Succinctness of Cosafety Fragments of LTL via Combinatorial Proof Systems

Luca Geatti[1(✉)], Alessio Mansutti[2(✉)], and Angelo Montanari[1(✉)]

[1] University of Udine, Udine, Italy
{luca.geatti,angelo.montanari}@uniud.it
[2] IMDEA Software Institute, Madrid, Spain
alessio.mansutti@imdea.org

Abstract. This paper focuses on succinctness results for fragments of Linear Temporal Logic with Past (LTL) devoid of binary temporal operators like *until*, and provides methods to establish them. We prove that there is a family of *cosafety* languages $(\mathcal{L}_n)_{n\geq 1}$ such that $\mathcal{L}_n$ can be expressed with a *pure future formula* of size $\mathcal{O}(n)$, but it requires formulae of size $2^{\Omega(n)}$ to be captured with *past formulae*. As a by-product, such a succinctness result shows the optimality of the *pastification algorithm* proposed in *[Artale et al., KR, 2023]*.

We show that, in the considered case, succinctness cannot be proven by relying on the classical automata-based method introduced in *[Markey, Bull. EATCS, 2003]*. In place of this method, we devise and apply a *combinatorial proof system* whose deduction trees represent LTL formulae. The system can be seen as a proof-centric (one-player) view on the games used by Adler and Immerman to study the succinctness of CTL.

Keywords: Temporal logics · LTL · Succinctness · Proof systems.

1 Introduction

Linear Temporal Logic with Past (LTL [17,23]) is the *de-facto* standard language for the specification, verification, and synthesis of reactive systems [19]. Concerning these reasoning tasks, two fundamental subsets of LTL-definable languages come into play, namely, *safety* and *cosafety* languages. Safety languages express properties stating that "something bad never happens"; cosafety languages, instead, express the fact that "something good will eventually happen". The crucial feature of cosafety (resp., safety) languages is that checking a *finite prefix* of an infinite trace suffices to establish whether the entire trace belongs (resp., does not belong) to the language. Such an ability of reducing reasoning over infinite words to the finite case plays a fundamental role in lowering the complexity of reasoning tasks [16]. Because of this, while LTL was commonly interpreted over infinite traces, recent work mainly considers its finite trace semantics [8,18,22].

In what follows, given a set of temporal operators S, we write $\mathsf{LTL}[S]$ for the set of all LTL formulae in *negation normal form* whose temporal operators are restricted to those in S. Similarly, we denote with $\mathsf{F}(\mathsf{LTL}[S])$ the set of formulae of the form $\mathsf{F}(\alpha)$, with $\alpha \in \mathsf{LTL}[S]$. Here, F is the *future* modality (a.k.a. *eventually*).

N. Kobayashi and J. Worrell (Eds.): FoSSaCS 2024, LNCS 14575, pp. 95–115, 2024.
https://doi.org/10.1007/978-3-031-57231-9_5

There are two notable syntactic characterizations of the cosafety languages of LTL. The first one is a *pure future* characterization given by the logic $\mathsf{LTL}[\mathsf{X},\mathsf{U}]$ featuring modalities *next* X and *until* U. The second one is an *eventually pure past*[3] characterisation given by the logic $\mathsf{F}(\mathsf{pLTL})$, where pLTL is the *pure past* fragment of LTL, that is, the restriction of LTL to past modalities. Analogous characterizations have been provided for safety languages.

As for applications, $\mathsf{F}(\mathsf{pLTL})$ is considered to be much more convenient than $\mathsf{LTL}[\mathsf{X},\mathsf{U}]$, because, starting from an (eventually) pure past formula of size n, it is possible to build an equivalent deterministic finite automaton of singly exponential size in n [7]. In the case of $\mathsf{LTL}[\mathsf{X},\mathsf{U}]$, such an automaton may have size doubly exponential in n [16]. This computational advantage of pure past formulae originated a recent line of research that focuses on the *pastification problem*, i.e., the problem of translating an input *pure future* formula for a cosafety (or safety) language into an equivalent *pure past* (equivalently, *eventually pure past*) formula. While the best known algorithm for $\mathsf{LTL}[\mathsf{X},\mathsf{U}]$ is triply exponential [7], a singly exponential pastification algorithm to transform $\mathsf{LTL}[\mathsf{X},\mathsf{F}]$ formulae into $\mathsf{F}(\mathsf{LTL}[\mathsf{Y},\widetilde{\mathsf{Y}},\mathsf{O}])$ ones has been recently developed in [4]. Here, modalities *yesterday* Y and *once* O are the "temporal reverses" of modalities X and F, respectively, whereas the *weak yesterday* operator $\widetilde{\mathsf{Y}}$ is the dual of Y (we formally define the semantics of all these modalities in Section 2). No super-polynomial lower bounds for these pastification problems are known.

While the above two characterisations of cosafety languages have been thoroughly studied in the last decades in terms of expressiveness [6] and complexity [2], their *succinctness* is still poorly understood. To the best of our knowledge, the only known result is the one in [3] showing that $\mathsf{F}(\mathsf{pLTL})$ *can* be exponentially more succinct than $\mathsf{LTL}[\mathsf{X},\mathsf{U}]$ — note that lower bounds to pastification problems require the opposite direction.[4]

In this paper, we study the succinctness of $\mathsf{LTL}[\mathsf{F}]$ against $\mathsf{F}(\mathsf{LTL}[\mathsf{Y},\widetilde{\mathsf{Y}},\mathsf{O},\mathsf{H}])$, where H is the dual of O, as well as the succinctness of their *reverse logics* [3], that is, the succinctness of $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ against $\mathsf{LTL}[\mathsf{X},\widetilde{\mathsf{X}},\mathsf{F},\mathsf{G}]$. For these fragments of LTL, we establish the following two results.

Theorem 1. $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ *can be exponentially more succinct than* $\mathsf{LTL}[\mathsf{X},\widetilde{\mathsf{X}},\mathsf{F},\mathsf{G}]$.

Theorem 2. $\mathsf{LTL}[\mathsf{F}]$ *can be exponentially more succinct than* $\mathsf{F}(\mathsf{LTL}[\mathsf{Y},\widetilde{\mathsf{Y}},\mathsf{O},\mathsf{H}])$.

The two theorems prove an *incomparability result* about the succinctness of the characterizations of cosafety languages in the pure future and eventually pure past fragments of LTL. Theorem 1 and Theorem 2 hold for both the finite and infinite trace semantics of LTL (however, due to lack of space, we report the proof of Theorem 1 only in the case of finite traces). As a corollary, Theorem 2 implies that the pastification algorithm proposed in [4] is optimal.

[3] "Eventually pure past" refers to formulae of the form $\mathsf{F}(\alpha)$, with α pure past formula.

[4] A logic $\mathbb{L}$ *can be exponentially more succinct* than a logic $\mathbb{L}'$ whenever there is a family of languages $(\mathcal{L}_n)_{n\geq 1}$ such that $\mathcal{L}_n$ can be expressed in $\mathbb{L}$ with a formula of size polynomial in n, whereas expressing $\mathcal{L}_n$ in $\mathbb{L}'$ requires formulae of size $2^{\Omega(n)}$.

Corollary 1. *The pastification of* $\mathsf{LTL}[\mathsf{X},\mathsf{F}]$ *into* $\mathsf{F}(\mathsf{LTL}[\mathsf{Y},\widetilde{\mathsf{Y}},\mathsf{O},\mathsf{H}])$ *is in* $2^{\Theta(n)}$.

To prove Theorem 1, we devise and apply a *combinatorial proof system.*[5] Given two sets of finite traces A and B, with the proof system one can establish whether there is a formula φ in $\mathsf{LTL}[\mathsf{X},\widetilde{\mathsf{X}},\mathsf{F},\mathsf{G}]$ that *separates* A from B, that is, φ is satisfied by all traces in A (written $A \models \varphi$) and violated by all traces in B (written $B \perp\!\!\!\perp \varphi$). A proof obtained by applying k rules of the proof system corresponds to the existence of one such separating formula φ of size k.

The proposed combinatorial proof system can be seen as a reformulation in terms of proofs of the games introduced by Adler and Immerman to show that CTL^+ is $\Theta(n)!$ more succinct than CTL [1]. They are two-player games that extend Ehrenfeucht–Fraïssé games for quantifier depth in a way that captures the notion of formula size instead. However, unlike Ehrenfeucht–Fraïssé ones, in Adler–Immerman games one of the two players (the duplicator) has always a trivial strategy. With our proof system, we show that removing the duplicator from the game yields a natural one-player game based on building proofs.

To prove Theorem 1 by applying the proposed proof system, we provide, for every $n \geq 1$, a formula Φ_n in $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ of size linear in n and two sets of traces $\mathbf{A}_n$ and $\mathbf{B}_n$ such that $\mathbf{A}_n \models \Phi_n$ and $\mathbf{B}_n \perp\!\!\!\perp \Phi_n$, and then we show that the smallest deduction tree that separates $\mathbf{A}_n$ from $\mathbf{B}_n$ has size at least 2^n. This implies that all formulas of $\mathsf{LTL}[\mathsf{X},\widetilde{\mathsf{X}},\mathsf{F},\mathsf{G}]$ capturing Φ_n are of size at least 2^n.

Once Theorem 1 is established, one can prove Theorem 2 by "reversing" the direction of time, building correspondences between formulae of $\mathsf{LTL}[\mathsf{F}]$ and $\mathsf{FLTL}[\mathsf{O}]$, and between formulae of $\mathsf{F}(\mathsf{LTL}[\mathsf{Y},\widetilde{\mathsf{Y}},\mathsf{O},\mathsf{H}])$ and $\mathsf{LTL}[\mathsf{X},\widetilde{\mathsf{X}},\mathsf{F},\mathsf{G}]$.

In the context of LTL, the main technique to prove "future against past" succinctness discrepancies is arguably the automata method introduced by Markey in [20]. At its core, such a method exploits the fact that pure future formulae of LTL can be translated into nondeterministic Büchi automata of exponential size, and thus no property requiring a doubly exponential size automaton can be represented succinctly. The introduction of our proof system raises the question of whether Markey's method can be applied to establish our succinctness results. We prove that it cannot be used in our context. In order to obtain such a result, the key observation is that, given a cosafety formula $\mathsf{F}\psi$, a *deterministic* Büchi automaton (DBA) for $\mathsf{F}\psi$ of size ℓ, and a prefix Π consisting of k temporal operators among X, F, and G, the minimal DBA for the formula $\Pi\mathsf{F}\psi$ has size polynomial in k and ℓ.

Synopsis. Section 2 introduces the necessary background. Section 3 discusses the languages we use to prove Theorem 1. Section 4 introduces the combinatorial proof system. In Section 5 we prove Theorem 1. In Section 6 we prove Theorem 2 and Corollary 1. The limits of the automata-based method to prove succinctness lower bounds are discussed in Section 7. Related and future work are discussed in Section 8. An extended version of the paper, complete of all proofs, can be found in [13].

[5] We use the term "combinatorial" for our proof system to conform with the terminology from the Workshop "Combinatorial Games in Finite Model Theory", LICS'23.

2 Preliminaries

In this section, we introduce background knowledge on LTL focusing on finite traces. All definitions admit a natural extension to the setting of infinite traces.

