The complexity of linear temporal verification for continuous counter systems

Michael Blondin

Université de Sherbrooke

The complexity of linear temporal verification for continuous counter systems

Michael Blondin

Joint work with Alex Sansfaçon-Buchanan and Philip Offtermatt

Slides based on those of Alex

Université de Sherbrooke

Constant-rate multi-mode systems (MMS) Alur et al. HSCC'12

An *MMS* is a finite set $M \subseteq \mathbb{R}^d$

An *MMS* is a finite set $M \subseteq \mathbb{R}^d$

For example, $M = \{\boldsymbol{m}_1, \boldsymbol{m}_2, \boldsymbol{m}_3\}$ where

$\boldsymbol{m}_1$	$\boldsymbol{m}_2$	$\boldsymbol{m}_3$
$(1, 2)$	$(1, 0)$	$(-1, -1)$

Constant-rate multi-mode systems (MMS) — Alur et al. HSCC'12

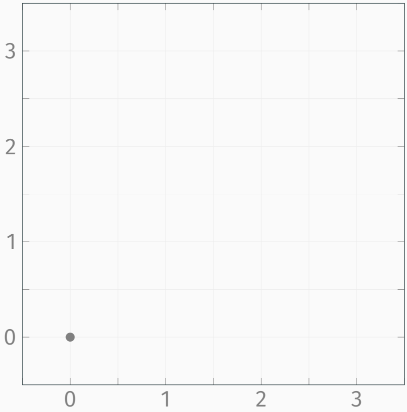
An *MMS* is a finite set $M \subseteq \mathbb{R}^d$

For example, $M = \{\boldsymbol{m}_1, \boldsymbol{m}_2, \boldsymbol{m}_3\}$ where

$\boldsymbol{m}_1$	$\boldsymbol{m}_2$	$\boldsymbol{m}_3$
$(1, 2)$	$(1, 0)$	$(-1, -1)$

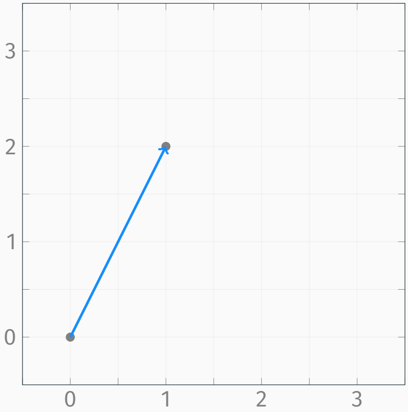
Introduced by Alur et al. to reason about problems related to green scheduling and energy peak-consumption reduction

Schedules and executions



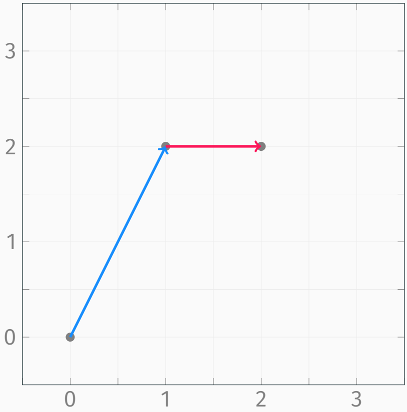
$\pi =$

Schedules and executions



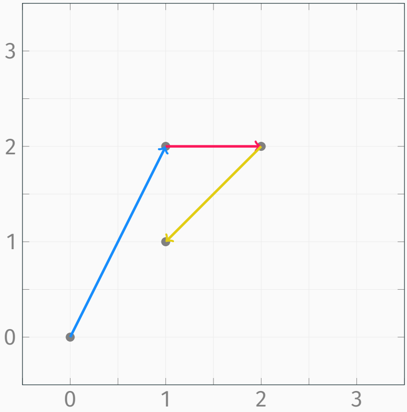
$\pi = 1.0\boldsymbol{m}_1$

Schedules and executions



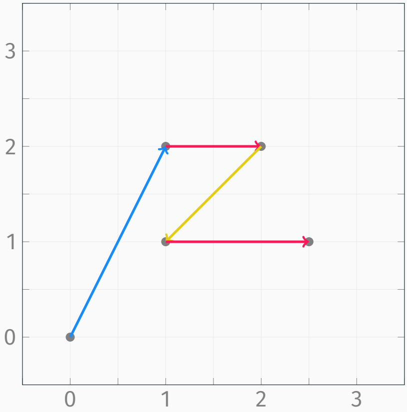
$\pi = 1.0\boldsymbol{m}_1 \; 1.0\boldsymbol{m}_2$

Schedules and executions



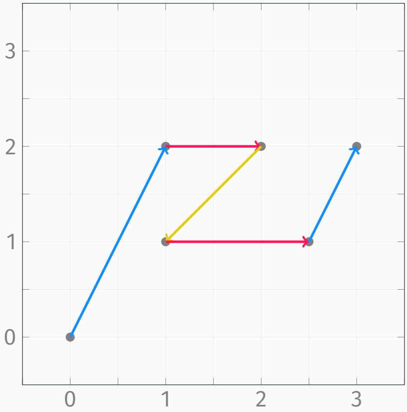
$\pi = 1.0\boldsymbol{m}_1\ 1.0\boldsymbol{m}_2\ 1.0\boldsymbol{m}_3$

