LTL over Finite Words Can Be Exponentially More Succinct Than Pure-Past LTL, and *vice versa*

Alessandro Artale
Free University of Bozen-Bolzano, Italy

Luca Geatti
University of Udine, Italy

Nicola Gigante
Free University of Bozen-Bolzano, Italy

Andrea Mazzullo
Free University of Bozen-Bolzano, Italy

Angelo Montanari
University of Udine, Italy

Abstract

Linear Temporal Logic over finite traces ($\mathsf{LTL_f}$) has proved itself to be an important and effective formalism in formal verification as well as in artificial intelligence. Pure past $\mathsf{LTL_f}$ (pLTL) is the logic obtained from $\mathsf{LTL_f}$ by replacing each (future) temporal operator by a corresponding past one, and is naturally interpreted at the end of a finite trace. It is known that each property definable in $\mathsf{LTL_f}$ is also definable in pLTL, and *vice versa*. However, despite being extensively used in practice, to the best of our knowledge, there is no systematic study of their *succinctness*.

In this paper, we investigate the succinctness of $\mathsf{LTL_f}$ and pLTL. First, we prove that pLTL can be exponentially more succinct than $\mathsf{LTL_f}$ by showing that there exists a property definable with a pLTL formula of size n such that the size of all $\mathsf{LTL_f}$ formulas defining it is at least *exponential* in n. Then, we prove that $\mathsf{LTL_f}$ can be exponentially more succinct than pLTL as well. This result shows that, although being expressively equivalent, $\mathsf{LTL_f}$ and pLTL are incomparable when succinctness is concerned. In addition, we study the succinctness of $\mathsf{Safety\text{-}LTL}$ (the syntactic safety fragment of LTL over infinite traces) with respect to its canonical form $\mathsf{G}(\mathsf{pLTL})$, whose formulas are of the form $\mathsf{G}(\alpha)$, G being the *globally* operator and α a pLTL formula. We prove that $\mathsf{G}(\mathsf{pLTL})$ can be exponentially more succinct than $\mathsf{Safety\text{-}LTL}$, and that the same holds for the dual cosafety fragment.

2012 ACM Subject Classification Theory of computation → Modal and temporal logics; Theory of computation → Logic and verification

Keywords and phrases Temporal Logic, Succinctness, LTLf, Finite Traces, Pure past LTL

Digital Object Identifier 10.4230/LIPIcs.TIME.2023.2

Funding Luca Geatti and Angelo Montanari acknowledge the support from the 2022 Italian INdAM-GNCS project *"Elaborazione del Linguaggio Naturale e Logica Temporale per la Formalizzazione di Testi"*, ref. no. `CUP_E55F22000270001`. Andrea Mazzullo and Angelo Montanari acknowledge the support of the MUR PNRR project FAIR - Future AI Research (`PE00000013`) funded by the NextGenerationEU. Nicola Gigante acknowledges the support of the PURPLE project, in the context of the AIPlan4EU project's First Open Call for Innovators.

1 Introduction

In this paper, we study the succinctness of Linear Temporal Logic over finite words ($\mathsf{LTL_f}$) with respect to pure past $\mathsf{LTL_f}$ (pLTL) and prove two lower bounds that show the incomparability of $\mathsf{LTL_f}$ and pLTL as far as succinctness is concerned. In addition, we investigate some succinctness properties of the safety and cosafety fragments of Linear Temporal Logic over infinite words (LTL) with respect to their canonical forms (resp., $\mathsf{G}(\mathsf{pLTL})$ and $\mathsf{F}(\mathsf{pLTL})$).

$\mathsf{LTL_f}$ is a modal logic that extends classic Boolean Logic with *temporal modalities* for reasoning about time and it is interpreted over *finite* sequences of states (called traces or words). $\mathsf{LTL_f}$ is extensively used in many areas of Artificial Intelligence (AI), like automated synthesis [10, 12, 23], planning [4–6], and business process management [19, 20]. In last years, also pLTL, the *pure past* version of $\mathsf{LTL_f}$, gained momentum in AI. As a matter of fact, while all properties expressible in $\mathsf{LTL_f}$ are also expressible in pLTL and *vice versa* (pLTL and $\mathsf{LTL_f}$ have been shown to be expressively equivalent [9, 16, 25]), some properties like, e.g., those characterizing planning problems ("*to reach a goal while always obeying to a safety rule*") are more natural and easy to express using past modalities [16]. Moreover, pLTL has been advocated as a suitable declarative, logic programming language [3, 14]. Last but not least, arguably the most important feature of pLTL is enjoying a compilation into deterministic finite automata of singly exponential size [8, 9], a result that cannot be achieved for $\mathsf{LTL_f}$ [11].

In spite of the success of $\mathsf{LTL_f}$ and pLTL, to the best of our knowledge, there is no systematic study of their *succinctness*, that is, the study of which properties (if any) are definable in one logic with formulas of small, polynomial size, but such that all formulas in the other logic would require exponential size or more. The importance of studying succinctness is twofold. On the one hand, it is an important theoretical tool, that joins the study of computational complexity and expressive power (cf. e.g. the work by Hella and Vilander [15], comparing first-order logic with basic modal logic and μ-calculus in terms of succinctness, by means of formula size games). On the other hand, it may help in choosing the right formalism when solving problems like model checking and reactive synthesis.

The main contributions of the paper are the following ones.

First, we prove that pLTL *can be exponentially more succinct than* $\mathsf{LTL_f}$, that is, there exists a family of properties definable with pLTL formulas of size n such that the size of all $\mathsf{LTL_f}$ formulas defining them is at least exponential in n.

Second, by exploiting the fact that each trace recognized by a pLTL formula is the *reverse*[1] of a trace recognized by an $\mathsf{LTL_f}$ formula, we derive that $\mathsf{LTL_f}$ *can be exponentially more succinct than* pLTL as well. This has three important consequences:

1. it shows that, despite being expressively equivalent, $\mathsf{LTL_f}$ and pLTL are *incomparable* when succinctness is concerned;
2. it confirms the conjecture formulated in [2], derived from the complexity gap between the realizability problem of $\mathsf{LTL_f}$, which is 2EXPTIME-complete, and pLTL, which is EXPTIME-complete;
3. it proves that *any* translation from $\mathsf{LTL_f}$ to pLTL (and *vice versa*), for which we only have a triply exponential upper bound [9], has at least an exponential complexity in the size of the initial formula.

Third, we study the succinctness of the syntactic safety fragment of LTL over infinite traces (denoted as Safety-LTL) with respect to its *canonical form* G(pLTL), which is the set of formulas of the form $\mathsf{G}(\alpha)$, where G is the *globally* modality of LTL and α is a pLTL formula [7]. We show that G(pLTL) *can be exponentially more succinct than* Safety-LTL. By a duality argument, we derive the same result for the syntactic cosafety fragment of LTL (coSafety-LTL) and its canonical form (F(pLTL)). Whether Safety-LTL (resp., coSafety-LTL) can be exponentially more succinct than G(pLTL) (resp., F(pLTL)) is, to the best of our knowledge, still an open question.

[1] By "reverse" of a trace σ, we mean the trace obtained by σ considering its last state as the first one.

The paper is organized as follows. In Section 2, we provide the necessary background. Sections 3 and 4 prove, respectively, that pLTL can be exponentially more succinct than $\mathsf{LTL_f}$, and *vice versa*. In Section 5, we show the succinctness of $\mathsf{G}(\mathsf{pLTL})$ and $\mathsf{F}(\mathsf{pLTL})$ with respect to the safety and cosafety fragments of LTL, respectively. We conclude with Section 6, where we recap the results of the paper and we point out some future research directions.

