On the Size of Shortest Modal Descriptions

Santiago Figueira [1]

Departamento de Computación, FCEyN, Universidad de Buenos Aires and CONICET
Pabellón I, Ciudad Universitaria
(C1428EGA), Buenos Aires, Argentina

Daniel Gorín [2]

Departamento de Computación, FCEyN, Universidad de Buenos Aires
Pabellón I, Ciudad Universitaria
(C1428EGA), Buenos Aires, Argentina

Abstract

We address the problems of *separation* and *description* in some fragments of modal logics. The former consists in finding a formula that is true in some given subset of the domain and false in another. The latter is a special case when one separates a singleton from the rest. We are interested in the shortest *size* of both separations and descriptions. This is motivated by applications in computational linguistics. Lower bounds are given by considering the minimum size of Spoiler's strategies in the classical Ehrenfeucht-Fraïssé game. This allows us to show that the size of such formulas is not polynomially bounded (with respect to the size of the finite input model). Upper bounds for these problems are also studied. Finally we give a fine hierarchy of succinctness for separation over the studied logics.

Keywords: Modal logic, referring expression, shortest formula size, lower bound, Ehrenfeucht-Fraïssé, succinctness.

1 Introduction

We informally say that a formula φ *describes* an element e in the domain of some model $\mathcal{M}$ whenever φ is true when evaluated at e and false when evaluated at every other point in the domain of $\mathcal{M}$. One can then define the *description problem* as that of finding a description for a given e, if such description exists. [3]

---

[3] We are being deliberately unspecific about the logic in question here, since one can in principle define a description problem for any logic with suitable semantics.

This a fundamental problem in the *Generation of Referring Expressions* (GRE), a key task in the field of Natural Language Generation with continuous active research (see [6,7,8,16,17] among others). GRE is the generation of *noun-phrases* that refer unequivocally to certain objects in the context of conversation. The description problem amounts to finding the relevant features that identify an object (e.g., $kid \land \langle carries \rangle (ball \land red)$) the outcome of which can be given to a *surface realization* module that deals with the generation of an equivalent expression in natural language (e.g., "the kid carrying a red ball").

In this paper we focus on the description problem for the basic modal logic $\mathcal{ML}$ and some of its syntactical fragments. Multi-modal versions of these languages have been previously considered in the context of GRE because of their combination of expressiveness and good computational behavior [3].

In particular, we are interested in the computational complexity of the description problem for modal languages. This question was initially addressed in [3] where it is shown that a standard algorithm for computing bisimulation minimization [15,12] can be adapted to compute an $\mathcal{ML}$-description for every equivalence class in the minimized model (we revisit this idea in Section 5). Since bisimulation minimization can be done in polynomial time, if new formulas are built by combining, in constant-time, formulas that were computed in previous iterations, the resulting algorithm will run in polynomial time too.

Can we conclude that the description problem for $\mathcal{ML}$ can be solved in polynomial time? One must be careful here. In order to implement formula constructors (such as $\land$, $\Box$, etc.) as constant-time operations one needs to resort to pointers or similar mechanisms based on aliasing; the upshot of this is that we will be computing $\mathcal{ML}$-descriptions that are *compactly represented* as direct acyclic graphs (DAG). The size of these DAGs is, by construction, bounded by a polynomial in the size of the model, but it is not clear, in principle, that such a bound exists with respect to the expansion of these DAGs to full-blown trees.

It is shown in [2] that for certain class of models this algorithm can lead to DAGs whose expansions cannot be bounded by a polynomial (cf. Section 5). However, every element in that class of models has a description of linear size. That example only proves that this algorithm may compute very degenerate solutions, but already shows that one has to be careful about complexity claims for this problem. [4]

The description problem is a particular instance of a more general problem: given a model $\mathcal{M}$ and two non-empty sets C, D (of the domain of $\mathcal{M}$), find a formula that is true at every element in C and false at every element in D. We call this the *separation problem*. In this article we show that no polynomial can bound the size of the solutions to the separation and description problems. More precisely, we give exponential lower bounds for the worst-case size of solutions for the separation problem for C and D singleton sets. We show similar lower bounds when weak fragments of $\mathcal{ML}$ (such as the one without negation) are used.

The article is structured as follows. We begin in Section 2 by introducing the notation

---

[4] Surface realization algorithms do not exploit subformula sharing, but will produce a noun-phrase that is proportional (typically, linear) in length to the size of the formula.

we will use throughout the paper. In Section 3 we present the tool we will use to establish lower bound results: *uniform strategy trees*. These formalize a strategy for Spoiler in an Ehrenfeucht-Fraïssé game in a way such that the size of a minimum winning uniform strategy tree corresponds to the size of the minimum formula that separates the elements in the initial position of the game. Using these, we give, in Section 4, exponential lower bounds for the separation problem using different fragments of $\mathcal{ML}$ formulas. In Section 5 we give an upper bound for this problem that is slightly higher than the lower bound of Section 4. Though the upper and lower bounds are almost tight, the question of which are the optimal bounds remains open. Finally, in Section 6 we relate these results with the standard notion of succinctness, and use them to form a hierarchy for the studied fragments of $\mathcal{ML}$ in terms of it. Conclusions and future work are presented in Section 7.

2   Preliminaries

We will work on the basic modal language, presented for convenience in negation normal form. Results can be trivially extended to multi-modal languages.

Definition 2.1 [Syntax] The language of the basic modal logic $\mathcal{ML}$ is given by the following grammar:

$$\varphi ::= \top \mid \bot \mid p \mid \neg p \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \Diamond\varphi \mid \Box\varphi$$

where $p, q, r \ldots$ are propositional symbols. A *literal* is formula of the form $\top$, $\bot$, p or $\neg p$. $\mathcal{ML}^{\Diamond\wedge\neg}$ is the fragment of $\mathcal{ML}$ with no occurrences of $\Box$ and $\vee$. $\mathcal{ML}^{\Diamond\wedge}$ is the fragment of $\mathcal{ML}^{\Diamond\wedge\neg}$ with no literals of the form $\neg p$.

For $\varphi \in \mathcal{ML}$, we use $\overline{\varphi}$ to denote the negation of φ: $\overline{\top} = \bot$, $\overline{p} = \neg p$, $\overline{\varphi \wedge \psi} = \overline{\varphi} \vee \overline{\psi}$, $\overline{\Diamond\varphi} = \Box\overline{\varphi}$, etc. We use $d(\varphi)$ for the *modal depth* of φ, i.e., the maximum number of nested modalities occurring in φ.

To measure *formula size*, we define $|\varphi|$ as the number of literals (counting repetitions) that occur in φ; therefore, we have $|\varphi \vee \psi| = |\varphi \wedge \psi| = |\varphi| + |\psi|$; $|\neg\varphi| = |\varphi|$; $|\Diamond\varphi| = |\Box\varphi| = |\varphi|$; $|p| = 1$ for propositional symbols p; and $|\top| = |\bot| = 1$. Of course there are other reasonable notions of formula size, for instance those counting modal or boolean operators, or even parenthesis. But any reasonable measure of size $||\cdot||$ should satisfy $||\Diamond\varphi|| \geq ||\varphi||, ||\varphi \wedge \psi|| \geq ||\varphi|| + ||\psi||$, etc. and, therefore, will satisfy $||\varphi|| \geq |\varphi|$. Hence all the lower bounds presented in this work will hold for any reasonable definition of size.

As usual, formulas are interpreted using Kripke models $\mathcal{M} = \langle W, R, V \rangle$ where W is a non-empty carrier set, R is a binary relation on W and V maps proposition symbols to subsets of W. We use $sucs_{\mathcal{M}}(w)$ for $\{w' \mid (w, w') \in R\}$ (or simply $sucs(w)$ if $\mathcal{M}$ is clear from context). The size of the finite model $\mathcal{M} = \langle W, R, V \rangle$ (with finite domain W and finite valuation V), denoted $|\mathcal{M}|$, is taken to be $|W| + |R| + |V|$. Here $|V|$ denotes the size of the set of all pairs (w, p) where where p ranges over a finite set of propositional symbols and $w \in V(p)$, while $|R|$ is the number of pairs $(w, v) \in R$.

Definition 2.2 [Semantics] Given $\mathcal{M} = \langle W, R, V \rangle$, the satisfaction relation $\models$ is inductively defined as:

$$\mathcal{M}, w \models \top$$

$$\mathcal{M}, w \models p \quad \text{iff } w \in V(p)$$

$$\mathcal{M}, w \models \neg p \quad \text{iff } w \notin V(p)$$

$$\mathcal{M}, w \models \varphi \wedge \psi \text{ iff } \mathcal{M}, w \models \varphi \text{ and } \mathcal{M}, w \models \psi$$

$$\mathcal{M}, w \models \varphi \vee \psi \text{ iff } \mathcal{M}, w \models \varphi \text{ or } \mathcal{M}, w \models \psi$$

$$\mathcal{M}, w \models \Diamond\varphi \quad \text{iff } \mathcal{M}, w' \models \varphi \text{ for some } w' \in sucs_{\mathcal{M}}(w)$$

$$\mathcal{M}, w \models \Box\varphi \quad \text{iff } \mathcal{M}, w' \models \varphi \text{ for every } w' \in sucs_{\mathcal{M}}(w)$$

If $C \subseteq W$, we write $\mathcal{M}, C \models \varphi$ when $\mathcal{M}, w \models \varphi$ for every $w \in C$. When $\mathcal{M}$ is fixed or clear from context, we shall use the shorter versions $w \models \varphi$ and $C \models \varphi$.

The *description problem* can be seen as a particular case of the more general problem of finding a formula that *separates* two arbitrary sets.