Schedules and executions



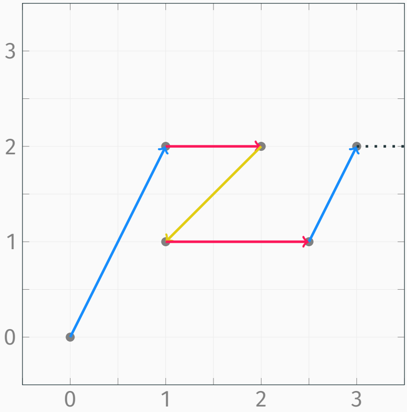
$\pi = 1.0\boldsymbol{m}_1\ 1.0\boldsymbol{m}_2\ 1.0\boldsymbol{m}_3\ 1.5\boldsymbol{m}_2$

Schedules and executions

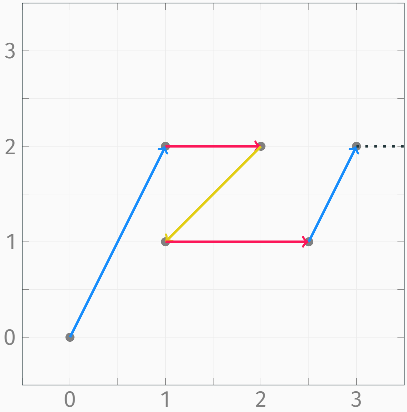


$$\pi = 1.0\boldsymbol{m}_1 \; 1.0\boldsymbol{m}_2 \; 1.0\boldsymbol{m}_3 \; 1.5\boldsymbol{m}_2 \; 0.5\boldsymbol{m}_1$$

Schedules and executions

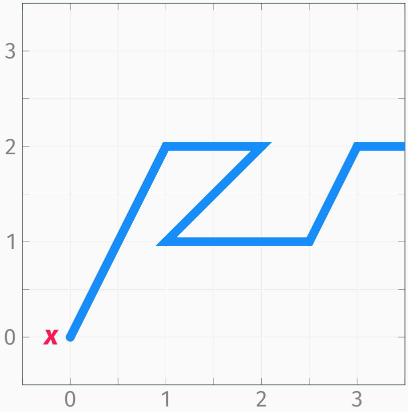


$$\pi = 1.0\boldsymbol{m}_1\ 1.0\boldsymbol{m}_2\ 1.0\boldsymbol{m}_3\ 1.5\boldsymbol{m}_2\ 0.5\boldsymbol{m}_1\ \cdots$$



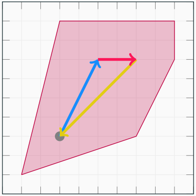
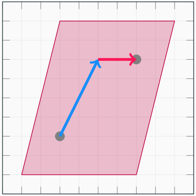
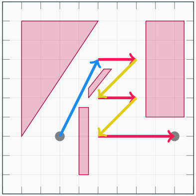
Schedule: $\pi = \alpha_1 \boldsymbol{m}_{i_1}\ \alpha_2 \boldsymbol{m}_{i_2}\ \cdots$ where $\alpha_i \in \mathbb{R}_{>0}$ and $\sum \alpha_i = \infty$

Schedules and executions

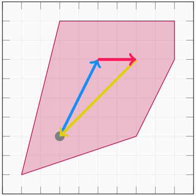
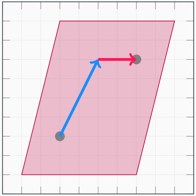
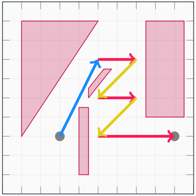


Execution: $\sigma : \mathbb{R}_{\geq 0} \to \mathbb{R}^d$ with $\sigma(0) = \boldsymbol{x}$

Known results

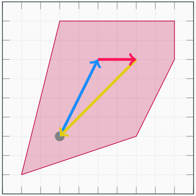
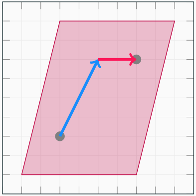
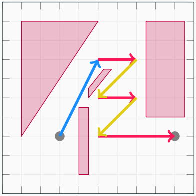
Safe scheduling	Safe reachability	Safe planning
Can always remain within a zone?	Can reach target within a zone?	Can reach target while avoiding obstacles?
		
PTIME-complete (Alur et al. HSCC'12)		Decidable (Krishna et al. ATVA'17)

Known results

Safe scheduling	Safe reachability	Safe planning
Can always remain within a zone?	Can reach target within a zone?	Can reach target while avoiding obstacles?
		