2 Background

In this section, we give the necessary background on linear-time temporal logic and finite-state automata.

2.1 Linear-time Temporal Logic

Given a set Σ of proposition letters, an $\mathsf{LTL{+}P}$ *formula* ϕ is generated as follows:

$$\begin{aligned} \phi \coloneqq\ & p \mid \neg p \mid \phi \vee \phi \mid \phi \wedge \phi && \text{Boolean connectives}\\ & \mid \mathsf{X}\phi \mid \widetilde{\mathsf{X}}\phi \mid \phi \,\mathsf{U}\, \phi \mid \phi \,\mathsf{R}\, \phi && \text{future modalities}\\ & \mid \mathsf{Y}\phi \mid \widetilde{\mathsf{Y}}\phi \mid \phi \,\mathsf{S}\, \phi \mid \phi \,\mathsf{T}\, \phi && \text{past modalities} \end{aligned}$$

where $p \in \Sigma$, and we call: X, *next*; $\widetilde{\mathsf{X}}$, *weak next*; U, *until*; R, *releases*; Y, *yesterday*; $\widetilde{\mathsf{Y}}$, *weak yesterday*; S, *since*; T, *triggers*. Note that, *w.l.o.g.*, our definition of $\mathsf{LTL{+}P}$ considers formulas already in Negation Normal Form (NNF), that is, negations are applied only to proposition letters. For any formula ϕ, the *size* of ϕ (denoted with $|\phi|$) is the size of the (smallest) syntax tree of ϕ.

Let $\sigma \in (2^\Sigma)^+ \cup (2^\Sigma)^\omega$ be a *word over* 2^Σ (or *trace over* 2^Σ). We define the *length* of σ as $|\sigma| = n$, if $\sigma = \langle \sigma_0, \dots, \sigma_{n-1} \rangle \in (2^\Sigma)^+$ (in this case we say that σ is a *finite* trace); or $|\sigma| = \omega$, if $\sigma \in (2^\Sigma)^\omega$ (in this case we say that σ is an *infinite* trace). We call any subset of $(2^\Sigma)^*$ a *language of finite words over* 2^Σ. Similarly, a *language of infinite words over* 2^Σ is any subset of $(2^\Sigma)^\omega$.

The *satisfaction* of an $\mathsf{LTL{+}P}$ formula ϕ by σ at time $0 \le i < |\sigma|$, denoted by $\sigma, i \models \phi$, is defined as follows:

- $\sigma, i \models p$ iff $p \in \sigma_i$;
- $\sigma, i \models \neg p$ iff $p \notin \sigma_i$;
- $\sigma, i \models \phi_1 \vee \phi_2$ iff $\sigma, i \models \phi_1$ or $\sigma, i \models \phi_2$;
- $\sigma, i \models \phi_1 \wedge \phi_2$ iff $\sigma, i \models \phi_1$ and $\sigma, i \models \phi_2$;
- $\sigma, i \models \mathsf{X}\phi$ iff $i + 1 < |\sigma|$ and $\sigma, i + 1 \models \phi$;
- $\sigma, i \models \widetilde{\mathsf{X}}\phi$ iff either $i + 1 = |\sigma|$ or $\sigma, i + 1 \models \phi$;
- $\sigma, i \models \mathsf{Y}\phi$ iff $i > 0$ and $\sigma, i - 1 \models \phi$;
- $\sigma, i \models \widetilde{\mathsf{Y}}\phi$ iff either $i = 0$ or $\sigma, i - 1 \models \phi$;
- $\sigma, i \models \phi_1 \,\mathsf{U}\, \phi_2$ iff there exists $i \le j < |\sigma|$ such that $\sigma, j \models \phi_2$, and $\sigma, k \models \phi_1$ for all k, with $i \le k < j$;
- $\sigma, i \models \phi_1 \,\mathsf{S}\, \phi_2$ iff there exists $j \le i$ such that $\sigma, j \models \phi_2$, and $\sigma, k \models \phi_1$ for all k, with $j < k \le i$;
- $\sigma, i \models \phi_1 \,\mathsf{R}\, \phi_2$ iff either $\sigma, j \models \phi_2$ for all $i \le j < |\sigma|$, or there exists $i \le k < |\sigma|$ such that $\sigma, k \models \phi_1$ and $\sigma, j \models \phi_2$ for all $i \le j \le k$;
- $\sigma, i \models \phi_1 \,\mathsf{T}\, \phi_2$ iff either $\sigma, j \models \phi_2$ for all $0 \le j \le i$, or there exists $k \le i$ such that $\sigma, k \models \phi_1$ and $\sigma, j \models \phi_2$ for all $i \ge j \ge k$.

We say that σ is a *model* of ϕ (written as $\sigma \models \phi$) iff $\sigma, 0 \models \phi$. The *language of infinite* (resp., *finite*) *traces* of ϕ, denoted by $\mathcal{L}(\phi)$, is the set of traces $\sigma \in (2^\Sigma)^\omega$ (resp., $\sigma \in (2^\Sigma)^+$) such that $\sigma \models \phi$.

We use the standard shortcuts for $\top \coloneqq p \vee \neg p$, $\bot \coloneqq p \wedge \neg p$ (for some $p \in \Sigma$) and other temporal operators: $\mathsf{F}\phi \coloneqq \top \mathbin{\mathsf{U}} \phi$ (*eventually*), $\mathsf{G}\phi \coloneqq \bot \mathbin{\mathsf{R}} \phi$ (*globally*), $\mathsf{O}\phi \coloneqq \top \mathbin{\mathsf{S}} \phi$ (*once*), and $\mathsf{H}\phi \coloneqq \bot \mathbin{\mathsf{T}} \phi$ (*historically*).

From now on, given a linear-time temporal logic $\mathbb{L}$, with some abuse of notation, we denote with $\mathbb{L}$ also the set of formulas of $\mathbb{L}$. A *pure future* (resp., *pure past*) *formula* is an LTL+P formula without occurrences of past (resp., future) modalities. We denote by LTL (resp., pLTL) the set of pure future (resp., pure past) formulas. In the following, we use the subscript f to denote a logic interpreted on finite traces. Thus, e.g., with LTL$_\mathsf{f}$ we denote LTL interpreted on finite traces. Note that, if ϕ belongs to pLTL (*i.e.* pure past fragment of LTL+P), then we interpret ϕ only on *finite words* and we say that $\sigma \in (2^\Sigma)^+$ is a model of ϕ if and only if $\sigma, |\sigma| - 1 \models \phi$, that is, each ϕ in pLTL is interpreted at the *last* state of a finite word. It holds that LTL$_\mathsf{f}$ and pLTL are expressively equivalent.

▶ **Proposition 1** (see [9, 16, 25]). *For any alphabet Σ and for any language $\mathcal{L} \subseteq \Sigma^\omega$, it holds that: there exists a formula $\phi \in$ LTL$_\mathsf{f}$ such that $\mathcal{L}(\phi) = \mathcal{L}$ iff there exists a formula $\phi' \in$ pLTL such that $\mathcal{L}(\phi') = \mathcal{L}$.*

In the following, we denote by Safety-LTL (also called the syntactic safety fragment of LTL) the set of LTL formulas whose temporal operators are restricted to $\widetilde{\mathsf{X}}$, G, and R [7, 21, 24]. Similarly, we define coSafety-LTL (the syntactic cosafety fragment of LTL) as the set of LTL formulas whose temporal operators are restricted to X, F, and U. Finally, we denote by G(pLTL) (resp., F(pLTL)) the set of LTL+P formulas of the form $\mathsf{G}\alpha$ (resp., $\mathsf{F}\alpha$), with $\alpha \in$ pLTL. A fundamental theorem by Chang, Manna, and Pnueli [7], based on the results found by Zuck [25], establishes the expressive equivalence of Safety-LTL with G(pLTL), and of coSafety-LTL and F(pLTL), when interpreted over infinite traces.