Definition 2.3 Let $\mathcal{M} = \langle W, R, V \rangle$ and let $C, D \subseteq W$ be two non-empty sets. We say that φ *separates C and D in* $\mathcal{M}$ whenever $\mathcal{M}, C \models \varphi$ and $\mathcal{M}, D \models \overline{\varphi}$. When φ separates $\{w\}$ and $W \setminus \{w\}$ in $\mathcal{M}$, we say that φ *is a description for w*. For $c, d \in W$, by 'φ separates c and d' we mean 'φ separates $\{c\}$ and $\{d\}$'.

3   Games, strategies and shortest description size

The standard way of establishing lower bounds on formula size is using Adler-Immerman games [1] (for other techniques and logics see [9,14,18]). In these games, one of the players tries to build a tree that induces a formula of the same size separating two models (or points in a model), while an opponent tries to prevent it. The latter has an optimal strategy in these games, so one only needs to show that the former has a strategy that beats it. This is essentially the technique we will employ.

In order to make this paper self-contained, we will define in this section all the machinery needed. But we shall do it with a slight twist. The trees constructed during Adler-Immerman games can be reinterpreted as decision trees that act as winning strategies for Spoiler in classical Ehrenfeucht-Fraïssé games. This means that while existence of a winning strategy for Duplicator in these games can be used to give lower bounds on the number of nested modalities needed for some task, the minimum size of (certain formalization of) a strategy for Spoiler can be used to give lower bounds on the size of a formula.

We start then defining the classical n-turn Ehrenfeucht-Fraïssé game for $\mathcal{ML}$. Instead of being played on two Kripke models, we find it convenient to define it on two elements of the same Kripke model. Since modal truth is invariant under disjoint unions, no generality is lost.

Definition 3.1 The n-turn Ehrenfeucht-Fraïssé game over model $\mathcal{M} = \langle W, R, V \rangle$ starting on $(w, v) \in W^2$ (notation: $\mathcal{G}_{\mathcal{M}}(w, v, n)$) is played between two players, Spoiler and Duplicator. The rules of the games are:

p_l: Spoiler picks a p such that $w \in V(p)$ and $v \notin V(p)$ and wins.

p_r: Spoiler picks a p such that $v \in V(p)$ and $w \notin V(p)$ and wins.

R_l: If $n > 0$, Spoiler may pick a $w' \in sucs(w)$; then Duplicator must respond choosing a $v' \in sucs(v)$ or otherwise loses. In the first case, the turn ends and they continue to play $\mathcal{G}_{\mathcal{M}}(w', v', n-1)$.

R_r: If $n > 0$, Spoiler may pick a $v' \in sucs(v)$; then Duplicator must respond choosing a $w' \in sucs(w)$ or otherwise loses. In the first case, the turn ends and they continue to play $\mathcal{G}_{\mathcal{M}}(w', v', n-1)$.

Duplicator wins whenever Spoiler cannot play. We write $\mathcal{G}_{\mathcal{M}}^{\diamond \wedge \neg}(w, v, n)$ for the variation of $\mathcal{G}_{\mathcal{M}}(w, v, n)$ without rule R_r, and $\mathcal{G}_{\mathcal{M}}^{\diamond \wedge}(w, v, n)$ for the one that additionally drops rule p_r. We will write $\mathcal{G}(w, v, n)$ when the model is clear from context.

Rules p_l and p_r are not typically part of presentations of Ehrenfeucht-Fraïssé games for $\mathcal{ML}$; they include, instead, the additional constraint on rules R_l and R_r that w' and v' must agree propositionally. Our formulation is clearly equivalent.

Informally, a *strategy* for the game $\mathcal{G}_{\mathcal{M}}(w, v, n)$ is a way of playing in which a player's moves are determined by the previous ones. It is a *winning* strategy for player P when P, following the commands of the strategy, wins the game independently of the opponent's moves. Before turning into the formal definition of *strategy* and *winning strategy* let us mention some is well-known results (see [5, Chapter 3] for more details) regarding winning strategies, bisimilarity and modal equivalence. The following are equivalent:

- Duplicator has a winning strategy for $\mathcal{G}_{\mathcal{M}}(w, v, n)$;
- For every formula φ of $\mathcal{ML}$ with modal depth n, $\mathcal{M}, w \models \varphi$ iff $\mathcal{M}, v \models \varphi$

Hence, w and v are modally equivalent in $\mathcal{M}$ if and only if, for every n, Duplicator has a winning strategy for $\mathcal{G}_{\mathcal{M}}(w, v, n)$. If we drop the restriction of n-rounds and allow for infinite games, then a winning strategy for player P denotes a way of playing in such a way that P can always answer to his opponent's move. Let $\mathcal{G}_{\mathcal{M}}(w, v)$ denote this infinite game with no limit in the number of rounds. Then Duplicator has a winning strategy for $\mathcal{G}_{\mathcal{M}}(w, v)$ (that is, one that prevents Spoiler from reaching any of his winning states) if and only if u and v are bisimilar in $\mathcal{M}$. If u and v are bisimilar in $\mathcal{M}$ then u and v are modally equivalent in $\mathcal{M}$. The converse is not true for arbitrary $\mathcal{M}$ but it holds when $\mathcal{M}$ is finitely branching (this is known as the Hennessy-Milner Theorem [4,10]).

Any strategy for $\mathcal{G}_{\mathcal{M}}(w, v, n)$ can be formalized in a more or less straightforward way using a simple lookup table. However, we are interested in a formalization that will ultimately allow us to correlate strategy and formula size. We therefore formalize Spoiler's strategies using a form of decision tree, which we call *uniform strategy trees*.

Definition 3.2 Let $\mathcal{M} = \langle W, R, V \rangle$ be some fixed model. A *uniform strategy tree* for Spoiler is an annotated tree. We write $x \to y$ to mean that nodes x and y are linked by

an edge and use $x \xrightarrow{C} y$ when the edge is annotated with a non-empty $C \subseteq W$. Nodes can be of six different types: those of type 1 and 2 are annotated with a non-empty set $C \subseteq W$ and are denoted $\langle C \rangle$ and $[C]$, respectively; those of type 3 and 4 are annotated with a proposition symbol p and denoted (p) and $(\bar{p})$; finally, those of type 5 and 6, denoted $(\wedge)$ and $(\vee)$, are not annotated (that is, they do not have any other information apart from the type itself).

Let $C, D \subseteq W$ be non-empty sets; we say that a uniform strategy tree with root x is *winning for* $\mathcal{G}_{\mathcal{M}}(C, D)$ whenever these inductive conditions hold:

(i) If $x = \langle E \rangle$, then we must have $E \cap sucs(\{w\}) \neq \emptyset$ for all $w \in C$, and if $sucs(D) \neq \emptyset$, then $x \to y$ for some y that is winning for $\mathcal{G}_{\mathcal{M}}(E, sucs(D))$.

(ii) If $x = [E]$, then we must have $E \cap sucs(\{w\}) \neq \emptyset$ for all $w \in D$, and if $sucs(C) \neq \emptyset$, then $x \to y$ for some y that is winning for $\mathcal{G}_{\mathcal{M}}(sucs(C), E)$.

(iii) If $x = (p)$, then we must have $C \cap V(p) = C$ and $D \cap V(p) = \emptyset$.

(iv) If $x = (\bar{p})$, then we must have $C \cap V(p) = \emptyset$ and $D \cap V(p) = D$.

(v) If $x = (\wedge)$, then $D = \bigcup\{A \mid \exists y, x \xrightarrow{A} y$ and y is winning for $\mathcal{G}_{\mathcal{M}}(C, A)\}$.

(vi) If $x = (\vee)$, then $C = \bigcup\{A \mid \exists y, x \xrightarrow{A} y$ and y is winning for $\mathcal{G}_{\mathcal{M}}(A, D)\}$.

When a uniform strategy tree that is winning for $\mathcal{G}_{\mathcal{M}}(C, D)$ has no nodes of type 2 nor 6 we say that it is winning for $\mathcal{G}_{\mathcal{M}}^{\Diamond \wedge \neg}(C, D)$; if it doesn't have nodes of type 4 either, we say that it is also winning for $\mathcal{G}_{\mathcal{M}}^{\Diamond \wedge}(C, D)$. Again, we will drop the model when clear from context and say, e.g., that a strategy is *winning for* $\mathcal{G}(C, D)$.

The size $|s|$ of a uniform strategy tree s is the number of *leaf nodes* in s; its depth $d(s)$ is the maximum number of nested nodes of type 1 or 2.

Notice that every uniform strategy tree s has, by definition, a finite height. Therefore, it has a finite size if and only if every node is finitely branching.

Readers familiar with Adler-Immerman games may recognize in conditions (i)–(vi) the rules of the modal version of these games (for formulas in negation normal form). Since in Adler-Immerman games Duplicator has an optimal strategy, it is not surprising that we can give a static characterization of them.

The first thing we need to show is that winning uniform strategy trees indeed constitute winning strategies.

