Some Exponential Lower Bounds on Formula-size in Modal Logic

Hans van Ditmarsch[1]

CNRS, LORIA Université de Lorraine

Jie Fan[2]

Department of Philosophy
Peking University

Wiebe van der Hoek[3]

Department of Computer Science
University of Liverpool

Petar Iliev[4]

CNRS, LORIA, Université de Lorraine

Abstract

We present two families of exponential lower bounds on the size of modal formulae and use them to establish the following succinctness results. We show that the logic of contingency (ConML) is exponentially more succinct than basic modal logic (ML). We strengthen the known proofs that the so-called public announcement logic (PAL) in a signature containing at least two different diamonds and one propositional symbol is exponentially more succinct than ML by showing that this is already true for signatures that contain only one diamond and one propositional symbol. As a corollary of these results, we obtain an alternative proof of the fact that modal circuits are exponentially more succinct than ML-formulae.

Keywords: lower bounds on formula size, succinctness of modal logics, public announcement logic, modal logic of contingency.

[1] hans.van-ditmarsch@loria.fr

[2] fanjie@pku.edu.cn

[3] wiebe@csc.liv.ac.uk

[4] petar.iliev@loria.fr

1 Introduction

Unlike computational complexity theory where the question of proving lower bounds on the size of Boolean formulae and Boolean circuits computing a given Boolean function is a central challenge of the whole field, it seems that the general problem of proving lower bounds on formula and circuit-size in modal logic (ML) has not attracted much attention outside the field of temporal logics.

As far as we know, the first lower bound on the size of ML-formulae comes from [17] where it is shown that any ML-formula locally corresponding to the first-order condition $\forall y \forall z((xRy \wedge xRz) \rightarrow (yRz \vee zRy \vee y = z))$ contains at least two different propositional variables. After that, the interest in proving such bounds was focussed mainly on temporal logics with one of the first results, derivable from [15], being that there is a sequence of first-order formulae with three variables $\varphi_1, \varphi_2, \ldots$ for which there is a polynomial p with the property that, for $n \geq 1$, the length of φ_n is less or equal to $p(n)$, but there is no sequence of temporal formulae $\psi_1, \psi_2, \ldots$ such that φ_i is equivalent on ω-words to ψ_i and the lengths of the formulae $\psi_1, \psi_2, \ldots$ can be bounded from above by an elementary function of their indices, i.e., first-order logic is non-elementarily more succinct than temporal logic on ω-words. Two lower bounds on the size of temporal formulae deserve special mention. The first one [18] is that every computation tree logic (or μ-calculus) formula expressing the property *there is a path along which there are n positions v_1, v_2, …, v_n (not necessary in this order) satisfying the propositions p_1, p_2, …, p_n respectively* must have size at least $\frac{2^n}{\sqrt{n}}$. This estimate was improved in [1] to $n!$. The second is from [4], in which it is shown that every formula of the linear-time temporal logic (LTL) expressing the property *for any two positions v and w on a path π, if $\pi, v \models p_i$ iff $\pi, w \models p_i$ for any $1 \leq i \leq n$, then $\pi, v \models p_{n+1}$ if and only if $\pi, w \models p_{n+1}$, i.e., if v and w agree on the first n propositions, then they agree on p_{n+1} too* has size at least 2^n. Using the property *any position v on a path π that agrees with the initial position v_0 on p_1, p_2, …, p_n must also agree on p_{n+1}* it was proven in [9] that any LTL formula that expresses this property, contains only future temporal operators, and is evaluated at the initial position of the path has size at least 2^n.

In contrast to temporal logic, results on lower bounds on formula size in the general setting of modal logic seem to be scarce. Besides [17], we would like to mention the following articles. In [11], a modal language with 5 boxes: $[R]$, $[id]$, $[\neg S]$, $[S_1 \cap S_2]$, and $[S^-]$ is studied. As usual, R is an atomic binary relation on the underlying Kripke structure. The more complex modalities are obtained from the identity relation id, the complement $\neg S$ of S, the intersection $S_1 \cap S_2$ of S_1 and S_2, and the converse S^- of the relation S. The authors proved that, for any n, any formula in this language that defines the property *the carrier set of the Kripke model has cardinality at least* 2^n has size at least 2^{n-1}.

Ehrenfeucht-Fraïssé games were used in [13] to show that every ML-formula that “says” *there is a point that satisfies the proposition p reachable from the current point in at most* 2^n *steps* must have modal depth of at least 2^n.

In [16], it is shown, among other things, that, for any $n \geq 1$ and any k such that $2^n < k \leq 2^{n+1}$, any ML-formula that modally defines the property *the current point has less than k successors* contains at least $n+1$ different propositional symbols.

Two other papers that establish lower bounds on the size of ML-formulae are [5] and [10]. In the former, using a technique developed in [1], the authors proved an exponential lower bound on the size of certain ML-formulae. In the latter, it is established that, on the class of all Kripke models (**K**), there is no equivalence-preserving translation from public announcement logic (a conservative extension of ML that has a popular epistemic interpretation and is usually denoted by PAL) to ML that produces an equivalent ML-formula of sub-exponential length for every PAL-formula, i.e., PAL is exponentially more succinct than ML on **K** provided that there are at least two different boxes $[a]$, $[b]$ and one propositional symbol in both logics. If there are at least four boxes and four propositional symbols in both logics, it was shown in [6] that PAL is exponentially more succinct than ML on $\mathbf{S}_5$-models, too.

In the present paper,

- we strengthen the result from [10] in yet another way by proving that the presence of two different boxes is not necessary to show that PAL is exponentially more succinct than ML on **K** (one box is enough);
- we prove that the logic of contingency [12] which is strictly less expressive on **K** than ML is nevertheless exponentially more succinct than ML on **K**;
- we use the above results to give an alternative proof of the fact that modal circuits are exponentially more succinct than modal formulae [8].

2 Technical Preliminaries

Let P be a countable set of propositional symbols. We fix a modal signature $S = \{P, \neg, \wedge, \vee, \Box, \Diamond\}$. The definition of modal formulae over S, Kripke models and the truth of a formula φ in a point m of a Kripke model $\mathcal{M}$, written $(\mathcal{M}, m) \models \varphi$ is standard [2]. *Boolean* formulae are modal formulae that do not contain the $\Box$ and $\Diamond$ operators. We call the pair $(\mathcal{M}, m)$ a *pointed model.* Sets of pointed models are denoted $\mathbb{A}, \mathbb{B} \ldots$. We write $\mathbb{M} \models \varphi$ to mean that $(\mathcal{M}, w) \models \varphi$ for all $(\mathcal{M}, w) \in \mathbb{M}$. We would like to stress that $\mathbb{M}$ may be empty. In this case, it is trivially true that, for all $(\mathcal{M}, w) \in \emptyset$, we have $(\mathcal{M}, w) \models \varphi$ and therefore, $\emptyset \models \varphi$. Abusing notation, we write $m \in \mathcal{M}$ to mean that m is a node or a point of the carrier set of $\mathcal{M}$. The binary relation in the Kripke model $\mathcal{M}$ that is used to interpret the operators $\Box$ and $\Diamond$ is denoted $R^{\mathcal{M}}$ or simply R when no confusion arises. We assume that the reader is familiar with the notion of bisimulation between pointed models and that two bisimilar pointed models satisfy the same formulae over S [2]. We proceed now to the definition of our main technical tool that is motivated as follows.

Our goal is to prove exponential lower bounds on the size of formulae φ that express a certain property P of pointed models. By "size", we mean the length of φ as a string over the alphabet S. Intuitively, it will be helpful if we have a tool that is tailored simultaneously to some useful approximation of our notion of size of the formulae φ and to the fact that they can differentiate between models that have the property P and those that do not. One such tool can be found by defining the length of any φ to be the number of nodes of its syntax tree T_φ. In addition, we must add some new features to syntax trees in order to be able to reason about formulae that differentiate between Kripke models that have a given property from the ones that do not. Extended syntax trees were introduced in [7] in the setting of first order logic and can be used as a formalisation of the above intuition [5].

As its name suggests, an extended syntax tree of a modal formula φ is just the usual syntax tree of φ where, apart from a syntax label that is a symbol from S, each node has a semantic label that is a pair of sets of pointed models $\langle \mathbb{M}, \mathbb{N} \rangle$. A node η with a semantic label $\langle \mathbb{M}, \mathbb{N} \rangle$ will be denoted $\mathbb{M} \circ \mathbb{N}$ when no confusion arises. The pointed models in $\mathbb{M}$ are called *the models on the left of* η. Similarly, the pointed models in $\mathbb{N}$ are called *the models on the right of* η. To simplify our exposition, we write $\texttt{lft}(\eta)$ to mean the set of models on the left of η (in this case, $\texttt{lft}(\eta) = \mathbb{M}$) and, similarly, $\texttt{rght}(\eta)$ to mean the set of models on the right of η (in the present case, $\texttt{rght}(\eta) = \mathbb{N}$).

We begin by defining a number of useful operations on pointed models.

Definition 2.1 Let $(\mathcal{M}, w)$ be a pointed model and $\mathbb{M}$ be a (not necessarily non-empty) set of pointed models. Then

- $$\Box(\mathcal{M}, w) = \{(\mathcal{M}, v) \mid v \in \mathcal{M} \text{ and } wR^{\mathcal{M}}v\}.$$

  Intuitively, $\Box(\mathcal{M}, w)$ is the set of all pointed models that can be reached from w by making one $R^{\mathcal{M}}$-step. Note that if there is no point $v \in \mathcal{M}$ such that $wR^{\mathcal{M}}v$, then $\Box(\mathcal{M}, w) = \emptyset$.
- If $(\mathcal{M}, w) \models \Diamond\psi$, then there is at least one $v \in \mathcal{M}$ such that $wR^{\mathcal{M}}v$ and $(\mathcal{M}, v) \models \psi$. We construct the non-empty set of all such pointed models

  $$\Diamond\psi(\mathcal{M}, w) = \{(\mathcal{M}, v) \mid v \in \mathcal{M} \text{ such that } wR^{\mathcal{M}}v \text{ and } (\mathcal{M}, v) \models \psi\}.$$
- $\Box(\mathbb{M})$ is defined as

  $$\Box(\mathbb{M}) = \bigcup_{(\mathcal{M}, w) \in \mathbb{M}} \Box(\mathcal{M}, w).$$

  It is obvious that $\Box(\mathbb{M})$ is empty when $\Box(\mathcal{M}, w) = \emptyset$ for each $(\mathcal{M}, w) \in \mathbb{M}$ or when $\mathbb{M} = \emptyset$.
- If $\mathbb{M} \models \Diamond\psi$, then, we form the set of pointed models

  $$\Diamond\psi(\mathbb{M}) = \bigcup_{(\mathcal{M}, w) \in \mathbb{M}} \Diamond\psi(\mathcal{M}, w).$$

[5] Readers familiar with [1] can easily see that these trees can be thought of as closed game trees for suitably defined versions of the Adler-Immerman games introduced in [1]. Essentially the same tool was used in [5] under the name *uniform strategy trees*.