We now define what it means, for two linear-time temporal logics $\mathbb{L}$ and $\mathbb{L}'$, that $\mathbb{L}$ *can be exponentially more succinct than* $\mathbb{L}'$. We use the Ω-notation $f(n) \in \Omega(g(n))$ to denote that the function f is asymptotically bounded from below by g. Similarly, we use the $\mathcal{O}$-notation $f(n) \in \mathcal{O}(g(n))$ to denote that f is asymptotically bounded from above by g.

▶ **Definition 2.** *Given two linear-time temporal logics $\mathbb{L}$ and $\mathbb{L}'$, we say that $\mathbb{L}$* can be exponentially more succinct than $\mathbb{L}'$ over infinite trace *(resp.,* over finite traces*) iff there exists an alphabet Σ and a family of languages $\{\mathcal{L}_n\}_{n>0} \subseteq (2^\Sigma)^\omega$ (resp., $\{\mathcal{L}_n\}_{n>0} \subseteq (2^\Sigma)^*$) such that, for any $n > 0$:*

- *there exists a formula $\phi \in \mathbb{L}$ over Σ such that its language over infinite traces (resp., over finite traces) is $\mathcal{L}_n$ and $|\phi| \in \mathcal{O}(n)$; and*
- *for all formulas $\phi' \in \mathbb{L}'$ over Σ, if the language of ϕ' over infinite traces (resp., finite traces) is $\mathcal{L}_n$, then $|\phi'| \in 2^{\Omega(n)}$.*

2.2 Finite-state Automata

A *Nondeterministic Finite Automaton* (NFA, for short) is a tuple $A = (2^\Sigma, Q, I, \Delta, F)$, where: 2^Σ is a finite (nonempty) *alphabet*; Q is a finite set of *states*; $I \subseteq Q$ is the set of *initial states*; $\Delta \subseteq Q \times 2^\Sigma \times Q$ is the *transition relation*; $F \subseteq Q$ is the set of *final states*. We define the *size of* $\mathcal{A}$, denoted with $|\mathcal{A}|$, as the number of its states ($|Q|$).

A *run* π of $\mathcal{A}$ over the word $\sigma = \langle \sigma_0, \sigma_1, \ldots, \sigma_{n-1} \rangle \in (2^\Sigma)^*$ is a finite sequence of states $\pi = \langle q_0, q_1, \ldots, q_n \rangle$ such that $(q_i, \sigma_i, q_{i+1}) \in \Delta$, for all $0 \leq i < n-1$. A run $\pi = \langle q_0, q_1, \ldots, q_n \rangle$ is *accepting* if q_n is a final state of $\mathcal{A}$, that is $q_n \in F$.

Given an NFA $\mathcal{A} = (2^\Sigma, Q, I, \Delta, F)$, a word $\sigma \in (2^\Sigma)^*$ is *accepted by* $\mathcal{A}$ iff there exists an accepting run of $\mathcal{A}$ over σ. The *language of* $\mathcal{A}$, denoted with $\mathcal{L}(\mathcal{A})$, is the set (finite) words accepted by $\mathcal{A}$.

For each $\mathsf{LTL_f{+}P}$ formula ϕ of size n over the set of proposition letters Σ, we can effectively build an NFA whose language is exactly $\mathcal{L}(\phi)$ and its size is at most exponential in n [11].

▶ **Proposition 3** (see [11]). *For any formula ϕ of* $\mathsf{LTL_f{+}P}$ *of size n, there exists an* NFA $\mathcal{A}$ *such that $\mathcal{L}(\phi) = \mathcal{L}(\mathcal{A})$ and $|\mathcal{A}| \in 2^{\mathcal{O}(n)}$.*

3 pLTL can be exponentially more succinct than LTL$_\mathsf{f}$

In this section, we prove the first main result of this paper, *i.e.* that pLTL can be exponentially more succinct than $\mathsf{LTL_f}$.

Let $\Sigma = \{p_0, p_1, \dots, p_n\}$ be a finite set of proposition letters. Consider the following family of languages over the alphabet 2^Σ, where $n > 0$.

$$A_n \coloneqq \{\sigma \in (2^\Sigma)^+ \mid \exists k > 0 \,.\, (\bigwedge_{i=0}^{n} (p_i \in \sigma_k \leftrightarrow p_i \in \sigma_0))\} \tag{1}$$

For any $n > 0$, the language A_n is the set of finite words over 2^Σ containing a position which agrees with the initial state on the evaluation of all proposition letters in Σ (cf. Figure 1).

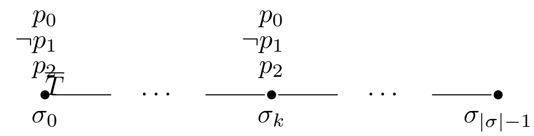

Figure 1 Example of a word σ in A_2.

We shall prove that all formulas of $\mathsf{LTL_f}$ defining A_n are at least of size exponential in n. Conversely, as shown by the following lemma, A_n can be expressed in pLTL with formulas of linear size in n, for any $n > 0$.

▶ **Lemma 4.** *For any $n > 0$, there exists a formula $\phi \in$ pLTL such that $\mathcal{L}(\phi) = A_n$ and $|\phi| \in \mathcal{O}(n)$.*

Proof. For any $n > 0$, we define the formula ϕ_{A_n} as

$$\mathsf{O}(\bigwedge_{i=0}^{n} (p_i \leftrightarrow \mathsf{YO}(\widetilde{\mathsf{Y}}\bot \wedge p_i)))$$

Note the crucial role of the *weak yesterday* operator, and in particular of the subformula $\widetilde{\mathsf{Y}}\bot$, for hooking the initial state of a word. We prove that $\mathcal{L}(\phi_{A_n}) = A_n$. For any $\sigma \in (2^\Sigma)^+$ and for any $n > 0$, it holds that $\sigma \in A_n$ if and only if $\exists k > 0 \,.\, \bigwedge_{i=1}^{n}(\sigma_k \models p_i \leftrightarrow \sigma_0 \models p_i$. This, in turn, is equivalent to $\exists k < |\sigma| \,.\, (k \neq 0 \wedge \bigwedge_{i=1}^{n}(\sigma_k \models p_i \leftrightarrow \sigma_0 \models p_i))$ and thus to $\exists k < |\sigma| \,.\, \bigwedge_{i=1}^{n}(\sigma_k \models p_i \leftrightarrow (\exists h \,.\, (h < k \wedge h = 0 \wedge \sigma_h \models p_i)))$. Therefore, $\sigma \models \mathsf{O}(\bigwedge_{i=0}^{n}(p_i \leftrightarrow \mathsf{YO}(\widetilde{\mathsf{Y}}\bot \wedge p_i)))$. Clearly, $|\phi_{A_n}| \in \mathcal{O}(n)$. ◀

To prove that A_n is not expressible in $\mathsf{LTL_f}$ with formulas of size less than $2^{\Omega(n)}$ (for any $n > 0$), we make use of an auxiliary family of languages. For each $n > 0$, we define the language B_n over the alphabet 2^Σ with $\Sigma = \{p_0, p_1, \dots, p_n\}$ as follows:

$$B_n \coloneqq \{\sigma \in (2^\Sigma)^+ \mid \exists h \geq 0 \,.\, \exists k > h \,.\, (\bigwedge_{i=0}^{n} (p_i \in \sigma_k \leftrightarrow p_i \in \sigma_h))\}$$

For any $n > 0$, B_n is the set of finite words over 2^Σ containing two (distinct) positions that agree on the interpretation of all the proposition letters in Σ (cf. Figure 4). Clearly, $A_n \subseteq B_n$, for any $n > 0$.