Let Σ be a finite alphabet. We denote by Σ^* the set of all finite words over Σ and by Σ^+ the subset of finite non-empty words. We use the term *trace* as a synonym of word. A *language* $\mathcal{L}$ over Σ is a subset of Σ^*. Let $\sigma = \langle w_0, w_1, \ldots, w_n \rangle$ be a word in Σ^*. We denote by $|\sigma|$ the *length* of σ, that is, $n+1$. A *position* in σ is an element in the set $\text{pos}(\sigma) := [0, n]$. For every $i \in \text{pos}(\sigma)$, we denote by $\sigma[i] \in \Sigma$ the letter w_i, and by $\sigma[i\rangle$ the word $\langle w_i, \ldots, w_n \rangle$. We say that position j of σ has *type* $\tau \in \Sigma$ whenever $\sigma[j] = \tau$. Given two traces σ_1 and σ_2, we write $\sigma_1 \sqsubseteq \sigma_2$ whenever σ_1 is a *suffix* of σ_2, that is, there is $j \in \text{pos}(\sigma_2)$ such that $\sigma_1 = \sigma_2[j\rangle$. Given a word $\sigma' \in \Sigma^*$, we denote the *concatenation of* σ' *to* σ as $\sigma \cdot \sigma'$, or simply $\sigma\sigma'$. Given two languages $\mathcal{L}$ and $\mathcal{L}'$, we define $\mathcal{L} \cdot \mathcal{L}' := \{\sigma \cdot \sigma' \mid \sigma \in \mathcal{L}, \sigma' \in \mathcal{L}'\}$. We sometimes apply the concatenation to a word and a language; in these cases the word is implicitly converted into a singleton language, e.g., $\sigma \cdot \mathcal{L} := \{\sigma\} \cdot \mathcal{L}$. With $A \subseteq_{\mathit{fin}} B$ we denote the fact that A is a *finite* subset of the set B.

Linear Temporal Logic with Past. In the following, we introduce syntax and semantics of Linear Temporal Logic with Past (LTL) restricted to those operators that we are going to use throughout the paper. In particular, we omit the future operators *until* and *release*, and their past counterparts (*since* and *triggers*). Let $\mathcal{AP}$ be a finite set of atomic propositions. The syntax of the formulae over $\mathcal{AP}$ is generated by the following grammar:

$$\begin{array}{rlr} \varphi := & p \mid \neg p \mid \varphi \vee \varphi \mid \varphi \wedge \varphi & \text{Boolean connectives} \\ & \mid \mathsf{X}\varphi \mid \widetilde{\mathsf{X}}\varphi \mid \mathsf{F}\varphi \mid \mathsf{G}\varphi & \text{future operators} \\ & \mid \mathsf{Y}\varphi \mid \widetilde{\mathsf{Y}}\varphi \mid \mathsf{O}\varphi \mid \mathsf{H}\varphi & \text{past operators} \end{array}$$

where $p \in \mathcal{AP}$. The temporal operators are respectively called: X, *next*; $\widetilde{\mathsf{X}}$, *weak next*; F, *future*; G, *globally*; Y, *yesterday*; $\widetilde{\mathsf{Y}}$, *weak yesterday*; O, *once*; H, *historically.* For the rest of the paper, we let $\mathbb{OP} := \{\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}, \mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}\}$.

For every formula φ, we define the *size of* φ, denoted by $\text{size}(\varphi)$, inductively defined as follows: (i) $\text{size}(p) := 1$ and $\text{size}(\neg p) := 1$, (ii) $\text{size}(\otimes\varphi) := \text{size}(\varphi) + 1$, for $\otimes \in \mathbb{OP}$, and (iii) $\text{size}(\varphi_1 \oplus \varphi_2) := \text{size}(\varphi_1) + \text{size}(\varphi_2) + 1$ for $\oplus \in \{\vee, \wedge\}$.

We focus on the interpretation of LTL formulae over *finite non-empty traces* over the alphabet $2^{\mathcal{AP}}$. From now on, we set the alphabet Σ to be $2^{\mathcal{AP}}$. Given a word $\sigma \in \Sigma^+$, the *satisfaction* of a formula φ by σ at time point / position $i \in \text{pos}(\sigma)$, denoted by $\sigma, i \models \varphi$, is defined as follows:

1. $\sigma, i \models p$ iff $p \in \sigma[i]$;
2. $\sigma, i \models \neg p$ iff $p \notin \sigma[i]$;
3. $\sigma, i \models \varphi_1 \vee \varphi_2$ iff $\sigma, i \models \varphi_1$ or $\sigma, i \models \varphi_2$;
4. $\sigma, i \models \varphi_1 \wedge \varphi_2$ iff $\sigma, i \models \varphi_1$ and $\sigma, i \models \varphi_2$;
5. $\sigma, i \models \mathsf{X}\varphi$ iff $i + 1 < |\sigma|$ and $\sigma, i + 1 \models \varphi$;
6. $\sigma, i \models \widetilde{\mathsf{X}}\varphi$ iff either $i + 1 = |\sigma|$ or $\sigma, i + 1 \models \varphi$;

7. $\sigma, i \models \mathsf{F}\varphi$ iff there exists $i \leq j < |\sigma|$ such that $\sigma, j \models \varphi$;
8. $\sigma, i \models \mathsf{G}\varphi$ iff for all $i \leq j < |\sigma|$, it holds $\sigma, j \models \varphi$;
9. $\sigma, i \models \mathsf{Y}\varphi$ iff $i > 0$ and $\sigma, i-1 \models \varphi$;
10. $\sigma, i \models \widetilde{\mathsf{Y}}\varphi$ iff either $i = 0$ or $\sigma, i-1 \models \varphi$;
11. $\sigma, i \models \mathsf{O}\varphi$ iff there exists $0 \leq j \leq i$ such that $\sigma, j \models \varphi$;
12. $\sigma, i \models \mathsf{H}\varphi$ iff for all $0 \leq j \leq i$, it holds $\sigma, j \models \varphi$.

For every formula φ, we say that a trace σ satisfies φ, written $\sigma \models \varphi$, if $\sigma, 0 \models \varphi$. The *language* of φ, denoted by $\mathcal{L}(\varphi)$, is the set of words $\sigma \in \Sigma^+$ such that $\sigma \models \varphi$. Given two formulae φ and ψ, we say that φ is *equivalent* to ψ, written $\varphi \equiv \psi$, whenever $\mathcal{L}(\varphi) = \mathcal{L}(\psi)$.

Fragments of LTL. Given a set of operators $S \subseteq \mathbb{OP}$, we denote by $\mathsf{LTL}[S]$ the set of formulae only using temporal operators from S. When dealing with a concrete S, we omit the curly brackets and write, e.g., $\mathsf{LTL}[\mathsf{X}, \mathsf{F}]$ instead of $\mathsf{LTL}[\{\mathsf{X}, \mathsf{F}\}]$. Whenever S contains only future operators (resp., past operators), the logic $\mathsf{LTL}[S]$ is called a *pure future* (resp., *pure past*) fragment of LTL. Finally, we denote by $\mathsf{F}(\mathsf{LTL}[S])$ (resp., $\mathsf{G}(\mathsf{LTL}[S])$) the set of formulae of the form $\mathsf{F}(\alpha)$ (resp., $\mathsf{G}(\alpha)$), where α is a formula of $\mathsf{LTL}[S]$. A language $\mathcal{L} \subseteq \Sigma^*$ is a *cosafety language* whenever $\mathcal{L} = K \cdot \Sigma^*$, for some $K \subseteq \Sigma^*$. A language $\mathcal{L}$ is a *safety language* whenever its complement $\overline{\mathcal{L}}$ is a cosafety language. For every formula φ in the fragments $\mathsf{LTL}[\mathsf{X}, \mathsf{F}]$ and $\mathsf{F}(\mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}])$, it holds that $\mathcal{L}(\varphi)$ is a cosafety language. Similarly, for every formula φ in the fragments $\mathsf{LTL}[\widetilde{\mathsf{X}}, \mathsf{G}]$ and $\mathsf{G}(\mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}])$, it holds that $\mathcal{L}(\varphi)$ is a safety language.

The pastification problem. Given two sets $S \subseteq \{\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}\}$ and $S' \subseteq \{\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}\}$, the *pastification problem for* $\mathsf{LTL}[S]$ *into* $\mathsf{F}(\mathsf{LTL}[S'])$ asks, given an input formula $\varphi \in \mathsf{LTL}[S]$, to return a formula ψ from $\mathsf{F}(\mathsf{LTL}[S'])$ such that $\varphi \equiv \psi$. An algorithm for the pastification problem is said to be of *k-exponential size* (for $k \in \mathbb{N}$ fixed) whenever the output formula ψ is such that $\text{size}(\psi) \in \exp_2^k(\text{poly}(\text{size}(\varphi)))$, where $\exp^k(.)$ is the k-th iteration of the base-2 tetration function given by $\exp^0(n) = n$ and $\exp^{i+1}(n) = 2^{\exp^i(n)}$. In [4], an exponential time, 1-exponential size, pastification algorithm for $\mathsf{LTL}[\mathsf{X}, \mathsf{F}]$ into $\mathsf{F}(\mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}])$ is presented.

Succinctness. Given two sets $S, S' \subseteq \mathbb{OP}$, we say that $\mathsf{LTL}[S]$ *can be exponentially more succinct than* $\mathsf{LTL}[S']$ if there is a family of languages $(\mathcal{L}_n)_{n \geq 1}$ such that, for every $n \geq 1$, $\mathcal{L}_n \subseteq \Sigma_n^+$, for some alphabet Σ_n, and:

- there is $\varphi \in \mathsf{LTL}[S]$ such that $\mathcal{L}(\varphi) = \mathcal{L}_n$ and $\text{size}(\varphi) \in \text{poly}(n)$, and
- for every $\psi \in \mathsf{LTL}[S']$, if $\mathcal{L}(\psi) = \mathcal{L}_n$ then $\text{size}(\psi) \in 2^{\Omega(n)}$.

It is worth noticing that the above-given syntax for LTL is already in *negation normal form*, that is, negation may only appear in front of atomic propositions. Allowing negations to occur freely in the formula neither increase expressiveness nor succinctness, as the grammar above is already closed under dual operators, e.g., $\mathsf{G}\varphi \equiv \neg\mathsf{F}\neg\varphi$, and the size of a formula does not depend on the number of negations occurring in literals. Because of this, all results given in the paper continue to hold when negation is added to the language.

3 A problematic cosafety language for $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$

We now describe the property that we will exploit to prove that $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ can be exponentially more succinct than $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$ (Theorem 1). More precisely, we define a family of $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ formulae $(\Phi_n)_{n \geq 1}$ such that, for every $n \geq 1$, Φ_n has size in $\mathcal{O}(n)$ and captures a property requiring a formula of size at least 2^n to be expressed in $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$ (as we will see in Section 5).