It is easy to see that $\Diamond\psi(\mathbb{M}) = \emptyset$ iff $\mathbb{M} = \emptyset$.

We are ready to define extended syntax trees of ML-formulae. For convenience, we are working with *formulae in negation normal form*, i.e., formulae in which $\neg$ can appear only in front of propositional symbols. From now on, and unless otherwise stated, a *formula* means an ML-formula in negation normal form. As usual, a *literal* l is a propositional symbol p or its negation $\neg p$. We denote the set of all literals by LIT. For a symbol $s \in LIT \cup \{\wedge, \vee, \Box, \Diamond\}$, we write $synl(\eta) = s$ to mean that the node η has the syntax label s.

Definition 2.2 [Extended Syntax Trees] For any formula φ and any sets of pointed models $\mathbb{M}$ and $\mathbb{N}$ such that $\mathbb{M} \models \varphi$ and $\mathbb{N} \models \neg\varphi$, the extended syntax tree $T_\varphi^{\mathbb{M}\circ\mathbb{N}}$ is defined recursively on the structure of φ as follows:

(φ is a literal l): $T_l^{\mathbb{M}\circ\mathbb{N}}$ has a single node $r = \mathbb{M} \circ \mathbb{N}$ such that $synl(r) = l$.

(φ is $\psi_1 \wedge \psi_2$): $T_{\psi_1\wedge\psi_2}^{\mathbb{M}\circ\mathbb{N}}$ has a root $r = \mathbb{M}\circ\mathbb{N}$ and $synl(r) = \wedge$. The left successor of r is the root $\mathbb{M} \circ \mathbb{N}_1$ of $T_{\psi_1}^{\mathbb{M}\circ\mathbb{N}_1}$. The right successor of r is the root $\mathbb{M} \circ \mathbb{N}_2$ of $T_{\psi_2}^{\mathbb{M}\circ\mathbb{N}_2}$ where the sets $\mathbb{N}_1$ and $\mathbb{N}_2$ are defined as follows. $\mathbb{N}_1 = \{(\mathcal{N}, v) \in \mathbb{N} \mid (\mathcal{N}, v) \models \neg\psi_1\}$ and $\mathbb{N}_2 = \{(\mathcal{N}, v) \in \mathbb{N} \mid (\mathcal{N}, v) \models \neg\psi_2\}$. Hence, while $\mathbb{M} \models (\psi_1 \wedge \psi_2)$, we have $\mathbb{N}_1 \models \neg\psi_1$ and $\mathbb{N}_2 \models \neg\psi_2$ and thus $\mathbb{N} \models \neg(\psi_1 \wedge \psi_2)$. We would like to stress that $\mathbb{N} = \mathbb{N}_1 \cup \mathbb{N}_2$ does not imply $\mathbb{N}_1 \cap \mathbb{N}_2 = \emptyset$.

(φ is $\psi_1 \vee \psi_2$): $T_{\psi_1\vee\psi_2}^{\mathbb{M}\circ\mathbb{N}}$ has a root $r = \mathbb{M}\circ\mathbb{N}$ and $synl(r) = \vee$. The left successor of r is the root $\mathbb{M}_1 \circ \mathbb{N}$ of $T_{\psi_1}^{\mathbb{M}_1\circ\mathbb{N}}$. The right successor of r is the root $\mathbb{M}_2 \circ \mathbb{N}$ of $T_{\psi_2}^{\mathbb{M}_2\circ\mathbb{N}}$ where $\mathbb{M}_1 = \{(\mathcal{M}, v) \in \mathbb{M} \mid (\mathcal{M}, v) \models \psi_1\}$ and $\mathbb{M}_2 = \{(\mathcal{M}, v) \in \mathbb{M} \mid (\mathcal{M}, v) \models \psi_2\}$.

Therefore, $\mathbb{M}_1 \models \psi_1$ and $\mathbb{M}_2 \models \psi_2$ while $\mathbb{N} \models \neg(\psi_1 \vee \psi_2)$. Again, $\mathbb{M}_1$ and $\mathbb{M}_2$ may have a non-empty intersection.

(φ is $\Box\psi$): $T_{\Box\psi}^{\mathbb{M}\circ\mathbb{N}}$ has a root $r = \mathbb{M} \circ \mathbb{N}$ and $synl(r) = \Box$. The unique successor of r is the root $\Box(\mathbb{M}) \circ \Diamond\neg\psi(\mathbb{N})$ of $T_\psi^{\Box(\mathbb{M})\circ\Diamond\neg\psi(\mathbb{N})}$. It is obvious that $\Box(\mathbb{M}) \models \psi$ and $\Diamond\neg\psi(\mathbb{N}) \models \neg\psi$.

(φ is $\Diamond\psi$): $T_{\Diamond\psi}^{\mathbb{M}\circ\mathbb{N}}$ has a root $r = \mathbb{M} \circ \mathbb{N}$ and $synl(r) = \Diamond$. The unique successor of r is the root $\Diamond\psi(\mathbb{M}) \circ \Box(\mathbb{N})$ of $T_\psi^{\Diamond\psi(\mathbb{M})\circ\Box(\mathbb{N})}$. It is obvious that $\Diamond\psi(\mathbb{M}) \models \psi$ and $\Box(\mathbb{N}) \models \neg\psi$.

As we said above, the size of a formula φ is its length as a word over S or, equivalently, the number of nodes in it syntax tree. However, for our purposes, defining the size of φ as the number of leaves of its syntax tree will suffice.

Definition 2.3 The size of a formula φ, denoted $|\varphi|$, is the number of leaves of an extended syntax tree $T_\varphi^{\mathbb{M}\circ\mathbb{N}}$ for some (any) sets of pointed models $\mathbb{M}$ and $\mathbb{N}$ such that $\mathbb{M} \models \varphi$ and $\mathbb{N} \models \neg\varphi$.

It is obvious that $|\varphi|$ is the number of not necessarily different literals occurring in φ. Note that, for every φ (even for contradictions like $p \wedge \neg p$), we can find a pair $\langle\mathbb{M}, \mathbb{N}\rangle$ such that $\mathbb{M} \models \varphi$ and $\mathbb{N} \models \neg\varphi$. This, of course, is done by taking $\mathbb{M} = \mathbb{N} = \emptyset$. In fact, we can identify the usual syntax tree T_φ of φ with the extended syntax tree $T_{\varphi}^{\emptyset\circ\emptyset}$. It is clear that extended syntax trees contain finitely many nodes; in particular, they do not have infinitely long branches.

Example 2.4 The extended syntax tree $T_{\Diamond b\wedge\Diamond\neg b}^{\langle\mathbb{A}\circ\mathbb{D}\rangle}$ is shown in Figure 1. Pointed models occurring in the semantic labels of the tree-nodes are the pairs consisting of the relevant Kripke model $\mathcal{A}$, $\mathcal{B}$ or $\mathcal{C}$ and the nodes marked by $\triangleright$ and $\triangleleft$. Hence, $\mathbb{A}$ consists of the pointed model on the left of the root of the tree which has syntax label $\wedge$ while $\mathbb{D}$ is on the right and contains the pointed models based on the Kripke models $\mathcal{B}$ and $\mathcal{C}$. Black circles denote the points where the atom **b** is true; white circles denote points that do not satisfy any proposition.

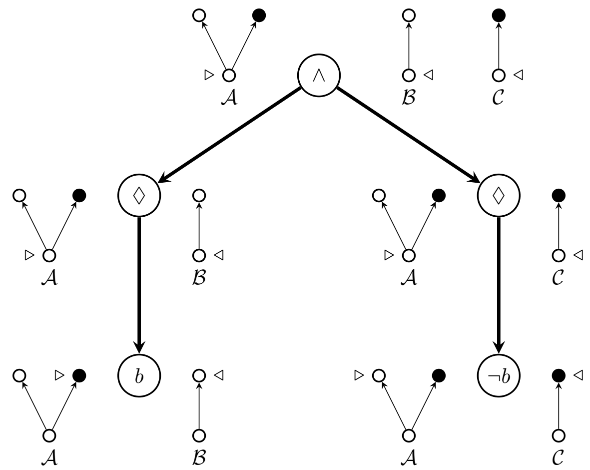


Fig. 1. An extended syntax tree of $\Diamond b \wedge \Diamond\neg b$.

The meaning of the next obvious proposition is that even if the formula $\neg\varphi$ is not in negation normal form, we can always find an equivalent to it formula in negation normal form that has the same number of literals.