Figure 2 Example of a word σ in B_2.

We now show that, if A_n was expressible in $\mathsf{LTL_f}$ in space less than exponential in n, then the property B_n would be expressible in $\mathsf{LTL_f}$ in space less than exponential as well.

▶ **Lemma 5.** *If there exists a formula of $\mathsf{LTL_f}$ for A_n of size less than exponential in n, then there exists a formula of $\mathsf{LTL_f}$ for B_n of size less than exponential in n.*

Proof. Let ψ_{A_n} be a formula of $\mathsf{LTL_f}$ for A_n of size less than exponential in n. Consider the formula $\mathsf{F}(\psi_{\mathsf{A}_n})$: we prove that its language is exactly B_n. For any $\sigma \in (2^\Sigma)^+$ and for any $n > 0$, it holds that $\sigma \models \mathsf{F}(\psi_{\mathsf{A}_n})$ iff $\exists k \geq 0 \,.\, \sigma_{[k,-]} \models \psi_{A_n}$, where $\sigma_{[k,-]}$ is the suffix of σ starting from i. This means: $\exists k \geq 0 \,.\, \exists h > k \,.\, (\bigwedge_{i=0}^{n}(\sigma_k \models p_i \leftrightarrow \sigma_h \models p_i))$. Equivalently, $\sigma \in B_n$. Moreover $\mathsf{F}(\psi_{\mathsf{A}_n})$ belongs to $\mathsf{LTL_f}$ and it is of size less than exponential in n. ◀

We show that there cannot exist formulas of $\mathsf{LTL_f}$ (and, in general, of $\mathsf{LTL_f}$+P) defining B_n whose size is less than exponential in n. In order to prove it, we first show that any NFA accepting B_n is of size at least *doubly exponential* in n.

▶ **Lemma 6.** *For any $n > 0$ and for any NFA $\mathcal{A}$ over the alphabet 2^Σ, if $\mathcal{L}(\mathcal{A}) = B_n$ then $|\,\mathcal{A}\,| \in 2^{2^{\Omega(n)}}$.*

Proof. Let $n > 0$ and let $\langle a_0, \ldots, a_{2^n-1}\rangle$ be any permutation of the 2^n subsets of $\{p_1, \ldots, p_n\}$ (note that this set does not include the proposition letter $p_0 \in \Sigma$). Let K be any subset of $\{0, \ldots, 2^n - 1\}$ and let $\overline{K}$ be the complement set of K. We define b_i^K in this way: $b_i^K \coloneqq a_i$, if $i \in \overline{K}$; and $b_i^K \coloneqq a_i \cup \{p_0\}$, otherwise. We define σ_K as the sequence $\langle b_0^K, b_1^K, \ldots, b_{2^n-1}^K\rangle$.

Suppose by contradiction that there exists an NFA $\mathcal{A}$ for B_n of size less than doubly exponential in n. Consider the words $\sigma_K \cdot \sigma_K$ (obtained by concatenating σ_K with itself), $\sigma_{\overline{K}} \cdot \sigma_{\overline{K}}$ (the concatenation of $\sigma_{\overline{K}}$ with itself), and $\sigma_K \cdot \sigma_{\overline{K}}$ (the concatenation of σ_K with $\sigma_{\overline{K}}$). By construction, both $\sigma_K \cdot \sigma_K$ and $\sigma_{\overline{K}} \cdot \sigma_{\overline{K}}$ contain (at least) two positions that agree on the interpretation of all symbols in Σ and thus they both belong to B_n, while $\sigma_K \cdot \sigma_{\overline{K}}$ contains no such positions and so it does *not* belong to B_n. Therefore, for any $K \subseteq \{0, \ldots, 2^n - 1\}$:

1. $\sigma_K \cdot \sigma_K$ is accepted by $\mathcal{A}$;
2. $\sigma_{\overline{K}} \cdot \sigma_{\overline{K}}$ is accepted by $\mathcal{A}$;
3. $\sigma_K \cdot \sigma_{\overline{K}}$ is *not* accepted by $\mathcal{A}$.

Now let π (resp., π') be any *accepting* run of $\mathcal{A}$ over the word $\sigma_K \cdot \sigma_K$ (resp., $\sigma_{\overline{K}} \cdot \sigma_{\overline{K}}$). Let q (resp., q') be the 2^n-th state of π (resp., π'). Suppose that $q = q'$ and let π'' be the sequence obtained by appending the suffix of π' starting from its 2^n-th state to the prefix of π of length $2^n - 1$, *i.e.*: $\pi'' \coloneqq \langle \pi_0, \ldots, \pi_{2^n-1}, \pi'_{2^n}, \pi'_{2^n+1}, \ldots\rangle$. By construction, π'' is an *accepting run* of the automaton $\mathcal{A}$ over the word $\sigma_K \cdot \sigma_{\overline{K}}$, which is a contradiction. Therefore, the 2^n-th states of π and π' must be distinct. This means that the automaton $\mathcal{A}$ has to contain at least a state for choice of $K \subseteq \{0, \ldots, 2^n - 1\}$. Since there are 2^{2^n} of such possible choices, this means that $\mathcal{A}$ has to contain at least $2^{2^{\Omega(n)}}$ states. ◀

By exploiting the singly exponential translation of $\mathsf{LTL_f{+}P}$ formulas into equivalent NFAs (Proposition 3), we can prove that (for any $n > 0$) the language B_n is not expressible in $\mathsf{LTL_f{+}P}$ (and, in particular, in $\mathsf{LTL_f}$) in space less than exponential.

▶ **Lemma 7.** *For any formula $\phi \in \mathsf{LTL_f{+}P}$, if $\mathcal{L}(\phi) = B_n$ then $|\phi| \in 2^{\Omega(n)}$.*

Proof. Suppose by contradiction that this does not hold, *i.e.* there exists a formula $\phi \in \mathsf{LTL_f{+}P}$ such that $\mathcal{L}(\phi) = B_n$ and $|\phi|$ is less than exponential in n. Then, by Proposition 3, it holds that there exists an NFA $\mathcal{A}$ such that $\mathcal{L}(\mathcal{A}) = B_n$ and $|\mathcal{A}|$ is *less than doubly exponential* in n, which is a contradiction with Lemma 6. ◀

Directly from Lemmas 5 and 7, it follows that the family of languages A_n cannot be expressed in $\mathsf{LTL_f}$ with formulas of size less than exponential in n.

▶ **Theorem 8.** *For any $n > 0$ and for any formula $\phi \in \mathsf{LTL_f}$, if $\mathcal{L}(\phi) = A_n$ then $|\phi| \in 2^{\Omega(n)}$.*

The following corollary is a direct consequence of Lemma 4 and Theorem 8.

▶ **Corollary 9.** pLTL *can be exponentially more succinct than* $\mathsf{LTL_f}$*.*

Comparison with Markey's proof about LTL+P and LTL

In [18], Markey proves that LTL+P can be exponentially more succinct than LTL. In particular, he exploits the result by Etessami, Vardi, and Wilke [13] that there are no Büchi automata of size less than doubly exponential for the family of languages I_n (for all $n > 0$), defined as the language of *infinite* traces in which any two positions that agree on $p_1, \ldots, p_n$, agree also on p_0.