Theorem 3.3 *If there exists a uniform strategy tree for Spoiler with $d(s) \leq n$ that is winning for $\mathcal{G}_{\mathcal{M}}^{\star}(C, D)$, then Spoiler wins every game $\mathcal{G}_{\mathcal{M}}^{\star}(w, v, n)$ with $w \in C$, $v \in D$ (for $\mathcal{G}^{\star} \in \{\mathcal{G}, \mathcal{G}^{\Diamond \wedge \neg}, \mathcal{G}^{\Diamond \wedge}\}$).*

Proof. We proceed by induction on the tree, so let x be its root. If $x = (p)$ or $x = (\bar{p})$, then by definition, Spoiler can play p according to rule p_l or p_r, respectively, and win immediately. If $x = \langle E \rangle$, then Spoiler may play according to rule R_l, picking (non-deterministically) some $w' \in E$ that is an R-successor of w (observe that since $E \cap sucs(\{w\}) \neq \emptyset$, some such successor exists); if Duplicator answers with some $v' \in sucs(D)$, then $sucs(D) \neq \emptyset$ and there must exist some $x \to y$ such that y is winning for $\mathcal{G}(E, sucs(D))$, and by inductive hypothesis, Spoiler wins every instance of $G(w', v', n -$

1). If $x = [E]$, Spoiler may play according to rule R_r and we reason analogously. Suppose now $x = (\wedge)$; for some A with $v \in A$ we must have $x \xrightarrow{A} y$ and since y is winning for $\mathcal{G}(C, A)$, we conclude that Spoiler wins every instance of $\mathcal{G}(w, v, n)$. The case for $x = (\vee)$ is analogous.   □

We will now prove again the well-known Ehrenfeucht-Fraïssé Theorem for $\mathcal{ML}$ but paying attention not only to the modal depth of formulas but also to their sizes. For the rest of this section, we assume a fixed but otherwise arbitrary model $\mathcal{M} = \langle W, R, V \rangle$.

Lemma 3.4 *Let* $C, D \subseteq W$ *be non-empty. If* $\varphi \in \mathcal{L}^\star$ *separates* C *and* D *in* $\mathcal{M}$*, then Spoiler has a uniform strategy tree* s *that is winning for* $\mathcal{G}_\mathcal{M}^\star(C, D)$*, with* $|s| \leq |\varphi|$ *and* $d(s) \leq d(\varphi)$ *(for* $(\mathcal{L}^\star, \mathcal{G}^\star) \in \{(\mathcal{ML}, \mathcal{G}), (\mathcal{ML}^{\Diamond\wedge\neg}, \mathcal{G}^{\Diamond\wedge\neg}), (\mathcal{ML}^{\Diamond\wedge}, \mathcal{G}^{\Diamond\wedge})\})$*.*

Proof. We proceed by induction on φ. Since C and D are non-empty, we cannot have $\varphi = \top$ nor $\varphi = \bot$. If $\varphi = p$, then a leaf-node (p) suffices while $(\overline{p})$ works in case $\varphi = \neg p$. If $\varphi = \Diamond\psi$ then we know there exists a $E \subseteq sucs(C)$ such that $E \cap sucs(\{w\}) \neq \emptyset$ for all $w \in C$ and $E \models \psi$. In case $sucs(D) = \emptyset$, we can use $\langle E \rangle$ (or $\langle sucs(C) \rangle$) as strategy tree. Otherwise, by inductive hypothesis, there is a strategy with root y that is winning for $\mathcal{G}(E, sucs(D))$, $|y| \leq |\psi|$ and $d(y) \leq d(\psi)$. Therefore, the uniform strategy tree with root $x = \langle E \rangle$ such that $x \to y$ must be winning for $\mathcal{G}(C, D)$, $|x| = |y| \leq |\psi| = |\Diamond\psi|$ and $d(x) = 1 + d(y) \leq 1 + d(\psi) = d(\Diamond\psi)$. The case for $\varphi = \Box\psi$ is analogous. Suppose now that $\varphi = \psi_1 \wedge \cdots \wedge \psi_k$ and let $F_i = \{v \in D \mid v \models \overline{\psi_i}\}$. Observe that $D = \bigcup_{i=1}^{k} F_i$, so for some i, $F_i \neq \emptyset$. For each $1 \leq i \leq k$, if $F_i \neq \emptyset$ then there exists, by inductive hypothesis, a uniform strategy tree y_i that is winning for $\mathcal{G}(C, F_i)$. Therefore, the uniform strategy tree whose root x is $(\wedge)$ and such that $x \xrightarrow{F_i} y_i$ for every $F_i \neq \emptyset$, is winning for $\mathcal{G}(C, D)$. Observe also that, by inductive hypothesis, $|x| \leq \sum_{i=i}^{k} |\psi_i| = |\varphi|$ and, similarly, $d(x) \leq d(y)$. The case for $\varphi = \Box(\psi_1 \vee \cdots \vee \psi_k)$ is analogous.   □

Theorem 3.5 (Ehrenfeucht-Fraïssé Theorem) *If Duplicator has some strategy that is winning for* $\mathcal{G}_\mathcal{M}^\star(w, v, n)$*, then for every* $\varphi \in \mathcal{L}^\star$ *with* $d(\varphi) \leq n$*,* $\mathcal{M}, w \models \varphi$ *implies* $\mathcal{M}, v \models \varphi$ *(*$(\mathcal{L}^\star, \mathcal{G}^\star) \in \{(\mathcal{ML}, \mathcal{G}), (\mathcal{ML}^{\Diamond\wedge\neg}, \mathcal{G}^{\Diamond\wedge\neg}), (\mathcal{ML}^{\Diamond\wedge}, \mathcal{G}^{\Diamond\wedge})\}$*).*

Proof. Suppose $w \models \varphi$ and $v \not\models \varphi$. This means that φ separates w and v and, by Lemma 3.4, Spoiler has a uniform strategy tree s with $d(s) \leq n$ that is winning for $\mathcal{G}^\star(\{w\}, \{v\})$. Therefore, by Theorem 3.3, Spoiler wins every instance of $\mathcal{G}^\star(w, v, n)$, so Duplicator cannot have a winning strategy.   □

Observe that the condition "$\mathcal{M}, w \models \varphi$ implies $\mathcal{M}, v \models \varphi$" is equivalent to "$\mathcal{M}, w \models \varphi$ iff $\mathcal{M}, v \models \varphi$" in $\mathcal{ML}$, but not in $\mathcal{ML}^{\Diamond\wedge}$ nor $\mathcal{ML}^{\Diamond\wedge\neg}$, since they are not closed under negation.

For the converse of Lemma 3.4 we need the additional requirement that s is finitely branching.

Lemma 3.6 *If Spoiler has a uniform strategy tree* s *of finite size that is winning for* $\mathcal{G}^\star(C, D)$*, then there exists a* $\varphi \in \mathcal{ML}^\star$ *such that* $|\varphi| \leq |s|$ *and* $d(\varphi) \leq d(s)$ *that separates* C *and* D *(for* $(\mathcal{L}^\star, \mathcal{G}^\star) \in \{(\mathcal{ML}, \mathcal{G}), (\mathcal{ML}^{\Diamond\wedge\neg}, \mathcal{G}^{\Diamond\wedge\neg}), (\mathcal{ML}^{\Diamond\wedge}, \mathcal{G}^{\Diamond\wedge})\}$*).*

Proof. We proceed by induction on s, and let x be its root. If $x = (p)$, then p trivially satisfies $C \models p$ and $D \models \bar{p}$. Similarly, $\neg p$ can handle the case $x = (\bar{p})$. In case $x = \langle E \rangle$, then either $sucs(D) = \emptyset$ and $\Diamond \top$ is the formula we need or else there exists an y that is winning for $\mathcal{G}(E, sucs(D))$ and, by inductive hypothesis for some ψ we have $E \models \psi$, $sucs(D) \models \overline{\psi}$, $|\psi| \leq |y|$ and $(\psi) \leq d(y)$. Clearly, $D \models \Box \overline{\psi}$, $|\Diamond \psi| \leq |x|$, $d(\Diamond \psi) \leq d(x)$ and, because $E \cap sucs(\{w\}) = \emptyset$ for every $w \in C$, we can also conclude $C \models \Diamond \psi$. The case for for $x = [E]$ is analogous. Suppose now that $x = (\wedge)$. Since s has finite size, there can be only finitely many y such that $x \xrightarrow{A} y$ (there is at least one y since D is non-empty). For every such y there exists, by inductive hypothesis, a formula φ_y such that $C \models \varphi_y$ and $A \models \overline{\varphi_y}$; by taking φ to be the conjunction of all such φ_y, we get $C \models \varphi$ and $D \models \overline{\varphi}$. The case for $x = (\vee)$ is symmetrical. $\square$

We are now ready to give the main result of this section.

Definition 3.7 We say that a uniform strategy tree s that is winning for $\mathcal{G}^\star_\mathcal{M}(C, D)$ is *minimum* whenever for any other uniform strategy tree s' winning for $\mathcal{G}^\star_\mathcal{M}(C, D)$, $|s| \leq |s'|$ (for $\mathcal{G}^\star \in \{\mathcal{G}, \mathcal{G}^{\Diamond \wedge \neg}, \mathcal{G}^{\Diamond \wedge}\}$). Similarly, a formula $\varphi \in \mathcal{ML}^\star$ that separates C and D in $\mathcal{M}$ is *minimum* whenever for any $\psi \in \mathcal{ML}^\star$ that separates C and D in $\mathcal{M}$, $|\varphi| \leq |\psi|$ (for $\mathcal{ML}^\star \in \{\mathcal{ML}, \mathcal{ML}^{\Diamond \wedge \neg}, \mathcal{ML}^{\Diamond \wedge}\}$).

Theorem 3.8 *If s is a minimum uniform strategy tree winning for $\mathcal{G}^\star_\mathcal{M}(C, D)$ and $\varphi \in \mathcal{ML}^\star$ is a minimum formula that separates C and D in $\mathcal{M}$, then $|s| = |\varphi|$ (for $(\mathcal{L}^\star, \mathcal{G}^\star) \in \{(\mathcal{ML}, \mathcal{G}), (\mathcal{ML}^{\Diamond \wedge \neg}, \mathcal{G}^{\Diamond \wedge \neg}), (\mathcal{ML}^{\Diamond \wedge}, \mathcal{G}^{\Diamond \wedge})\}$).*

Proof. By Lemma 3.6, there exists a ψ that separates C and D such that $|\psi| \leq |s|$, and since φ is minimum, we know $|\varphi| \leq |\psi| \leq |s|$. Now, by Lemma 3.4, there exists an s' that is winning for $\mathcal{G}^\star(C, D)$ with $|s'| \leq |\varphi|$, and since s is minimum we conclude $|s| \leq |s'| \leq |\varphi| \leq |\psi| \leq |s|$. $\square$

A simple inspection of Definition 3.2 shows that if a uniform tree strategy is winning for $\mathcal{G}(C, D)$, then it is also winning for $\mathcal{G}(C', D')$ for every non-empty $C' \subseteq C$ and $D' \subseteq D$. This shows that in order to give a lower bound for the description problem for w it suffices to guarantee that w has a description and give a lower bound for the size of a formula that separates $\{w\}$ and D for some D with $w \notin D$. This will be pursued in Section 4 and the following results will be useful.

