prefixes of models of  $\mathcal{A}$ . The details of the construction are described in the full version.  $\square$

We conclude the section with the comment that Theorem 6 is optimal in the sense that there are modally separable formulae which require a large separator. We actually show the following stronger statement implying that, over words,  $\mu\text{ML}$  is exponentially more succinct than ML.

**Proposition 4.** *There is a sequence of  $\mu\text{ML}$ -formulae  $(\varphi_n)_{n \in \mathbb{N}}$  of size polynomial in  $n$  such that each  $\varphi_n$  is equivalent to a ML-formula but every  $\psi \in \text{ML}$  equivalent to  $\varphi_n$  has size at least  $2^n$ .*

The proof is entirely standard. The main idea is that, already in PDL one can stipulate (with a small formula) a finite word of exponential length. Clearly, any ML-formula expressing this requires exponential size. The only difficulty is doing it with a fixed signature: instead of encoding  $i$ -bit counters using  $i$  propositions, we use just two propositions and encode numbers in  $i$  consecutive points.

## 6. Conclusion and Open Problems

We have studied the problem of deciding separability of  $\mu\text{ML}$ -formulae by fixpoint free formulae from ML, and computing separators if they exist. Our results cover several interesting classes of models such as trees, finite trees, and words. Due to the great expressivity of  $\mu\text{ML}$  the results remain valid in the presence of ontologies.

A notably missing case is the class of trees of fixed outdegree  $d$  independent from formulae. This is surprisingly different from the classes we studied. The key difficulty here lies in the fact that the implication (iii)  $\Rightarrow$  (ii) from Proposition 1 is not true over such trees.

An intriguing challenge left for future study is to look at extensions of  $\mu\text{ML}$  and/or ML. Natural extensions are inverse modalities, the universal modality, graded modalities, and constants (corresponding to inverse roles, the universal role, counting quantifiers, and nominals in DL speech). We expect the adaptation to inverse modalities to be only minor. Also graded modalities look innocent if they are allowed both in the larger logic and in the separator logic. If we only extend  $\mu\text{ML}$  with graded modalities and ask for separators in ML (without graded modalities), we would have to combine our techniques with the ones from [10], which is potentially challenging. We expect universal modality and/or constants to pose more technical difficulties as well. Intuitively, adding a universal modality or constants leads to the loss of the strong locality underlying Proposition 1.