One could, in principle, use I_n interpreted over finite trace (let us call it $I_n^{<\omega}$) to prove that any NFA recognizing $I_n^{<\omega}$ is at least of doubly exponential size in n, and use it as a base for proving that pLTL can be exponentially more succinct than $\mathsf{LTL_f}$. This would require to restate and reprove the theorem by Etessami, Vardi, and Wilke [13] to work over finite traces. While we believe this is possible, we followed a simpler (and more useful) path by showing that there is another family of properties, in our case B_n (which is arguably simpler than I_n and $I_n^{<\omega}$), for which each NFA explodes double-exponentially.

4 $\mathsf{LTL_f}$ can be exponentially more succinct than pLTL

In this section, we show the second main result of this paper, *i.e.* that $\mathsf{LTL_f}$ can be exponentially more succinct than pLTL. Together with Corollary 9, this shows that $\mathsf{LTL_f}$ and pLTL, despite being expressively equivalent, are *incomparable* when succinctness is considered.

4.1 The Reverse Lemma

We first define the notions of *reverse language* and *reverse logic*. Given an alphabet Σ and a language $\mathcal{L} \subseteq (2^\Sigma)^+$ of finite words over 2^Σ, we define the *reverse language* of $\mathcal{L}$ as the set:

$$\mathcal{L}^- = \{\sigma' \in (2^\Sigma)^+ \mid \sigma'_i = \sigma_{n-i}, \text{ for } \sigma = \sigma_0 \ldots \sigma_n \in \mathcal{L} \text{ and } 0 \leq i \leq n\}.$$

We then define *reverse logics* as follows.

▶ **Definition 10** (Reverse Logics). *Given two linear-time temporal logics $\mathbb{L}$ and $\mathbb{L}^-$, we say that $\mathbb{L}^-$ is a* reverse logic *of $\mathbb{L}$ iff:*

1. *for any formula $\phi \in \mathbb{L}$, there is a formula $\phi' \in \mathbb{L}^-$ so that $\mathcal{L}(\phi) = \mathcal{L}(\phi')^-$ and $|\phi'| = |\phi|$;*
2. *for any formula $\phi' \in \mathbb{L}^-$, there is a formula $\phi \in \mathbb{L}$ so that $\mathcal{L}(\phi') = \mathcal{L}(\phi)^-$ and $|\phi| = |\phi'|$.*

Clearly, being a reverse logic is a symmetric property: $\mathbb{L}$ is a reverse logic of $\mathbb{L}^-$ iff $\mathbb{L}^-$ is a reverse logic of $\mathbb{L}$.

As an example, consider the logic pLTL and any formula $\phi \in \mathsf{pLTL}$. By replacing in ϕ the temporal operators Y, $\widetilde{\mathsf{Y}}$, S, and T with X, $\widetilde{\mathsf{X}}$, U, and R, respectively, one obtains a formula ϕ' such that: (i) it belongs to $\mathsf{LTL_f}$; (ii) its size is $|\phi|$; (iii) it is such that $\mathcal{L}(\phi) = \mathcal{L}(\phi')^-$. Therefore, $\mathsf{LTL_f}$ is a reverse logic of pLTL, and *vice versa*.

The next lemma proves that, for any two linear-time temporal logics $\mathbb{L}$ and $\mathbb{L}^-$ such that $\mathbb{L}$ is a reverse logic of $\mathbb{L}^-$, if a language $\mathcal{L}$ with a compact definition in $\mathbb{L}$ is not succinctly definable in $\mathbb{L}^-$, then $\mathcal{L}^-$ (*i.e.*, the reverse language of $\mathcal{L}$) is compactly definable in $\mathbb{L}^-$, but its definitions exponentially blow-up in $\mathbb{L}$.

▶ **Lemma 11** (Reverse Lemma)**.** *Let $\mathbb{L}$ and $\mathbb{L}^-$ be two linear-time temporal logics such that $\mathbb{L}^-$ is a reverse logic of $\mathbb{L}$. Moreover, let $\phi \in \mathbb{L}$ be such that, for every $\psi \in \mathbb{L}^-$, $\mathcal{L}(\psi) = \mathcal{L}(\phi)$ implies $|\psi| \in 2^{\Omega(|\phi|)}$. Then, for some $\phi' \in \mathbb{L}^-$, we have: (i) $\mathcal{L}(\phi') = \mathcal{L}(\phi)^-$; (ii) $|\phi'| = |\phi|$; and (iii) for every $\psi \in \mathbb{L}$, $\mathcal{L}(\psi) = \mathcal{L}(\phi')$ implies $|\psi| \in 2^{\Omega(|\phi'|)}$.*

Proof. Suppose that $\phi \in \mathbb{L}$ is a formula such that, for any $\psi \in \mathbb{L}^-$, if $\mathcal{L}(\psi) = \mathcal{L}(\phi)$ then $|\psi| \in 2^{\Omega(|\phi|)}$. Now suppose by contradiction that for any formula $\phi' \in \mathbb{L}'$, such that $\mathcal{L}(\phi') = \mathcal{L}(\phi)^-$ and $|\phi'| = |\phi|$, there exists a formula $\psi \in \mathbb{L}$ such that $\mathcal{L}(\psi) = \mathcal{L}(\phi')$ and $|\psi|$ is sub-exponential in $|\phi'|$. Since $\mathbb{L}^-$ is a reverse logic of $\mathbb{L}$, this means that for any formula $\phi' \in \mathbb{L}^-$, such that $\mathcal{L}(\phi') = \mathcal{L}(\phi)^-$ and $|\phi'| = |\phi|$, there exists a formula $\phi_R \in \mathbb{L}$ and a formula $\psi_R \in \mathbb{L}^-$ such that:

- $\mathcal{L}(\phi_R)^- = \mathcal{L}(\phi')$ and thus $\mathcal{L}(\phi_R) = \mathcal{L}(\phi)$;
- $\mathcal{L}(\psi_R)^- = \mathcal{L}(\phi')$ and thus $\mathcal{L}(\psi_R) = \mathcal{L}(\phi)$;
- $|\phi_R| = |\phi'|$ and thus $|\phi_R| = |\phi|$;
- $|\psi_R| = |\psi|$ and thus $|\psi_R|$ is sub-exponential in $|\phi_R|$.

It follows that for any $\phi_R \in \mathbb{L}$ there exists $\psi_R \in \mathbb{L}'$ such that $\mathcal{L}(\phi_R) = \mathcal{L}(\psi_R)$ and $|\psi_R|$ is sub-exponential in ϕ_R. But this is a contradiction with the hypothesis. Therefore, it has to hold that there exists a formula $\phi' \in \mathbb{L}'$, such that $\mathcal{L}(\phi') = \mathcal{L}(\phi)^-$ and $|\phi'| = |\phi|$ and, for all $\psi \in \mathbb{L}$, if $\mathcal{L}(\psi) = \mathcal{L}(\phi')$ then $|\psi| \in 2^{\Omega(|\phi'|)}$. ◀

From Lemma 11, one obtains a concrete family of languages that are definable with $\mathsf{LTL_f}$ formulas of polynomial size but such that any pLTL formula for them requires at least an exponential amount of space. In particular, for any $n > 0$, recall A_n from the previous section, and consider A_n^- (cf. Figure 3):

$$A_n^- \coloneqq \{\sigma \in (2^\Sigma)^+ \mid \exists k < |\sigma| - 1 \,.\, (\bigwedge_{i=0}^{n} (p_i \in \sigma_k \leftrightarrow p_i \in \sigma_{|\sigma|-1}))\}$$

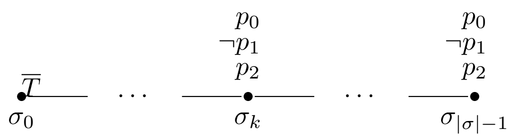

Figure 3 Example of a word σ in A_2^-.