PTIME-complete* (Alur et al. HSCC'12)		Decidable* (Krishna et al. ATVA'17)

* When each zone is a bounded closed convex polytope

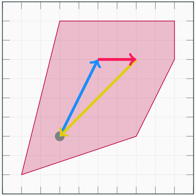
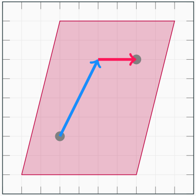
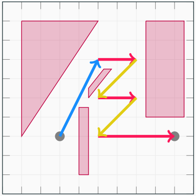
Known results

Safe scheduling	Safe reachability	Safe planning
Can always remain within a zone?	Can reach target within a zone?	Can reach target while avoiding obstacles?
		
PTIME-complete* (Alur et al. HSCC'12)		Decidable* (Krishna et al. ATVA'17)

* When each zone is a bounded closed convex polytope

$$\text{Zone} = \{\mathbf{x} \in \mathbb{R}^d : \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$$

Known results

Safe scheduling	Safe reachability	Safe planning
Can always remain within a zone?	Can reach target within a zone?	Can reach target while avoiding obstacles?
		
PTIME-complete* (Alur et al. HSCC'12)		Decidable (Krishna et al. ATVA'17)

* Safe reachability with Zone $= \mathbb{R}^d_{\geq 0}$ is also PTIME-complete by work on continuous VAS / Petri nets (Fraca and Haddad PN'13)

Our questions

- Can we unify these results?
- What are the decidable problems?
- What is their complexity?

Linear temporal logic (LTL): syntax

$$\varphi ::= \mathit{true} \mid Z \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathsf{F}\varphi \mid \mathsf{G}\varphi \mid \varphi \,\mathsf{U}\, \varphi$$

Linear temporal logic (LTL): syntax

$$\varphi ::= \mathit{true} \mid Z \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathsf{F}\varphi \mid \mathsf{G}\varphi \mid \varphi \; \mathsf{U} \; \varphi$$

Zones: closed convex polytopes

Linear temporal logic (LTL): syntax

$$\varphi ::= \mathit{true} \mid Z \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathsf{F}\varphi \mid \mathsf{G}\varphi \mid \varphi \;\mathsf{U}\; \varphi$$

Finally φ holds

Linear temporal logic (LTL): syntax

$$\varphi ::= \mathit{true} \mid Z \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathsf{F}\varphi \mid \mathsf{G}\varphi \mid \varphi\ \mathsf{U}\ \varphi$$

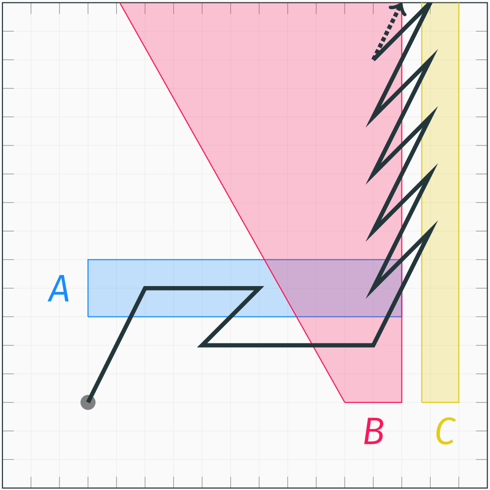
Globally φ holds

Linear temporal logic (LTL): syntax

$$\varphi ::= \mathit{true} \mid Z \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathsf{F}\varphi \mid \mathsf{G}\varphi \mid \varphi\ \mathsf{U}\ \varphi'$$

