Succinctness of Cosafety Fragments of LTL via Combinatorial Proof Systems

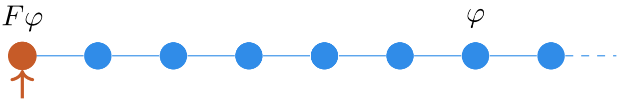
Luca Geatti[1] **Alessio Mansutti**[2] Angelo Montanari[1]

[1] University of Udine, Udine, Italy

[2] IMDEA Software Institute, Madrid, Spain

Linear-time Temporal Logics

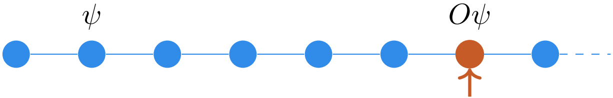
LTL:



φ will eventually hold

F(pLTL) – Eventually Past LTL:

Set of formulae of the form $F(\varphi)$ with φ only using past temporal operators.



ψ once held

$F(\mathrm{pLTL})$: comparison with $\mathrm{coSafety\ LTL}$

Expressive power:

$F(\mathrm{pLTL})$ is equivalent to the cosafety fragment of LTL.

Cosafety language:

$\mathcal{L} = K \cdot \Sigma^\omega$ for some $K \subseteq \Sigma^*$.

"something good will eventually happen"

Complexity:

	$\mathrm{coSafety\ LTL}$	$F(\mathrm{pLTL})$
Realizability	2EXPTIME	EXPTIME
	without the Until/Since operators:	
Realizability	EXPTIME	EXPTIME

$F(\mathrm{pLTL})$: comparison with $\mathrm{coSafety\ LTL}$

Expressive power:

$F(\mathrm{pLTL})$ is equivalent to the cosafety fragment of LTL.

Cosafety language:

$\mathcal{L} = K \cdot \Sigma^\omega$ for some $K \subseteq \Sigma^*$.

"something good will eventually happen"

Complexity:

	$\mathrm{coSafety\ LTL}$	$F(\mathrm{pLTL})$
Realizability	2EXPTIME	EXPTIME
	without the Until/Since operators:	
Realizability	EXPTIME	EXPTIME

Question: What is the cost of translating $\mathrm{coSafety\ LTL}$ into $F(\mathrm{pLTL})$?

Translating from coSafety LTL to F(pLTL)

- In triply-exponential time [De Giacomo et al., IJCAI'21].
- and in time $2^{O(n)}$ when Until/Since are removed [Artale et al., KR'23].
- Before our work, only trivial lower bounds were known.

Translating from coSafety LTL to $F(\mathrm{pLTL})$

- In triply-exponential time [De Giacomo et al., IJCAI'21].
- and in time $2^{O(n)}$ when Until/Since are removed [Artale et al., KR'23].
- Before our work, only trivial lower bounds were known.

Theorem

There is a family of cosafety languages $(\mathcal{L}_n)_{n\geq 1}$ such that, for every $n \geq 1$,

- *$\mathcal{L}_n$ is expressible with a formula φ_n of $\mathrm{LTL}[F]$ having size polynomial in n. The formula φ_n is in negation normal from.*
- *Every formula of $F(\mathrm{pLTL})$, without Since operator, expressing $\mathcal{L}_n$ has size $2^{\Omega(n)}$.*

This result holds for both languages on finite and infinite words.

Translating from coSafety LTL to $F(\mathrm{pLTL})$

$2^{\Theta(n)}$

- In triply-exponential time [De Giacomo et al., IJCAI'21].
- and in time $2^{\cancel{\mathcal{O}(n)}}$ when Until/Since are removed [Artale et al., KR'23].
- Before our work, only trivial lower bounds were known.

Theorem

There is a family of cosafety languages $(\mathcal{L}_n)_{n\geq 1}$ such that, for every $n \geq 1$,

- *$\mathcal{L}_n$ is expressible with a formula φ_n of $\mathrm{LTL}[F]$ having size polynomial in n. The formula φ_n is in negation normal from.*
- *Every formula of $F(\mathrm{pLTL})$, without Since operator, expressing $\mathcal{L}_n$ has size $2^{\Omega(n)}$.*

This result holds for both languages on finite and infinite words.