For each $n > 0$, A_n^- can be expressed in $\mathsf{LTL_f}$ in space linear in n with the formula

$$\mathsf{F}(\bigwedge_{i=0}^{n} (p_i \leftrightarrow \mathsf{XF}(\widetilde{\mathsf{X}}\bot \wedge p_i))).$$

However, since $\mathsf{LTL_f}$ is a reverse logic of pLTL, by Lemma 11 every formula of pLTL for A_n^- requires an amount of space at least exponential in n. This leads directly to the following.

▶ **Theorem 12.** *For any $n > 0$ and for any formula $\phi \in$ pLTL, if $\mathcal{L}(\phi) = A_n^-$ then $|\phi| \in 2^{\Omega(n)}$.*

▶ **Corollary 13.** LTL$_\mathsf{f}$ *can be exponentially more succinct than* pLTL*.*

4.2 Some meaningful implications of the incomparability

We have shown that the logics LTL$_\mathsf{f}$ and pLTL, despite being expressively equivalent (Proposition 1), are incomparable when succinctness is considered. Here below, we point out some implications of this incomparability that are worth discussing.

Succinctness and Realizability

Realizability is the problem of establishing whether there is a strategy implementing a given formula. That is, given a formula $\phi \in$ LTL$_\mathsf{f}$ (resp., $\phi \in$ pLTL) over a set of variables $\mathcal{C} \cup \mathcal{U}$ (with $\mathcal{C}$ and $\mathcal{U}$ sets of *controllable* and *uncontrollable* variables, respectively), the realizability problem of LTL$_\mathsf{f}$ (resp., pLTL) is the problem of establishing whether there exists a strategy $s : (2^{\mathcal{U}})^+ \to 2^{\mathcal{C}}$ such that, for all sequences $\langle \mathsf{U}_0, \mathsf{U}_1, \ldots \rangle \in (2^{\mathcal{U}})^+$, it holds that there exists $k \in \mathbb{N}$ so that the prefix from 0 up to k of $\langle \mathsf{U}_0 \cup s(\langle \mathsf{U}_0 \rangle), \mathsf{U}_1 \cup s(\langle \mathsf{U}_0, \mathsf{U}_1 \rangle), \ldots \rangle$ is a model of ϕ.

Despite having the same expressive power, LTL$_\mathsf{f}$ and pLTL have different complexity for the realizability problem: while LTL$_\mathsf{f}$ realizability is 2EXPTIME-complete [12], pLTL realizability is EXPTIME-complete [2]. This is due to the fact that, starting from any LTL$_\mathsf{f}$ formula ϕ of size n, it is not possible to construct a *Deterministic Finite Automaton* (DFA) recognizing $\mathcal{L}(\phi)$ of singly exponential size in n, whereas for pLTL formulas this is possible, thanks to the fact that "since past already happened", there is no need to introduce nondeterminism [2,8,9].

In [2], the exponential gap between the two complexities, and the fact that LTL$_\mathsf{f}$ and pLTL are expressively equivalent, led to the conjecture that any translation from LTL$_\mathsf{f}$ formulas to equivalent pLTL ones requires at least an exponential blowup in the size of the resulting formulas. The results proved in this paper (in particular Theorem 12) *confirm* this conjecture: any translation in pLTL of the LTL$_\mathsf{f}$ formula $\mathsf{F}(\bigwedge_{i=0}^{n}(p_i \leftrightarrow \mathsf{XF}(\widetilde{\mathsf{X}}\bot \wedge p_i)))$, which defines the language A_n^-, requires at least an exponential blowup.

Succinctness helps in choosing the most convenient formalism for realizability

The succinctness results between LTL$_\mathsf{f}$ and pLTL can help in choosing the right formalism to express a property when the time complexity of realizability is considered. As a matter of fact, consider the family of languages A_n (Equation (1)), and suppose one wants to solve the realizability problem for A_n, for a given partition of the variables $p_0, \ldots, p_n$ into controllable and uncontrollable. There are two possibilities:

(i) either formalize A_n in pLTL (in linear size) and use pLTL realizability algorithms (which are singly exponential in the worst case);

(ii) or formalize the language in LTL$_\mathsf{f}$ (with at least an exponential blowup, by Theorem 8) and use LTL$_\mathsf{f}$ realizability algorithms (which are doubly exponential in the worst case).

While the former point requires only a *singly exponential* amount of time in the worst case, the latter requires a *triply exponential* amount of time, in the worst case. This shows how the results on the succinctness of LTL$_\mathsf{f}$ and pLTL can tremendously help choosing the best performing algorithm.

The fact that LTL$_\mathsf{f}$ can be exponentially more succinct than pLTL has an important implication as well. The realizability problem for the family of language A_n^- has the same worst-case time complexity (doubly exponential in n) irrespectively of whether we choose to formalize A_n^- in LTL$_\mathsf{f}$ or we choose pLTL as the target formalism. In other words, the family of languages A_n^- cancels out the advantages of the past in realizability.

Translation of LTL$_f$ into pLTL

Recall that LTL$_f$ and pLTL are expressively equivalent (Proposition 1). To the best of our knowledge, the most efficient translation of LTL$_f$ into pLTL is the one reported in [9] that performs the following steps:

1. build the corresponding NFA for the initial formula;
2. determinize the NFA into a DFA;
3. build a pLTL formula from the DFA using the Krohn-Rhodes cascaded decomposition [17];

Since all three steps may introduce an exponential blow up in the worst case, the whole translation is *triply exponential* in the size of the initial formula. Maler and Pnueli prove that this translation (in particular the third step) has an exponential lower bound [17]. In this respect, Corollary 13 proves that *any* translation from LTL$_f$ to pLTL (not only the above one) has at least an exponential lower bound.

5 Succinctness of safety and cosafety fragments of LTL+P

In this section, we show our last main results, *i.e.* that, when interpreted over infinite traces, G(pLTL) can be exponentially more succinct than Safety-LTL, and that F(pLTL) can be exponentially more succinct than coSafety-LTL.

5.1 G(pLTL) can be exponentially more succinct than Safety-LTL

The proof of this case follows from the result by Markey that LTL+P can be exponentially more succinct than LTL, when interpreted over infinite traces [18]. In the following, we show the details of the proof.

Let $\Sigma = \{p_0, \ldots, p_n\}$ be a set finite set of proposition symbols. Consider the family of languages M_n over the alphabet 2^Σ proposed by Markey in [18]: for each $n > 0$, M_n comprises all and only those infinite traces in which any position of the trace that agrees on $p_1, \ldots, p_n$ with the initial state also agrees on p_0. Formally, for each $n > 0$, we define:

$$M_n \coloneqq \{\sigma \in (2^\Sigma)^\omega \mid \forall k > 0 (\forall i, 1 \leq i \leq n\, (p_i \in \sigma_k \leftrightarrow p_i \in \sigma_0) \leftrightarrow (p_0 \in \sigma_k \leftrightarrow p_0 \in \sigma_0))\}$$

In [18], Markey proves that, for any $n > 0$, any formula of LTL expressing M_n is at least of size exponential in n. Since Safety-LTL is a proper subfragment of LTL (*i.e.* each Safety-LTL formula is also an LTL formula), it follows that, for any $n > 0$, any formula of Safety-LTL expressing M_n is at least of size exponential in n.