Let $n \geq 1$. We consider the alphabet of $2n + 2$ distinct atomic propositions $\mathcal{AP} := \{\widetilde{p}, \widetilde{q}\} \cup P \cup Q$, with $P := \{p_1, \dots, p_n\}$ and $Q := \{q_1, \dots, q_n\}$. For all $n \geq 1$, the formula Φ_n of $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ is defined as follows:

$$\Phi_n := \mathsf{F}\left(\widetilde{q} \wedge \bigwedge_{i=1}^{n} \left(\left(q_i \wedge \mathsf{O}(\widetilde{p} \wedge p_i)\right) \vee \left(\neg q_i \wedge \mathsf{O}(\widetilde{p} \wedge \neg p_i)\right)\right)\right).$$

Observe that, for every $n \geq 1$, $\mathrm{size}(\Phi_n)$ belongs to $\mathcal{O}(n)$. The formula Φ_n is satisfied by those traces $\sigma \in \Sigma^+$ where there is a position $j \in \mathrm{pos}(\sigma)$ such that (i) $\widetilde{q} \in \sigma[j]$ and (ii) for every $i \in [1, n]$ there is a position $k_i \in [0, j]$ such that $\widetilde{p} \in \sigma[k_i]$ and $q_i \in \sigma[j]$ *if and only if* $p_i \in \sigma[k_i]$. Notice that each $k_i \in [0, j]$ depends on an index $i \in [1, n]$. Therefore, for distinct $i, j \in [1, n]$ the positions k_i and k_j might differ. This feature is crucial to get a language which has a compact definition in $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$, but is hard to capture for $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$.

As a matter of fact, requiring the various k_i to coincide yields a formula Ψ_n characterising the property: "the trace σ has two positions $j \geq k$ such that $\widetilde{p} \in \sigma[k]$, $\widetilde{q} \in \sigma[j]$ and, for every $i \in [1, n]$, $q_i \in \sigma[j]$ if and only if $p_i \in \sigma[k]$". This formula is known to require exponential size in LTL [20], and therefore in $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ as well. In a sense, the asymmetry obtained by relaxing the uniqueness of the position k above is what makes Φ_n being easily expressible in $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$, but difficult to characterise in $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$. The same trick, applied to position j instead of position k, can be used to obtain a family of formulae that can be represented in an exponentially more succinct way in $\mathsf{LTL}[\mathsf{F}]$ than in $\mathsf{F}(\mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}])$. This form of "temporal duality" is what we will ultimately exploit in Section 6 to prove Theorem 2.

The following lemma shows that Φ_n can be expressed in $\mathsf{LTL}[\mathsf{F}]$ (and thus in $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$ as well) with a formula of exponential size.

Lemma 1. *For every $n \geq 1$, there is a formula Φ'_n in $\mathsf{LTL}[\mathsf{F}]$ such that $\Phi'_n \equiv \Phi_n$ and $\mathrm{size}(\Phi'_n) < 2^{n+1}(n+2)^2$.*

Proof sketch. Given $\tau \in 2^P$, we write $\overline{\tau}$ for the element of 2^Q such that $p_i \in \tau$ if and only if $q_i \in \overline{\tau}$, for every $i \in [1, n]$. Then, the formula Φ'_n is defined as follows:

$$\Phi'_n := \bigvee_{\tau \in 2^P} \left(\bigwedge_{p \in \tau} \mathsf{F}\big(\widetilde{p} \wedge p \wedge \mathsf{F}(\widetilde{q} \wedge \psi_{\overline{\tau}})\big) \wedge \bigwedge_{p \in P \setminus \tau} \mathsf{F}\big(\widetilde{p} \wedge \neg p \wedge \mathsf{F}(\widetilde{q} \wedge \psi_{\overline{\tau}})\big)\right),$$

where $\psi_{\overline{\tau}} := (\bigwedge_{q \in \overline{\tau}} q \wedge \bigwedge_{q \in Q \setminus \overline{\tau}} \neg q)$. □

4 A combinatorial proof system for $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$

In this section, we introduce the proof system that we will later employ to prove Theorem 1, and discuss its connection with Adler–Immerman games [1].

Further notation. Let $A \subseteq \Sigma^+$, with $\Sigma := 2^{\mathcal{AP}}$ for some set of propositions $\mathcal{AP}$. We define $A^{\mathsf{X}} := \{\sigma[1\rangle : \sigma \in A \text{ s.t. } |\sigma| \geq 2\}$, i.e., the set of non-empty traces obtained from A by stepping each trace one position to the right. We define $A^{\mathsf{G}} := \{\sigma[j\rangle : \sigma \in A \text{ and } j \in \text{pos}(\sigma)\}$, i.e., the set of all suffixes of the traces in A. We say that a map $f : A \to \mathbb{N}$ is a *future point* for A whenever $f(\sigma) \in \text{pos}(\sigma)$ for every $\sigma \in A$. We write F_A for the set of all maps that are future points for A. Given a future point f for A and $\sigma \in A$ with $f(\sigma) = i$, we define $\sigma^f := \sigma[i\rangle$ and $A^f := \{\sigma^f : \sigma \in A\}$. Note that, by definition, $A^{\mathsf{G}} = \bigcup_{f \in \mathrm{F}_A} A^f$.

For a formula φ of LTL, we write $A \models \varphi$ whenever $(\sigma, 0) \models \varphi$ for every $\sigma \in A$, and $A \perp\!\!\!\perp \varphi$ whenever $(\sigma, 0) \not\models \varphi$ for every $\sigma \in A$. Given two sets of traces $A, B \subseteq \Sigma^+$ we say that φ *separates* A from B whenever $A \models \varphi$ and $B \perp\!\!\!\perp \varphi$. We write $\langle \cdot, \cdot \rangle_S \subseteq \Sigma^+ \times \Sigma^+$ for the *separable relation on* $S \subseteq \mathbb{OP}$, i.e., the binary relation holding on pairs (A, B) whenever there is some formula from $\mathsf{LTL}[S]$ that separates A from B. Note that, when A and B are finite sets and $\mathsf{X} \in S$, deciding whether $\langle A, B \rangle_S$ holds is trivial.

Lemma 2. *Let $A, B \subseteq \Sigma^+$ and $S \subseteq \mathbb{OP}$. Then, $\langle A, B \rangle_S$ implies $A \cap B = \varnothing$. Moreover, if A and B are finite sets and $\mathsf{X} \in S$, $A \cap B = \varnothing$ implies $\langle A, B \rangle_S$.*

Proof sketch. For the first statement, clearly if $A \cap B \neq \varnothing$ then it is not possible to separate A from B. To prove the second statement, one defines a disjunction φ of formulae, each characterising an element in A. For instance, for $\mathcal{AP} = \{p, q\}$, the trace $\{p\}\{q\}$ can be characterised with the formula $(p \wedge \neg q) \wedge \mathsf{X}(q \wedge \neg p \wedge \widetilde{\mathsf{X}}\bot)$, where $\bot := p \wedge \neg p$. Then, φ separates A from B. □

We mainly consider the relation $\langle \cdot, \cdot \rangle_S$ with S being the set $\{\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}\}$, and thus from now on simply write $\langle \cdot, \cdot \rangle$ when considering this concrete choice of S.

4.1 The proof system

The combinatorial proof system that we define is a natural-deduction-style proof system. It is made of several inference rules of the form $\frac{H_1 \; H_2 \; \dots \; H_n}{C}$, to be read as "if the hypotheses $H_1, \dots, H_n$ hold, then the consequence C holds". As usual, proofs within the proof system have a tree-like presentation. An example of such a *deduction tree* is given in Figure 2, where $a := \{p\}$ and $b := \varnothing$, with $p \in \mathcal{AP}$. This is a deduction tree for the *term* $\langle \{abaa, aaaa\}, \{aaab\} \rangle$, which we call the *root* of the deduction tree. In Figure 2, to the root it is *applied* the rule Or, with hypotheses $\langle \{abaa\}, \{aaab\} \rangle$ and $\langle \{aaaa\}, \{aaab\} \rangle$. In turn, these two hypotheses are derived in the deduction tree by eventually reaching applications to the rule Atomic. A deduction tree is always *closed*: all maximal paths from the root ends with an application of the rule Atomic. This means that a rule of the proof system must be applied to each term $\langle A, B \rangle$ appearing in the tree. We call a tree a *partial deduction tree* if this property is not enforced, namely when there might be unproven terms $\langle A, B \rangle$. The *size* of a deduction tree is the number of rules in it. For instance, the tree in Figure 2 has size 5.

We define the inference rules of the proof system in Figure 1. Let us briefly describe these rules. The Atomic rule allows deriving $\langle A, B \rangle$ if every trace in A

$$\textsc{Atomic}\ \frac{A \models \alpha \quad B \perp\!\!\!\perp \alpha}{\langle A, B\rangle}\ \alpha \text{ literal} \qquad \textsc{Or}\ \frac{\langle A_1, B\rangle \quad \langle A_2, B\rangle}{\langle A_1 \uplus A_2, B\rangle} \qquad \textsc{And}\ \frac{\langle A, B_1\rangle \quad \langle A, B_2\rangle}{\langle A, B_1 \uplus B_2\rangle}$$

$$\textsc{Next}\ \frac{\langle A^{\mathsf{X}}, B^{\mathsf{X}}\rangle \quad A \subseteq \Sigma \cdot \Sigma^{+}}{\langle A, B\rangle} \qquad \textsc{WeakNext}\ \frac{\langle A^{\mathsf{X}}, B^{\mathsf{X}}\rangle \quad B \subseteq \Sigma \cdot \Sigma^{+}}{\langle A, B\rangle}$$

$$\textsc{Future}\ \frac{\langle A^{f}, B^{\mathsf{G}}\rangle}{\langle A, B\rangle}\ f \in \mathrm{F}_A \qquad \textsc{Globally}\ \frac{\langle A^{\mathsf{G}}, B^{f}\rangle}{\langle A, B\rangle}\ f \in \mathrm{F}_B$$

Fig. 1. The combinatorial proof system. Here, $A, B \subseteq \Sigma^{+}$.

$$\textsc{Or}\ \dfrac{\textsc{Next}\ \dfrac{\textsc{Atomic}\ \dfrac{\{baa\} \models \neg p \quad \{aab\} \perp\!\!\!\perp \neg p}{\langle \{baa\}, \{aab\}\rangle}}{\langle \{abaa\}, \{aaab\}\rangle} \qquad \dfrac{\dfrac{\{aaaa, aaa, aa, a\} \models p \quad \{b\} \perp\!\!\!\perp p}{\langle \{aaaa, aaa, aa, a\}, \{b\}\rangle}\ \textsc{Atomic}}{\langle \{aaaa\}, \{aaab\}\rangle}\ \textsc{Globally}}{\langle \{abaa, aaaa\}, \{aaab\}\rangle}$$

Fig. 2. A deduction tree proving $\langle \{abaa, aaaa\}, \{aaab\}\rangle$. Here, $a := \{p\}$ and $b := \varnothing$.

satisfies some literal α and every trace in B violates α. The Or rule corresponds the case of A being separable from B via a formula of the form $\varphi_1 \vee \varphi_2$. In this and the rule And, $\uplus$ stands for the union of disjoint sets. Intuitively, Or can be applied by proving that φ_1 separates a set $A_1 \subseteq A$ from B *and* that φ_2 separates the set $A \setminus A_1$ from B. The Next rule allows separating A from B with a formula of the form $\mathsf{X}\varphi$, by checking whether the sets obtained by stepping all traces in A and B to next time point are separable by φ. The condition $A \subseteq \Sigma \cdot \Sigma^{+}$ is necessary to ensure that all traces in A have a next time point. The Future rule separates A from B by following this principle: if the set obtained by choosing one suffix for every trace in A is separable from the set of all suffixes of the traces in B, then there is a formula of the form $\mathsf{F}\varphi$ separating A from B. The rules And, WeakNext and Globally are designed to be duals of the rules Or, Next and Future, respectively.