**Acknowledgements** We are grateful to the anonymous referees for their valuable remarks.

## References

- [1] V. R. Pratt, A decidable mu-calculus: Preliminary report, in: Proceedings of 22nd Annual Symposium on Foundations of Computer Science (FOCS), IEEE Computer Society, 1981, pp. 421–427. URL: <https://doi.org/10.1109/SFCS.1981.4>. doi:10.1109/SFCS.1981.4.
- [2] D. Kozen, Results on the propositional mu-calculus, *Theor. Comput. Sci.* 27 (1983) 333–354. URL: [https://doi.org/10.1016/0304-3975\(82\)90125-6](https://doi.org/10.1016/0304-3975(82)90125-6). doi:10.1016/0304-3975(82)90125-6.
- [3] M. J. Fischer, R. E. Ladner, Propositional dynamic logic of regular programs, *J. Comput. Syst. Sci.* 18 (1979) 194–211. URL: [https://doi.org/10.1016/0022-0000\(79\)90046-1](https://doi.org/10.1016/0022-0000(79)90046-1). doi:10.1016/0022-0000(79)90046-1.
- [4] D. Janin, I. Walukiewicz, On the expressive completeness of the propositional mu-calculus with respect to monadic second order logic, in: Proceedings of Concurrency Theory (CONCUR), 1996, pp. 263–277.
- [5] C. Baier, J. Katoen, Principles of model checking, MIT Press, 2008.
- [6] M. Otto, Eliminating recursion in the  $\mu$ -calculus, in: Proceedings of 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS), volume 1563 of *Lecture Notes in Computer Science*, Springer, 1999, pp. 531–540. URL: [https://doi.org/10.1007/3-540-49116-3\\_50](https://doi.org/10.1007/3-540-49116-3_50). doi:10.1007/3-540-49116-3\_50.
- [7] K. Lehtinen, S. Quickert, Deciding the first levels of the modal mu alternation hierarchy by formula construction, in: Proceedings of Annual Conference on Computer Science Logic CSL, volume 41 of *LIPICs*, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015, pp. 457–471. URL: <https://doi.org/10.4230/LIPICs.CSL.2015.457>. doi:10.4230/LIPICs.CSL.2015.457.
- [8] J. C. Jung, J. Kołodziejcki, Modal separability of fixpoint formulae, 2024. arXiv:2406.01497.
- [9] T. Place, M. Zeitoun, Separating regular languages with first-order logic, *Log. Methods Comput. Sci.* 12 (2016). URL: [https://doi.org/10.2168/LMCS-12\(1:5\)2016](https://doi.org/10.2168/LMCS-12(1:5)2016). doi:10.2168/LMCS-12(1:5)2016.
- [10] L. Kuijter, T. Tan, F. Wolter, M. Zakharyashev, Separating counting from non-counting in fragments of two-variable first-order logic (extended abstract), in: Proc. of DL 2024, 2024.
- [11] J. C. Jung, F. Wolter, Living without beth and craig: Definitions and interpolants in the guarded and two-variable fragments, in: Proceedings of Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), IEEE, 2021, pp. 1–14. URL: <https://doi.org/10.1109/LICS52264.2021.9470585>. doi:10.1109/LICS52264.2021.9470585.
- [12] A. Artale, J. C. Jung, A. Mazzullo, A. Ozaki, F. Wolter, Living without beth and craig: Definitions and interpolants in description and modal logics with nominals and role inclusions, *ACM Trans. Comput. Log.* 24 (2023) 34:1–34:51. URL: <https://doi.org/10.1145/3597301>. doi:10.1145/3597301.
- [13] D. M. L. Martins, Reverse engineering database queries from examples: State-of-the-art, challenges, and research opportunities, *Inf. Syst.* 83 (2019) 89–100. URL: <https://doi.org/10.1016/j.is.2019.03.002>. doi:10.1016/J.IS.2019.03.002.
- [14] J. C. Jung, C. Lutz, H. Pulcini, F. Wolter, Logical separability of labeled data examples under ontologies, *Artif. Intell.* 313 (2022) 103785. URL: <https://doi.org/10.1016/j.artint.2022.103785>. doi:10.1016/J.ARTINT.2022.103785.
- [15] P. Blackburn, J. F. A. K. van Benthem, F. Wolter (Eds.), Handbook of Modal Logic, volume 3 of *Studies in logic and practical reasoning*, North-Holland, 2007. URL: <https://www.sciencedirect.com/bookseries/studies-in-logic-and-practical-reasoning/vol/3/suppl/C>.
- [16] M. Y. Vardi, Reasoning about the past with two-way automata, in: Proceedings of International Colloquium Automata, Languages and Programming (ICALP), volume 1443 of *Lecture Notes in Computer Science*, Springer, 1998, pp. 628–641. URL: <https://doi.org/10.1007/BFb0055090>. doi:10.1007/BFb0055090.
- [17] M. Bojańczyk, W. Czerwiński, Automata Toolbox, University of Warsaw, 2018. URL: <https://www.mimuw.edu.pl/~bojan/papers/toolbox.pdf>.
- [18] M. Marx, Interpolation in modal logic, in: International Conference on Algebraic Methodology and Software Technology, 1999. URL: <https://api.semanticscholar.org/CorpusID:13397497>.
- [19] A. Visser, Uniform interpolation and layered bisimulation, in: Gödel '96: Logical Foundations

of Mathematics, Computer Science and Physics - Kurt Gödel's Legacy, Lecture Notes in Logic, Springer, 1996, p. 139–164. doi:10.1017/9781316716939.010.

- [20] G. D'Agostino, M. Hollenberg, Uniform interpolation, automata and the modal  $\mu$ -calculus, in: Proceedings of Advances in Modal Logic, CSLI Publications, 1996, pp. 73–84.
- [21] A. P. Sistla, E. M. Clarke, The complexity of propositional linear temporal logics, J. ACM 32 (1985) 733–749. URL: <https://doi.org/10.1145/3828.3837>. doi:10.1145/3828.3837.
- [22] T. French, W. van der Hoek, P. Iliev, B. P. Kooi, On the succinctness of some modal logics, Artif. Intell. 197 (2013) 56–85. doi:<https://doi.org/10.1016/j.artint.2013.02.003>.
- [23] A. Potthoff, First-order logic on finite trees, in: Theory and Practice of Software Development, 1995.