Proposition 3.9 *If s is a uniform strategy tree winning for $\mathcal{G}^\star(C, D)$ whose root is of the form $\langle E \rangle$ (resp. $[E]$), then there exists a uniform strategy tree s' with root $\langle E' \rangle$ (resp. $[E']$) that is winning for $\mathcal{G}^\star(C, D)$ and such that $|s| = |s'|$ and $E' \subseteq sucs(C)$ (resp. $E' \subseteq sucs(D)$), for $\mathcal{G}^\star \in \{\mathcal{ML}, \mathcal{ML}^{\Diamond \wedge \neg}, \mathcal{ML}^{\Diamond \wedge}\}$.*

Proof. Let $x = \langle E \rangle$ be the root of s; define $x' = \langle E \cap sucs(C) \rangle$ and set $x' \to y$ for every y such that $x \to y$. Since C is not empty and for every $w \in C$, $E \cap sucs(w) \neq \emptyset$, we know $E \cap sucs(C)$ is not empty. If $x \to y$ then y is winning for $\mathcal{G}(E, sucs(D))$ and, by the observation above, y is also winning for $\mathcal{G}(E \cap sucs(C), sucs(D))$ and therefore x' is winning for $\mathcal{G}(C, D)$. The case for $[E]$ is analogous. $\square$

Proposition 3.10 *If s is a uniform strategy tree that is winning for $\mathcal{G}^\star(C,D)$ ($\mathcal{G}^\star \in \{\mathcal{G}, \mathcal{G}^{\diamond\wedge\neg}, \mathcal{G}^{\diamond\wedge}\}$), then $C \cap D = \emptyset$.*

Proposition 3.11 *Let $\mathcal{G}^\star \in \{\mathcal{G}, \mathcal{G}^{\diamond\wedge\neg}, \mathcal{G}^{\diamond\wedge}\}$ and let s be a winning strategy for $\mathcal{G}^\star(C,D)$ of minimum size. If D is singleton (resp. C is singleton) and the root of s is of type $(\wedge)$ (resp. of type $(\vee)$) then there is a winning strategy s' for $\mathcal{G}^\star(C,D)$ such that $|s| = |s'|$ and the root of s' is not of type $(\wedge)$ (resp. of type $(\vee)$).*

Proof. If s is of type $(\wedge)$ and D is singleton then $s \xrightarrow{D} s_1$ is the only possible beginning of s (here s_1 is the only child of s because s is minimum). Like s, the subtree s_1 is winning for $\mathcal{G}^\star(C,D)$ and $|s| = |s_1|$. Let n be the least such that $s \xrightarrow{D} s_1 \xrightarrow{D} s_2 \ldots s_{n-1} \xrightarrow{D} s_n$ and s_n is not of type $(\wedge)$. By a simple induction one can show that the subtree s_n of s is winning for $\mathcal{G}^\star(C,D)$ and $|s| = |s_n|$. The case for C singleton is analogous. $\square$

Proposition 3.12 *If s is a uniform strategy tree winning for $\mathcal{G}(C,D)$, then there exists a winning strategy tree s' for $\mathcal{G}(D,C)$ and $|s| = |s'|$.*

Proof. We obtain s' from s by applying on each node of s substitution σ, where $\sigma = [\langle E \rangle \mapsto [E], [E] \mapsto \langle E \rangle, (p) \mapsto (\overline{p}), (\overline{p}) \mapsto (p), (\wedge) \mapsto (\vee), (\vee) \mapsto (\wedge)]$. By a trivial induction, s is winning for $\mathcal{G}(C,D)$ iff $s' = \sigma(s)$ is winning for $\mathcal{G}(D,C)$. $\square$

4   Lower bound for the size of modal descriptions

We say that, for a modal logic $\mathcal{L}$, the size of the $\mathcal{L}$-separation problem is bounded by f if for all finite models $\mathcal{M} = \langle W, R, V \rangle$ and non-empty $C, D \subseteq W$ if there is an $\mathcal{L}$-formula that separates C and D then there is one such formula of size at most $f(|\mathcal{M}|)$. We say it is polynomially bounded when it is bounded by some polynomial. Similarly, we say that f is a lower bound for the size of the $\mathcal{L}$-separation problem when there are infinitely many models $\mathcal{M} = \langle W, R, V \rangle$ and non-empty $C, D \subseteq W$ such that an $\mathcal{L}$-formula $\varphi_\mathcal{M}$ separates C and D, and all such formulas have size at least $f(|\mathcal{M}|)$. We say that the size of the $\mathcal{L}$-separation problem has an exponential lower bound when there is a fixed $b > 1$ such that b^x is a lower bound for the $\mathcal{L}$-separation problem. The notions are analogously defined for the $\mathcal{L}$-description problem.

In general one cannot conclude that an a in the domain of $\mathcal{M}$ has exclusively $\mathcal{L}$-descriptions of size at least $f(\mathcal{M})$ from the fact that a is $\mathcal{L}$-separable from some b exclusively by formulas of size at least $f(|\mathcal{M}|)$ (a could have no $\mathcal{L}$-descriptions at all). However, the implication is true when the a in question does have an $\mathcal{L}$-description.

In this section we show that, for $\mathcal{L} \in \{\mathcal{ML}, \mathcal{ML}^{\diamond\wedge\neg}, \mathcal{ML}^{\diamond\wedge}\}$, the size of the $\mathcal{L}$-separation and $\mathcal{L}$-description problems has an exponential lower bound and therefore it cannot be polynomially bounded. We use the machinery introduced in Section 3 to show lower bounds on Spoiler's winning uniform strategy trees and hence the size of the corresponding formulas.

Theorem 4.1 *There is a recursive family of acyclic finite models with two distinguished points $(\mathcal{M}_n, a_n, b_n)_{n \in \mathbb{N}}$ such that $|\mathcal{M}_n| \in O(n)$ and the size of the shortest $\mathcal{ML}$-formula φ_n that separates a_n from b_n in $\mathcal{M}_n$ is exponential in n. Furthermore, there exists an $\mathcal{ML}$-description of a_n in $\mathcal{M}_n$.*
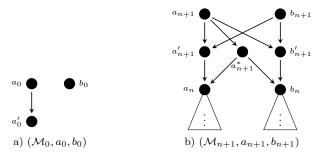
Fig. 1. Recursive family of models $(\mathcal{M}_n, a_n, b_n)_{n \in \mathbb{N}}$

Proof. The definition of $(\mathcal{M}_n, a_n, b_n)_{n \in \mathbb{N}}$ is shown in Figure 1. It is not hard to see that for all n, $\mathcal{M}_n$ is acyclic and $|\mathcal{M}_n| \in O(n)$. We now show by induction on n that for all n there exists a minimum uniform strategy tree s_n such that $|s_n| = 2^n$. Since $|\mathcal{M}_n| \in O(n)$, we conclude from this that $|s_n|$ is exponential in $|\mathcal{M}|$ and by Theorem 3.8 the minimum formula φ_n that separates a_n from b_n is exponential in $|\mathcal{M}|$ too.

For $n = 0$, we have that $s_0 = \langle \{a_0'\} \rangle$ is clearly winning for $\mathcal{G}(\{a_0\}, \{b_0\})$ and since $|s_0| = 1$, s_0 is minimum. Now assume s_n is a minimum uniform strategy tree that is winning for $\mathcal{G}(\{a_n\}, \{b_n\})$ and let s_{n+1} be a minimum winning strategy tree for $\mathcal{G}(\{a_n\}, \{b_n\})$. We do a case analysis of s_{n+1} to rule out possibilities and ensure that s_{n+1} is unique (up-to redundant occurrences of nodes of type $(\wedge)$, see below) and exponentially large.

In what follows, we avoid subscript $n+1$ for convenience (e.g., we write a' for a'_{n+1}). The reader can track the name of the nodes we use and the shape of the resulting strategy in Figure 2. We use the convention for nodes of type $\langle \cdot \rangle$ and $[\cdot]$ guaranteed by Proposition 3.9.

The first thing to observe is that since $\mathcal{M}_n \models \neg p$, for all p, no nodes of type (p) or $(\bar{p})$ can occur in s_{n+1} at all. Secondly, observe that using Proposition 3.11, we can assume without loss of generality that the root of s_{n+1} is not of type $(\wedge)$ or $(\vee)$. Next we can rule out also the case $s_{n+1} = [E] \to x$, for some $E \subseteq \{a', b'\}$, for that would imply that x is winning for $\mathcal{G}(\{a', b'\}, E)$, which contradicts Proposition 3.10.

We can assume, therefore that $s_{n+1} = \langle E \rangle \to x$, for some non-empty $E \subseteq \{a', b', a^*\}$ and that x is winning for $\mathcal{G}(E, \{a', b'\})$. But by Proposition 3.10, we may conclude $a' \notin E$ and $b' \notin E$. Hence, we must have $E = \{a^*\}$.