Translating from coSafety LTL to F(pLTL)

$2^{\Theta(n)}$

- In triply-exponential time [De Giacomo et al., IJCAI'21].
- and in time $2^{\cancel{O(n)}}$ when Until/Since are removed [Artale et al., KR'23].
- Before our work, only trivial lower bounds were known.

Theorem

There is a family of cosafety languages $(\mathcal{L}_n)_{n \geq 1}$ such that, for every $n \geq 1$,

- *$\mathcal{L}_n$ is expressible with a formula φ_n of $F(\mathrm{pLTL}[O])$ having size polynomial in n. The formula φ_n is in negation normal from.*
- *Every formula of LTL, without Until operator, expressing $\mathcal{L}_n$ has size $2^{\Omega(n)}$.*

This result holds for both languages on finite and infinite words.

Translating from coSafety LTL to $F(\text{pLTL})$

$2^{\Theta(n)}$

- In triply-exponential time [De Giacomo et al., IJCAI'21].

Proof technique: combinatorial proof systems (1-player games).

No proofs of size $< k$ for a property $P \implies P$ requires formulae of size $\geq k$.

Theorem

There is a family of cosafety languages $(\mathcal{L}_n)_{n\geq 1}$ such that, for every $n \geq 1$,

- *$\mathcal{L}_n$ is expressible with a formula φ_n of $F(\text{pLTL}[O])$ having size polynomial in n. The formula φ_n is in negation normal from.*
- *Every formula of LTL, without Until operator, expressing $\mathcal{L}_n$ has size $2^{\Omega(n)}$.*

This result holds for both languages on finite and infinite words.

LTL on finite traces, without the Until operator

Syntax: $\varphi, \psi \;:=\; p \mid \neg p \mid \varphi \vee \psi \mid \varphi \wedge \psi \mid X\psi \mid \widetilde{X}\varphi \mid F\varphi \mid G\varphi$

Structure: non-empty finite words over a (possibly infinite) alphabet $\Sigma := 2^{\mathcal{AP}}$, where $\mathcal{AP}$ is a set of atomic propositions.

Semantics: Let $\boldsymbol{w} = w_0 \dots w_n$ be a finite word in Σ^+. Then,

$$
\begin{aligned}
\boldsymbol{w} &\models p && \iff && p \in w_0 \\
\boldsymbol{w} &\models \neg p && \iff && p \notin w_0 \\
\boldsymbol{w} &\models \varphi \vee \psi && \iff && \boldsymbol{w} \models \varphi \text{ or } \boldsymbol{w} \models \psi \\
\boldsymbol{w} &\models \varphi \wedge \psi && \iff && \boldsymbol{w} \models \varphi \text{ and } \boldsymbol{w} \models \psi \\
\boldsymbol{w} &\models X\varphi && \iff && n \geq 1 \text{ and } w_1 \dots w_n \models \varphi \\
\boldsymbol{w} &\models \widetilde{X}\varphi && \iff && n = 0 \text{ or } w_1 \dots w_n \models \varphi \\
\boldsymbol{w} &\models F\varphi && \iff && w_j \dots w_n \models \varphi \text{ for some } j \in [0, n] \\
\boldsymbol{w} &\models G\varphi && \iff && w_j \dots w_n \models \varphi \text{ for every } j \in [0, n]
\end{aligned}
$$

Lower bounds via combinatorial proof systems

Let $\boldsymbol{A}, \boldsymbol{B} \subseteq \Sigma^+$. We write $\langle \boldsymbol{A}, \boldsymbol{B} \rangle$ whenever $\boldsymbol{A}$ and $\boldsymbol{B}$ are separable, i.e., there is a formula φ (a separator) such that

- $\boldsymbol{A} \models \varphi$: for every $\boldsymbol{w} \in \boldsymbol{A}$, $\boldsymbol{w} \models \varphi$, and
- $\boldsymbol{B} \perp\!\!\!\perp \varphi$: for every $\boldsymbol{w} \in \boldsymbol{B}$, $\boldsymbol{w} \not\models \varphi$.