▶ **Lemma 14.** *For any $n > 0$ and any formula $\phi \in$ Safety-LTL, if $\mathcal{L}(\phi) = M_n$ then $|\phi| \in 2^{\Omega(n)}$.*

However, for each $n > 0$, there is a formula in G(pLTL) of size linear in n expressing M_n, such as the following:

$$\mathsf{G}((\bigwedge_{i=1}^{n} (p_i \leftrightarrow \mathsf{O}(\widetilde{\mathsf{Y}}\bot \wedge p_i))) \leftrightarrow (p_0 \leftrightarrow \mathsf{O}(\widetilde{\mathsf{Y}}\bot \wedge p_0)).)$$

Note again the crucial role of the subformula $\widetilde{\mathsf{Y}}\bot$ for hooking the initial state of the trace. This theorem directly follows.

▶ **Theorem 15.** G(pLTL) *can be exponentially more succinct than* Safety-LTL.

5.2 F(pLTL) can be exponentially more succinct than coSafety-LTL

We now dualize the previous result to the cosafety case, and obtain the succinctness lower bound of F(pLTL) with respect to coSafety-LTL. We first prove a more general result on "dual" temporal logics (which we define here below), and then we instantiate the result to the specific case of F(pLTL) and G(pLTL). We define *dual logics* as follows.

▶ **Definition 16** (Dual Logics). *Given two linear-time temporal logics $\mathbb{L}$ and $\overline{\mathbb{L}}$, we say that $\overline{\mathbb{L}}$ is a* dual logic *of $\mathbb{L}$ iff:*
1. *for any formula $\phi \in \overline{\mathbb{L}}$, the transformation in* negation normal form *of $\neg\phi$ (denoted as $\mathrm{nnf}(\neg\phi)$) belongs to $\mathbb{L}$, and*
2. *for any formula $\phi \in \mathbb{L}$, the transformation in* negation normal form *of $\neg\phi$ (denoted as $\mathrm{nnf}(\neg\phi)$) belongs to $\overline{\mathbb{L}}$.*

As for the case of reverse logics, also being a dual logic is a symmetric property.

The following lemma proves that *duality* (as for Definition 16) preserves succinctness.

▶ **Lemma 17** (Duality Lemma). *For any linear-time temporal logics $\mathbb{L}$ and $\mathbb{L}'$, if $\mathbb{L}$ can be exponentially more succinct than $\mathbb{L}'$, then $\overline{\mathbb{L}}$ can be exponentially more succinct than $\overline{\mathbb{L}'}$, where $\overline{\mathbb{L}}$ (resp., $\overline{\mathbb{L}'}$) is a dual logic of $\mathbb{L}$ (resp., $\mathbb{L}'$).*

Proof. Since by hypothesis $\mathbb{L}$ can be exponentially more succinct than $\mathbb{L}'$, there exists a formula $\phi \in \mathbb{L}$ of size n such that, for all $\phi' \in \mathbb{L}'$, if $\mathcal{L}(\phi') = \mathcal{L}(\phi)$ than $|\phi'| \in 2^{\Omega(n)}$.

Let $\overline{\phi}$ be the negation normal form of $\neg\phi$, *i.e.* $\overline{\phi} = \mathrm{nnf}(\neg\phi)$. By definition:
1. $\overline{\phi}$ belongs to $\overline{\mathbb{L}}$;
2. $\mathcal{L}(\overline{\phi}) = \mathcal{L}(\neg\phi)$; and
3. $|\overline{\phi}| \in \mathcal{O}(n)$.

Suppose by contradiction that the thesis does not hold, that is, for all formulas $\overline{\psi} \in \overline{\mathbb{L}}$ of size $s = |\overline{\psi}|$ there exists a formula $\overline{\psi'} \in \overline{\mathbb{L}'}$ such that $\mathcal{L}(\overline{\psi}) = \mathcal{L}(\overline{\psi'})$ and $|\overline{\psi'}|$ is less than exponential in s. In particular, for $\overline{\psi} := \overline{\phi}$, this means that there exists a formula $\overline{\psi'}$ in $\overline{\mathbb{L}'}$ such that $\mathcal{L}(\overline{\psi'}) = \mathcal{L}(\overline{\phi})$ and $|\overline{\psi'}|$ is less than exponential in n (recall that n is the size of $\overline{\phi}$).

Now, let χ' be the negation normal form of $\neg\overline{\psi'}$ in negated normal form. It holds that:
1. χ' is a formula in $\mathbb{L}$;
2. $\mathcal{L}(\chi') = \mathcal{L}(\phi)$; and
3. $|\chi'| \in \mathcal{O}(|\overline{\psi'}|)$.

Since the size of $\overline{\psi'}$ is less than exponential in n, the size of $|\chi'|$ is less than exponential in n as well. This means that $\mathcal{L}(\phi)$ can be defined in $\mathbb{L}'$ with a formula of size less than exponential in n, which is a contradiction with the hypothesis. ◀

Since, by definition, F(pLTL) and coSafety-LTL are dual logics of G(pLTL) and Safety-LTL, respectively, by Lemma 17 and Theorem 15, this result follows.

▶ **Theorem 18.** F(pLTL) *can be exponentially more succinct than* coSafety-LTL.

5.3 Open Problems

To complete the picture, we give a conjecture on the succinctness of the (co)safety fragments of LTL. To the best of our knowledge, it is still an open question whether coSafety-LTL (resp., Safety-LTL) can be exponentially more succinct than F(pLTL) (resp., G(pLTL)).

We conjecture that coSafety-LTL can be $n!$ (n factorial) more succinct than F(pLTL). Let $\Sigma = \{p_i\}_{i=1}^n \cup \{q_i\}_{i=1}^n$ be a finite alphabet. Consider the following family of languages C_n over the alphabet Σ, where $n > 0$:

$$C_n := \{\sigma \in (2^\Sigma)^\omega \mid \exists k \geq 0 \,.\, \bigwedge_{i=1}^{n} (\exists h > k \,.\, (q_i \in \sigma_h \wedge \forall k \leq l < h \,.\, p_i \in \sigma_l))\}.$$

For any $n > 0$, C_n comprises the infinite traces for which there exists a time point k such that, for each $i \in \{1, \ldots, n\}$, q_i will eventually be realized in the future of k and p_i holds until (and excluding) that point (cf. Figure 4).

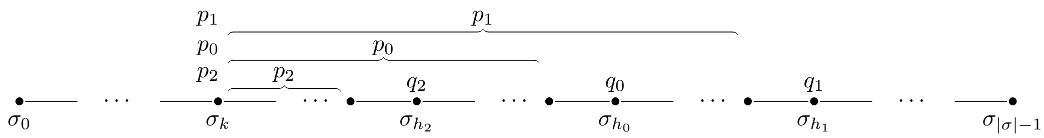

Figure 4 Example of a word σ in C_2.

While C_n is definable in coSafety-LTL with a formula of linear size in n, for example $\mathsf{F}(\bigwedge_{i=1}^{n}(p_i \mathbin{\mathsf{U}} q_i))$, using $\mathsf{F}(\text{pLTL})$ one is forced to enumerate all possible orders between $q_1, \ldots, q_n$ with a formula of this type:

$$\mathsf{F}\Big(\bigvee_{\pi \in \Pi} (q_{\pi(1)} \wedge \mathsf{Y}(p_{\pi(1)}) \mathbin{\mathsf{S}} (p_{\pi(1)} \wedge q_{\pi(2)} \wedge \mathsf{Y}(p_{\pi(1)} \wedge p_{\pi(2)}) \mathbin{\mathsf{S}} (p_{\pi(1)} \wedge p_{\pi(2)} \wedge q_{\pi(3)} \wedge \ldots \mathsf{Y}(\bigwedge_{i=1}^{n-1} p_{\pi(i)}) \mathbin{\mathsf{S}} (q_{\pi(n)} \wedge \bigwedge_{i=1}^{n} p_{\pi(i)})) \ldots))\Big)$$

where Π is the set of permutations of $\{1, \ldots, n\}$. This, in turn, forces the formula to be at least of size $n!$.