By using the proof system one can derive whether a pair of (finite or infinite) sets of traces (A, B) is in the separable relation $\langle \cdot, \cdot \rangle$. Because of Lemma 2, this is not, however, a particularly useful application. Instead, the proof system is to be used to derive non-trivial lower (or upper) bounds on the size of the minimal formula that separates A from B. This is done by studying the sizes of the possible deduction trees of $\langle A, B\rangle$ in the proof system.

For instance, the deduction tree of Figure 2 shows that there is a formula φ having $\text{size}(\varphi) = 5$ and separating $\{abaa, aaaa\}$ from $\{aaab\}$. This formula is found by simply reading bottom-up, starting from the root, the rules in the deduction tree, associating to each rule the homonymous operator of LTL. In the case of the tree in Figure 2 we have $\varphi := (\mathsf{X}\neg p) \vee \mathsf{G}p$. Note that the formula φ is not the smallest separating formula, because the formula $\mathsf{XXG}p$ also separates $\{abaa, aaaa\}$ from $\{aaab\}$ and corresponds to a tree of size 4.

The correspondence between deduction trees and formulae is formalised in the next theorem (we remark that A and B below do not need to be finite sets).

Theorem 3. *Consider* $A, B \subseteq \Sigma^+$*. Then, the term* $\langle A, B\rangle$ *has a deduction tree of size* k *if and only if there is a formula* φ *of* $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$ *separating* A *from* B *and such that* $\mathrm{size}(\varphi) = k$*.*

Proof sketch. We leave to the reader the proof of the left to right direction of the theorem (shown by induction on k), as it is not required to establish lower bounds on the sizes of formulae, and focus instead on the right to left direction.

Consider a $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$ formula φ that separates A and B. We construct a deduction tree of size $\mathrm{size}(\varphi)$. We proceed by structural induction on φ.

base case: φ literal. The deduction tree consists of a single rule Atomic.

induction step, case: $\varphi = \varphi_1 \vee \varphi_2$. Define $A_1 := \{a \in A : a \models \varphi_1\}$ and $A_2 := A \setminus A_1$. From $A \models \varphi$ and $B \perp\!\!\!\perp \varphi$ we get $A_i \models \varphi_i$ and $B \perp\!\!\!\perp \varphi_i$ for both $i \in \{1, 2\}$. By induction hypothesis $\langle A_i, B\rangle$ has a deduction tree of size $\mathrm{size}(\varphi_i)$. By applying the rule And, we obtain a deduction tree for $\langle A, B\rangle$ having size $\mathrm{size}(\varphi_1) + \mathrm{size}(\varphi_2) + 1 = \mathrm{size}(\varphi)$.

induction step, case: $\varphi = \mathsf{X}\psi$. Since $A \models \mathsf{X}\psi$, for every $\sigma \in A$ we have $|\sigma| \geq 2$ and $(\sigma, 1) \models \psi$. By definition of A^{X}, $A \subseteq \Sigma \cdot \Sigma^+$ and $A^{\mathsf{X}} \models \psi$. From $B \perp\!\!\!\perp \mathsf{X}\psi$, for every $\sigma' \in B$, if $|\sigma'| \geq 2$ then $(\sigma', 1) \not\models \psi$. By definition of B^{X}, we have $B^{\mathsf{X}} \perp\!\!\!\perp \psi$. By induction hypothesis, $\langle A^{\mathsf{X}}, B^{\mathsf{X}}\rangle$ has a deduction tree of size $\mathrm{size}(\psi)$. We apply the rule Next to obtain a deduction tree of $\langle A, B\rangle$ of size $\mathrm{size}(\psi) + 1 = \mathrm{size}(\varphi)$.

induction step: $\varphi = \mathsf{F}\psi$. Since $A \models \mathsf{F}\psi$, for every $\sigma \in A$ there is $j_\sigma \in \mathrm{pos}(\sigma)$ such that $(\sigma, j_\sigma) \models \psi$. Let $f \in \mathrm{F}_A$ be the map given by $f(\sigma) = j_\sigma$ for every $\sigma \in A$. We have $A^f \models \psi$. We show that $B^{\mathsf{G}} \perp\!\!\!\perp \psi$. *Ad absurdum*, suppose there is $\sigma_1 \in B^{\mathsf{G}}$ such that $\sigma_1 \models \psi$. By definition of B^{G} there is $\sigma_2 \in B$ such that $\sigma_1 \sqsubseteq \sigma_2$. Then, $(\sigma_2, 0) \models \mathsf{F}\psi$. However, this contradicts the fact that $B \perp\!\!\!\perp \mathsf{F}\psi$. Therefore, $B^{\mathsf{G}} \perp\!\!\!\perp \psi$. By induction hypothesis, $\langle A^f, B^{\mathsf{G}}\rangle$ has a deduction tree of size $\mathrm{size}(\psi)$. By applying the rule Future, we obtain a deduction tree for $\langle A, B\rangle$ of size $\mathrm{size}(\psi) + 1 = \mathrm{size}(\varphi)$.

induction step, cases $\varphi = \varphi_1 \wedge \varphi_2$, $\varphi = \widetilde{\mathsf{X}}\psi$ and $\varphi = \mathsf{G}\psi$. The cases for $\varphi = \varphi_1 \wedge \varphi_2$, $\varphi = \widetilde{\mathsf{X}}\psi$ and $\varphi = \mathsf{G}\psi$ are analogous to the cases $\varphi = \varphi_1 \vee \varphi_2$, $\varphi = \mathsf{X}\psi$ and $\varphi = \mathsf{F}\psi$, respectively. □

The right to left direction of Theorem 3 implies the following corollary that highlights how our proof system is used for formulae sizes lower bounds.

Corollary 2. *Consider a formula* φ *in* $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$*. Suppose that (i) there are* $A, B \subseteq \Sigma^+$ *such that* φ *separates* A *from* B*, and (ii) every deduction tree of* $\langle A, B\rangle$ *has size at least* k*. Then,* $\mathrm{size}(\varphi) \geq k$*.*

4.2 Connections with the Adler–Immerman games

As outlined in Section 1, our proof system can be seen as an adaptation of the games for CTL introduced by Adler and Immerman in [1]. We now illustrate this connection. Readers that are mostly interested in seeing our proof system in action may want to skip to Section 5.

The *Adler–Immerman games* extend the classical Ehrenfeucht–Fraïssé games in order to bound the *sizes* of the formulae that separate two (sets of) structures, instead of their quantifier depths. As in the Ehrenfeucht–Fraïssé games, the Adler–Immerman games are two-player games between a *spoiler* and a *duplicator*. The game arena is a pair of sets of structures (A, B), and at each round of the game the spoiler choses a rule r to play (there is one rule for each Boolean connective and operator of the logic) and plays on one set of structures accordingly to what r dictates. The duplicator replies on the other set, again accordingly to r. The goal of the spoiler is to separate A from B (i.e., to show $\langle A, B\rangle$ in the context of CTL) in fewer rounds as possible, whereas the duplicator must prolong the game as much as she can. The length of the minimal game corresponds to the size of the minimal formula separating A from B. The main difference between an Adler–Immerman game and an Ehrenfeucht–Fraïssé game is that, in the former, in each round the duplicator is allowed to make copies of the structures in the set she is playing on, and to play differently in each of these copies. This extra power given to the duplicator is why the games end up capturing the notion of size of a formula.

In the setting of the Adler–Immerman games, the rule for the operator F in LTL would be spelled as follows: *"For each structure $\sigma \in A$, the spoiler moves to a future position of σ (i.e., $\sigma[j\rangle$ for some $j \in pos(\sigma)$). The duplicator answers by first making as many copies of elements in B as she wants, and then selects a future position for each of these copies"*. Because she can make copies, the duplicator has a trivial optimal strategy: at each round, copy the structures in B as much as possible, choosing a different position in each copy. The rule for F the simplifies to *"For each structure $\sigma \in A$, the spoiler moves to a future position of σ. The duplicator answers with B^{G}"*, which corresponds to our rule Future.

While Adler and Immerman discuss the fact that the duplicator has a trivial optimal strategy, they do not restate the games with only one player (mainly to not lose the similarity with the Ehrenfeucht–Fraïssé games). Our work shows that removing the duplicator yields a natural one-player game based on building proofs within a proof system. We think that this proof-system view has a few merits over the games. When proving lower bounds, it reduces the clumsiness of discussing the various moves of the spoiler and the replies of the duplicator. The combinatorics is of course still there, but not the players, and this substantially simplifies the exposition. Second, the proof system resembles the way in which one reasons about the *algorithmic* problem of separating A from B. For instance, the algorithm presented in [21] uses decision trees for solving this problem. These decision trees, when they encode a formula from $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$, can be easily translated into proofs in our proof system. We discuss more this line of work connected with LTL formulae learning and explainable planning in Section 8.

5 The exponential lower bound for $\boldsymbol{\Phi_n}$

In this section, we show that, for every $n \geq 1$, all formulae of $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$ characterising the $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ formula Φ_n defined in Section 3 have size at least 2^n.

According to the definition of Φ_n, we consider a set of $2n+2$ distinct atomic propositions $\mathcal{AP} := \{\widetilde{p}, \widetilde{q}\} \cup P \cup Q$, with $P := \{p_1, \dots, p_n\}$ and $Q := \{q_1, \dots, q_n\}$; and $\Sigma := 2^{\mathcal{AP}}$. Throughout the section, let $\alpha(n) := 2^{n+1}(n+2)^2$, i.e., the upper bound given in Lemma 1 for one of these formulae.

Following Corollary 2, to prove the exponential lower bound we

1. define $\mathbf{A}, \mathbf{B} \subseteq \Sigma^+$ such that Φ_n separates $\mathbf{A}$ from $\mathbf{B}$ (Section 5.1), and
2. prove that every deduction tree for $\langle \mathbf{A}, \mathbf{B} \rangle$ has size at least 2^n (Section 5.2).

5.1 Setting up the sets of traces A and B

We define the sets of *types* $T_P := \{\tau \in \Sigma : \widetilde{p} \in \tau \text{ and } \tau \subseteq P \cup \{\widetilde{p}\}\}$ and $T_Q := \{\tau \in \Sigma : \widetilde{q} \in \tau \text{ and } \tau \subseteq Q \cup \{\widetilde{q}\}\}$. Similarly to what done in the proof of Lemma 1, we write $\overline{(\cdot)}$ for the involution on $T_P \cup T_Q$ sending a type $\tau \in T_Q$ into the (only) type $\overline{\tau} \in T_P$ with $q_i \in \tau$ if and only if $p_i \in \overline{\tau}$, for every $i \in [1, n]$.