Lower bounds via combinatorial proof systems

Let $\boldsymbol{A}, \boldsymbol{B} \subseteq \Sigma^+$. We write $\langle \boldsymbol{A}, \boldsymbol{B} \rangle$ whenever $\boldsymbol{A}$ and $\boldsymbol{B}$ are separable, i.e., there is a formula φ (a separator) such that

- $\boldsymbol{A} \models \varphi$: for every $\boldsymbol{w} \in \boldsymbol{A}$, $\boldsymbol{w} \models \varphi$, and
- $\boldsymbol{B} \perp\!\!\!\perp \varphi$: for every $\boldsymbol{w} \in \boldsymbol{B}$, $\boldsymbol{w} \not\models \varphi$.

Combinatorial proof system: Set of proof rules to establish whether $\langle \boldsymbol{A}, \boldsymbol{B} \rangle$.

$$\text{Rule1}\;\dfrac{\text{Rule2}\;\dfrac{\text{Axiom}\;\dfrac{}{\langle \boldsymbol{A}_1, \boldsymbol{B}_1 \rangle}}{\langle \boldsymbol{A}_2, \boldsymbol{B}_2 \rangle} \qquad \dfrac{}{\langle \boldsymbol{A}_3, \boldsymbol{B}_3 \rangle}\;\text{Axiom}}{\langle \boldsymbol{A}, \boldsymbol{B} \rangle}$$

Desired property (for lower bounds): If there is a separator for $\boldsymbol{A}$ and $\boldsymbol{B}$ having size k, then $\langle \boldsymbol{A}, \boldsymbol{B} \rangle$ has a proof of size k. (in fact, we get an if-and-only-if)

The combinatorial proof system, via an example

Consider the alphabet $2^{\{p\}} = \{\varnothing, \{p\}\}$. For simplicity, let $a := \{p\}$ and $b := \varnothing$.

$$\langle \{abaa, aaaa\}, \{aaab\} \rangle$$

The combinatorial proof system, via an example

Consider the alphabet $2^{\{p\}} = \{\varnothing, \{p\}\}$. For simplicity, let $a := \{p\}$ and $b := \varnothing$.

$$\textsc{Or}\ \frac{\langle \{abaa\}, \{aaab\} \rangle \qquad \langle \{aaaa\}, \{aaab\} \rangle}{\langle \{abaa, aaaa\}, \{aaab\} \rangle}$$

$$\textsc{Or}\ \frac{\langle \boldsymbol{A}_1, \boldsymbol{B} \rangle \quad \langle \boldsymbol{A}_2, \boldsymbol{B} \rangle}{\langle \boldsymbol{A}_1 \cup \boldsymbol{A}_2, \boldsymbol{B} \rangle}$$

The combinatorial proof system, via an example

Consider the alphabet $2^{\{p\}} = \{\varnothing, \{p\}\}$. For simplicity, let $a := \{p\}$ and $b := \varnothing$.

$$\textsc{Or}\ \dfrac{\textsc{Next}\ \dfrac{\langle\{baa\},\{aab\}\rangle}{\langle\{abaa\},\{aaab\}\rangle} \qquad \langle\{aaaa\},\{aaab\}\rangle}{\langle\{abaa, aaaa\},\{aaab\}\rangle}$$

$$\textsc{Next}\ \dfrac{\langle \boldsymbol{A}^X, \boldsymbol{B}^X\rangle \qquad \boldsymbol{A} \subseteq \Sigma \cdot \Sigma^+}{\langle \boldsymbol{A}, \boldsymbol{B}\rangle}$$

$\boldsymbol{A}^X := \{\boldsymbol{w} \in \Sigma^+ : w_0 \cdot \boldsymbol{w} \in \boldsymbol{A}$ for some $w_0 \in \Sigma\}$

The combinatorial proof system, via an example

Consider the alphabet $2^{\{p\}} = \{\varnothing, \{p\}\}$. For simplicity, let $a := \{p\}$ and $b := \varnothing$.

$$\dfrac{\dfrac{\dfrac{\{baa\} \models \neg p \qquad \{aab\} \perp\!\!\!\perp \neg p}{\langle \{baa\}, \{aab\} \rangle}\,\textsc{Atomic}}{\langle \{abaa\}, \{aaab\} \rangle}\,\textsc{Next} \qquad \langle \{aaaa\}, \{aaab\} \rangle}{\langle \{abaa, aaaa\}, \{aaab\} \rangle}\,\textsc{Or}$$

$$\textsc{Atomic}\ \dfrac{\boldsymbol{A} \models \alpha \quad \boldsymbol{B} \perp\!\!\!\perp \alpha}{\langle \boldsymbol{A}, \boldsymbol{B} \rangle}\ \alpha \text{ literal}$$

The combinatorial proof system, via an example

Consider the alphabet $2^{\{p\}} = \{\varnothing, \{p\}\}$. For simplicity, let $a := \{p\}$ and $b := \varnothing$.