We now perform a similar case analysis on x. By Proposition 3.11, we may assume that x is not of type $(\vee)$. If $x = \langle F \rangle \to y$, for some non-empty $F \subseteq \{a_n, b_n\}$, then y would have to be winning for $\mathcal{G}(F, \{a_n, b_n\})$ which contradicts Proposition 3.10. Similarly, we can see that we cannot have $x = [F] \to y$.

Therefore, we can assume that x is of type $(\wedge)$, winning for $\mathcal{G}(\{a^*\}, \{a', b'\})$ and minimum. Notice that we can ignore, without loss of generality, the case where x has only one successor y with $x \overset{\{a', b'\}}{\to} y$ (for in that case y would also be a minimum uniform strategy tree for $\mathcal{G}(\{a^*\}, \{a', b'\})$). We conclude, then, that x has two children y_1 and y_2 such that $x \overset{\{a'\}}{\to} y_1$ and $x \overset{\{b'\}}{\to} y_2$. Furthermore, y_1 is winning for $\mathcal{G}(\{a^*\}, \{a'\})$ and y_2 is winning for $\mathcal{G}(\{a^*\}, \{b'\})$.
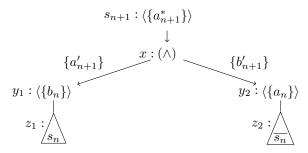
Fig. 2. s_{n+1}, minimum uniform strategy tree winning for $\mathcal{G}_{\mathcal{M}_{n+1}}(\{a_{n+1}\}, \{b_{n+1}\})$.

Using again Proposition 3.11, we conclude that y_1 and y_2 are not of type $(\wedge)$ nor $(\vee)$. If $y_1 = [G_1] \to z_1$, the only possibility for G_1 is $\{a_n\}$. So z_1 would have to be winning for $\mathcal{G}(\{a_n, b_n\}, \{a_n\})$ contradicting Proposition 3.10.

Hence we must have $y_1 = \langle E_1 \rangle \to z_1$ and $y_2 = \langle E_1 \rangle \to z_2$ with $E_i \subseteq \{a_n, b_n\}$; z_1 must be winning for $\mathcal{G}(\{E_1\}, \{a_n\})$ and z_2 for $\mathcal{G}(\{E_2\}, \{b_n\})$. By Proposition 3.10, $a_n \notin E_1$ and $b_n \notin E_2$, so $E_1 = \{b_n\}$ and $E_2 = \{a_n\}$.

For s to be minimum, both z_1 and z_2 have to be minimum strategies winning for $\mathcal{G}(\{b_n\}, \{a_n\})$ and $\mathcal{G}(\{a_n\}, \{b_n\})$ respectively. We can therefore assume that $z_1 = s_n$ and, by inductive hypothesis, $|z_1| = 2^n$. Using Proposition 3.12, we conclude $|z_2| = 2^n$ and since $|s_{n+1}| = |z_1| + |z_2|$, we obtain $|s_{n+1}| = 2^{n+1}$.

By Lemma 3.6, there exists a separating formula φ_n associated to each s_n. It is not hard to see that they are $\varphi_0 := \Diamond\top$ and $\varphi_{n+1} := \Diamond(\Diamond\varphi_n \wedge \Diamond\overline{\varphi_n})$. But observe that φ_n is stronger than $\Diamond^{2n+1}\top$ and since clearly $\mathcal{M}_n, w \not\models \Diamond^{2n+1}\top$ for all w other than a_n and b_n, we have that φ_n is a description for a_n. $\square$

Corollary 4.2 *The size of the $\mathcal{ML}$-separation and $\mathcal{ML}$-description problems has an exponential lower bound and therefore it is not polynomially bounded.*

An inspection of the proof of Theorem 4.1 reveals that any $\mathcal{ML}$-formula that separates a_n and b_n in $\mathcal{M}_n$ (with $n > 1$) must use $\Box$ and $\vee$. This already implies that one cannot separate a_n and b_n in $\mathcal{M}_n$ using the logics $\mathcal{ML}^{\Diamond\wedge}$ or $\mathcal{ML}^{\Diamond\wedge\neg}$ (one could alternatively show that consistently playing a'_n constitutes a winning strategy for Duplicator).

In order to show that the size of the $\mathcal{L}$-separation and $\mathcal{L}$-description problem for $\mathcal{L} \in \{\mathcal{ML}^{\Diamond\wedge}, \mathcal{ML}^{\Diamond\wedge\neg}\}$ has an exponential lower bounds, we need to find another family of models. The models in this case turned out to be somehow more complex.

Theorem 4.3 *There is a recursive family of acyclic finite models with two distinguished points $(\mathcal{N}_n, a_n, b_n)_{n\in\mathbb{N}}$ such that $|\mathcal{N}_n| \in O(n)$ and the size of the shortest $\mathcal{ML}^{\Diamond\wedge\neg}$-formula ψ_n that separates a_n and b_n in $\mathcal{N}_n$ is exponential in n. Furthermore, there exists an $\mathcal{ML}^{\Diamond\wedge\neg}$-description of a_n in $\mathcal{N}_n$.*

Proof. The definition of $(\mathcal{N}_n, a_n, b_n)_{n\in\mathbb{N}}$ is given in Figure 3. Notice that now the models interpret a propositional variable p. It is not hard to see that for all n, $\mathcal{N}_n$ is acyclic and $|\mathcal{N}_n| \in O(n)$. One proceeds as in the proof of Theorem 4.1, and shows by induction on n that the minimum uniform strategy tree winning for $\mathcal{G}^{\Diamond\wedge\neg}(\{a_n\}, \{b_n\})$ has size, in this case, $2^n 3 - 2$, which is the closed form of $|s_0| = 1; |s_{n+1}| = 2|s_n| + 2$.
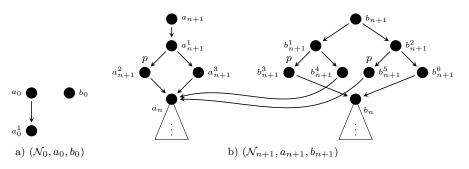
Fig. 3. Recursive family of models $(\mathcal{N}_n, a_n, b_n)_{n \in \mathbb{N}}$

Details can be found in Appendix A.   □

Corollary 4.4 *The size of the $\mathcal{ML}^{\diamond \wedge \neg}$-separation and $\mathcal{ML}^{\diamond \wedge \neg}$-description problems has an exponential lower bound and therefore it is not polynomially bounded.*

The proof of Theorem 4.3 shows that atomic negation is necessary to separate a_n and b_n in $\mathcal{N}_n$. Therefore, there is no $\mathcal{ML}^{\diamond \wedge}$-formula separating a_n and b_n in $\mathcal{N}_n$. However a simple modification of the models $(\mathcal{N}_n)_{n \in \mathbb{N}}$ in the proof of Theorem 4.3 shows the same result for $\mathcal{ML}^{\diamond \wedge}$ instead of $\mathcal{ML}^{\diamond \wedge \neg}$.

Theorem 4.5 *There is a recursive family of acyclic finite models with two distinguished points $(\mathcal{N}'_n, a_n, b_n)_{n \in \mathbb{N}}$ such that $|\mathcal{N}'_n| \in O(n)$ and the size of the shortest $\mathcal{ML}^{\diamond \wedge}$-formula ψ' that separates a_n and b_n in $\mathcal{N}'_n$ is exponential in n. Furthermore, there exists an $\mathcal{ML}^{\diamond \wedge}$-description of a_n in $\mathcal{N}_n$.*

Proof. Define $\mathcal{N}'_n$ in the same way as $\mathcal{N}_n$ in the proof of Theorem 4.3, but introduce a second propositional variable q and set $V(q) = \{a_n^3, b_n^4, b_n^6\}$ in $\mathcal{N}'_n$; that is, make q true in all the nodes of the third level of $\mathcal{N}_n$ where p was false. The proof is completely analogous.   □

Corollary 4.6 *The size of the $\mathcal{ML}^{\diamond \wedge}$-separation and $\mathcal{ML}^{\diamond \wedge}$-description problems has an exponential lower bound and therefore it is not polynomially bounded.*

5   Upper bound for the size of modal descriptions

In the previous section we found an exponential lower bound for the size of a modal formula that describes an element of the domain, more precisely, $O(b^{|\mathcal{M}|})$ for $b \in (1, 2]$. We will now analyze the complexity of a simple algorithm that computes such formulas in order to find an upper bound for this problem. We will see that its complexity is $O(2^{|\mathcal{M}|^2} \cdot |\mathcal{M}|)$, so, while the lower bound is not tight it is still reasonable. We expect that tighter upper bounds can be obtained by considering better algorithms.

Assume a fixed *finite* model $\mathcal{M} = \langle W, R, V \rangle$. We define now a simple procedure which maintains, at each step s, a relation $\sim_s \subseteq W \times W$ and a map $f_s : W \to \mathcal{ML}$ that satisfy the following invariant: "if $w \not\sim_s v$, then Spoiler has a winning strategy for the game $\mathcal{G}(w, v, d(f_s(w)))$; witnessed by the fact that $w \models f_s(w)$ and $v \not\models f_s(w)$". The algorithm computes the largest such $\sim_s$, which, of course, corresponds to the maximum

autobisimulation on $\mathcal{M}$ (in fact, it can be seen as a variation of Hopcroft's bisimulation algorithm [11]). The procedure goes as follows:

- *Step* 0: Let $P(v) := \{p \mid v \in V(p)\}$ and $\overline{P}(v) := \{\neg p \mid v \notin V(p)\}$. Define

$$f_0(w) := \bigwedge (P(w) \cup \overline{P}(w)) \qquad \text{for all } w \in W$$
$$\sim_0 := \{(u, v) \mid P(u) = P(v)\}$$

- *Step* $s+1$: Pick any two $u, v \in W$ with $u \sim_s v$ that satisfy Condition 1 or 2 below and proceed accordingly. If no such elements exist, stop.