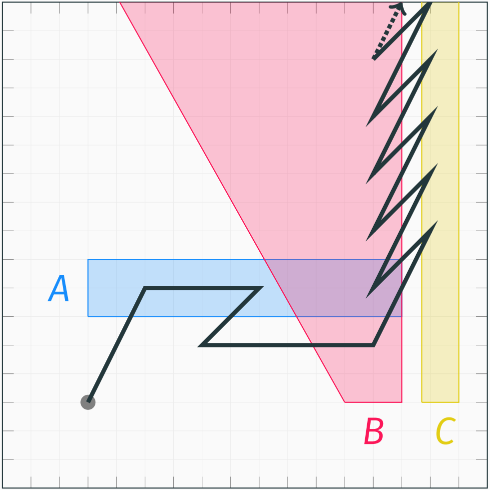
φ holds until φ' holds

Linear temporal logic (LTL): informal semantics



Linear temporal logic (LTL): informal semantics

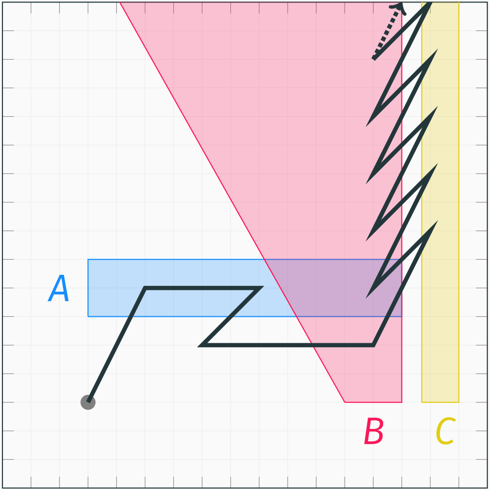
$\sigma \models \mathsf{F}\, A$



Linear temporal logic (LTL): informal semantics

$\sigma \models \mathsf{F}\, A$

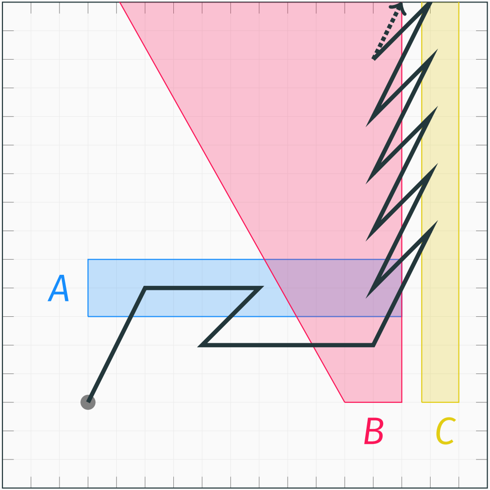
$\sigma \not\models \mathsf{G}\, A$



$\sigma \models \mathsf{F}\, A$

$\sigma \not\models \mathsf{G}\, A$

$\sigma \models \mathsf{F}\,(A \wedge \mathsf{F}\, B)$



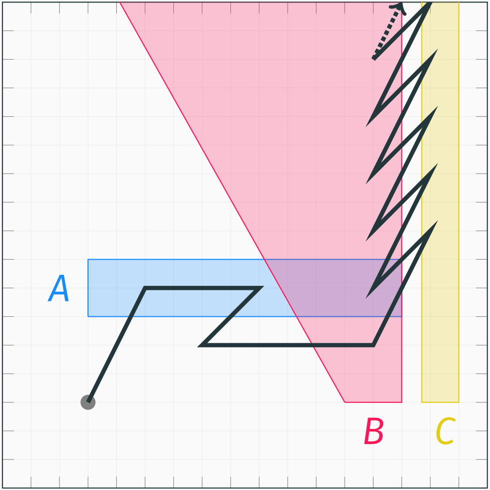
Linear temporal logic (LTL): informal semantics