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

$$
\textsc{Globally}\ \dfrac{\langle \boldsymbol{A}^G, \boldsymbol{B}^f\rangle}{\langle \boldsymbol{A}, \boldsymbol{B}\rangle}\ f \in F_{\boldsymbol{B}}
$$

$F_{\boldsymbol{B}}$ is the set of all functions $f \colon \{\boldsymbol{w} \in B\} \to \{\boldsymbol{w}' : \boldsymbol{w}' \text{ suffix of } \boldsymbol{w}\}$

$\boldsymbol{B}^f := \{f(\boldsymbol{w}) : \boldsymbol{w} \in \boldsymbol{B}\}$

$\boldsymbol{A}^G := \{\boldsymbol{w}' : \boldsymbol{w}' \text{ suffix of some } \boldsymbol{w} \in \boldsymbol{A}\}$

The combinatorial proof system, via an example

Consider the alphabet $2^{\{p\}} = \{\varnothing, \{p\}\}$. For simplicity, let $a := \{p\}$ and $b := \varnothing$.

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

$$
\textsc{Atomic}\ \dfrac{\boldsymbol{A} \models \alpha \qquad \boldsymbol{B} \perp\!\!\!\perp \alpha}{\langle \boldsymbol{A}, \boldsymbol{B} \rangle}\ \alpha \text{ literal}
$$

The combinatorial proof system, via an example

Consider the alphabet $2^{\{p\}} = \{\varnothing, \{p\}\}$. For simplicity, let $a := \{p\}$ and $b := \varnothing$.

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

$\{abaa, aaaa\}$ and $\{aaab\}$ are separated by the formula $(X\neg p) \vee (Gp)$

The combinatorial proof system

$$\textsc{Atomic}\ \frac{\boldsymbol{A} \models \alpha \quad \boldsymbol{B} \perp\!\!\!\perp \alpha}{\langle \boldsymbol{A}, \boldsymbol{B} \rangle}\ \alpha \text{ literal} \qquad \textsc{Or}\ \frac{\langle \boldsymbol{A}_1, \boldsymbol{B} \rangle \quad \langle \boldsymbol{A}_2, \boldsymbol{B} \rangle}{\langle \boldsymbol{A}_1 \cup \boldsymbol{A}_2, \boldsymbol{B} \rangle} \qquad \textsc{And}\ \frac{\langle \boldsymbol{A}, \boldsymbol{B}_1 \rangle \quad \langle \boldsymbol{A}, \boldsymbol{B}_2 \rangle}{\langle \boldsymbol{A}, \boldsymbol{B}_1 \cup \boldsymbol{B}_2 \rangle}$$

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

$$\textsc{Future}\ \frac{\langle \boldsymbol{A}^f, \boldsymbol{B}^G \rangle}{\langle \boldsymbol{A}, \boldsymbol{B} \rangle}\ f \in F_{\boldsymbol{A}} \qquad \textsc{Globally}\ \frac{\langle \boldsymbol{A}^G, \boldsymbol{B}^f \rangle}{\langle \boldsymbol{A}, \boldsymbol{B} \rangle}\ f \in F_{\boldsymbol{B}}$$

Theorem

Consider $\boldsymbol{A}, \boldsymbol{B} \subseteq \Sigma^+$. The term $\langle \boldsymbol{A}, \boldsymbol{B} \rangle$ has a proof of size k if and only if $\boldsymbol{A}$ and $\boldsymbol{B}$ are separated by a formula φ of LTL without the Until operator satisfying size$(\varphi) = k$.