Proposition 2.5 *For any* ML*-formula in negation normal form φ, there is a formula in negation normal form $\overline{\varphi}$ such that $\overline{\varphi}$ is equivalent to $\neg\varphi$ on* **K** *and $|\varphi| = |\overline{\varphi}|$.*

Proof. The desired formula $\overline{\varphi}$ is obtained from $\neg\varphi$ by pushing $\neg$ inside φ using DeMorgan's laws and the equivalences $\neg\Diamond\psi \equiv \Box\neg\psi$ and $\neg\Box\psi \equiv \Diamond\neg\psi$. $\Box$

Proposition 2.6 *For any pair of sets of pointed models $\langle\mathbb{M},\mathbb{N}\rangle$, any formula φ such that $\mathbb{M} \models \varphi$ and $\mathbb{N} \models \neg\varphi$ has size at least n iff any formula ψ such that $\mathbb{N} \models \psi$ and $\mathbb{M} \models \neg\psi$ has size at least n.*

Proof. Let us assume that $|\varphi| \geq n$ for any formula φ for which $\mathbb{M} \models \varphi$ and $\mathbb{N} \models \neg\varphi$ but that there is a ψ such that $\mathbb{N} \models \psi$ and $\mathbb{M} \models \neg\psi$ and $|\psi| < n$. Then $\mathbb{M} \models \overline{\psi}$ and $\mathbb{N} \models \neg\overline{\psi}$. Using Proposition 2.5, we see that $|\overline{\psi}| = |\psi| < n$ and, thus, arrive at a contradiction. The other direction is similar. $\Box$

Proposition 2.7 *Let $\langle \mathbb{M}, \mathbb{N} \rangle$ and $\langle \mathbb{M}_1, \mathbb{N}_1 \rangle$ be two pairs of sets of pointed models such that, for every $(\mathcal{M}, m) \in \mathbb{M}$, there is a bisimilar model $(\mathcal{M}_1, m_1) \in \mathbb{M}_1$ and, for every $(\mathcal{N}, n) \in \mathbb{N}$, there is a bisimilar model $(\mathcal{N}_1, n_1) \in \mathbb{N}_1$. If every formula φ such that $\mathbb{M} \models \varphi$ and $\mathbb{N} \models \neg\varphi$ has size at least n, then every formula ψ for which $\mathbb{M}_1 \models \psi$ and $\mathbb{N}_1 \models \neg\psi$ has size at least n.*

Proof. The proof follows immediately from the fact that for every formula ψ for which $\mathbb{M}_1 \models \psi$ and $\mathbb{N}_1 \models \neg\psi$, we have that $\mathbb{M} \models \psi$ and $\mathbb{N} \models \neg\psi$. $\square$

Proposition 2.8 *If the pointed models $(\mathcal{A}, a)$ and $(\mathcal{B}, b)$ are bisimilar, then there is no formula φ such that its extended syntax tree T contains a node η for which $(\mathcal{A}, a) \in \mathtt{lft}(\eta)$ and $(\mathcal{B}, b) \in \mathtt{rght}(\eta)$.*

Proof. According to Definition 2.2, every node η of T is a root of a sub-tree T_1 that is an extended syntax tree of a sub-formula ψ of φ such that $\mathtt{lft}(\eta) \models \psi$ and $\mathtt{rght}(\eta) \models \neg\psi$. Thus, assuming that there are two bisimilar pointed models $(\mathcal{A}, a) \in \mathtt{lft}(\eta)$ and $(\mathcal{B}, b) \in \mathtt{rght}(\eta)$ leads to a contradiction because bisimilar pointed models satisfy the same formulae. $\square$

Proposition 2.9 *If every formula φ such that $\mathbb{M} \models \varphi$ and $\mathbb{N}_1 \cup \ldots \cup \mathbb{N}_k \models \neg\varphi$ has size at least n, then $(|\varphi_1| + |\varphi_2| + \ldots + |\varphi_k|) \geq n$ for any k formulae $\varphi_1, \ldots, \varphi_k$ such that $\mathbb{M} \models \varphi_1 \wedge \ldots \wedge \varphi_k$ and $\mathbb{N}_i \models \neg\varphi_i$.*

Proof. Let $\varphi_1, \ldots \varphi_k$ be such that $\mathbb{M} \models \varphi_1 \wedge \ldots \wedge \varphi_k$ and $\mathbb{N}_i \models \neg\varphi_i$. Therefore, $\mathbb{M} \models \varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_k$ and $\mathbb{N}_1 \cup \ldots \cup \mathbb{N}_k \models \neg(\varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_k)$. The result follows immediately from our assumption. $\square$

3 Main Results

In this section, we are going to prove some lower bounds on the size of formulae which we will later use to show that public announcement logic (PAL) and contingency logic (ConML) are exponentially more succinct than ML. To this end, we begin with a very brief introduction to these two logics.

PAL [14] is a conservative extension of ML in which formulae of the form $[\varphi]\psi$ are allowed. In a natural way, we can introduce a dual $\langle\varphi\rangle$ of the operator $[\varphi]$ by stipulating that $\langle\varphi\rangle$ is an abbreviation of $\neg[\varphi]\neg$. Intuitively, a formula $[\varphi]\psi$ is true in a pointed model $(\mathcal{M}, m)$ if after removing all points that do not satisfy the formula φ, the formula ψ is true at the point m in the resulting new model. For every PAL-formula, there is an equivalent ML-formula that can be obtained by following the rewriting rules below [14]. For any pointed model $(\mathcal{M}, m)$,

$$\begin{array}{lll}
(\mathcal{M}, m) \models \langle\varphi\rangle p & \text{iff} & (\mathcal{M}, m) \models \varphi \wedge p; \\
(\mathcal{M}, m) \models \langle\varphi\rangle(\psi_1 \wedge \psi_2) & \text{iff} & (\mathcal{M}, m) \models \langle\varphi\rangle\psi_1 \wedge \langle\varphi\rangle\psi_2; \\
(\mathcal{M}, m) \models \langle\varphi\rangle\neg\psi & \text{iff} & (\mathcal{M}, m) \models \varphi \wedge \neg\langle\varphi\rangle\psi; \\
(\mathcal{M}, m) \models \langle\varphi\rangle\Diamond\psi & \text{iff} & (\mathcal{M}, m) \models \varphi \wedge \Diamond\langle\varphi\rangle\psi; \\
(\mathcal{M}, m) \models \langle\varphi_1\rangle\langle\varphi_2\rangle\psi & \text{iff} & (\mathcal{M}, m) \models \langle\langle\varphi_1\rangle\varphi_2\rangle\psi.
\end{array}$$

ConML was introduced in [12]. It extends Boolean logic with formulae $\triangle\varphi$. For our purposes, it is enough to say that for every pointed model $(\mathcal{M}, m)$, we have $(\mathcal{M}, m) \models \triangle\varphi$ iff $(\mathcal{M}, m) \models \Box\varphi \vee \Box\neg\varphi$. It is known [3] that ConML is strictly less expressive on $\mathbf{K}$ than ML.

Here, we are going to show that on $\mathbf{K}$, there is no equivalence-preserving translation from either PAL or ConML to ML that produces an equivalent ML-formula of sub-exponential length. To this end, we are going to exhibit two infinite sequences $\delta_1, \delta_2, \ldots$ and $\theta_1, \theta_2, \ldots$ of PAL and ConML-formulae respectively and show that every ML-formula ψ_n that is equivalent to δ_n or θ_n on $\mathbf{K}$ has size at least 2^n whereas the lengths of δ_n and θ_n are linear in n.

Definition 3.1 Let the sequences $\delta_1, \delta_2, \ldots$ and $\theta_1, \theta_2, \ldots$ be defined as follows.

$$
\begin{array}{llll}
\delta_1 \stackrel{\text{def}}{=} & \Diamond\mathbf{b} \wedge \Diamond\neg\mathbf{b} & \theta_1 \stackrel{\text{def}}{=} & \triangle\mathbf{b} \\
\vdots & & \vdots & \\
\delta_{n+1} \stackrel{\text{def}}{=} & \langle\delta_n\rangle\delta_1 & \theta_n \stackrel{\text{def}}{=} & \underbrace{\triangle\ldots\triangle}_{n \text{ times}}\mathbf{b} \\
\vdots & & \vdots &
\end{array}
$$

In order to apply extended syntax trees to show that every ML-formula ψ that is equivalent to δ_n or θ_n on $\mathbf{K}$ has size at least 2^n, we must exhibit two sets of pointed models $\mathbb{M}$ and $\mathbb{N}$ such that $\mathbb{M} \models \psi$ whereas $\mathbb{N} \models \neg\psi$ and show that the extended syntax tree of ψ with root $\mathbb{M} \circ \mathbb{N}$ has at least 2^n leaves. We begin by defining the required sets of pointed models for the formulae δ_n.

Definition 3.2 The set $\mathbb{A}^1$ consists of the pointed model $(\mathcal{A}^1, a^1)$ shown on the left of the leftmost dotted line in Figure 2. The set $\mathbb{B}^1$ contains the two pointed models $(\mathcal{B}^1_1, b^1_1)$ and $(\mathcal{B}^1_2, b^1_2)$ between the leftmost dotted line and the thick vertical line. **B**lack nodes satisfy the proposition $\mathbf{b}$ whereas white nodes

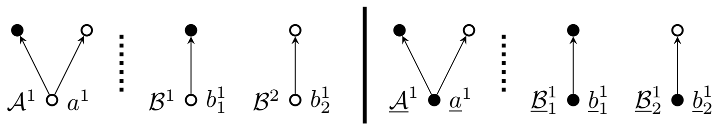


Fig. 2. The sets of models $\mathbb{A}^1$, $\mathbb{B}^1$, $\underline{\mathbb{A}}^1$, and $\underline{\mathbb{B}}^1$.

do not satisfy any proposition. An arrow in a model $\mathcal{M}$ coming from a point m_1 and pointing to a point m_2 means that $m_1 R^{\mathcal{M}} m_2$. The sets $\underline{\mathbb{A}}^1$, and $\underline{\mathbb{B}}^1$ are shown on the right of the thick vertical line. The only difference between the pointed model $(\mathcal{A}^1, a^1)$ and $(\underline{\mathcal{A}}^1, \underline{a}^1)$ is that in the latter the point $\underline{a}^1$ satisfies $\mathbf{b}$; similarly for the pointed models $(\mathcal{B}^1_i, b^1_i)$ and $(\underline{\mathcal{B}}^1_i, \underline{b}^1_i)$ where $1 \leq i \leq 2$.

Let us suppose that the sets $\mathbb{A}^n$, $\mathbb{B}^n$, $\underline{\mathbb{A}}^n$, and $\underline{\mathbb{B}}^n$ have been constructed. For any pointed model $(\mathcal{A}^n, a^n)$, $(\mathcal{B}^n_k, b^n_k)$, $(\underline{\mathcal{A}}^n, \underline{a}^n)$, and $(\underline{\mathcal{B}}^n_k, \underline{b}^n_k)$, we call the points a^n, b^n_k, $\underline{a}^n$, and $\underline{b}^n_k$ the *root* of the respective model. The set $\mathbb{A}^{n+1}$ consists of the pointed model $(\mathcal{A}^{n+1}, a^{n+1})$ (shown in the Figure 3 on the left of the dotted vertical line) and built from the models in $\mathbb{A}^n \cup \mathbb{B}^n \cup \underline{\mathbb{A}}^n \cup \underline{\mathbb{B}}^n$ as follows. We take the pointed models $(\mathcal{A}^n, a^n), (\underline{\mathcal{A}}^n, \underline{a}^n), (\mathcal{B}_i^n, b_i^n), (\underline{\mathcal{B}}_i^n, \underline{b}_i^n)$, where $1 \le i \le 2$, and connect each of the roots of these models to the point a^{n+1} as shown.