$\sigma \models \mathsf{F}\, A$

$\sigma \not\models \mathsf{G}\, A$

$\sigma \models \mathsf{F}\,(A \wedge \mathsf{F}\, B)$

$\sigma \not\models \mathsf{F}\,(B \wedge \mathsf{F}\,(A \wedge \neg B))$



Linear temporal logic (LTL): informal semantics

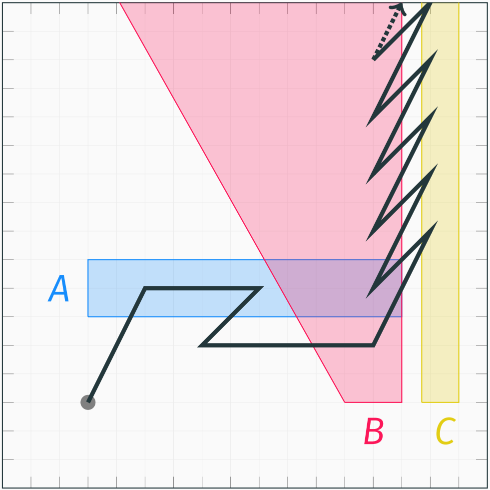
$\sigma \models \mathsf{F}\, A$

$\sigma \not\models \mathsf{G}\, A$

$\sigma \models \mathsf{F}\,(A \land \mathsf{F}\, B)$

$\sigma \not\models \mathsf{F}\,(B \land \mathsf{F}\,(A \land \neg B))$

$\sigma \models (\neg C)\ \mathsf{U}\ B$



Linear temporal logic (LTL): informal semantics

$\sigma \models \mathsf{F}\, A$

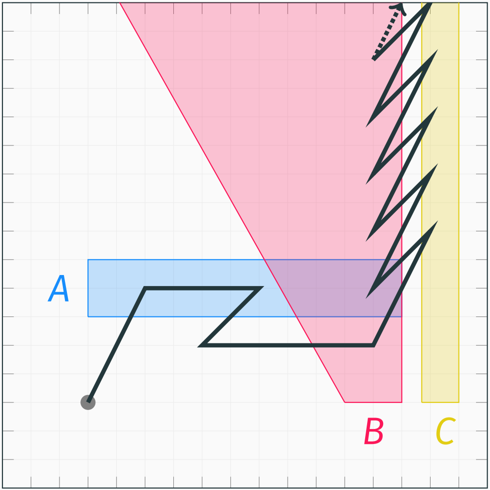
$\sigma \not\models \mathsf{G}\, A$

$\sigma \models \mathsf{F}\,(A \wedge \mathsf{F}\, B)$

$\sigma \not\models \mathsf{F}\,(B \wedge \mathsf{F}\,(A \wedge \neg B))$

$\sigma \models (\neg C)\ \mathsf{U}\ B$

$\sigma \not\models A\ \mathsf{U}\ B$



Linear temporal logic (LTL): informal semantics

$\sigma \models \mathsf{F}\, A$

$\sigma \not\models \mathsf{G}\, A$

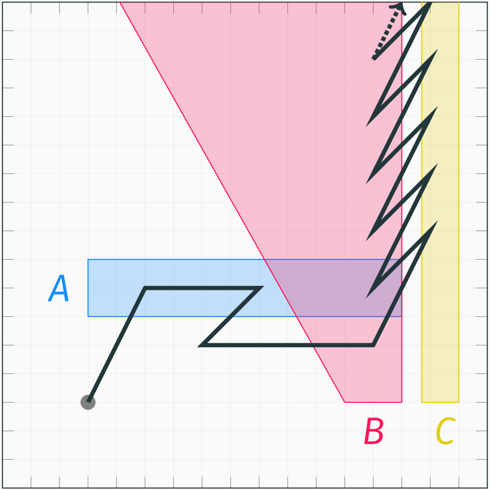
$\sigma \models \mathsf{F}\,(A \wedge \mathsf{F}\, B)$

$\sigma \not\models \mathsf{F}\,(B \wedge \mathsf{F}\,(A \wedge \neg B))$

$\sigma \models (\neg C)\ \mathsf{U}\ B$

$\sigma \not\models A\ \mathsf{U}\ B$

$\sigma \models (\mathsf{GF}\, B) \wedge (\mathsf{GF}\, C)$



Linear temporal logic (LTL): informal semantics

$\sigma \models \mathsf{F}\, A$

$\sigma \not\models \mathsf{G}\, A$

$\sigma \models \mathsf{F}\,(A \wedge \mathsf{F}\, B)$

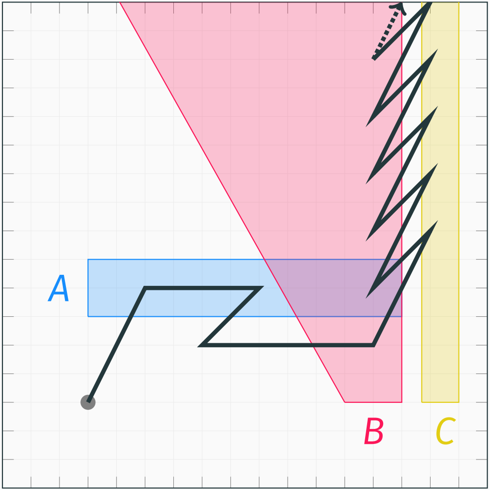
$\sigma \not\models \mathsf{F}\,(B \wedge \mathsf{F}\,(A \wedge \neg B))$

$\sigma \models (\neg C)\ \mathsf{U}\ B$

$\sigma \not\models A\ \mathsf{U}\ B$

$\sigma \models (\mathsf{GF}\, B) \wedge (\mathsf{GF}\, C)$

$\sigma \not\models \mathsf{GF}(B \wedge C)$



Linear temporal logic (LTL): formal semantics

$$\sigma, \tau \models \mathit{true} \iff \mathit{true}$$

$$\sigma, \tau \models Z \iff \sigma(\tau) \in Z$$

Time $\tau \in \mathbb{R}_{\geq 0}$

Execution $\sigma : \mathbb{R}_{\geq 0} \to \mathbb{R}^d$

Linear temporal logic (LTL): formal semantics

$$\sigma, \tau \models \mathit{true} \iff \mathit{true}$$

$$\sigma, \tau \models Z \iff \sigma(\tau) \in Z$$

$$\sigma, \tau \models \neg\varphi \iff \neg(\sigma, \tau \models \varphi)$$

Linear temporal logic (LTL): formal semantics

$$\begin{aligned}
\sigma, \tau \models \mathit{true} &\iff \mathit{true} \\
\sigma, \tau \models Z &\iff \sigma(\tau) \in Z \\
\sigma, \tau \models \neg\varphi &\iff \neg(\sigma, \tau \models \varphi) \\
\sigma, \tau \models \varphi \wedge \varphi' &\iff (\sigma, \tau \models \varphi) \wedge (\sigma, \tau \models \varphi')
\end{aligned}$$

Linear temporal logic (LTL): formal semantics

$$
\begin{aligned}
\sigma, \tau \models \mathit{true} &\iff \mathit{true} \\
\sigma, \tau \models Z &\iff \sigma(\tau) \in Z \\
\sigma, \tau \models \neg\varphi &\iff \neg(\sigma, \tau \models \varphi) \\
\sigma, \tau \models \varphi \wedge \varphi' &\iff (\sigma, \tau \models \varphi) \wedge (\sigma, \tau \models \varphi') \\
\sigma, \tau \models \varphi \vee \varphi' &\iff (\sigma, \tau \models \varphi) \vee (\sigma, \tau \models \varphi')
\end{aligned}
$$