Throughout the section, we fix a (arbitrary) strict total order $\prec$ on the elements of T_Q. Then, we denote by $\mathcal{E} \in (\varnothing^{\alpha(n)} \cdot T_Q)^{2^n} \cdot \varnothing^{\alpha(n)}$ the (only) finite word enumerating all elements in T_Q, with respect to the order $\prec$. Note that, in $\mathcal{E}$, between any two subsequent elements of T_Q there are $\alpha(n)$ positions of type $\varnothing$. This "padding" added to the enumeration is required to handle the rules NEXT and WEAKNEXT. Given $\tau \in T_Q$, we write $\mathcal{E}|_{-\tau}$ for the trace obtained from $\mathcal{E}$ by eliminating the only position of type τ, together with the $\alpha(n)$ positions of type $\varnothing$ preceding it. So, $\mathcal{E}_{-\tau}$ belongs to $(\varnothing^{\alpha(n)} \cdot T_Q)^{2^n-1} \cdot \varnothing^{\alpha(n)}$.

For instance, consider the case of $n = 2$, so $Q = \{q_1, q_2\}$ and $\alpha(n) = 128$. Suppose $\{\widetilde{q}\} \prec \{\widetilde{q}, q_1\} \prec \{\widetilde{q}, q_2\} \prec \{\widetilde{q}, q_1, q_2\}$ to be the strict order on T_Q. Then,

$$\mathcal{E} = \varnothing^{128} \cdot \{\widetilde{q}\} \cdot \varnothing^{128} \cdot \{\widetilde{q}, q_1\} \cdot \varnothing^{128} \cdot \{\widetilde{q}, q_2\} \cdot \varnothing^{128} \cdot \{\widetilde{q}, q_1, q_2\} \cdot \varnothing^{128},$$
$$\mathcal{E}|_{-\{\widetilde{q}, q_2\}} = \varnothing^{128} \cdot \{\widetilde{q}\} \cdot \varnothing^{128} \cdot \{\widetilde{q}, q_1\} \cdot \varnothing^{128} \cdot \{\widetilde{q}, q_1, q_2\} \cdot \varnothing^{128}.$$

For the rest of the paper, we denote with $\mathbf{A}$ and $\mathbf{B}$ the sets:

$$\mathbf{A} := \{\varnothing^j \cdot \overline{\tau} \cdot \mathcal{E} : j \in \mathbb{N}, \tau \in T_Q\}, \qquad \mathbf{B} := \{\varnothing^j \cdot \overline{\tau} \cdot (\mathcal{E}|_{-\tau}) : j \in \mathbb{N}, \tau \in T_Q\}.$$

Lemma 3. *The formula Φ_n separates* $\mathbf{A}$ *from* $\mathbf{B}$.

Proof. Let $j \in \mathbb{N}$ and $\tau \in T_Q$. In a nutshell, the fact that $\varnothing^j \cdot \overline{\tau} \cdot \mathcal{E} \models \Phi_n$ follows from the fact that τ occurs in $\mathcal{E}$, and from the position corresponding to τ one can refer back to $\overline{\tau}$ and find in this way a position satisfying p_i if and only if $q_i \in \tau$, for every $i \in [1, n]$. However, since τ is removed from $\mathcal{E}|_{-\tau}$, we see that $b := \varnothing^j \cdot \overline{\tau} \cdot (\mathcal{E}|_{-\tau}) \not\models \Phi_n$: indeed, $b[j] = \overline{\tau}$ corresponds to the only position in b satisfying $\widetilde{p}$, but τ does not appear in b (since it does not appear in $\mathcal{E}|_{-\tau}$). Therefore, $\mathbf{A} \models \Phi_n$ and $\mathbf{B} \perp\!\!\!\perp \Phi_n$. □

5.2 Separating A from B requires an exponential proof

We now show that every deduction tree for $\langle \mathbf{A}, \mathbf{B} \rangle$ has size at least 2^n. To do so, we use a relation $\approx$ that, roughly speaking, states what elements $(a, b) \in \mathbf{A}^{\mathsf{G}} \times \mathbf{B}^{\mathsf{G}}$ are similar enough to require a non-trivial proof in order to be separated using the proof system. Formally, for $a, b \in \Sigma^+$, we write $a \approx b$ whenever:

a and b are in the language $\varnothing^u \cdot \rho \cdot \varnothing^{\alpha(n)} \cdot \Sigma^*$, for some $u \in \mathbb{N}$ and $\rho \in T_Q \cup T_P$.

The central issue in the proof of the lower bound is counting how many of these pairs $a \approx b$ are preserved when applying the rules of the proof system. This count is done inductively on the size of the deduction tree, and allows us to derive the following lemma.

Lemma 4. *Let $r_1, t_1, \dots, r_m, t_m \in \mathbb{N}$ and let $\tau_1, \dots, \tau_m \in T_Q$ be pairwise distinct sets. Consider $A \subseteq \mathbf{A}^{\mathsf{G}}$, $B := \{(\varnothing^{t_i} \cdot \overline{\tau_i} \cdot \mathcal{E}|_{-\tau_i})[r_i\rangle : i \in [1, m]\}$, and $C := \{(a, b) \in A \times B : a \approx b\}$. Every deduction tree for $\langle A, B\rangle$ has size at least $|C| + 1$.*

Proof. Below, suppose that $\langle A, B\rangle$ has a deduction tree (else the statement is trivially true). In particular, let $\mathcal{T}$ be a minimal deduction tree for $\langle A, B\rangle$, and assume it has size s. Note that the hypothesis that $\tau_1, \dots, \tau_m$ are distinct implies $|B| \le 2^n$, which in turn implies $|C| < 2^n$ (by definition of $\approx$, for every $b \in B$ there is at most one $a \in \mathbf{A}^{\mathsf{G}}$ such that $a \approx b$). Then, w.l.o.g. we can assume $s \le \alpha(n)$; otherwise the lemma follows trivially.

During the proof, we write $\prec$ for the strict total order on elements of T_Q used to construct the trace $\mathcal{E}$ enumerating T_Q. Before continuing the proof of the lemma, we highlight a useful property of the elements of C.

Claim 1. Let $(a, b) \in C$ and $i \in [1, m]$ with $b = (\varnothing^{t_i} \cdot \overline{\tau_i} \cdot \mathcal{E}|_{-\tau_i})[r_i\rangle$. Then, $b = \varnothing^u \cdot \rho \cdot \varnothing^{\alpha(n)} \cdot \sigma$, for some $u \in \mathbb{N}$, $\rho \in \{\overline{\tau_i}\} \cup \{\tau \in T_Q : \tau \prec \tau_i\}$ and $\sigma \in \Sigma^*$.

In a nutshell, this claim tells us that for every $(a, b) \in C$ we have $b \not\sqsubseteq \mathcal{E}$.

Let us go back to the proof of Lemma 4. If $A = \varnothing$ or $m = 0$ then $C = \varnothing$ and the lemma follows trivially. Below, let us assume $A \neq \varnothing$ and $m \ge 1$. We prove the statement by induction on the size s of $\mathcal{T}$.

In the base case $s = 1$, $\mathcal{T}$ is a simple application of the rule Atomic. This means that for every $a \in A$ and $b \in B$ we have $a[0] \neq b[0]$. By definition of $\approx$, this implies $C = \varnothing$, and therefore $s \ge |C| + 1$.

Let us then consider the induction step $s \ge 2$. Note that if $|C| \le 1$ then the statement follows trivially. Hence, below, we assume $|C| \ge 2$. We split the proof depending on the rule applied to the root $\langle A, B\rangle$ of $\mathcal{T}$. Since $s \ge 2$, this rule cannot be Atomic. We omit the cases for Or and And, as they simply follow the induction hypothesis, and focus on the rules related to temporal operators.

- **case: rules** Next **and** WeakNext**.** We consider Next and WeakNext together, as both require $\langle A^{\mathsf{X}}, B^{\mathsf{X}}\rangle$. Perhaps surprisingly, this case is non-trivial. The main difficulty stems from the fact that $C' := \{(a, b) \in A^{\mathsf{X}} \times B^{\mathsf{X}} : a \approx b\}$ might in principle even be empty, and thus applying the induction hypothesis on $\langle A^{\mathsf{X}}, B^{\mathsf{X}}\rangle$ is unhelpful for concluding that $s \ge |C| + 1$. We now show how to circumvent this issue. The minimal deduction tree for $\langle A^{\mathsf{X}}, B^{\mathsf{X}}\rangle$ has size $s - 1$. Within this deduction tree, consider the maximal partial deduction tree $\mathcal{T}'$ rooted at $\langle A^{\mathsf{X}}, B^{\mathsf{X}}\rangle$ and made solely of applications of the rules And, Or, Next, and WeakNext. Let $\langle A_1, B_1\rangle, \dots, \langle A_q, B_q\rangle$ be the leafs of such a tree. Let $j \in [1, q]$. In the tree $\mathcal{T}$, to $\langle A_j, B_j\rangle$ it is applied a rule among Atomic, Future and Globally. Let $\xi_j \ge 1$ be the number of Next and WeakNext rules used in the path of $\mathcal{T}$ from $\langle A, B\rangle$ to $\langle A_j, B_j\rangle$. Note that, from $s \leq \alpha(n)$, we have $\xi_j \leq \alpha(n)$. We define the following two sets C_j and N_j, whose role is essentially to "track" the evolution of pairs in C with respect to $A_j \times B_j$:

$$\begin{aligned} C_j &\coloneqq \{(a[\xi_j\rangle, b[\xi_j\rangle) \in A_j \times B_j : (a,b) \in C,\, a[\xi_j\rangle \approx b[\xi_j\rangle\},\\ N_j &\coloneqq \{(a[\xi_j\rangle, b[\xi_j\rangle) \in A_j \times B_j : (a,b) \in C,\, a[\xi_j\rangle \not\approx b[\xi_j\rangle\}. \end{aligned}$$

The minimal deduction tree for $\langle A_j, B_j\rangle$ has size $s_j \geq |C_j| + 1$; by induction hypothesis. Claims 2 to 4 below highlight a series of properties on the sets C_j and N_j from which we derive $s \geq |C| + 1$.

Claim 2. For every $j \in [1, q]$, if $C_j \cup N_j \neq \varnothing$ then the rule applied to $\langle A_j, B_j\rangle$ in $\mathcal{T}$ is either Future or Globally.

As already said, the rule applied to $\langle A_j, B_j\rangle$ is among the rules Atomic, Future and Globally. Then, showing that $a[0] = b[0]$ for every $(a, b) \in C_j \cup N_j$ excludes the rule Atomic.

Claim 3. For every $j \in [1, q]$, $|N_j| \leq 1$.

The proof of this claim is by contradiction, assuming the existence of distinct $(a_1, b_1), (a_2, b_2) \in N_j$. In the proof, we analyse structural properties of the traces a_1, a_2, b_1 and b_2, and consider several cases depending on such properties (for instance, one case split depends on whether $a_1 \sqsubseteq a_2$). In all these cases, we reach a contradiction with either $(a_1, b_1) \neq (a_2, b_2)$ or Claim 2.

Claim 4. $|C| \leq \sum_{j=1}^{q} |C_j \cup N_j|$.