  · *Condition 1:* For some $u' \in sucs(u)$, there is no element $v' \in sucs(v)$ such that $u' \sim_s v'$. In that case, set

  $$f_{s+1}(u) := f_s(u) \wedge \Diamond f_s(v')$$
  $$f_{s+1}(v) := f_s(v) \wedge \Box \overline{f_s(v')}$$
  $$f_{s+1}(x) := f_s(x) \qquad \text{for any } x \notin \{u, v\}$$
  $$\sim_{s+1} := \sim_s \setminus \{(u, v), (v, u)\}$$

  · *Condition 2:* For some $v' \in sucs(v)$, there is no element $u' \in sucs(u)$ such that $u' \sim_s v'$. In that case, set

  $$f_{s+1}(u) := f_s(u) \wedge \Box \overline{f_s(v')}$$
  $$f_{s+1}(v) := f_s(v) \wedge \Diamond f_s(v')$$
  $$f_{s+1}(x) := f_s(x) \qquad \text{for any } x \notin \{u, v\}$$
  $$\sim_{s+1} := \sim_s \setminus \{(u, v), (v, u)\}$$

Clearly the invariant holds after *Step 0*, and assuming it holds at the beginning of *Step* $s+1$, it is straightforward to see that whenever Condition 1 holds, then Spoiler wins any $\mathcal{G}(u, v, n)$ (for certain n) by playing first u' according to rule R_l, so the invariant is maintained (similar for Condition 2). The procedure is guaranteed to terminate and, if it does by stage k then, because of the invariant, $f_k(u)$ is a description for u whenever $u \not\sim_k v$ for $u \neq v$.

Notice that this procedure does not compute a *minimum* description for u (this problem appears to be harder). Even more, it is shown in [2] that for the class of converse well-founded, linear models, there exist executions of this procedure [5] that lead to formulas whose size cannot be bound by a polynomial. However, every element in a model in this class has a modal formula description of linear size.

In any case, the analysis of this simple procedure will give us an upper bound for the size of the *minimum* modal description of an element.

The first thing to observe is that the procedure terminates at most by stage $\frac{1}{2}(|W|^2 - |W|)$. This is because at each step s we have $|\sim_{s+1}| = |\sim_s| - 2$, $|\sim_0| \leq |W|^2$ and for every $w \in W$ and s, $w \sim_s w$.

Let $M(s) = \max\{|f_s(v)| \mid v \in W\}$. It is clear that $M(0) \in O(|V|)$ and (since the witnesses u and v are distinct) we have $M(s+1) \leq 2 \cdot M(s)$. We conclude that

---

[5] The procedure is non-deterministic on the choice of u and v on *Step* $s+1$.

$M(s) \in O(2^s \cdot |V|)$. Therefore, if $v \in W$ has a modal description, the one computed by this procedure has size at most $M(\frac{1}{2}(|W|^2 - |W|))$.

Theorem 5.1 *Let $\mathcal{M} = \langle W, R, V \rangle$ and $v \in W$. If $\varphi \in \mathcal{ML}$ is a minimum description of v in $\mathcal{M}$, then $|\varphi| \in O(2^{\frac{1}{2}|W|^2} \cdot |V|)$.*

Obtaining an upper bound for $\mathcal{ML}^{\diamond\wedge}$ and $\mathcal{ML}^{\diamond\wedge\neg}$ is not difficult. The main difference is that the simulation notion for these two logics is no longer symmetric. Hence we have to treat (u, v) and (v, u) separately. For $\mathcal{ML}^{\diamond\wedge\neg}$ the same procedure applies with the following three modifications: a) since Spoiler cannot play according to rule R_r, only *Condition 1* is considered; b) only the value for u has to be updated in f_{s+1} (i.e., $f_{s+1}(v) = f_s(v)$); c) $\sim_{s+1}$ has to be defined as $\sim_s \setminus \{(u, v)\}$. For $\mathcal{ML}^{\diamond\wedge}$ the same three modifications have to be made, plus d) replace $=$ by $\subseteq$ in the initialization of $\sim_0$.

The procedure for $\mathcal{ML}^{\diamond\wedge}$ or $\mathcal{ML}^{\diamond\wedge\neg}$ terminates at most by step $|W|^2 - |W|$ and the same analysis as the one explained for $\mathcal{ML}$ applies in this case. Hence the upper bound for the size of minimum descriptions in $\mathcal{ML}^{\diamond\wedge}$ or $\mathcal{ML}^{\diamond\wedge\neg}$ is $O(|W^2| - |W|)$.

Theorem 5.2 *Let $\mathcal{M} = \langle W, R, V \rangle$ and $v \in W$. If $\varphi \in \mathcal{ML}^{\diamond\wedge\neg}$ (or $\varphi \in \mathcal{ML}^{\diamond\wedge}$) is a minimum description of v in $\mathcal{M}$, then $|\varphi| \in O(2^{|W|^2} \cdot |V|)$.*

6 Succinctness for separation

The bounds established in the previous sections resemble classical succinctness results. It is therefore interesting to analyze in which way they differ.

Succinctness deals with how short a formula can be found to express a given property. It is especially important when studying two equally expressive logics $\mathcal{L}$ and $\mathcal{L}'$. In these situations, succinctness is the foremost quantitative measure to distinguish $\mathcal{L}$ and $\mathcal{L}'$. Informally speaking, if we find a an infinite collection of properties Φ_n, each expressible in $\mathcal{L}$ with a formula φ_n of size $f(n)$ and all the formulas of $\mathcal{L}'$ expressing the same property Φ_n (that is, all the $\mathcal{L}'$-formulas semantically equivalent to φ_n) are much larger than $f(n)$, then we say that $\mathcal{L}$ is more succinct than $\mathcal{L}'$.

In this section we will see that the results of Section 4 can also be used to distinguish the three logics considered (and, more generally, other logics) by means of a quantitative measure. Informally, if an $\mathcal{L}$-formula φ_n of size $f(n)$ separates C_n and D_n in $\mathcal{M}_n$ (for an infinite collection of models $\mathcal{M}_n$) and C_n and D_n are also separable in $\mathcal{L}'$, but only with $\mathcal{L}'$-formulas much larger than $f(n)$ then we say that $\mathcal{L}$ is more *succinct for separation* than $\mathcal{L}'$. It is interesting to observe, though, that this notion is not a form of succinctness as described above: the short $\mathcal{L}$-formula φ_n need not be semantically equivalent to none of the exponentially larger ones of $\mathcal{L}$.

We will ultimately show that $\mathcal{ML}$ is more succinct for separation than $\mathcal{ML}^{\diamond\wedge\neg}$, which in turn is more succinct for separation than $\mathcal{ML}^{\diamond\wedge}$. We shall do this in a rather formal (arguably, too formal) way, but this will allow us to close this section drawing some promising connections with the field of Algorithmic Information Theory.

Let us turn to the formal analysis. We say that $\mathcal{L} \geq_p \mathcal{L}'$ if there is a truth-preserving translation T mapping $\mathcal{L}'$-formulas into $\mathcal{L}$-formulas and p is a polynomial such that $|T(\varphi)| \leq p(|\varphi|)$ for all $\mathcal{L}'$-formula φ.

Knowing $\mathcal{L} \geq_p \mathcal{L}'$ tells us, for example, that every two sets separated by a formula φ of $\mathcal{L}'$ can be separated by a formula of $\mathcal{L}$ which is not much larger than φ. But $\mathcal{L}$ might in principle separate sets in a much shorter way.

Definition 6.1 Given a modal logic $\mathcal{L}$ and a suitable Kripke model $\mathcal{H} = \langle W, R, V \rangle$, let $S_{\mathcal{H}}^{\mathcal{L}} : \mathcal{P}(W)^2 \to \mathbb{N}$ be the *separation complexity* of $\mathcal{L}$ within $\mathcal{H}$. For $C, D \subseteq W$, $S_{\mathcal{H}}^{\mathcal{L}}(C, D)$ is defined as the size of the shortest $\mathcal{L}$-formula which separates C and D in $\mathcal{H}$ in case there is such separator or ∞ otherwise.

Clearly if $\mathcal{L} \geq_p \mathcal{L}'$ then the following hold:

$$S_{\mathcal{H}}^{\mathcal{L}} \leq p \circ S_{\mathcal{H}}^{\mathcal{L}'} \tag{1}$$

$$S_{\mathcal{H}}^{\mathcal{L}}(C, D) = S_{\mathcal{H}}^{\mathcal{L}}(D, C) \tag{2}$$

$$S_{\mathcal{H}}^{\mathcal{L}}(C', D') \leq S_{\mathcal{H}}^{\mathcal{L}}(C, D) \text{ (whenever } C' \subseteq C \text{ and } D' \subseteq D). \tag{3}$$

The notion of size of the $\mathcal{L}$-separation problem bounded by f introduced in Section 4 can be restated in the following way: the size of the $\mathcal{L}$-separation problem is bounded by f if for all finite model $\mathcal{H} = \langle W, R, V \rangle$ and $C, D \subseteq W$ if $S_{\mathcal{H}}^{\mathcal{L}}(C, D) < \infty$ then $S_{\mathcal{H}}^{\mathcal{L}}(C, D) \leq f(|\mathcal{H}|)$.