The combinatorial proof system

$$\textsc{Atomic}\ \frac{\boldsymbol{A} \models \alpha \quad \boldsymbol{B} \perp\!\!\!\perp \alpha}{\langle \boldsymbol{A}, \boldsymbol{B} \rangle}\ \alpha \text{ literal} \qquad \textsc{Or}\ \frac{\langle \boldsymbol{A}_1, \boldsymbol{B} \rangle \quad \langle \boldsymbol{A}_2, \boldsymbol{B} \rangle}{\langle \boldsymbol{A}_1 \cup \boldsymbol{A}_2, \boldsymbol{B} \rangle} \qquad \textsc{And}\ \frac{\langle \boldsymbol{A}, \boldsymbol{B}_1 \rangle \quad \langle \boldsymbol{A}, \boldsymbol{B}_2 \rangle}{\langle \boldsymbol{A}, \boldsymbol{B}_1 \cup \boldsymbol{B}_2 \rangle}$$

Observation: For propositional logic, the proof system with rules Atomic, Or and And correspond to the communication games by Karchmer and Wigderson.

- originally introduced for both (i) size lower bounds of formulae and (ii) depth of Boolean circuits (STOC'88)
- still actively studied in circuit complexity (see the KRW conjecture).

Consider $\boldsymbol{A}, \boldsymbol{B} \subseteq \Sigma^+$. The term $\langle \boldsymbol{A}, \boldsymbol{B} \rangle$ has a proof of size k if and only if $\boldsymbol{A}$ and $\boldsymbol{B}$ are separated by a formula φ of LTL without the Until operator satisfying size$(\varphi) = k$.

Using the combinatorial proof system

Goal

Find a family of formulae $\{\varphi_n\}_{n\geq 1}$ in $F(\mathrm{pLTL}[O])$ and a family of pairs of sets of words $\{(\boldsymbol{A}_n, \boldsymbol{B}_n)\}_{n\geq 1}$ such that, for every $n \geq 1$,

- φ_n has size polynomial in n,
- $\boldsymbol{A}_n \subseteq \mathcal{L}_n$ and $\boldsymbol{B}_n \cap \mathcal{L}_n = \varnothing$, and $\langle \boldsymbol{A}_n, \boldsymbol{B}_n \rangle$ requires a proof of size $2^{\Omega(n)}$.

Ad Break: LTL formula learning tools are great!

Finding simple definitions for φ_n and $(\boldsymbol{A}_n, \boldsymbol{B}_n)$ was not a fun endeavour.

Tools for LTL formula learning helped us a lot!

Skip Ad ▶|

Ad Break: LTL formula learning tools are great!

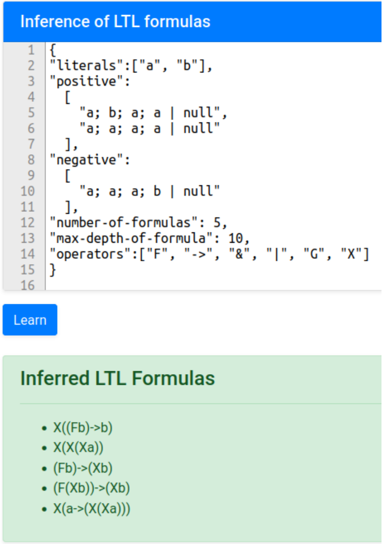
Finding simple definitions for φ_n and $(\boldsymbol{A}_n, \boldsymbol{B}_n)$ was not a fun endeavour.

Tools for LTL formula learning helped us a lot!

Samples2LTL (Neider and Gavran):
Given in input two finite sets $\boldsymbol{A}$ and $\boldsymbol{B}$ of words, finds a (minimal) separator.

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

$\{abaa, aaaa\}$ and $\{aaab\}$ are separated by $(X\neg p) \vee (Gp)$, but also by $XXXp$



Using the combinatorial proof system

Goal

Find a family of formulae $\{\varphi_n\}_{n\geq 1}$ in $F(\mathrm{pLTL}[O])$ and a family of pairs of sets of words $\{(\boldsymbol{A}_n, \boldsymbol{B}_n)\}_{n\geq 1}$ such that, for every $n \geq 1$,

- φ_n has size polynomial in n,
- $\boldsymbol{A}_n \subseteq \mathcal{L}_n$ and $\boldsymbol{B}_n \cap \mathcal{L}_n = \varnothing$, and $\langle \boldsymbol{A}_n, \boldsymbol{B}_n \rangle$ requires a proof of size $2^{\Omega(n)}$.