The claim follows as soon as one proves the following two statements:

1. for every $(a, b) \in C$ there is $j \in [1, q]$ such that $(a[\xi_j\rangle, b[\xi_j\rangle) \in C_j \cup N_j$,
2. for all distinct $(a_1, b_1), (a_2, b_2) \in C$, we have $(a_1[\ell\rangle, b_1[\ell\rangle) \neq (a_2[\ell\rangle, b_2[\ell\rangle)$ for every $\ell \leq \alpha(n)$ (recall that $\xi_j \leq \alpha(n)$, for every $j \in [1, q]$).

Item 1 is by induction on the size of $\mathcal{T}'$. Similarly to Claim 3, the proof of Item 2 again requires to consider many cases, and uses properties of $\approx$, $\mathcal{E}$ and $\mathcal{E}|_{-\tau_i}$.

Thanks to Claims 3 and 4, one can then prove $s \geq |C| + 1$, concluding the proof for the rules Next and WeakNext:

$$\begin{aligned} s &\geq 1 + \textstyle\sum_{j=1}^{q} s_j && \text{by definition of } \mathcal{T} \text{ and } \mathcal{T}'\\ &\geq 1 + \textstyle\sum_{j=1}^{q} (|C_j| + 1) && \text{by } s_j \geq |C_j| + 1 \text{ (induction hypothesis)}\\ &\geq 1 + \textstyle\sum_{j=1}^{q} (|C_j \cup N_j|) && \text{by } |N_j| \leq 1 \text{ (Claim 3)}\\ &\geq |C| + 1 && \text{by } |C| \leq \textstyle\sum_{j=1}^{q} |C_j \cup N_j| \text{ (Claim 4).} \end{aligned}$$

• **case: rule** Future. Let $f \in \mathrm{F}_A$ be the future point used when applying this rule. Define $C' \coloneqq \{(a', b') \in A^f \times B^{\mathsf{G}} : a' \approx b'\}$. The minimal deduction tree for $\langle A^f, B^{\mathsf{G}}\rangle$ has size $s - 1$. By induction hypothesis, $s - 1 \geq |C'| + 1$, i.e., $s \geq |C'| + 2$. We divide the proof into two cases.

Case 1: *for every $a' \in A^f$, $a' \not\sqsubseteq \mathcal{E}$.* By definition of $\approx$, every $(a,b) \in C$ is such that a and b belong to the language $\varnothing^u \cdot \overline{\tau_i} \cdot \varnothing^{\alpha(n)} \cdot \Sigma^*$ for some $u \in \mathbb{N}$, and $i \in [1,m]$. Since $a^f \not\sqsubseteq \mathcal{E}$, we must have $f(a) \leq u+1$. Then, $a^f \approx b[f(a)\rangle$. Note that distinct $(a,b) \in C$ concern distinct $\overline{\tau_i}$ with $i \in [1,m]$, and therefore, together with $b[f(a)\rangle \in B^G$, one concludes that $|C'| \geq |C|$; and so $s \geq |C|+2$.

Case 2: *there is $a' \in A^f$ such that $a' \sqsubseteq \mathcal{E}$.* Let us denote with $\widetilde{a}$ the element in A^f such that $\widetilde{a} \sqsubseteq a$ for every $a \in A^f$. The existence of such an element follows directly from the fact that $a' \sqsubseteq \mathcal{E}$ for some $a' \in A^f$.

Let $I \subseteq [1,m]$ be the subset of those indices $i \in [1,m]$ for which there is a pair $(a',b') \in C$ such that $b' = (\varnothing^{t_i} \cdot \overline{\tau_i} \cdot \mathcal{E}|_{\tau_i})[r_i\rangle$. Without loss of generality, suppose $I = \{1,\dots,q\}$ for some $q \leq m$, and that $\tau_1 \prec \tau_2 \prec \cdots \prec \tau_q$; recall that all these types are pairwise distinct. By definition of $\approx$, for every $b' \in B$ there is at most one $a' \in \mathbf{A}^{\mathsf{G}}$ such that $a' \approx b'$, and therefore $q = |C|$. To conclude the proof it suffices to show $|C'| \geq q-1$. We do so by establishing a series of claims. Recall that we are assuming $|C| \geq 2$, so in particular C and I are non-empty.

Claim 5. There are $u \in \mathbb{N}$, $\rho \in T_Q$ and $\sigma \in (2^Q)^*$ s.t. $\widetilde{a} = \varnothing^u \cdot \rho \cdot \varnothing^{\alpha(n)} \cdot \sigma$. Moreover, $\rho \preceq \tau_i$ for every $i \in I$.

The first statement of this claim is established from the definition of $\widetilde{a}$. The second statement is proven by contradiction. In particular, assuming that there is $i \in I$ such that $\tau_i \prec \rho$ yields a contradiction with Claim 1.

Below, we write u, ρ and σ for the objects appearing in Claim 5. Note that, from $\tau_1 \prec \cdots \prec \tau_q$, the second statement of Claim 5 implies $\rho \prec \tau_2 \cdots \prec \tau_q$. For $i \in [2,q]$, let (a'_i, b'_i) denote the pair in C such that $b'_i = (\varnothing^{t_i} \cdot \overline{\tau_i} \cdot \mathcal{E}|_{\rho_i})[r_i\rangle$.

Claim 6. For each $i \in [2,q]$ there is $\ell \in \mathbb{N}$ such that $\widetilde{a} \approx b''_i$ with $b''_i := b'_i[\ell\rangle$. Moreover, every type in $\{\tau_2,\dots,\tau_q\} \setminus \{\tau_i\}$ appears in some position of b''_i.

This claim is proven using Claims 1 and 5 and properties of $\mathcal{E}|_{-\tau_i}$.

Since all types $\tau_2,\dots,\tau_q$ are pairwise distinct, from the second statement in Claim 6 we conclude that $b''_i \neq b''_j$ for every two distinct $i,j \in I \setminus \{i_1\}$. Then, the first statement in Claim 6 entails $|C'| \geq q-1$.

• **case: rule** GLOBALLY. Let $f \in \mathrm{F}_A$ be the future point used when applying this rule. The minimal deduction tree for $\langle A^{\mathsf{G}}, B^f \rangle$ has size $s-1$. We define $C' := \{(a',b') \in A^{\mathsf{G}} \times B^f : a' \approx b'\}$. By induction hypothesis, $s-1 \geq |C'|+1$, i.e., $s \geq |C'|+2$. To conclude the proof it suffices to show that $|C'| \geq |C|-1$ (in fact, we prove $|C'| \geq |C|$). Let $\{(a_1,b_1),\dots,(a_{|C|},b_{|C|})\} = C$.

Claim 7. For every $j \in [1,|C|]$, b_j^f is not a suffix of $\mathcal{E}$. More precisely, given $i \in [1,m]$ such that $b_j = (\varnothing^{t_i} \cdot \overline{\tau_i} \cdot \mathcal{E}|_{-\tau_i})[r_i\rangle$, we have $b_j^f = \varnothing^u \cdot \rho \cdot \varnothing^{\alpha(n)} \cdot \sigma$, for some $u \in \mathbb{N}$, $\rho \in \{\overline{\tau_i}\} \cup \{\tau \in T_P : \tau \prec \tau_i\}$ and $\sigma \in \Sigma^*$.

See the similarities between this claim and Claim 1. The first statement is proven by contradiction, deriving an absurdum with the existence of $\mathcal{T}$. The second statement follows from the definition of $\mathcal{E}|_{-\tau_i}$.

Starting from Claim 7, we conclude that (i) for every $j \neq k \in [1, |C|]$, $b^f_j \neq b^f_k$, and (ii) for every $j \in [1, |C|]$ there is $\ell \in \mathbb{N}$ such that $a_j[\ell\rangle \approx b^f_j$. This directly implies $|C'| \geq |C|$. This concludes both the proof of the case Globally and the proof of the lemma. □

Together, Lemmas 3 and 4 yield an exponential lower bound for all formulae of $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$ characterising Φ_n (Lemma 5), which in turn implies Theorem 1.

Lemma 5. *Let* $\Psi_n \in \mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$*. If* $\Psi_n \equiv \Phi_n$ *then* $\mathrm{size}(\Psi_n) \geq 2^n$*.*

Proof. We define the sets $A = \{\overline{\tau} \cdot \mathcal{E} : \tau \in T_Q\}$ and $B = \{\overline{\tau} \cdot \mathcal{E}_{-\tau} : \tau \in T_Q\}$. Observe that $A \subseteq \mathbf{A} \subseteq \mathbf{A}^{\mathsf{G}}$ and $B \subseteq \mathbf{B}$. Let $C = \{(a, b) \in A \times B : a \approx b\}$. From the definition of $\approx$, $|C| = 2^n$. We apply Lemma 4, and conclude that the minimal deduction tree for $\langle A, B\rangle$ has size at least 2^n (in fact, $2^n + 1$). Since $A \subseteq \mathbf{A}$ and $B \subseteq \mathbf{B}$, the same holds for the minimal deduction tree for $\langle \mathbf{A}, \mathbf{B}\rangle$. Then, the theorem follows from Corollary 2 and Lemma 3. □

While we do not prove it formally, we claim that Theorem 1 also holds for infinite traces. It is in fact quite simple to see this: all traces in $\mathbf{A}$ and $\mathbf{B}$, have a suffix of the form $\varnothing^{\alpha(n)}$. Roughly speaking, these suitably long suffixes are added to make the far-end of the traces in $\mathbf{A}$ and $\mathbf{B}$ indistinguishable at the level of formulae, so that they cannot be used in deduction trees to separate $\mathbf{A}$ from $\mathbf{B}$. Then, to prove Theorem 1 for infinite traces, it suffices to update the proof system to handle these structures and consider an infinite suffix $\varnothing^{\omega}$ instead. The proof of Lemma 4 goes through with no significant change.

A second observation: traces in $\mathbf{A}$ and $\mathbf{B}$ are closed under taking arbitrary long prefixes of the form $\varnothing^{j}$. This feature is not used to prove Lemma 5 (see the definition of A and B in the proof). However, these prefixes play a role in the next section, when studying the succinctness of $\mathsf{F}(\mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}])$ on infinite traces.

6 Theorem 2: a 2^n lower bound for LTL[F] pastification

In this section, we rely on Lemma 5 to prove Theorem 2 and Corollary 1.

Theorem 2 is proven by relying on a "future–past duality" between future and past fragments of LTL. Given a trace $\sigma \in \Sigma^+$ we define the *reverse of* σ, written σ^-, as the trace satisfying $\sigma^-[i] = \sigma[|\sigma| - i]$ for every $i \in \mathrm{pos}(\sigma)$. The *reverse of a language* $\mathcal{L} \subseteq \Sigma^+$ is defined as the language $\mathcal{L}^- := \{\sigma^- : \sigma \in \mathcal{L}\}$. Clearly, $(\mathcal{L}^-)^- = \mathcal{L}$. Given a set of temporal operators $S \subseteq \{\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}\}$, we write S^- for the set of temporal operators among $\{\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}\}$ such that S^- contains Y (resp. $\widetilde{\mathsf{Y}}$; O; H) if and only if S contains X (resp. $\widetilde{\mathsf{X}}$; F; G). For finite traces, the following lemma, proves that if there is a family of languages $(\mathcal{L}_n)_{n \geq 1}$ that can be compactly defined in $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ but explodes in $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$, then the family $(\mathcal{L}_n^-)_{n \geq 1}$ can be compactly defined in $\mathsf{LTL}[\mathsf{F}]$ but explodes in $\mathsf{F}(\mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}])$.