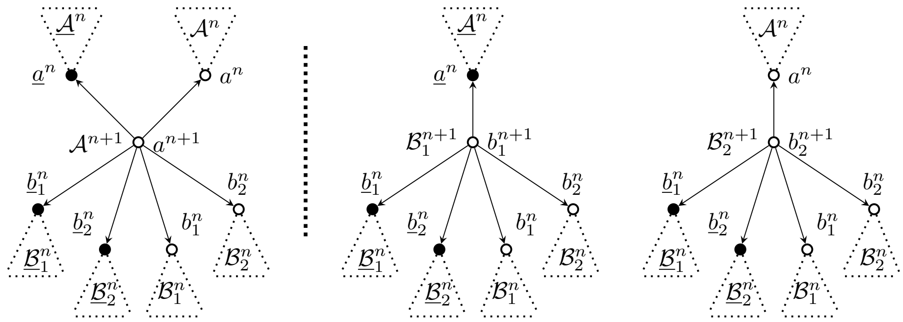
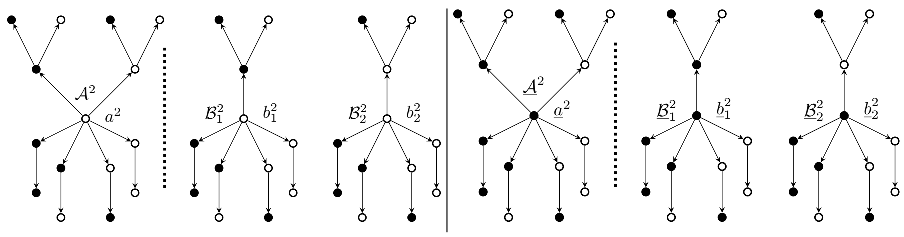


Fig. 3. The sets of models $\mathbb{A}^{n+1}$ and $\mathbb{B}^{n+1}$.

The set $\mathbb{B}^{n+1}$ contains the pointed models $(\mathcal{B}_1^{n+1}, b_1^{n+1})$ and $(\mathcal{B}_2^{n+1}, b_2^{n+1})$ shown on the right of the dotted line.

The sets of pointed models $\underline{\mathbb{A}}^{n+1} = \{(\underline{\mathcal{A}}^{n+1}, \underline{a}^{n+1})\}$ and $\underline{\mathbb{B}}^{n+1} = \{(\underline{\mathcal{B}}_1^{n+1}, \underline{b}_1^{n+1}), (\underline{\mathcal{B}}_2^{n+1}, \underline{b}_2^{n+1})\}$ are obtained from the models in the sets $\mathbb{A}^{n+1}$ and $\mathbb{B}^{n+1}$ by making the roots of the relevant models satisfy the proposition $\mathbf{b}$.

Example 3.3 The sets $\mathbb{A}^2$, $\mathbb{B}^2$ (on the left of the thick vertical line) and $\underline{\mathbb{A}}^2$, $\underline{\mathbb{B}}^2$ (on the right) are given below.



Note how the models from $\mathbb{A}^1$, $\mathbb{B}^1$ and $\underline{\mathbb{A}}^1$, $\underline{\mathbb{B}}^1$ shown in Figure 2 were used in the construction of the models $(\mathcal{A}^2, a^2)$, $(\mathcal{B}_1^2, b_1^2)$ and $(\mathcal{B}_2^2, b_2^2)$. Intuitively, $\underline{\mathbb{A}}^2$ consists of the pointed model obtained from $(\mathcal{A}^2, a^2)$ by making its root a^2 black. Similarly, $\underline{\mathbb{B}}^2$ consists of the models obtained from $(\mathcal{B}_1^2, b_1^{n+1})$ and $(\mathcal{B}_2^2, b_2^2)$ by making b_1^2 and b_2^2 black.

Proposition 3.4 *Consider the formulae $\delta_1, \delta_2, \ldots$ from Definition 3.1. Then $\mathbb{A}^n \models \delta_n$ and $\mathbb{B}^n \models \neg\delta_n$.*

Proof. Given the geometric shape of the models, it is easy to establish by induction the following facts.

(i) For every δ_n, there is an equivalent ML-formula δ'_n that can be obtained from δ_n by applying the rewriting rules for PAL-formulae. Namely, we have the recursively defined sequence of formulae

$$\delta'_1 = \delta_1 \text{ and } \delta'_{n+1} = \delta'_n \wedge \Diamond(\mathbf{b} \wedge \delta'_n) \wedge \Diamond(\neg\mathbf{b} \wedge \delta'_n).$$

(ii) $\mathbb{A}^1 \models \delta'_1$ and $\underline{\mathbb{A}}^1 \models \delta'_1$ whereas $\mathbb{B}^1 \models \neg\delta'_1$ and $\underline{\mathbb{B}}^1 \models \neg\delta'_1$.

(iii) If $n > 1$, then
- $\mathbb{A}^n \models \delta'_j$ and $\underline{\mathbb{A}}^n \models \delta'_j$ for every j such that $1 \leq j \leq n$;
- $(\mathcal{B}^n_1, b^n_1) \models \neg\Diamond(\neg\mathbf{b} \wedge \delta'_n)$ and $(\underline{\mathcal{B}}^n_1, \underline{b}^n_1) \models \neg\Diamond(\neg\mathbf{b} \wedge \delta'_n)$;
- $(\mathcal{B}^n_2, b^n_2) \models \neg\Diamond(\mathbf{b} \wedge \delta'_n)$ and $(\underline{\mathcal{B}}^n_2, \underline{b}^n_2) \models \neg\Diamond(\mathbf{b} \wedge \delta'_n)$.

The result follows immediately from the items above.

□

Next, we define suitable sets of pointed models for the formulae $\theta_1, \theta_2, \ldots$ from Definition 3.1.

Definition 3.5 Using the conventions established in Definition 3.2, namely, that black nodes satisfy the proposition **b**, white nodes do not satisfy any proposition, and arrows represent relations, the sets of pointed models $\mathbb{C}^n$ and $\mathbb{D}^n$ are defined recursively as shown in Figures 4, 5, and 6 where $(\mathcal{D}^n_1, d^n_1), \ldots, (\mathcal{D}^n_k, d^n_k)$ are all the pointed models in $\mathbb{D}^n$.

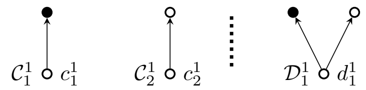


Fig. 4. The sets of models $\mathbb{C}^1$ (on the left of the dotted line) and $\mathbb{D}^1$ (on the right).

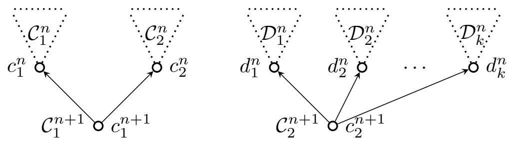


Fig. 5. The set of models $\mathbb{C}^{n+1}$.

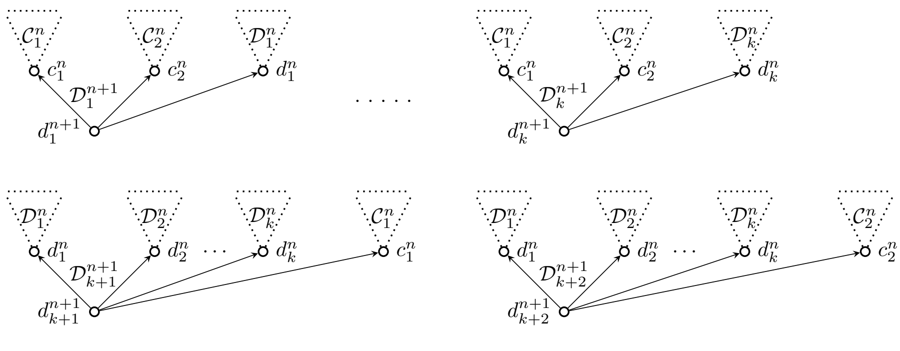


Fig. 6. The set of models $\mathbb{D}^{n+1}$.

Example 3.6 Figure 7 shows the sets $\mathbb{C}^2$, consisting of the pointed models $(\mathcal{C}^2_1, c^2_1)$ and $(\mathcal{C}^2_2, c^2_2)$ and $\mathbb{D}^2$, consisting of $(\mathcal{D}^2_1, d^2_1)$, $(\mathcal{D}^2_2, d^2_2)$, and $(\mathcal{D}^2_3, d^2_3)$. Note how the models in $\mathbb{C}^1$ and $\mathbb{D}^1$ from Figure 4 are used in the construction of the sets $\mathbb{C}^2$ and $\mathbb{D}^2$.

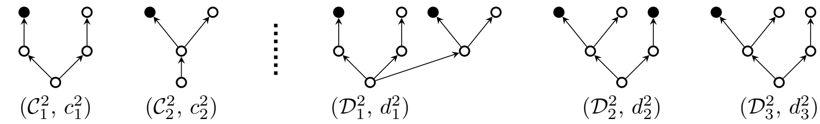


Fig. 7. The set of models $\mathbb{C}^2$ and $\mathbb{D}^2$.

Proposition 3.7 *For any formula θ_n as defined in Definition 3.1, we have $\mathbb{C}^n \models \theta_n$ and $\mathbb{D}^n \models \neg\theta_n$.*

Proof. The truth of the statement follows easily from the geometric shape of the models in $\mathbb{C}^n$ and $\mathbb{D}^n$ and from the fact that, for every θ_n, there is an equivalent ML-formula θ'_n defined recursively as follows $\theta'_1 = \Box\mathbf{b} \vee \Box\neg\mathbf{b}$ and $\theta'_{n+1} = \Box\theta'_n \vee \Box\neg\theta'_n$. $\Box$

We are ready now to prove our main lower-bound results formulated in Theorem 3.8 and Theorem 3.9 below.

Theorem 3.8 (First Lower Bound on ML-formulae) *Let the sets $\mathbb{A}^n$ and $\mathbb{B}^n$ be as defined in Definition 3.2. Any formula ψ such that $\mathbb{A}^n \models \psi$ and $\mathbb{B}^n \models \neg\psi$ has size at least 2^n.*

Theorem 3.9 (Second Lower Bound on ML-formulae) *Let the sets $\mathbb{C}^n$ and $\mathbb{D}^n$ be as defined in Definition 3.5. Any formula φ such that $\mathbb{C}^n \models \varphi$ and $\mathbb{D}^n \models \neg\varphi$ has size at least 2^n.*

The proofs of both theorems rely on a number of preliminary statements that revolve around similar ideas in both cases. We begin by establishing a convention that will simplify our arguments. Consider, for example, the Kripke model $\mathcal{C}^{n+1}_1$ from Figure 5. It is a tree with root c^{n+1}_1. The left successor of c^{n+1}_1 is the root c^n_1 of the model $\mathcal{C}^n_1$. It is obvious that the point c^n_1 in $\mathcal{C}^{n+1}_1$ and the point c^n_1 in $\mathcal{C}^n_1$ are bisimilar and satisfy the same ML-formulae. Since, in what follows, we are mainly interested in formulae-satisfiability, to increase readability, we are going to identify bisimilar pointed models. This allows us to substitute, e.g., the clearer $(\mathcal{C}^n_2, c^n_2)$ for the hard to read $(\mathcal{D}^{n+1}_{k+2}, c^n_2)$.