Using the combinatorial proof system

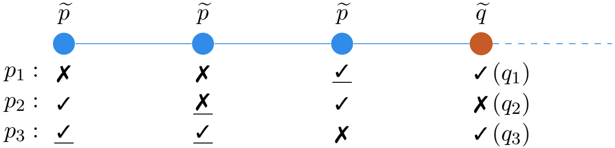
Goal

Find a family of formulae $\{\varphi_n\}_{n\geq 1}$ in $F(\mathrm{pLTL}[O])$ and a family of pairs of sets of words $\{(\boldsymbol{A}_n, \boldsymbol{B}_n)\}_{n\geq 1}$ such that, for every $n \geq 1$,

- φ_n has size polynomial in n,
- $\boldsymbol{A}_n \subseteq \mathcal{L}_n$ and $\boldsymbol{B}_n \cap \mathcal{L}_n = \varnothing$, and $\langle \boldsymbol{A}_n, \boldsymbol{B}_n \rangle$ requires a proof of size $2^{\Omega(n)}$.

Given $n \geq 1$, we consider atomic propositions $\widetilde{p}, \widetilde{q}, p_1, \ldots, p_n, q_1, \ldots, q_n$. Then,

$$\varphi_n := F\left(\widetilde{q} \land \bigwedge\nolimits_{i=1}^{n} \left(\left(q_i \land O(\widetilde{p} \land p_i)\right) \lor \left(\neg q_i \land O(\widetilde{p} \land \neg p_i)\right)\right)\right)$$



Finding $\boldsymbol{A}_n$ and $\boldsymbol{B}_n$

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

- Let $\mathcal{E}$ be a word enumerating $Q := \{S \subseteq \{\widetilde{q}, q_1, \ldots, q_n\} : \widetilde{q} \in S\}$
(for technical reason, in $\mathcal{E}$ after every element of Q we add exponentially many $\varnothing$)
- Let $\mathcal{E}|_{-\tau}$ be the word obtained from $\mathcal{E}$ by removing $\tau \in Q$
- Let $\overline{\tau} \subseteq \{\widetilde{p}, p_1, \ldots, p_n\}$ be the set obtained from $\tau \in Q$ by "replacing q with p".

Finding $\boldsymbol{A}_n$ and $\boldsymbol{B}_n$

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

- Let $\mathcal{E}$ be a word enumerating $Q := \{S \subseteq \{\widetilde{q}, q_1, \ldots, q_n\} : \widetilde{q} \in S\}$
  (for technical reason, in $\mathcal{E}$ after every element of Q we add exponentially many $\varnothing$)
- Let $\mathcal{E}|_{-\tau}$ be the word obtained from $\mathcal{E}$ by removing $\tau \in Q$
- Let $\overline{\tau} \subseteq \{\widetilde{p}, p_1, \ldots, p_n\}$ be the set obtained from $\tau \in Q$ by "replacing q with p".

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

Proposition

$\boldsymbol{A}_n \subseteq \mathcal{L}_n$ and $\boldsymbol{B}_n \cap \mathcal{L}_n = \varnothing$, and $\langle \boldsymbol{A}_n, \boldsymbol{B}_n \rangle$ requires a proof of size $2^{\Omega(n)}$.

Conclusion

- translating $\mathrm{coSafety\ LTL}$ into $F(\mathrm{pLTL})$, without Until/Since, requires $2^{\Theta(n)}$ time.
- The automata technique used by Markey (Bull. EATCS, 2003) to show that pLTL can be more succinct than LTL cannot be used to show the $2^{\Omega(n)}$ lower bound.

Conclusion

- translating $\mathrm{coSafety}\ \mathrm{LTL}$ into $F(\mathrm{pLTL})$, without Until/Since, requires $2^{\Theta(n)}$ time.
- The automata technique used by Markey (Bull. EATCS, 2003) to show that pLTL can be more succinct than LTL cannot be used to show the $2^{\Omega(n)}$ lower bound.

Combinatorial proof system for LTL:

- extension of Karchmer and Wigderson's communication games to LTL
- connected to recent games for bounding the number of quantifiers in first-order logic (see LICS'23 workshop *Combinatorial Games in Finite Model Theory*)
- LTL formula learning tools are very useful for exploring lower bounds.