▶ **Conjecture 19.** *For any $n > 0$, the language C_n is not expressible in* $\mathsf{F}(\text{pLTL})$ *with a formula of size less than $n!$.*

We conjecture the same for dual case of Safety-LTL and $\mathsf{G}(\text{pLTL})$.

6 Conclusions

We proved the incomparability between the succinctness of LTLf and of pLTL. We started by proving that the family of properties A_n admits a formalization in pLTL with formulas of linear size, while all formulas in LTLf for A_n are at least of exponential size. By using the Reverse Lemma, we derived that also the *vice versa* holds, that is, LTLf can be exponentially more succinct than pLTL. This result allowed us to confirm the conjecture left open in [2] about the lower bound for the complexity of translating LTLf into pLTL. We finally showed that $\mathsf{G}(\text{pLTL})$ and $\mathsf{F}(\text{pLTL})$ (*i.e.* the canonical forms of the safety and cosafety fragments of LTL) can be exponentially more succinct than Safety-LTL and coSafety-LTL, respectively.

The study of the maximal fragment of LTLf that does not incur in the exponential blow-up in the translation into pLTL is surely a problem worth studying, both for its theoretical implications and for its applications in reactive synthesis.

Proving Conjecture 19 is also an interesting future direction, which may require more sophisticated techniques for proving the lower bound, such as Ehrenfeucht-Fraïssé games [22] or Adler-Immermann games [1].

Finally, while we know that the lower bound between the translation of LTLf into pLTL is at least exponential, we have an upper bound which is triply exponential. The possibility of tighter lower bounds, or more efficient algorithms for this problem, is worth investigating.

References

1 Micah Adler and Neil Immerman. An n! lower bound on formula size. *ACM Transactions on Computational Logic (TOCL)*, 4(3):296–314, 2003.

2 Alessandro Artale, Luca Geatti, Nicola Gigante, Andrea Mazzullo, and Angelo Montanari. Complexity of safety and cosafety fragments of Linear Temporal Logic. In *Proc. of the 36th AAAI Conf. on Artificial Intelligence (AAAI-22)*. AAAI Press, 2023.

3 Howard Barringer, Michael Fisher, Dov Gabbay, Graham Gough, and Richard Owens. Metatem: A framework for programming in temporal logic. In *Stepwise Refinement of Distributed Systems Models, Formalisms, Correctness: REX Workshop, Mook, The Netherlands May 29–June 2, 1989 Proceedings*, pages 94–129. Springer, 1990.

4 Ronen I. Brafman and Giuseppe De Giacomo. Planning for LTLf/LDLf goals in non-markovian fully observable nondeterministic domains. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 1602–1608. ijcai.org, 2019.

5 Alberto Camacho, Jorge Baier, Christian Muise, and Sheila McIlraith. Finite ltl synthesis as planning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 28, pages 29–38, 2018.

6 Alberto Camacho, Eleni Triantafillou, Christian J. Muise, Jorge A. Baier, and Sheila A. McIlraith. Non-deterministic planning with temporally extended goals: LTL over finite and infinite traces. In Satinder Singh and Shaul Markovitch, editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3716–3724. AAAI Press, 2017.

7 Edward Y. Chang, Zohar Manna, and Amir Pnueli. Characterization of temporal property classes. In Werner Kuich, editor, *Proceedings of the 19th International Colloquium on Automata, Languages and Programming*, volume 623 of *Lecture Notes in Computer Science*, pages 474–486. Springer, 1992.

8 Alessandro Cimatti, Luca Geatti, Nicola Gigante, Angelo Montanari, and Stefano Tonetta. Extended bounded response LTL: a new safety fragment for efficient reactive synthesis. *Formal Methods in System Design*, pages 1–49, 2021.

9 Giuseppe De Giacomo, Antonio Di Stasio, Francesco Fuggitti, and Sasha Rubin. Pure-past linear temporal and dynamic logic on finite traces. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 4959–4965, 2021.

10 Giuseppe De Giacomo, Antonio Di Stasio, Lucas M. Tabajara, Moshe Y. Vardi, and Shufang Zhu. Finite-trace and generalized-reactivity specifications in temporal synthesis. In Zhi-Hua Zhou, editor, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 1852–1858. ijcai.org, 2021.

11 Giuseppe De Giacomo and Moshe Y. Vardi. Linear temporal logic and linear dynamic logic on finite traces. In Francesca Rossi, editor, *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pages 854–860. IJCAI/AAAI, 2013.

12 Giuseppe De Giacomo and Moshe Y. Vardi. Synthesis for LTL and LDL on finite traces. In Qiang Yang and Michael J. Wooldridge, editors, *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*, pages 1558–1564. AAAI Press, 2015.

13 Kousha Etessami, Moshe Y Vardi, and Thomas Wilke. First-order logic with two variables and unary temporal logic. *Information and computation*, 179(2):279–295, 2002.

14 Dov Gabbay. The declarative past and imperative future: Executable temporal logic for interactive systems. In *Temporal Logic in Specification: Altrincham, UK, April 8–10, 1987 Proceedings*, pages 409–448. Springer, 1989.

15 Lauri Hella and Miikka Vilander. Formula size games for modal logic and μ-calculus. *J. Log. Comput.*, 29(8):1311–1344, 2019.

16 Orna Lichtenstein, Amir Pnueli, and Lenore Zuck. The glory of the past. In *Workshop on Logic of Programs*, pages 196–218. Springer, 1985.

17 Oded Maler and Amir Pnueli. Tight bounds on the complexity of cascaded decomposition of automata. In *Proceedings [1990] 31st Annual Symposium on Foundations of Computer Science*, pages 672–682. IEEE, 1990.

18 Nicolas Markey. Temporal logic with past is exponentially more succinct. *Bull. EATCS*, 79:122–128, 2003.

19 Maja Pesic, Helen Schonenberg, and Wil MP Van der Aalst. Declare: Full support for loosely-structured processes. In *11th IEEE international enterprise distributed object computing conference (EDOC 2007)*, pages 287–287. IEEE, 2007.

20 Maja Pesic and Wil MP Van der Aalst. A declarative approach for flexible business processes management. In *Business Process Management Workshops: BPM 2006 International Workshops, BPD, BPI, ENEI, GPWW, DPM, semantics4ws, Vienna, Austria, September 4-7, 2006. Proceedings 4*, pages 169–180. Springer, 2006.

21 A. Prasad Sistla. On characterization of safety and liveness properties in temporal logic. In *Proceedings of the fourth annual ACM symposium on Principles of distributed computing*, pages 39–48, 1985.

22 Wolfgang Thomas. On the ehrenfeucht-fraïssé game in theoretical computer science. In *TAPSOFT'93: Theory and Practice of Software Development: 4th International Joint Conference CAAP/FASE Orsay, France, April 13–17, 1993 Proceedings*, pages 559–568. Springer, 2005.

23 Shufang Zhu, Giuseppe De Giacomo, Geguang Pu, and Moshe Y. Vardi. Ltl$f$ synthesis with fairness and stability assumptions. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 3088–3095. AAAI Press, 2020.

24 Shufang Zhu, Lucas M. Tabajara, Jianwen Li, Geguang Pu, and Moshe Y. Vardi. A Symbolic Approach to Safety LTL Synthesis. In Ofer Strichman and Rachel Tzoref-Brill, editors, *Proceedings of the 13th International Haifa Verification Conference*, volume 10629 of *Lecture Notes in Computer Science*, pages 147–162. Springer, 2017. `doi:10.1007/978-3-319-70389-3_10`.

25 Lenore Zuck. Past temporal logic. *Weizmann Institute of Science*, 67, 1986.