Let us first prove Theorem 3.8

Proposition 3.10 *For any $i \in \{1, 2\}$, no extended syntax tree contains a node η such that $synl(\eta) \in LIT \cup \{\Box\}$, $\mathbb{A}^n \subseteq \mathtt{lft}(\eta)$, and $(\mathcal{B}^n_i, b^n_i) \in \mathtt{rght}(\eta)$. The statement remains true if $\mathbb{A}^n$ and $(\mathcal{B}^n_i, b^n_i)$ are replaced with $\underline{\mathbb{A}}^n$ and $(\underline{\mathcal{B}}^n_i, \underline{b}^n_i)$, respectively.*

Proof. It is obvious that $(\mathcal{A}^n, a^n)$ and $(\mathcal{B}^n_i, b^n_i)$ satisfy the same Boolean formulae. Hence, there is no extended syntax tree that contains a node η such that $synl(\eta) \in LIT$, $\mathbb{A}^n \subseteq \mathtt{lft}(\eta)$, and $(\mathcal{B}^n_i, b^n_i) \in \mathtt{rght}(\eta)$.

Let us suppose that there is an extended syntax tree containing a node η such that $synl(\eta) = \Box$, $\mathbb{A}^n \subseteq \mathtt{lft}(\eta)$, and $(\mathcal{B}_i^n, b_i^n) \in \mathtt{rght}(\eta)$. We consider only the case $n > 1$ (the case $n = 1$ is similar). Then, the set $\Box(\mathbb{A}^n) = \{(\underline{\mathcal{A}}^{n-1}, \underline{a}^{n-1}), (\mathcal{A}^{n-1}, a^{n-1}), (\underline{\mathcal{B}}_1^{n-1}, \underline{b}_1^{n-1}), (\underline{\mathcal{B}}_2^{n-1}, \underline{b}_2^{n-1}), (\mathcal{B}_1^{n-1}, b_1^{n-1}), (\mathcal{B}_2^{n-1}, b_2^{n-1})\} \subseteq \mathtt{lft}(\eta_1)$, where η_1 is the successor of η. The geometry of $(\mathcal{B}_i^n, b_i^n)$ is such that at least one of the models in $\Box(\mathbb{A}^n)$ must appear on the right of η_1. Thus, we arrive at a contradiction with the help of Proposition 2.8. The proof for $\underline{\mathbb{A}}^n$ and $(\underline{\mathcal{B}}_i^n, \underline{b}_i^n)$ is the same. $\Box$

Proposition 3.11 *There is no extended syntax tree that contains a node η such that $synl(\eta) \in LIT \cup \{\Diamond, \Box\}$, $\mathbb{A}^n \subseteq \mathtt{lft}(\eta)$ and $\mathbb{B}^n \subseteq \mathtt{rght}(\eta)$. The statement remains true if $\mathbb{A}^n$ and $\mathbb{B}^n$ are replaced with $\underline{\mathbb{A}}^n$ and $\underline{\mathbb{B}}^n$, respectively.*

Proof. The fact that no node η for which $\mathbb{A}^n \subseteq \mathtt{lft}(\eta)$ and $\mathbb{B}^n \subseteq \mathtt{rght}(\eta)$ (or ($\underline{\mathbb{A}}^n \subseteq \mathtt{lft}(\eta)$ and $\underline{\mathbb{B}}^n \subseteq \mathtt{rght}(\eta)$) can have a syntax label that is either a literal or $\Box$ follows from Proposition 3.10. If $synl(\eta) = \Diamond$, then it is easily seen by consulting the relevant items from Definition 2.2 that the successor node of η would contain two bisimilar models one on the left and the other on the right which is impossible according to Proposition 2.8. $\Box$

Proposition 3.12 *Any extended syntax tree T with root r such that $\mathbb{A}^n \subseteq \mathtt{lft}(r)$ and $\mathbb{B}^n \subseteq \mathtt{rght}(r)$ contains a node η for which $synl(\eta) = \wedge$, $\mathbb{A}^n \subseteq \mathtt{lft}(\eta)$ and $\mathbb{B}^n \subseteq \mathtt{rght}(\eta)$; moreover, if η_1 and η_2 are the two successor of η, then $\mathbb{A}^n \subseteq \mathtt{lft}(\eta_1)$, $(\mathcal{B}_1^n, b_1^n) \in \mathtt{rght}(\eta_1)$, $(\mathcal{B}_2^n, b_2^n) \notin \mathtt{rght}(\eta_1)$ while $\mathbb{A}^n \subseteq \mathtt{lft}(\eta_2)$, $(\mathcal{B}_2^n, b_2^n) \in \mathtt{rght}(\eta_2)$, and $(\mathcal{B}_1^n, b_1^n) \notin \mathtt{rght}(\eta_2)$. The statement remains true if $\mathbb{A}^n$, $\mathbb{B}^n$, and $(\mathcal{B}_i^n, b_i^n)$ for $i \in \{1, 2\}$ are replaced with $\underline{\mathbb{A}}^n$, $\underline{\mathbb{B}}^n$, and $(\underline{\mathcal{B}}_i^n, \underline{b}_i^n)$, respectively.*

Proof. Let us assume that T does not have such a node η. We will show that T contains an infinite branch which is absurd. We saw already in Proposition 3.11 that $synl(r) \notin LIT \cup \{\Diamond, \Box\}$. Therefore, $synl(r) \in \{\vee, \wedge\}$. If $synl(r) = \vee$, then, since $\mathbb{A}^n$ contains just one model, we see that at least one of the successors r_1 and r_2, say r_1, of r is such that $\mathbb{A}^n \subseteq \mathtt{lft}(r_1)$ and $\mathbb{B}^n \subseteq \mathtt{rght}(r_1)$. If $r = \wedge$, since $\mathbb{B}^n$ contains two models, our assumption and the second item from Definition 2.2 imply that, again, at least one of the successors r_1 and r_2, say r_1, of r is such that $\mathbb{A}^n \subseteq \mathtt{lft}(r_1)$ and $\mathbb{B}^n \subseteq \mathtt{rght}(r_1)$. In either case, we can find a successor r_1 of the root r of T such that $\mathbb{A}^n \subseteq \mathtt{lft}(r_1)$ and $\mathbb{B}^n \subseteq \mathtt{rght}(r_1)$. It is obvious that this reasoning can be applied to the node r_1. Hence, we can find the desired infinite branch by starting at the root and “following” the nodes that contain the models $\mathbb{A}^n$ on the left and the models $\mathcal{B}^n$ on the right. $\Box$

Lemma 3.13 *For any extended syntax tree T with root r, the following hold.*

(i) *If $\mathbb{A}^{n+1} \subseteq \mathtt{lft}(r)$ and $(\mathcal{B}_1^{n+1}, b_1^{n+1}) \in \mathtt{rght}(r)$, then T contains a node η such that $synl(\eta) = \Diamond$, $\mathbb{A}^{n+1} \subseteq \mathtt{lft}(\eta)$ and $(\mathcal{B}_1^{n+1}, b_1^{n+1}) \in \mathtt{rght}(\eta)$; moreover, if η_1 is its successor, then $(\mathcal{A}^n, a^n) \in \mathtt{lft}(\eta_1)$ and $\{(\mathcal{B}_1^n, b_1^n), (\mathcal{B}_2^n, b_2^n)\} \subseteq \mathtt{rght}(\eta_1)$. The statement remains true if $\mathbb{A}^{n+1}$ and $(\mathcal{B}_1^{n+1}, b_1^{n+1})$ are replaced with $\underline{\mathbb{A}}^{n+1}$ and $(\underline{\mathcal{B}}_1^{n+1}, \underline{b}_1^{n+1})$, respectively.*

(ii) *If* $\mathbb{A}^{n+1} \subseteq \texttt{lft}(r)$ *and* $(\mathcal{B}_2^{n+1}, b_2^{n+1}) \in \texttt{rght}(r)$*, then* T *contains a node* η *such that* $synl(\eta) = \Diamond$*,* $\mathbb{A}^{n+1} \subseteq \texttt{lft}(\eta)$ *and* $(\mathcal{B}_2^{n+1}, b_2^{n+1}) \in \texttt{rght}(\eta)$*; moreover, if* η_1 *is its successor, then* $(\underline{\mathcal{A}}^n, \underline{a}^n) \in \texttt{lft}(\eta_1)$ *and* $\{(\underline{\mathcal{B}}_1^n, \underline{b}_1^n), (\underline{\mathcal{B}}_2^n, \underline{b}_2^n)\} \subseteq \texttt{rght}(\eta_1)$*. The statement remains true if* $\mathbb{A}^{n+1}$ *and* $(\mathcal{B}_2^{n+1}, b_2^{n+1})$ *are replaced with* $\underline{\mathbb{A}}^{n+1}$ *and* $(\underline{\mathcal{B}}_2^{n+1}, \underline{b}_2^{n+1})$*, respectively.*

Proof.