Lemma 6. *Let* $\mathcal{L} \subseteq \Sigma^+$*,* $S \subseteq \{\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}\}$*, and* φ *be a formula in* $\mathsf{F}(\mathsf{LTL}[S^-])$*. There is a formula* ψ *in* $\mathsf{F}(\mathsf{LTL}[S])$ *such that* $\mathcal{L}(\psi) = \mathcal{L}(\varphi)^-$ *and* $\mathrm{size}(\psi) = \mathrm{size}(\varphi)$*.*

Theorem 2 follows by applying Lemma 6 on the family of formulae $(\Phi_n)_{n\geq 1}$ defined in Section 3, and by relying on the exponential lower bounds of Lemma 5. The sequence of languages showing that $\mathsf{LTL}[\mathsf{F}]$ can be exponentially more succinct than $\mathsf{F}(\mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}])$ is given by $(\mathcal{L}(\Phi_n)^-)_{n\geq 1}$.

Next, we extend Theorem 2 to the case of infinite traces. As usual, let Σ^ω be the set of all infinite traces over the finite alphabet Σ. We denote with $\mathcal{L}^\omega(\varphi)$ the language of φ, when φ is interpreted over infinite traces (we refer the reader to, e.g., [2] for the semantics of LTL on infinite traces).

Lemma 7. *The family of languages of infinite traces* $(\mathcal{L}(\Phi_n)^- \cdot \Sigma^\omega)_{n\geq 1}$ *is such that, for every* $n \geq 1$*, (i) there is a formula* φ *of* $\mathsf{LTL}[\mathsf{F}]$ *such that* $\mathrm{size}(\varphi) \in \mathcal{O}(n)$ *and* $\mathcal{L}^\omega(\varphi) = \mathcal{L}(\Phi_n)^- \cdot \Sigma^\omega$*, and (ii) for every formula* ψ *in* $\mathsf{F}(\mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}])$*, if* $\mathcal{L}^\omega(\psi) = \mathcal{L}(\Phi_n)^- \cdot \Sigma^\omega$ *then* $\mathrm{size}(\psi) \geq 2^n$*.*

Item (i) in the lemma above follows by applying Lemma 6 and exploiting the fact that formulae φ in $\mathsf{LTL}[\mathsf{F}]$ satisfy $\mathcal{L}^\omega(\varphi) = \mathcal{L}(\varphi) \cdot \Sigma^\omega$ and $\mathcal{L}(\varphi) = \mathcal{L}(\varphi) \cdot \Sigma^*$ (cf. [2, Definition 5 and Lemma 5]). The proof of Item (ii) is instead quite subtle. One would like to use the hypothesis $\mathcal{L}^\omega(\psi) = \mathcal{L}(\Phi_n)^- \cdot \Sigma^\omega$ and that $\mathcal{L}(\psi)$ is a cosafety language to derive $\mathcal{L}(\psi) = \mathcal{L}(\Phi_n)^-$. However, note that nothing prevents $\mathcal{L}(\psi)$ to be equal to $\mathcal{L}(\Phi_n)^- \cdot \Sigma$, and as it stands we do not have bounds for characterising this language. We apply instead the following strategy. We consider the family of structures $A' \coloneqq \{a^- \cdot \varnothing^\omega : a \in \mathbf{A}\}$ and $B' \coloneqq \{b^- \cdot \varnothing^\omega : b \in \mathbf{B}\}$. Note that $A' \subseteq \mathcal{L}^\omega(\psi)$ and $B' \cap \mathcal{L}^\omega(\psi) = \varnothing$. Since ψ is of the form $\mathsf{F}(\alpha)$ with $\alpha \in \mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}, \mathsf{H}]$, we can, roughly speaking, study the effects of applying to A' and B' a variant of the rule Future for infinite words and that does not "forget the past", and then reverse all traces in the resulting sets $(A')^f$ and $(B')^{\mathsf{G}}$. In this way, we obtain two sets of finite traces $\widetilde{A} \subseteq \mathbf{A}$ and $\widetilde{B} \subseteq \mathbf{B}$ (this is where the prefixes $\varnothing^j$ discussed at the end of Section 5 play a role). We show that the hypotheses of Lemma 4 apply to $\widetilde{A}$ and a set $\widehat{B} \subseteq \widetilde{B}$ for which the set $\{(a, b) \in \widetilde{A} \times \widehat{B} : a \approx b\}$ has size at least $2^n - 1$. By Corollary 2, we get that α, and thus ψ, is of size at least 2^n.

Lemma 7 shows that Theorem 2 holds over infinite traces as well. Together with the $2^{\mathcal{O}(n)}$ upper bound for the pastification problem for $\mathsf{LTL}[\mathsf{X}, \mathsf{F}]$ into $\mathsf{F}(\mathsf{LTL}[\mathsf{Y}, \widetilde{\mathsf{Y}}, \mathsf{O}])$ established[6] in [4], this entails Corollary 1.

7 The automata method does not work for $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$

In this section we show that the classical method introduced by Markey in [20] to prove "future against past" succinctness discrepancies in fragments of LTL cannot be used to prove the results in Section 5, namely that $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ can be exponentially more succinct than $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$. Due to space constraints, we assume a basic familiarity with *non-deterministic Büchi automata* (NBAs) (and *deterministic Büchi automata*, DBAs), which are central tools in [20]. Moreover,

[6] To be more precise, in [4] the authors only provide a $2^{\mathcal{O}(n^2)}$ upper bound for their algorithm. Their analysis can however be easily improved to $2^{\mathcal{O}(n)}$.

we work on LTL on infinite traces, as done in [20], and write $\varphi \equiv_\omega \psi$ whenever $\mathcal{L}^\omega(\varphi) = \mathcal{L}^\omega(\psi)$. We write $\mathcal{L}^\omega(A)$ for the language of an NBA A.

Proposition 1 below summarises the method in [20], which is parametric on a fixed prefix Π of operators among X, F and G. Given two fragments F, F' of LTL, with F' pure future, to apply the method one has to provide the two families of formulae $(\Phi_n)_{n\geq 1} \in F$ and $(\Phi'_n)_{n\geq 1} \in F'$, as well as the family of minimal NBAs $(A_n)_{n\geq 1}$, satisfying the hypotheses of Proposition 1. In [20], this is done for $F = \mathsf{LTL}$ and F' set as the pure future fragment of LTL, using the prefix $\Pi = \mathsf{G}$.

Proposition 1 (Markey's method [20]). *Let F, F' be fragments of LTL, with F' pure future. Consider two families of formulae $(\Phi_n)_{n\geq 1} \in F$, $(\Phi'_n)_{n\geq 1} \in F'$, and a family of minimal NBAs $(A_n)_{n\geq 1}$, such that*

$$\text{size}(A_n) \in 2^{2^{\Omega(n)}}, \quad \text{size}(\Phi_n) \in poly(n), \quad \Phi_n \equiv_\omega \Phi'_n, \quad \mathcal{L}^\omega(\Pi(\Phi'_n)) = \mathcal{L}^\omega(A_n).$$

Then, $\text{size}(\Phi'_n) \in 2^{\text{size}(\Phi_n)^{\Omega(1)}}$ and F can be exponentially more succinct than F'.

The consequence $\text{size}(\Phi'_n) \in 2^{\text{size}(\Phi_n)^{\Omega(1)}}$ obtained in Proposition 1 follows directly from the fact that, from every pure future LTL formula φ, one can build an NBA A of size $2^{\mathcal{O}(\text{size}(\varphi))}$ such that $\mathcal{L}^\omega(A) = \mathcal{L}^\omega(\varphi)$ [26]. Then, the hypotheses $\text{size}(A_n) \in 2^{2^{\Omega(n)}}$ and $\mathcal{L}^\omega(\Pi(\Phi'_n)) = \mathcal{L}^\omega(A_n)$ imply $\text{size}(\Phi'_n) \in 2^{\Omega(n)}$.

To show that Proposition 1 cannot be used to derive that $F \coloneqq \mathsf{F}(\mathsf{LTL}[\mathsf{O}])$ can be exponentially more succinct than $F' \coloneqq \mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$, it suffices to show that no families $(\Phi_n)_{n\geq 1} \in F$, $(\Phi'_n)_{n\geq 1} \in F'$ and $(A_n)_{n\geq 1}$ achieve the hypotheses required by Proposition 1, no matter the temporal prefix Π. We do so by establishing that whenever $\text{size}(\Phi_n) \in \text{poly}(n)$ and $\Phi_n \equiv_\omega \Phi'_n$, the minimal *deterministic* Büchi automaton for $\mathcal{L}^\omega(\Pi(\Phi'_n))$ has size in $2^{\mathcal{O}(\text{poly}(n))}$. Therefore, no family of minimal NBAs $(A_n)_{n\geq 1}$ such that $\text{size}(A_n) \in 2^{2^{\Omega(n)}}$ can also satisfy the hypothesis $\mathcal{L}^\omega(\Pi(\Phi'_n)) = \mathcal{L}^\omega(A_n)$. Here is the formal statement:

Theorem 4. *Let Π be a prefix of k temporal operators among X, F and G. Let φ be a formula of $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$, and ψ be a formula of $\mathsf{LTL}[\mathsf{X}, \widetilde{\mathsf{X}}, \mathsf{F}, \mathsf{G}]$, with $\varphi \equiv_\omega \psi$. The minimal DBA for $\mathcal{L}^\omega(\Pi(\psi))$ is of size in $(k+1) \cdot 2^{\mathcal{O}(\text{size}(\varphi))}$.*

The proof of this theorem is divided into three steps.

As a first step, one shows that $\psi \equiv_\omega \mathsf{F}\psi$; which essentially follows from the fact that $\psi \equiv_\omega \varphi$ with $\varphi \in \mathsf{F}(\mathsf{LTL}[\mathsf{O}])$. Together with tautologies of LTL such as $\mathsf{FGF}\psi' \equiv_\omega \mathsf{GF}\psi'$, $\mathsf{FX}\psi' \equiv_\omega \mathsf{XF}\psi'$ and $\mathsf{GX}\psi' \equiv_\omega \mathsf{XG}\psi'$, the equivalence $\psi \equiv_\omega \mathsf{F}\psi$ let us rearrange Π into a prefix of the form either $\mathsf{X}^j\mathsf{GF}$ or $\mathsf{X}^j\mathsf{F}$, for some $j \leq k$. Let us focus on the former (more challenging) case of $\Pi = \mathsf{X}^j\mathsf{GF}$.

The second step required for the proof is to bound the size of the minimal DBA A recognising $\mathcal{L}^\omega(\mathsf{F}\psi)$. Thanks to the equivalences $\varphi \equiv_\omega \psi \equiv_\omega \mathsf{F}\psi$, such a DBA has size exponential in $\text{size}(\varphi)$ by the following lemma.