Linear temporal logic (LTL): formal semantics

$$
\begin{aligned}
\sigma, \tau \models \mathit{true} &\iff \mathit{true} \\
\sigma, \tau \models Z &\iff \sigma(\tau) \in Z \\
\sigma, \tau \models \neg\varphi &\iff \neg(\sigma, \tau \models \varphi) \\
\sigma, \tau \models \varphi \wedge \varphi' &\iff (\sigma, \tau \models \varphi) \wedge (\sigma, \tau \models \varphi') \\
\sigma, \tau \models \varphi \vee \varphi' &\iff (\sigma, \tau \models \varphi) \vee (\sigma, \tau \models \varphi') \\
\sigma, \tau \models \mathsf{F}\varphi &\iff \exists \tau' \geq \tau : \sigma, \tau' \models \varphi
\end{aligned}
$$

Linear temporal logic (LTL): formal semantics

$$
\begin{aligned}
\sigma, \tau \models \mathit{true} &\iff \mathit{true} \\
\sigma, \tau \models Z &\iff \sigma(\tau) \in Z \\
\sigma, \tau \models \neg\varphi &\iff \neg(\sigma, \tau \models \varphi) \\
\sigma, \tau \models \varphi \wedge \varphi' &\iff (\sigma, \tau \models \varphi) \wedge (\sigma, \tau \models \varphi') \\
\sigma, \tau \models \varphi \vee \varphi' &\iff (\sigma, \tau \models \varphi) \vee (\sigma, \tau \models \varphi') \\
\sigma, \tau \models \mathsf{F}\varphi &\iff \exists \tau' \geq \tau : \sigma, \tau' \models \varphi \\
\sigma, \tau \models \mathsf{G}\varphi &\iff \forall \tau' \geq \tau : \sigma, \tau' \models \varphi
\end{aligned}
$$

Linear temporal logic (LTL): formal semantics

$$\begin{aligned}
\sigma, \tau \models \mathit{true} &\iff \mathit{true} \\
\sigma, \tau \models Z &\iff \sigma(\tau) \in Z \\
\sigma, \tau \models \neg\varphi &\iff \neg(\sigma, \tau \models \varphi) \\
\sigma, \tau \models \varphi \wedge \varphi' &\iff (\sigma, \tau \models \varphi) \wedge (\sigma, \tau \models \varphi') \\
\sigma, \tau \models \varphi \vee \varphi' &\iff (\sigma, \tau \models \varphi) \vee (\sigma, \tau \models \varphi') \\
\sigma, \tau \models \mathsf{F}\varphi &\iff \exists \tau' \geq \tau : \sigma, \tau' \models \varphi \\
\sigma, \tau \models \mathsf{G}\varphi &\iff \forall \tau' \geq \tau : \sigma, \tau' \models \varphi \\
\sigma, \tau \models \varphi \,\mathsf{U}\, \varphi' &\iff \exists \tau' \geq \tau : (\sigma, \tau' \models \varphi') \wedge (\forall \tau'' \in [\tau, \tau') : \sigma, \tau'' \models \varphi)
\end{aligned}$$

Linear temporal logic (LTL): formal semantics

$$\sigma \models \varphi \iff \sigma, 0 \models \varphi$$

Model checking: problem

$\boldsymbol{x} \models_M \varphi$ iff there is a schedule π of M such that $\mathrm{exec}(\pi, \boldsymbol{x}) \models \varphi$

Model checking: problem

$\boldsymbol{x} \models_M \varphi$ iff there is a schedule π of M such that $\mathrm{exec}(\pi, \boldsymbol{x}) \models \varphi$

Model checking problem

Given: *MMS M, initial point $\boldsymbol{x}$, LTL formula φ*

Decide: *whether $\boldsymbol{x} \models_M \varphi$*

Model checking: problem

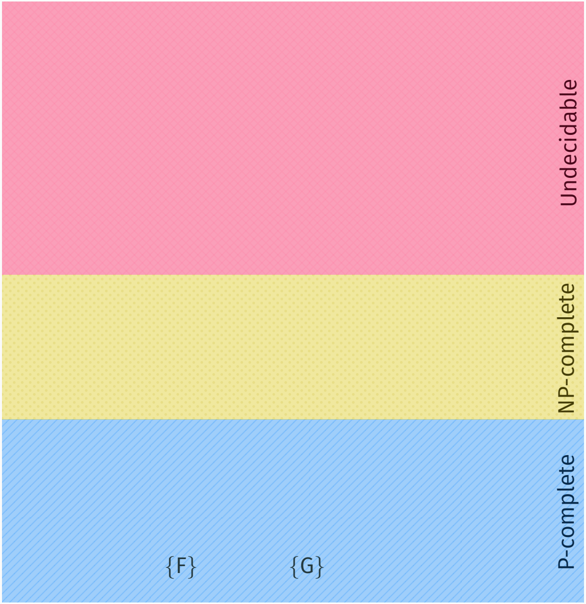
$\boldsymbol{x} \models_M \varphi$ iff there is a schedule π of M such that $\mathrm{exec}(\pi, \boldsymbol{x}) \models \varphi$