(i) Let us suppose that T does not have a node η such that $synl(\eta) = \Diamond$, $\mathbb{A}^{n+1} \subseteq \texttt{lft}(\eta)$ and $(\mathcal{B}_1^{n+1}, b_1^{n+1}) \in \texttt{rght}(\eta)$. We are going to show that, in this case, T contains an infinite branch which is absurd. Using Proposition 3.10 and our assumption, we see that $synl(r) \notin LIT \cup \{\Box, \Diamond\}$. Thus, $synl(r) \in \{\vee, \wedge\}$. Using reasoning identical to the one in the proof of Proposition 3.12, we can find at least one successor r_1 of r such that $\mathbb{A}^{n+1} \subseteq lft(r_1)$ and $(\mathcal{B}_1^{n+1}, b_1^{n+1}) \in rght(r_1)$. It is obvious that the same considerations can be applied to r_1, too. Thus the desired infinite branch is constructed by starting at r and following the nodes that contain $\mathbb{A}^{n+1}$ on the left and $(\mathcal{B}_1^{n+1}, b_1^{n+1})$ on the right. Hence, T must contain a node η, such that $synl(\eta) = \Diamond$, $\mathbb{A}^n \subseteq \texttt{lft}(\eta)$, and $(\mathcal{B}_1^{n+1}, b_1^{n+1}) \in \texttt{rght}(\eta)$. Let η_1 be its successor. According to the fifth item from Definition 2.2, we have $\Box(\mathcal{B}_1^{n+1}, b_1^{n+1}) \subseteq \texttt{rght}(\eta_1)$. Since $\Box(\mathcal{B}_1^{n+1}, b_1^{n+1}) = \{(\underline{\mathcal{A}}^n, \underline{a}^n), (\underline{\mathcal{B}}_1^n, \underline{b}_1^n), (\underline{\mathcal{B}}_2^n, \underline{b}_2^n), (\mathcal{B}_1^n, b_1^n), (\mathcal{B}_2^n, b_2^n)\}$, we see that none of these pointed models can appear on the right of η_1 according to Proposition 2.8. Given the geometry of $(\mathcal{A}^{n+1}, a^{n+1})$, we see that $(\mathcal{A}^n, a^n) \in \texttt{lft}(\eta_1)$. The proof for $\underline{\mathbb{A}}^{n+1}$ and $(\underline{\mathcal{B}}_1^{n+1}, \underline{b}_1^{n+1})$ is the same.

(ii) The proof of this item is completely analogous.

□

We are now ready to prove Theorem 3.8. We proceed by proving, with induction on n, the stronger statement below.

Any formula φ *such that* $\mathbb{A}^n \models \varphi$ *and* $\mathbb{B}^n \models \neg\varphi$ *has size at least* 2^n *and any formula* ψ *such that* $\underline{\mathbb{A}}^n \models \psi$ *and* $\underline{\mathbb{B}}^n \models \neg\psi$ *has size at least* 2^n*.*

Proof.

Base step: The fact that the extended syntax tree T of any formula φ such that $\mathbb{A}^1 \models \varphi$ and $\mathbb{B}^1 \models \neg\varphi$ has at least two leaves follows immediately from Proposition 3.12. The same is true about $\underline{\mathbb{A}}^1$ and $\underline{\mathbb{B}}^1$.

Induction step: The induction step depends on the following claim.

Claim For any $i \in \{1, 2\}$ and any $n \geq 1$, any formula φ such that $\mathbb{A}^{n+1} \models \varphi$ and $(\mathcal{B}_i^{n+1}, b_i^{n+1}) \models \neg\varphi$ has size at least 2^n. The statement remains true if $\mathbb{A}^{n+1}$ and $(\mathcal{B}_i^{n+1}, b_i^{n+1})$ are replaced with $\underline{\mathbb{A}}^{n+1}$ and $(\underline{\mathcal{B}}_i^{n+1}, \underline{b}_i^{n+1})$, respectively.

Proof. Let us consider the case $i = 2$. Let T be an extended syntax tree with root r for which $\mathbb{A}^{n+1} \subseteq \texttt{lft}(r)$ and $(\mathcal{B}_2^{n+1}, b_2^{n+1}) \in \texttt{rght}(r)$. Using the second item from Lemma 3.13, we see that T contains a node η_1 such that $\underline{\mathbb{A}}^n \subseteq \texttt{lft}(\eta_1)$ and $\{(\underline{\mathcal{B}}_1^n, \underline{b}_1^n), (\underline{\mathcal{B}}_2^n, \underline{b}_2^n)\} = \underline{\mathbb{B}}^n \subseteq \texttt{rght}(\eta_1)$. Applying the induction hypothesis and Proposition 2.7 to the subtree T_1 of T with root η_1, we see that T_1 and thus T has size at least 2^n. The proof of the case $i = 1$ is analogous modulo the fact that we use the first item of Lemma 3.13. This completes the proof of the claim. $\Box$

Let us complete now the proof of the induction step. Using Proposition 3.12, we see that the extended syntax tree T with root r, where $\mathbb{A}^{n+1} = \texttt{lft}(r)$ and $\mathbb{B}^{n+1} = \texttt{rght}(r)$, of any formula φ such that $\mathbb{A}^{n+1} \models \varphi$ and $\mathbb{B}^{n+1} \models \neg\varphi$ contains two different sub-trees T_1 and T_2 with roots r_1 and r_2 such that $\mathbb{A}^{n+1} = \texttt{lft}(r_1)$, $\{(\mathcal{B}_1^{n+1}, b_1^{n+1})\} = \texttt{rght}(r_1)$ and $\mathbb{A}^{n+1} = \texttt{lft}(r_2)$, $\{(\mathcal{B}_2^{n+1}, b_2^{n+1})\} = \texttt{rght}(r_2)$. It follows from the above Claim that both T_1 and T_2 have size at least 2^n. Thus, the size of T must be at least 2^{n+1}. Again, the proof about $\underline{\mathbb{A}}^{n+1}$ and $\underline{\mathbb{B}}^{n+1}$ is the same.

$\Box$

Next, we prove Theorem 3.9.

Consider the sets of pointed models from Definition 3.5.

Proposition 3.14 *For any $n \geq 0$,*

(i) *if $1 \leq i \leq k$, then no extended syntax tree contains a node η such that $synl(\eta) \in LIT \cup \{\Diamond\}$, $(\mathcal{C}_1^{n+1}, c_1^{n+1}) \in \texttt{lft}(\eta)$, and $(\mathcal{D}_i^{n+1}, d_i^{n+1}) \in \texttt{rght}(\eta)$;*

(ii) *if $k+1 \leq i \leq k+2$, then no extended syntax tree contains a node η such that $synl(\eta) \in LIT \cup \{\Diamond\}$, $(\mathcal{C}_2^{n+1}, c_2^{n+1}) \in \texttt{lft}(\eta)$ and $(\mathcal{D}_i^{n+1}, d_i^{n+1}) \in \texttt{rght}(\eta)$.*

Proof. It is obvious that all the pointed models in $\mathbb{C}^n$ and $\mathbb{D}^n$ satisfy the same Boolean formulae and, therefore, η cannot have a syntax label that is a literal. We consider only the case $synl(\eta) = \Diamond$ for the first item and we assume $n \geq 1$. The proofs for $(\mathcal{C}_1^1, c_1^1) \in \texttt{lft}(\eta)$ and $(\mathcal{D}_i^1, d_i^1) \in \texttt{rght}(\eta)$ and (ii) are analogous. Suppose that there is an extended syntax tree containing such a node η. Let η_1 be its successor. According to the fourth item from Definition 2.2, we have that $\Box(\mathcal{D}_i^{n+1}, d_i^{n+1}) \subseteq \texttt{rght}(\eta_1)$. Since $1 \leq i \leq k$, it is obvious that $\Box(\mathcal{D}_i^{n+1}, d_i^{n+1}) = \{(\mathcal{C}_1^n, c_1^n), (\mathcal{C}_2^n, c_2^n), (\mathcal{D}_i^n, d_i^n)\}$. Given the geometry of the pointed model $(\mathcal{C}_1^{n+1}, c_1^{n+1})$, either $(\mathcal{C}_1^n, c_1^n)$ or $(\mathcal{C}_2^n, c_2^n)$ must appear on the left of η_1. Using Proposition 2.8, we arrive at a contradiction. $\Box$

Proposition 3.15 *For any $(\mathcal{D}_i^n, d_i^n) \in \mathbb{D}^n$, there is no extended syntax tree that contains a node η such that $synl(\eta) \in LIT \cup \{\Diamond, \Box\}$, $\mathbb{C}^n \subseteq \texttt{lft}(\eta)$, and $(\mathcal{D}_i^n, d_i^n) \in \texttt{rght}(\eta)$.*

Proof. In the case of $synl(\eta) = \Diamond$ or $synl(\eta) \in LIT$, the statement follows from Proposition 3.14. If $synl(\eta) = \Box$, then it is easily seen that its successor η_1 has two bisimilar models one on the left and the other on the right. Hence, we arrive at a contradiction with the help of Proposition 2.8. $\Box$

Proposition 3.16 *For any $(\mathcal{D}_i^n, d_i^n) \in \mathbb{D}^n$, if T is an extended syntax tree with root r for which $\mathbb{C}^n \subseteq \texttt{lft}(r)$ and $(\mathcal{D}_i^n, d_i^n) \in \texttt{rght}(r)$, then T has a node η such that $synl(\eta) = \vee$, $\mathbb{C}^n \subseteq \texttt{lft}(\eta)$, and $(\mathcal{D}_i^n, d_i^n) \in \texttt{rght}(\eta)$; moreover, if η_1 and η_2 are the two successor of η, then $(\mathcal{C}_1^n, c_1^n) \in \texttt{lft}(\eta_1)$, $(\mathcal{C}_2^n, c_2^n) \notin \texttt{lft}(\eta_1)$,* *and* $(\mathcal{D}_i^n, d_i^n) \in \mathtt{rght}(\eta_1)$ *while* $(\mathcal{C}_2^n, c_2^n) \in \mathtt{lft}(\eta_2)$, $(\mathcal{C}_1^n, c_1^n) \notin \mathtt{lft}(\eta_2)$, *and* $(\mathcal{D}_i^n, d_i^n) \in \mathtt{rght}(\eta_2)$.

Proof. Let us assume that T does not have such a node η. We are going to show that T contains an infinite branch which is absurd. Indeed, using Proposition 3.15, we see that $synl(r) \notin LIT \cup \{\Box, \Diamond\}$. Therefore, either $synl(r) = \wedge$ or $synl(r) = \vee$. In the first case, at least one of the successors r_1 and r_2 of r, say r_1, will be such that $\mathbb{C}^n \subseteq \mathtt{lft}(r_1)$ and $(\mathcal{D}_i^n, d_i^n) \in \mathtt{rght}(r_1)$. In the second case, since the two successors r_1 and r_2 of r do not have the properties described in the statement, at least one of them, say r_1, must be such that $\mathbb{C}^n \subseteq \mathtt{lft}(r_1)$ and $(\mathcal{D}_i^n, d_i^n) \in \mathtt{rght}(r_1)$. In either case, we can find a successor r_1 of the root r of T such that $\mathbb{C}^n \subseteq \mathtt{lft}(r_1)$ and $(\mathcal{D}_i^n, d_i^n) \in \mathtt{rght}(r_1)$. It is obvious that this reasoning can be applied to the node r_1. Hence, we can find the desired infinite branch by starting at the root and "following" the nodes that contain the models $\mathbb{C}^n$ on the left and the model $(\mathcal{D}_i^n, d_i^n)$ on the right. $\Box$

To simplify the exposition of the proofs below, we write $\Box(\eta)$ to mean the successor of a node η in an extended syntax tree such that $synl(\eta) = \Box$.