The argument we have used in Section 4 to show that the size of the $\mathcal{L}$-separation problem (for $\mathcal{L} \in \{\mathcal{ML}, \mathcal{ML}^{\diamond\wedge\neg}, \mathcal{ML}^{\diamond\wedge}\}$) has an exponential lower bound is to exhibit a sequence $(\mathcal{H}_n, a_n, b_n)_{n \in \mathbb{N}}$ where a_n and b_n are elements of $\mathcal{H}_n$ such that there is $b > 1$ such that for all n, $\infty > S_{\mathcal{H}_n}^{\mathcal{L}}(\{a_n\}, \{b_n\}) > b^{|\mathcal{H}_n|}$. On the other hand, results of Section 5 show that for all $\mathcal{H} = \langle W, R, V \rangle$ and $a \in W$, $S_{\mathcal{H}}^{\mathcal{ML}}(\{a\}, W \setminus \{a\}) \in O(|V| \cdot 2^{\frac{1}{2}|W|^2})$ and $S_{\mathcal{H}}^{\mathcal{L}'}(\{a\}, W \setminus \{a\}) \in O(|V| \cdot 2^{|W|^2})$, for $\mathcal{L}' \in \{\mathcal{ML}^{\diamond\wedge}, \mathcal{ML}^{\diamond\wedge\neg}\}$.

We next introduce formally our notion of succinctness for separation that may be applied to logics which do not necessarily have the same expressive power.

Definition 6.2 Let $\mathcal{L} \geq \mathcal{L}'$. We say that $\mathcal{L}$ is *f-more succinct for separation* than $\mathcal{L}'$ if there is a sequence

$$(\mathcal{H}_n = \langle W_n, R_n, V_n \rangle, C_n \subseteq W_n, D_n \subseteq W_n)_{n \in \mathbb{N}} \tag{4}$$

of finite models with two distinguished sets such that C_n, D_n is separable in $\mathcal{L}'$ and there is a polynomial p such that for almost all n (that is for all n except finitely many), $S_{\mathcal{H}_n}^{\mathcal{L}'}(C_n, D_n) - S_{\mathcal{H}_n}^{\mathcal{L}}(C_n, D_n) > f(|\mathcal{H}_n|)$.

The idea is that $\mathcal{L}$ is f-more succinct for separation than $\mathcal{L}'$ when there is a sequence of examples (4) showing that the difference between $S_{\mathcal{H}_n}^{\mathcal{L}'}$ and $S_{\mathcal{H}_n}^{\mathcal{L}}$ grows faster than $f(|\mathcal{H}_n|)$. When f is of the form b^x, for a fixed $b > 1$, we simply say that $\mathcal{L}$ is exponentially more succinct for separation than $\mathcal{L}'$.

We ask if the additional expressive power of $\mathcal{ML}$ over $\mathcal{ML}^{\diamond\wedge\neg}$ is enough to be exponentially more succinct for separation than $\mathcal{ML}^{\diamond\wedge\neg}$, and if the additional expressive power of $\mathcal{ML}^{\diamond\wedge\neg}$ is enough to be exponentially more succinct for separation than $\mathcal{ML}^{\diamond\wedge}$. As we have anticipated, in both cases the answer is yes.

Theorem 6.3 $\mathcal{ML}$ *is exponentially more succinct for separation than* $\mathcal{ML}^{\diamond\wedge\neg}$.

Proof. Recall $(\mathcal{N}_n, a_n, b_n)_{n\in\mathbb{N}}$ from the proof of Theorem 4.3. For each n, a_n and b_n can be separated in $\mathcal{N}_n$ by χ_n, where $\chi_0 := \diamond\top$ and $\chi_{n+1} := \diamond\square\diamond\chi_n$. Clearly $|\chi_n| \in O(n)$, so $\mathcal{ML}$ is exponentially more succinct for separation than $\mathcal{ML}^{\diamond\wedge\neg}$. In fact, $\mathcal{ML}^{\diamond\wedge\neg}$ plus $\square$ is already exponentially more succinct for separation than $\mathcal{ML}^{\diamond\wedge\neg}$. Surprisingly χ_n does not use $\wedge$ or $\neg$. $\square$

Theorem 6.4 $\mathcal{ML}^{\diamond\wedge\neg}$ *is exponentially more succinct for separation than* $\mathcal{ML}^{\diamond\wedge}$.

Proof. Recall the proof of Theorem 4.5. Let $\mathcal{N}'_n = \langle W_n, R_n, V_n \rangle$. For a new propositional symbol r, define $\mathcal{N}''_n = \langle W_n, R_n, V_n[r \to \{b_n\}] \rangle$. It is easy to verify that the proof of Theorem 4.5 goes through with $\mathcal{N}''_n$ instead of $\mathcal{N}'_n$. Now elements a_n and b_n can be separated in $\mathcal{N}''_n$ by the constant-size formula $\theta = \neg r$ of $\mathcal{ML}^{\diamond\wedge\neg}$. Then $\mathcal{ML}^{\diamond\wedge\neg}$ is exponentially more succinct for separation than $\mathcal{ML}^{\diamond\wedge}$. $\square$

We close this section with a short digression. For a fixed and suitable model of computation M, the Kolmogorov complexity of a string σ relative to M, denoted $K_M(\sigma)$, is defined as the length of the shortest program which computes σ in M, or ∞ if there is no such program. (For more details on Kolmogorov Complexity Theory, see [13].) This underlying model of computation M may range from finite automata to Turing machines relativized to oracles. Here the *meaning* of a program is seen as the output it produces in the fixed model of computation M. Hence programs are seen as descriptors of strings, and the Kolmogorov complexity of a given string σ is just the length of the shortest description of σ within M. Informally speaking, stronger models of computation yield smaller Kolmogorov complexity. That is, if M is more powerful than M' then $K_M \leq K_{M'}$ (up to additive constant).

The notion of separation complexity S given in Definition 6.1 has some similarities with the classical Kolmogorov Complexity K. First, S needs some underlying language $\mathcal{L}$ and a suitable model $\mathcal{H}$. Second, in the context of logic, the *meaning* of a formula φ is given by its extension, that is by the set of points of $\mathcal{H}$ where φ is true. Hence formulas are seen as descriptors of elements of $\mathcal{H}$. As with K, if $\mathcal{L}$ is more expressive than $\mathcal{L}'$, in the sense of $\mathcal{L} \geq \mathcal{L}'$, then $S_{\mathcal{L}}$ is 'smaller' than $S_{\mathcal{L}'}$ in the sense of equation (1). But unlike programs which are simply executed in M to produce some output, for formulas evaluated in a fixed model $\mathcal{H}$ we may conceive different 'semantic tasks': here separation was analyzed, but one can conceive many others as well.

It is not the purpose of this article to study the resemblance of the *algorithmic* Kolmogorov Complexity with other *logical* description complexities. Although a fine analysis is needed, we want to point out that some results from the algorithmic side and the logical side may be somehow harmonized in a natural way.

7 Conclusions and future work

The line of research that motivated this work comes from the study of the computational complexity of the description problem for modal languages. We seek for efficient algorithms to compute modal descriptions, for various languages –including sub-boolean

ones. Is it true that the problem of finding an $\mathcal{L}$-description for a given element is computable in polynomial time? The answer depends in the way the output is represented. If one allows the output formula to be representable as a DAG then the answer is 'yes' [2]. But if we stick to the standard complexity computational model of Turing machines where 'compute a formula' means, literally, to write it down in the output tape then the answer is 'no': we have shown that the length of the output formula may be exponentially larger than the input model.

We have employed *classical* Ehrenfeucht-Fraïssé games as a theoretical tool for proving lower bounds on formula size. In this respect, our work is close to Adler and Immerman's [1], who propose a new kind of Ehrenfeucht-Fraïssé game to establish lower bounds for various kinds of logics. In their game, Spoiler can be seen as trying to construct what we have called a winning uniform strategy tree while Duplicator tries to identify deficiencies in it. The fact that Duplicator possesses an optimal strategy in this kind of games suggests, in our opinion, that the problem does not require a dynamic view in terms of games, but can be analyzed using the static notion of strategy over standard games.

We have only analyzed a few modal fragments, but the problem of the size of $\mathcal{L}$-descriptions is of course applicable to other logics. One can, for instance, study this problem for First Order Logic or Propositional Logic. These are two extremes, since $\mathcal{P} \leq \mathcal{ML}^{\Diamond\wedge} \leq \mathcal{ML}^{\Diamond\wedge\neg} \leq \mathcal{ML} \leq \mathcal{FO}^=$.

Consider $\mathcal{FO}^=$, the first-order logic with equality (over the modal correspondence language). It is well-known that one can characterize up-to-isomorphism any finite model $\mathcal{H}$ with domain $\{a_1 \ldots a_n\}$ using a sentence $\varphi \in \mathcal{FO}^=$ that is polynomial in the size of $\mathcal{H}$. Taking this as a basis, one can define for each a_i a formula of $\varphi_i(x) \in \mathcal{FO}^=$, with one free variable x, polynomial in $\mathcal{H}$ (that is there is a polynomial such that for all such $\mathcal{H}$, $|\varphi_i(x)| \leq p(|\mathcal{H}|)$), such that if a_i is $\mathcal{FO}^=$-describable then φ_i is a suitable description. In fact, this polynomial formula can be constructed in polynomial time.

Proposition 7.1 *The size of the $\mathcal{FO}^=$-description problem is polynomially bounded.*

We now go the the other extreme and regard Propositional Logic $\mathcal{P}$ as a fragment of $\mathcal{ML}$. For any finite Kripke model $\mathcal{H} = \langle W, R, V \rangle$ with $W = \{a_1, \ldots, a_n\}$ and Dom $V = \{p_1, \ldots, p_m\}$, we define, $\psi_k := \bigwedge_{a_k \in V(p_j)} p_j \wedge \bigwedge_{a_k \notin V(p_j)} \neg p_j$. Now if a_k is $\mathcal{P}$-describable in $\mathcal{H}$ then ψ_k is one such $\mathcal{P}$-description.