Model checking problem$_{X \subseteq \{F, G, U, \wedge, \vee, \neg\}}$

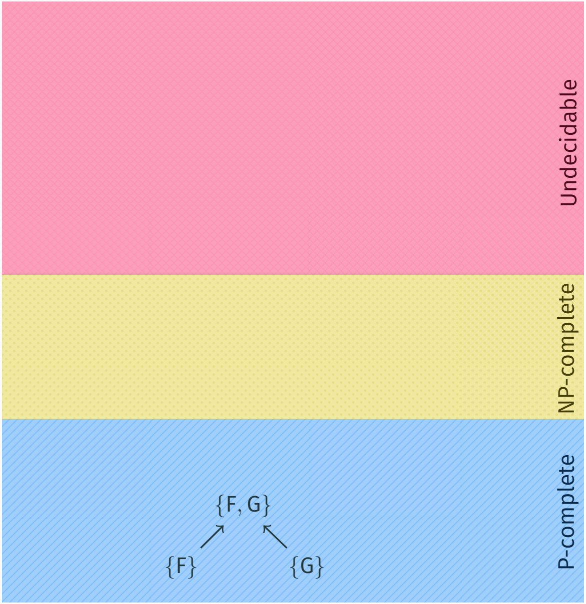
Given: *MMS M, initial point $\boldsymbol{x}$,*
LTL formula φ with operators only from X