Lemma 3.17 *For any extended syntax tree T with root r,*

(i) *if* $(\mathcal{C}_1^{n+1}, c_1^{n+1}) \in \mathtt{lft}(r)$ *and* $(\mathcal{D}_i^{n+1}, d_i^{n+1}) \in \mathtt{rght}(r)$, *where* $1 \leq i \leq k$, *then T contains a node η such that* $synl(\eta) = \Box$, $(\mathcal{C}_1^{n+1}, c_1^{n+1}) \in \mathtt{lft}(\eta)$, *and* $(\mathcal{D}_i^{n+1}, d_i^{n+1}) \in \mathtt{rght}(\eta)$;

(ii) *if* $k+1 \leq i \leq k+2$, $(\mathcal{C}_2^{n+1}, c_2^{n+1}) \in \mathtt{lft}(r)$ *and* $(\mathcal{D}_i^{n+1}, d_i^{n+1}) \in \mathtt{rght}(r)$, *then T contains a node η such that* $synl(\eta) = \Box$, $(\mathcal{C}_2^{n+1}, c_2^{n+1}) \in \mathtt{lft}(\eta)$, *and* $(\mathcal{D}_i^{n+1}, d_i^{n+1}) \in \mathtt{rght}(\eta)$;

(iii) *if T contains a node η such that* $synl(\eta) = \Box$, $(C_1^{n+1}, c_1^{n+1}) \in \mathtt{lft}(\eta)$, *and* $\{(\mathcal{D}_i^{n+1}, d_i^{n+1}), \ldots, (\mathcal{D}_j^{n+1}, d_j^{n+1})\} \subseteq \mathtt{rght}(\eta)$, *where* $1 \leq i \leq j \leq k$, *then* $\{(C_1^n, c_1^n), (C_2^n, c_2^n)\} \subseteq \mathtt{lft}(\Box(\eta))$ *and* $\{(\mathcal{D}_i^n, d_i^n), \ldots, (\mathcal{D}_j^n, d_j^n)\} \subseteq \mathtt{rght}(\Box(\eta))$;

(iv) *if T contains a node η such that* $synl(\eta) = \Box$, $(C_2^{n+1}, c_2^{n+1}) \in \mathtt{lft}(\eta)$, *and* $\{(\mathcal{D}_i^{n+1}, d_i^{n+1}), (\mathcal{D}_j^{n+1}, d_j^{n+1})\} \subseteq \mathtt{rght}(\eta)$, *where* $k+1 \leq i \leq j \leq k+2$, *then* $\{(\mathcal{D}_1^n, d_1^n), \ldots, (\mathcal{D}_k^n, d_k^n)\} \subseteq \mathtt{lft}(\Box(\eta))$ *and* $\{(C_l^n, c_l^n), (C_m^n, c_m^n)\} \subseteq \mathtt{rght}(\Box(\eta))$.

Proof.

(i) Let us assume that there is a syntax tree T that does not have a node η with the desired properties. We show that T contains an infinite branch which is absurd. Indeed, using the first item from Proposition 3.14, we see that $synl(r) \notin LIT \cup \{\Diamond\}$. According to our assumption $synl(r) \neq \Box$. Hence, either $synl(r) = \vee$ or $synl(r) = \wedge$. In either case, r has at least one successor r_1 such that $(\mathcal{C}_1^{n+1}, c_1^{n+1}) \in \mathtt{lft}(r_1)$ and $(\mathcal{D}_i^{n+1}, d_i^{n+1}) \in \mathtt{rght}(r_1)$. Thus, we can find the desired infinite branch by starting at r and "following" the nodes that contain the models $(\mathcal{C}_1^{n+1}, c_1^{n+1})$ on the left and $(\mathcal{D}_i^{n+1}, d_i^{n+1})$ on the right.

(ii) The proof is the same as the one above modulo using (ii) from Proposition 3.14.

(iii) It is obvious that $\Box(\mathcal{C}_1^{n+1}, c_1^{n+1}) = \{(C_1^n, c_1^n), (C_2^n, c_2^n)\}$. Using Definition 2.2, we see that $\{(C_1^n, c_1^n), (C_2^n, c_2^n)\} \subseteq \mathtt{lft}(\Box(\eta))$. It follows immediately from Proposition 2.8 that $(C_1^n, c_1^n) \notin \mathtt{rght}(\Box(\eta))$ and $(C_2^n, c_2^n) \notin \mathtt{rght}(\Box(\eta))$. Given the geometry of the models $(\mathcal{D}_i^{n+1}, d_i^{n+1}), \ldots, (\mathcal{D}_j^{n+1}, d_j^{n+1})$, we obtain $\{(\mathcal{D}_i^n, d_i^n), \ldots, (\mathcal{D}_j^n, d_j^n)\} \subseteq \mathtt{rght}(\Box(\eta))$.

(iv) Obviously, $\Box(\mathcal{C}_2^{n+1}, c_2^{n+1}) = \{(\mathcal{D}_1^n, d_1^n), \ldots, (\mathcal{D}_k^n, d_k^n)\}$. The fourth item from Definition 2.2 implies that $\Box((\mathcal{C}_2^{n+1}, c_2^{n+1})) \subseteq \mathtt{lft}(\Box(\eta))$. Using Proposition 2.8, we see that none of the pointed models in $\Box(\mathcal{C}_2^{n+1}, c_2^{n+1})$ can appear on the right of $\Box(\eta)$. Given the geometry of the models $(\mathcal{D}_{k+1}^{n+1}, d_{k+1}^{n+1})$ and $(\mathcal{D}_{k+2}^{n+1}, d_{k+2}^{n+1})$, we see that $\{(C_l^n, c_l^n), (C_m^n, c_m^n)\} \subseteq \mathtt{rght}(\Box(\eta))$.

$\Box$

We are ready now to prove Theorem 3.9. It follows immediately from the stronger statement below.

Theorem 3.18 *For any $n \geq 1$, any formula φ such that $\mathbb{C}^n \models \varphi$ and $\mathbb{D}^n \models \neg\varphi$ has size at least 2^n and any formula ψ such that $\mathbb{D}^n \models \psi$ and $\mathbb{C}^n \models \neg\psi$ has size at least 2^n.*

Proof. The proof proceeds by induction on n.

Base step: The fact that the extended syntax tree T of any formula φ such that $\mathbb{C}^1 \models \varphi$ and $\mathbb{D}^1 \models \neg\varphi$ has at least two leaves follows immediately from Propositions 3.16. Using this and Proposition 2.6, we see that the size of any formula ψ such that $\mathbb{D}^1 \models \psi$ and $\mathbb{C}^1 \models \neg\psi$ is at least 2.

Induction step: Let us consider the extended syntax tree T with a root $r = \mathbb{C}^{n+1} \circ \mathbb{D}^{n+1}$ of a formula φ such that $\mathbb{C}^{n+1} \models \varphi$ and $\mathbb{D}^{n+1} \models \neg\varphi$. It follows from Proposition 3.16, that we have two types of nodes.

(i) For the pointed model $(\mathcal{C}_1^{n+1}, c_1^{n+1})$ and any pointed model $(\mathcal{D}_i^{n+1}, d_i^{n+1})$, where $1 \leq i \leq k$, there is a node η in T such that $\{(\mathcal{C}_1^{n+1}, c_1^{n+1})\} = \mathtt{lft}(\eta)$ and $(\mathcal{D}_i^{n+1}, d_i^{n+1}) \in \mathtt{rght}(\eta)$.

(ii) For the pointed model $(\mathcal{C}_2^{n+1}, c_2^{n+1})$ and any pointed model $(\mathcal{D}_j^{n+1}, d_j^{n+1})$, where $k+1 \leq j \leq k+2$, there is a node ζ in T such that $\{(\mathcal{C}_2^{n+1}, c_2^{n+1})\} = \mathtt{lft}(\zeta)$ and $(\mathcal{D}_j^{n+1}, d_j^{n+1}) \in \mathtt{rght}(\zeta)$.

It is obvious that a node η and a node ζ cannot coincide. Hence η and ζ are the roots of two sub-trees T_η and T_ζ of T with no common nodes. Using item (i) from Lemma 3.17, we see that T_η contains a node η_1 such that $synl(\eta_1) = \Box$, $\{(\mathcal{C}_1^{n+1}, c_1^{n+1})\} = \mathtt{lft}(\eta_1)$, and $(\mathcal{D}_i^{n+1}, d_i^{n+1}) \in \mathtt{rght}(\eta_1)$; similarly, it follows from Lemma 3.17 (ii), that T_ζ contains a node ζ_2 such that $synl(\zeta_2) = \Box$, $\{(\mathcal{C}_2^{n+1}, c_2^{n+1})\} = \mathtt{lft}(\zeta_2)$, and $(\mathcal{D}_j^{n+1}, d_j^{n+1}) \in \mathtt{rght}(\zeta_2)$; Therefore, T has the shape shown in Figure 8. Namely,

- there are nodes $\eta_1^1 = (\mathcal{C}_1^{n+1}, c_1^{n+1}) \circ \mathbb{G}_1, \ldots, \eta_1^l = (\mathcal{C}_1^{n+1}, c_1^{n+1}) \circ \mathbb{G}_l$ such that each one of them has a syntax label $\Box$; what is more,

$\{(\mathcal{D}_1^{n+1}, d_1^{n+1}), \ldots, (\mathcal{D}_k^{n+1}, d_k^{n+1})\} \subseteq \mathbb{G}_1 \cup \ldots \cup \mathbb{G}_l$;

- there are nodes $\zeta_2^1 = (\mathcal{C}_2^{n+1}, c_2^{n+1}) \circ \mathbb{H}_1, \ldots, \zeta_2^m = (\mathcal{C}_2^{n+1}, c_2^{n+1}) \circ \mathbb{H}_m$ such that each one of them has a syntax label $\Box$; moreover, $\{(\mathcal{D}_{k+1}^{n+1}, d_{k+1}^{n+1}), (\mathcal{D}_{k+2}^{n+1}, d_{k+2}^{n+1})\} \subseteq \mathbb{H}_1 \cup \ldots \cup \mathbb{H}_m$.