Proposition 7.2 *The size of the $\mathcal{P}$-description problem is polynomially bounded.*

Propositions 7.1 and 7.2 are clearly opposed to Corollaries 4.2, 4.4 and 4.6. While the modal fragments studied in this article ($\mathcal{ML}$, $\mathcal{ML}^{\Diamond\wedge\neg}$ and $\mathcal{ML}^{\Diamond\wedge}$) do not have polynomially bounded descriptions problems two extreme logics in terms of expressivity do.

It is interesting to study the size of the description problem for other fragments not addressed in this article such that $\mathcal{ML}^{\Diamond\wedge\neg}$ plus $\Box$ but without $\vee$, or others with restrictions in the shape of nestings of $\wedge$ and $\vee$. Even for the logics considered here, it would be interesting to have a better understanding of the computational complexity of their description and separation problems. In particular, one would like to close the gap

between lower and upper bounds and determine the complexity of finding a minimum description or separation.

References

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

[2] Areces, C., S. Figueira and D. Gorín, *The question of expressiveness in the generation of referring expressions*, Technical report, University of Buenos Aires (2010).
URL http://arxiv.org/abs/1006.4621v1

[3] Areces, C., A. Koller and K. Striegnitz, *Referring expressions as formulas of description logic*, in: *Proc. of the 5th INLG*, Salt Fork, OH, USA, 2008, pp. 42–49.

[4] Blackburn, P., M. de Rijke and Y. Venema, "Modal Logic," Cambridge University Press, 2001.

[5] Blackburn, P., J. van Benthem and F. Wolter, "Handbook of Modal Logic, Volume 3 (Studies in Logic and Practical Reasoning)," Elsevier Science Inc., New York, NY, USA, 2006.

[6] Dale, R., *Cooking up referring expressions*, in: *Proc. of the 27th ACL*, 1989, pp. 68–75.

[7] Dale, R. and N. Haddock, *Generating referring expressions involving relations*, in: *Proc. of the 5th EACL*, 1991, pp. 161–166.

[8] Dale, R. and E. Reiter, *Computational interpretations of the Gricean maxims in the generation of referring expressions*, Cognitive Science **19** (1995).

[9] Dawar, A., M. Grohe, S. Kreutzer and N. Schweikardt, *Model theory makes formulas large*, in: *In Proceedings of the 34th International Colloquium on Automata, Languages and Programming*, 2007.

[10] Hennessy, M. and R. Milner, *Algebraic laws for indeterminism and concurrency*, Journal of the ACM **32** (1985), pp. 137–162.

[11] Hopcroft, J., "An $nlog(n)$ algorithm for minimizing states in a finite automaton," In Z. Kohave, editor, *Theory of Machines and Computations*, Academic Press, 1971.

[12] Kanellakis, P. and S. Smolka, *CCS expressions finite state processes, and three problems of equivalence*, Inf. Comput. **86** (1990), pp. 43–68.

[13] Li, M. and P. Vitányi, "An introduction to Kolmogorov complexity and its applications," Springer, 1997, 2nd edition.

[14] Markey, N., *Temporal logic with past is exponentially more succinct*, in: *Bulletin of the EATCS 79*, 2003, pp. 122–128.

[15] Paige, R. and R. Tarjan, *Three partition refinement algorithms*, SIAM J. Comput. **16** (1987), pp. 973–989.

[16] Stone, M., *On identifying sets*, in: *Proc. of the 1st INLG*, 2000, pp. 116–123.

[17] van Deemter, K., *Generating referring expressions: Boolean extensions of the incremental algorithm*, Computational Linguistics **28** (2002), pp. 37–52.

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

A   Proof of Theorem 4.3

The definition of $(\mathcal{N}_n, a_n, b_n)_{n \in \mathbb{N}}$ is given in Figure 3 (Section 4). Notice that now the models interpret a propositional variable p. It is not hard to see that for all n, $\mathcal{N}_n$ is acyclic and $|\mathcal{N}_n| \in O(n)$.

We proceed as in the proof of Theorem 4.1, and show by induction on n that the minimum uniform strategy tree winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a_n\}, \{g_n\})$ has size, in this case, $2^n 3 - 2$, which is the closed form of $|s_0| = 1; |s_{n+1}| = 2|s_n| + 2$.

For $n = 0$, we take $s_0 = \langle a_0^1 \rangle$ and it is clearly a minimum uniform strategy tree winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a_0\}, \{b_0\})$. Assume now that s_n is the minimum uniform strategy tree winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a_n\}, \{b_n\})$. We perform again a case analysis of s_n ruling out possibilities, but recall that uniform strategy trees for $\mathcal{G}^{\diamond \wedge \neg}$ comprise only nodes of type $\langle \cdot \rangle$, $(\wedge)$, (p) or $(\overline{p})$. The reader can track the name of the nodes and the general shape of the strategy in Figure A.1. Again, we avoid subscript $n + 1$ and write, for instance a^1 for a_{n+1}^1. We use the convention for nodes of type $\langle \cdot \rangle$ guaranteed by Proposition 3.9. We will write $\langle a \rangle$ for $\langle \{a\} \rangle$.

Since there are no propositional symbols true at a or b, the root of s_n is not of type (p) nor $(\overline{p})$. By Proposition 3.11, the only possibility is then $s_n = \langle a^1 \rangle$; so let x be the child of s, that is $\langle a^1 \rangle \to x$, where x is winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^1\}, \{b^1, b^2\})$. Again, x is not of type (p) nor $(\overline{p})$. Suppose then $x = \langle E \rangle \to y$, for a non-empty $E \subseteq \{a^2, a^3\}$ and a y winning for $\mathcal{G}^{\diamond \wedge \neg}(E, \{b^3, b^4, b^5, b^6\})$. This would imply that y is winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^2\}, \{b^5\})$ or for $\mathcal{G}^{\diamond \wedge \neg}(\{a^3\}, \{b^4\})$, which is absurd, so this possibility is discarded.

We conclude that x is of type $(\wedge)$. Again, we can ignore without loss of generality the case $x \overset{\{b^1, b^2\}}{\to} y$, and assume that x has two children y_1 and y_2, such that $x \overset{\{b^1\}}{\to} y_1$ and $x \overset{\{b^2\}}{\to} y_2$. Furthermore, y_1 is winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^1\}, \{b^1\})$ and y_2 is winning for $\mathcal{G}^{\diamond \wedge \neg}\{a^1\}, \{b^2\})$.

Again, we observe that y_1 and y_2 can only be of type $\langle \cdot \rangle$. Suppose $y_1 = \langle E_1 \rangle \to z_1$, for a non-empty $E_1 \subseteq \{a^2, a^3\}$ and a z_1 that is winning for $\mathcal{G}^{\diamond \wedge \neg}(E_1, \{b^3, b^4\})$. Since clearly there cannot be a uniform strategy tree winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^3\}, \{b^4\})$, we have $a^3 \notin E_1$ and, then $E_1 = \{a^2\}$ and z_1 is winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^2\}, \{b^3, b^4\})$. In a similar way, we conclude that $y_2 = \langle a^3 \rangle \to z_2$ with z_2 winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^3\}, \{b^5, b^6\})$.

Since p is true in b^3 and false in b^4, z_1 is not of type (p) nor $(\overline{p})$. The same can be said about z_2. Suppose $z_1 = \langle F \rangle \to h_1$; then necessarily $F = \{a_n\}$ and h_1 has to be winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a_n\}, \{a_n, b_n\})$ contradicting Proposition 3.10. Similarly, z_2 cannot be of type $\langle \cdot \rangle$ either.

Therefore, z_1 and z_2 must be of type $(\wedge)$ and, once again, we can assume without loss of generality that they have both two children each. Suppose $z_1 \overset{\{b^3\}}{\to} h_1$, $z_1 \overset{\{b^4\}}{\to} h_2$, $z_2 \overset{\{b^5\}}{\to} h_3$ and $z_2 \overset{\{b^6\}}{\to} h_4$ where h_1 is winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^2\}, \{b^3\})$, h_2 is winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^2\}, \{b^4\})$, h_3 is winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^3\}, \{b^5\})$ and h_4 is winning for $\mathcal{G}^{\diamond \wedge \neg}(\{a^3\}, \{b^6\})$. Furthermore, we assume all such strategies are minimum, so h_2 and h_3 are necessarily of type (p) and $(\overline{p})$ respectively. Since no propositional variable distinguishes a^2 from a^3, h_1 is not of type (p) nor $(\overline{p})$. And by Proposition 3.11 h_1 is not of type $(\wedge)$ either. The same can be said about h_4.
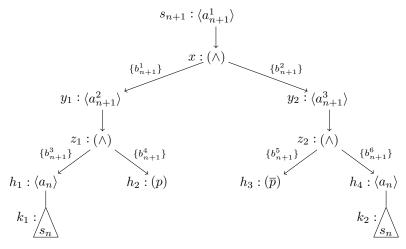
Fig. A.1. s_{n+1}, minimum uniform strategy tree winning for $\mathcal{G}_{\mathcal{N}_{n+1}}^{\Diamond\wedge\neg}(\{a_{n+1}\},\{b_{n+1}\})$.

Hence $h_1 = \langle a_n \rangle \to k_1$ and $h_4 = \langle a_n \rangle \to k_2$, where k_1 and k_2 are minimum strategies winning for $\mathcal{G}^{\Diamond\wedge\neg}(\{a_n\},\{b_n\})$, that is $k_1 = k_2 = s_n$. Therefore, we have that $|s_{n+1}| = 2|s_n| + 2$, so these uniform strategy trees cannot be polynomially bounded. As in the proof of Theorem 4.1, it is easy to see that the associated formulas $\psi_0 := \Diamond\top$ and $\psi_{n+1} := \Diamond(\Diamond(p \wedge \Diamond\psi_n) \wedge \Diamond(\neg p \wedge \Diamond\psi_n))$ describe a_n in $\mathcal{N}_n$.