Lemma 8. *Let φ in $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$. There is a DBA for $\mathcal{L}^\omega(\varphi)$ of size $2^{\mathcal{O}(\text{size}(\varphi))}$.*

Starting from A, the third and last step of the proof requires constructing a DBA for $\mathcal{L}^\omega(\mathsf{X}^j\mathsf{GF}\psi)$ of size in $(j+1)\cdot 2^{\mathcal{O}(\text{size}(\varphi))}$. The treatment for the prefix X^j is simple, so this step is mostly dedicated to constructing a DBA for $\mathcal{L}^\omega(\mathsf{GF}\psi)$. In the case of LTL, it is known that closing an NBA under the globally operator G might lead to a further exponential blow-up (in fact, this is one of the reasons Markey's method is possible in the first place). However, because φ is in $\mathsf{F}(\mathsf{LTL}[\mathsf{O}])$, and it is thus a cosafety language (and so ψ is a cosafety language too), it turns out that the size of the minimal DBA for $\mathcal{L}^\omega(\mathsf{GF}\psi)$ is in $\mathcal{O}(\text{size}(A))$.

Lemma 9. *Let ψ be in* LTL*, such that $\mathcal{L}^\omega(\psi)$ is a cosafety language. Let A be a* DBA *for $\mathcal{L}^\omega(\mathsf{F}\psi)$. The minimal* DBA *for $\mathcal{L}^\omega(\mathsf{GF}\psi)$ has size in $\mathcal{O}(\text{size}(A))$.*

Thanks to Lemma 9, the proof of Theorem 4 can be easily completed. To prove this lemma, one modifies A by redirecting all transitions exiting a final state so that they mimic the transitions exiting the initial state of the automaton. The reason why the obtained automaton recognises $\mathcal{L}^\omega(\mathsf{GF}\psi)$ uses in a crucial way the fact that ψ and $\mathsf{F}\psi$ are cosafety languages.

8 Related and Future Work

The proof systems we use in this work to establish Theorem 2 and Theorem 1 are strongly related to recent work in two seemingly disconnected areas of computer science: (i) combinatorial games for formulae lower bounds of first-order logics and (ii) learning of LTL formulae in explainable planning and program synthesis.

Combinatorial games. We have already discussed the connections between our proof system and the CTL^+ games by Adler and Immerman [1]. Recently, Fagin and coauthors [9,10] have looked at combinatorial games that allow to count the number of quantifiers required to express a property in first-order logic. While these games simplify Adler–Immerman games, they come with a drawback: by design, they implicitly look at how to express properties with first-order formulae in *prenex normal form*, and they are not able to give any bound on the quantifier-free part of such formulae. It seems then not possible to use these types of games in the context of LTL. One could in principle consider translations of LTL formulas into a prenex fragment of S1S. However, since S1S is both more expressive and more succinct than LTL [25], concluding that LTL[F] can be exponentially more succinct than $\mathsf{F}(\mathsf{LTL}[\mathsf{Y},\widetilde{\mathsf{Y}},\mathsf{O},\mathsf{H}])$ will ultimately require analysing structural properties of the quantifier-free part of the S1S formulae.

Closer to the case of LTL are the games on linear orders (implicitly) used by Grohe and Schweikartdt in [14]. These are formula-size games for a fixed number of variables of first-order logic. Using our notation, the method used to derive lower bounds in [14] relies on defining a function ω from terms of the form $\langle A, B\rangle$ to $\mathbb{N}$ that acts as a *sub-additive measure* with respect to the rules of the proof system. For instance, according to the rule Or, ω should satisfy $\omega(\langle A, B\rangle) \leq \omega(\langle A_1, B\rangle) + \omega(\langle A_2, B\rangle)$, whenever $A = A_1 \uplus A_2$. One can use a sub-additive measure ω to conclude that the minimal deduction tree for $\langle A, B\rangle$, if it exists, has size at least $\omega(\langle A, B\rangle)$. When restricted to the objects in Lemma 4, one can show that the function $\omega(\langle A, B\rangle) := |\{(a,b) \in A \times B : a \approx b\}| + 1$ is a sub-additive measure with respect to the rules Atomic, Or, And, Future and Globally (this is implicit in the proof of Lemma 4). However, it is not a sub-additive measure for the rules Next and WeakNext: as stressed in the proof, we might have $\omega(\langle A^{\mathsf{X}}, B^{\mathsf{X}}\rangle) = 1$ even for $\omega(\langle A, B\rangle)$ arbitrary large. This partially explains why the proof of Lemma 4 turned out to be quite involved.

In view of the optimality of the algorithm in [4] (Corollary 1), the main open problem regarding pastification is the optimality of the triply-exponential time procedure given in [7] for the pastification of $\mathsf{LTL}[\mathsf{X}, \mathsf{U}]$ into $\mathsf{F}(\mathsf{pLTL})$. As far as we are aware, no super-polynomial lower bound for this problem is known. Our proof system, properly extended with rules for the until and release operators, might be able to tackle this issue. Obtaining a matching triply-exponential lower bound might however be impossible: when restricted to propositional logic, our proof system is equivalent to the communication games introduced by Karchmer and Wigderson [15]. It is well-known that these games cannot prove super-quadratic lower bounds for formulae sizes, and one should expect similar limitations for temporal logics, albeit with respect to some function that is at least exponential.

LTL *formulae learning.* Motivated by the practical issue of understanding a complex system starting from its execution traces, several recent works study the algorithmic problem of finding an LTL formula separating two finite sets of traces, see e.g. [21,5,24,11,12]. In light of Theorem 3, this problem is equivalent to finding a proof in (possibly variations of) our combinatorial proof system. We believe that this simple observation will turn out to be quite fruitful for both the "combinatorial games" and the "formula learning" communities. From our experience, tools such as the one developed in [21,5,24] are quite useful when studying combinatorial lower bounds, as they can be used to empirically test whether families of structures are difficult to separate, before attempting a formal proof. In our case, we have used the tool in [21] while searching for the structures and formulae we ended up using in Section 5, and discarded several other candidates thanks to the evidences the tool gave us. On the other side of the coin, combinatorial proof systems can be seen as a common foundational framework for all these formulae-learning procedures. With this in mind, we believe that the techniques developed for proving lower bounds in works such as [14] might be of help for improving these procedures, for example using the minimization of a sub-additive measure as a heuristic.

Acknowledgements Luca Geatti and Angelo Montanari acknowledge the support from the 2023 Italian INdAM-GNCS project *"Analisi simbolica e numerica di sistemi ciberfisici"*, ref. no. CUP_E53C22001930001. Angelo Montanari acknowledges the support of the MUR PNRR project FAIR - Future AI Research (PE00000013) funded by the NextGenerationEU. Alessio Mansutti is supported by the César Nombela grant 2023-T1/COM-29001, funded by the Madrid Regional Government, and by the grant PID2022-138072OB-I00, funded by MCIN/AEI/10.13039/501100011033 (FEDER, EU).

References

1. M. Adler and N. Immerman. An n! lower bound on formula size. *ACM Transactions on Computational Logic*, 4(3):296–314, 2003.
2. A. Artale, L. Geatti, N. Gigante, A. Mazzullo, and A. Montanari. Complexity of safety and cosafety fragments of linear temporal logic. In *AAAI'23*, pages 6236–6244, 2023.
3. A. Artale, L. Geatti, N. Gigante, A. Mazzullo, and A. Montanari. LTL over finite words can be exponentially more succinct than pure-past LTL, and vice versa. In *TIME'23*, volume 278, pages 2:1–2:14, 2023.
4. A. Artale, L. Geatti, N. Gigante, A. Mazzullo, and A. Montanari. A singly exponential transformation of LTL[X, F] into pure past LTL. In *KR'23*, pages 65–74, 2023.
5. A. Camacho and S. A. McIlraith. Learning interpretable models expressed in linear temporal logic. In *ICAPS'19*, pages 621–630, 2019.
6. E. Y. Chang, Z. Manna, and A. Pnueli. Characterization of temporal property classes. In *ICALP'92*, pages 474–486, 1992.
7. G. De Giacomo, A. Di Stasio, F. Fuggitti, and S. Rubin. Pure-past linear temporal and dynamic logic on finite traces. In *IJCAI'21*, pages 4959–4965, 2021.
8. G. De Giacomo and M. Y. Vardi. Linear temporal logic and linear dynamic logic on finite traces. In *IJCAI'13*, pages 854–860, 2013.
9. R. Fagin, J. Lenchner, K. W. Regan, and N. Vyas. Multi-structural games and number of quantifiers. In *LICS'21*, pages 1–13, 2021.
10. R. Fagin, J. Lenchner, N. Vyas, and R. R. Williams. On the number of quantifiers as a complexity measure. In *MFCS'22*, pages 48:1–48:14, 2022.
11. M. Fortin, B. Konev, V. Ryzhikov, Y. Savateev, F. Wolter, and M. Zakharyaschev. Unique characterisability and learnability of temporal instance queries. In *KR'22*, 2022.
12. M. Fortin, B. Konev, V. Ryzhikov, Y. Savateev, F. Wolter, and M. Zakharyaschev. Reverse engineering of temporal queries mediated by LTL ontologies. In *IJCAI'23*, pages 3230–3238, 2023.
13. L. Geatti, A. Mansutti, and A. Montanari. Succinctness of Cosafety Fragments of LTL via Combinatorial Proof Systems (extended version). *arXiv, cs.LO/2401.09860*, 2024.
14. M. Grohe and N. Schweikardt. The succinctness of first-order logic on linear orders. *Log. Methods Comput. Sci.*, 1(1:6), 2005.
15. M. Karchmer. *Communication complexity - a new approach to circuit depth.* MIT Press, 1989.
16. O. Kupferman and M. Y. Vardi. Model checking of safety properties. *Formal Methods in System Design*, 19(3):291–314, 2001.
17. O. Lichtenstein, A. Pnueli, and L. Zuck. The glory of the past. In *Workshop on Logic of Programs*, pages 196–218, 1985.
18. F. M. Maggi, M. Montali, and R. Peñaloza. Temporal logics over finite traces with uncertainty. In *AAAI'20*, pages 10218–10225, 2020.
19. Z. Manna and A. Pnueli. *Temporal verification of reactive systems - safety.* Springer, 1995.
20. N. Markey. Temporal logic with past is exponentially more succinct, concurrency column. *Bull. EATCS*, 79:122–128, 2003.
21. D. Neider and I. Gavran. Learning linear temporal properties. In *FMCAD'18*, pages 1–10, 2018.

22. M. Pesic and W. M. P. van der Aalst. A declarative approach for flexible business processes management. In J. Eder and S. Dustdar, editors, *BPM'06*, pages 169–180, 2006.
23. A. Pnueli. The temporal logic of programs. In *FOCS (SFCS'77)*, pages 46–57, 1977.
24. R. Raha, R. Roy, N. Fijalkow, and D. Neider. Scalable anytime algorithms for learning fragments of linear temporal logic. In *TACAS'22*, pages 263–280, 2022.
25. L. J. Stockmeyer and A. R. Meyer. Cosmological lower bound on the circuit complexity of a small problem in logic. *J. ACM*, 49(6):753–784, 2002.
26. M. Y. Vardi and P. Wolper. An automata-theoretic approach to automatic program verification. In *LICS'86*, pages 322–331, 1986.