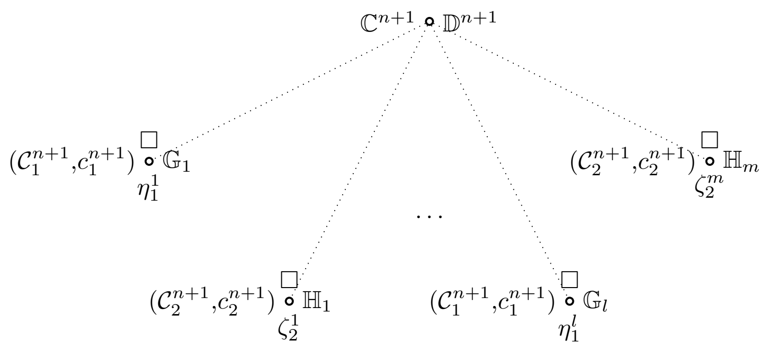


Fig. 8. An extended syntax tree T with root $\mathbb{C}^{n+1} \circ \mathbb{D}^{n+1}$.

Using item (iii) of Lemma 3.17, we see that, for any η_1^i, we have $\mathbb{C}^n = \{(C_1^n, c_1^n), (C_2^n, c_2^n)\} = \Box((\mathcal{C}_1^{n+1}, c_1^{n+1})) = \texttt{lft}(\Box(\eta_1^i))$ while $\mathbb{D}^n = \{(\mathcal{D}_1^n, d_1^n), \ldots, (\mathcal{D}_k^n, d_k^n)\} \subseteq \texttt{rght}(\Box(\eta_1^1)) \cup \ldots \cup \texttt{rght}(\Box(\eta_1^l))$. According to the induction hypothesis, any formula φ such that $\mathbb{C}^n \models \varphi$ and $\mathbb{D}^n \models \neg\varphi$ has size at least 2^n. Applying Proposition 2.7 and Proposition 2.9, we see that, for any formulae $\varphi_1, \ldots, \varphi_l$ for which $\texttt{lft}(\Box(\eta_1^1)) \models \varphi_1^1, \ldots, \texttt{lft}(\Box(\eta_1^l)) \models \varphi_l$ whereas $\texttt{rght}(\Box(\eta_1^1)) \models \neg\varphi_1, \ldots, \texttt{rght}(\Box(\eta_1^l)) \models \neg\varphi_l$, we have $(|\varphi_1| + \ldots + |\varphi_l|) \geq 2^n$.

Similarly, using item (iii) of Lemma 3.17, for any ζ_2^i, we see that $\mathbb{D}^n = \{(\mathcal{D}_1^n, d_1^n), \ldots, (\mathcal{D}_k^n, d_k^n)\} = \Box(\mathcal{C}_2^{n+1}, c_2^{n+1}) = \texttt{lft}(\Box(\zeta_2^i))$ while $\mathbb{C}^n \subseteq \texttt{rght}(\Box(\zeta_2^1)) \cup \ldots \cup \texttt{rght}(\Box(\zeta_2^m))$. Again, according to the induction hypothesis, any formula ψ such that $\mathbb{D}^n \models \psi$ and $\mathbb{C}^n \models \neg\psi$ has size at least 2^n. Applying Proposition 2.7 and Proposition 2.9, we see that, for any formulae $\psi_1, \ldots, \psi_m$ for which $\texttt{lft}(\Box(\zeta_2^1)) \models \psi_1, \ldots, \texttt{lft}(\Box(\zeta_m)) \models \psi_m$ whereas $\texttt{rght}(\Box(\zeta_2^1)) \models \neg\psi_1, \ldots, \texttt{rght}(\Box(\zeta_2^m)) \models \neg\psi_m$, we have $(|\psi_1| + \ldots + |\psi_m|) \geq 2^n$.

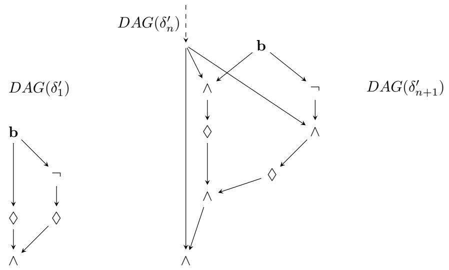
Thus, the number of leaves of any extended syntax tree with root $\mathbb{C}^{n+1} \circ \mathbb{D}^{n+1}$ is at least 2^{n+1}. Hence, any formula φ such that $\mathbb{C}^{n+1} \models \varphi$ and $\mathbb{D}^{n+1} \models \neg\varphi$ has size at least 2^{n+1}. It follows from Proposition 2.6, that any formula ψ such that $\mathbb{D}^{n+1} \models \psi$ and $\mathbb{C}^{n+1} \models \neg\psi$ has size at least 2^{n+1}.

$\Box$

As usual, we can represent ML-formulae compactly as *directed acyclic graphs* (*DAGs*) or *modal circuits*. The size of such a graph is the number of its edges. To the best of our knowledge, the fact that ML-formulae represented as *DAGs* are exponentially more succinct than ML-formulae in their tree representation seems to be taken for granted but we were unable to find a published proof although it can be easily obtained from the results in, e.g., [5], [6] or [10]. Nevertheless, we consider the corollary below to be yet another confirmation of a widely known folklore fact rather than an original new result.

Corollary 3.19 *Modal circuits are exponentially more succinct than* ML-*formulae on the class of all Kripke models* $\mathbf{K}$.

Proof. According to Theorem 3.8, every ML-formula that is equivalent to the formula δ_n from Definition 3.1 has size 2^n. In the proof of Proposition 3.4, we defined, for every δ_n, an equivalent ML-formula δ'_n as follows. $\delta'_1 = \delta_1$ and $\delta'_{n+1} = \delta'_n \wedge \Diamond(\mathbf{b} \wedge \delta'_n) \wedge \Diamond(\neg\mathbf{b} \wedge \delta'_n)$. Obviously, the formulae δ'_n can be represented as linearly growing *DAGs* as shown below. $\square$



4 Conclusion

With the benefit of hindsight, we can say that our Theorem 3.9 is related to Theorem 4.1 from [5]. The latter can be interpreted as showing that there is no sub-exponential equivalence preserving translation from the recursively defined formulae $\varphi_1 = \Diamond(\mathbf{b} \vee \neg\mathbf{b})$ and $\varphi_{n+1} = \Diamond\neg\triangle\varphi_n$ to ML, i.e., extending ML with formulae $\triangle\varphi$ leads to an exponential increase of succinctness with respect to ML. However, since $\Diamond$ is not definable in ConML on many classes of frames [3], this result cannot in general be used to show that ConML is exponentially more succinct than ML.

More results on the comparison between modal formulae and modal circuits for extensions of ML and a list of open problems about lower bounds on formula and circuit-size in modal logics can be found in [8].

A very general problem was pointed out in [16]. It consists of finding the shortest possible modal equivalents of modally definable first-order conditions. The potential importance of this question is witnessed by the fact that its initial study has led to an extension of the class of Sahlqvist formulae [16].

As far as our present work is concerned, it would be nice to compare in terms of succinctness PAL and ConML. We conjecture that PAL is exponentially more succinct than ConML on $\mathbf{K}$ even for modal languages with one diamond and one propositional symbol. We think that this remains true on the class of $\mathbf{S}_5$-models commonly used in epistemic logic but we do not know how many different diamonds and propositional symbols are needed. Additionally, we conjecture that ConML is not exponentially more succinct than PAL but that ML-circuits are exponentially more succinct than PAL-formulae.

Acknowledgements

Hans van Ditmarsch, Jie Fan, and Petar Iliev acknowledge support from European Research Council grant EPS 313360.

Hans van Ditmarsch is also affiliated to the Institute of Mathematical Sciences, Chennai, India as a research associate.

Jie Fan gratefully acknowledges support from the China Scholarship Council.

References

[1] Adler, M. and N. Immerman, *An n! lower bound on formula size*, ACM Trans. Comput. Logic **4** (2003), pp. 296–314.

[2] Blackburn, P., M. de Rijke and Y. Venema, "Modal Logic," CUP, 2001.

[3] Cresswell, M. J., *Necessity and contingency*, Studia Logica **47** (1988), pp. 145–149.

[4] Etessami, K., M. Vardi and T. Wilke, *First-order logic with two variables and unary temporal logic*, in: *Proceedings of LICS*, 1997, pp. 228–236.

[5] Figueira, S. and D. Gorín, *On the size of shortest modal descriptions*, Advances in Modal Logic **8** (2010), pp. 114–132.

[6] French, T., W. van der Hoek, P. Iliev and B. Kooi, *On the succinctness of some modal logics*, Artificial Intelligence **197** (2013), pp. 56–85.

[7] Grohe, M. and N. Schweikardt, *The succinctness of first-order logic on linear orders*, Logical Methods in Computer Science **1** (2005), pp. 1–25.

[8] Iliev, P., "On The Relative Succinctness of Some Modal Logics," PhD thesis, Department of Computer Science, University of Liverpool, 2013.

[9] Laroussinie, F., N. Markey and P. Schnoebelen, *Temporal logic with forgettable past*, in: *Proceedings of LICS*, 2002, pp. 383–392.

[10] Lutz, C., *Complexity and succinctness of public announcement logic*, in: P. Stone and G. Weiss, editors, *Proceedings of AAMAS*, 2006, pp. 137–144.

[11] Lutz, C., U. Sattler and F. Wolter, *Modal logic and the two-variable fragment*, in: L. Fribourg, editor, *Proceedings of CSL*, 2001, pp. 247–261.

[12] Montgomery, H. and R. Routley, *Contingency and non-contingency bases for normal modal logics*, Logique et Analyse **9** (1966), pp. 318–328.

[13] Otto, M., *Bisimulation invariance and finite models*, in: W. P. Z. Chatzidakis, P. Koepke, editor, *Lecture Notes in Logic, Logic Colloquium 2002*, 2006 pp. 276–298.

[14] Plaza, J., *Logics of public communications*, in: M. Emrich, M. Pfeifer, M. Hadzikadic and Z. Ras, editors, *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems: Poster Session Program* (1989), pp. 201–216.

[15] Stockmeyer, L. J., "The Complexity of Decision Problems in Automata Theory and Logic," PhD thesis MIT, 1974.

[16] Vakarelov, D., *Modal definability in languages with a finite number of propositional variables and a new extension of the Sahlqvist's class*, Advances in Modal Logic **4** (2003), pp. 499–518.

[17] van Benthem, J., "Modal logic and Classical Logic," Bibliopolis, 1983.

[18] Wilke, T., *CTL^+ is exponentially more succinct than CTL*, in: *Proceedings of FSTTCS*, 1999, pp. 110–121.