Decide: *whether $\boldsymbol{x} \models_M \varphi$*

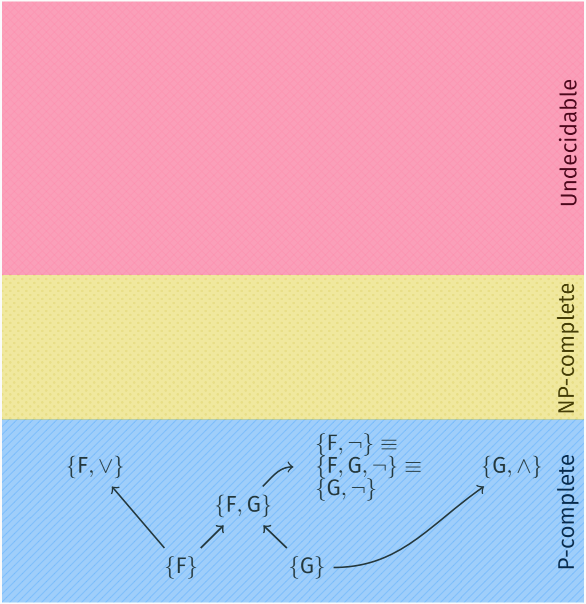
Model checking: our contribution



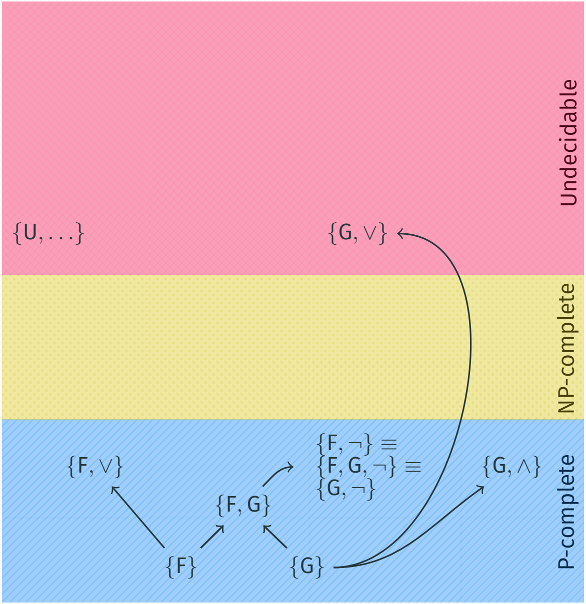
Model checking: our contribution



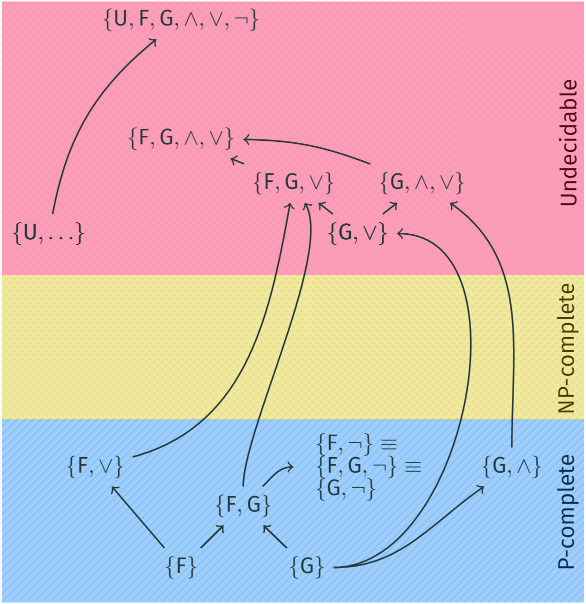
Model checking: our contribution



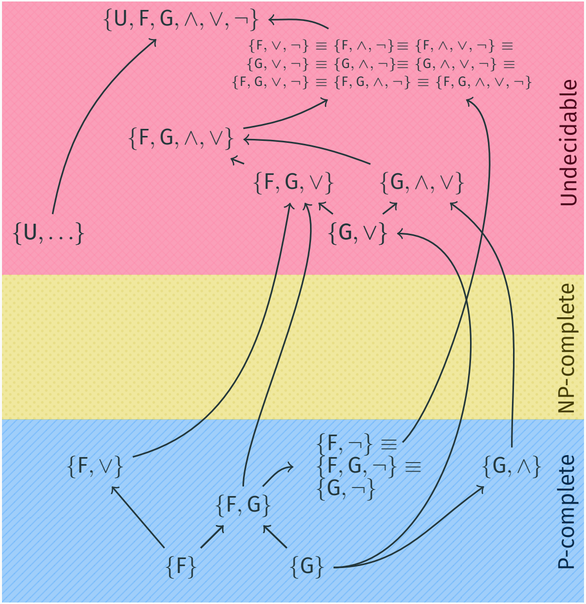
Model checking: our contribution



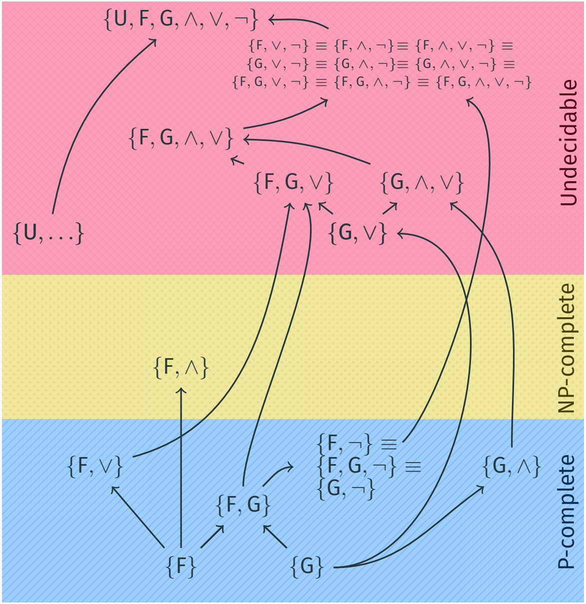
Model checking: our contribution



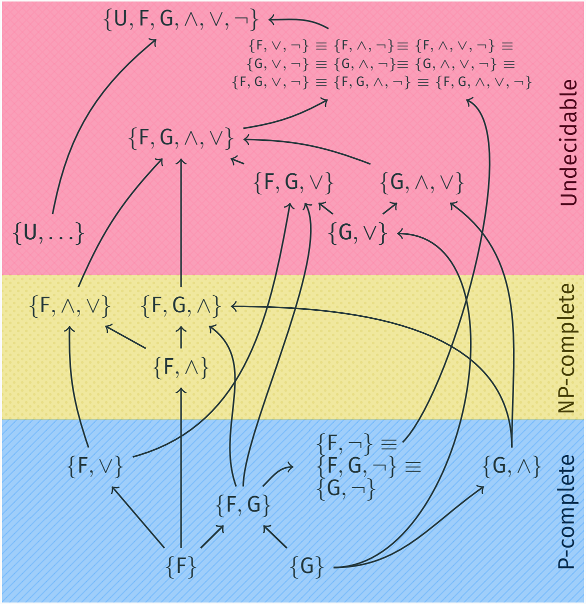
Model checking: our contribution



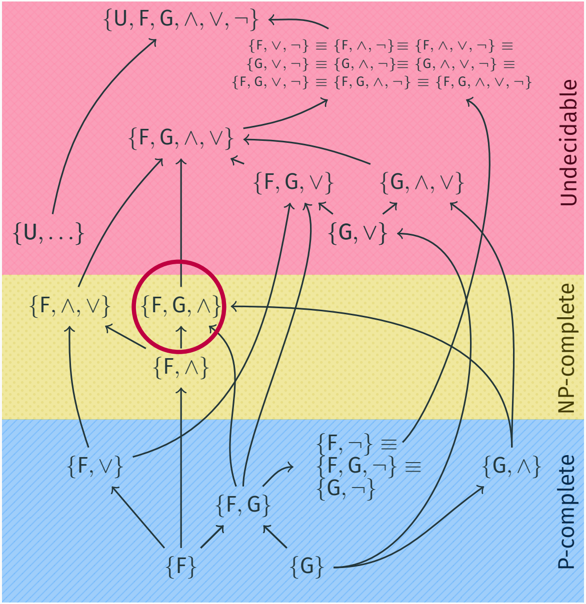
Model checking: our contribution



Model checking: our contribution



Model checking: our contribution



$\{\mathbf{F}, \mathbf{G}, \land\} \in$ NP: overview

1. Flatten formula φ

$$\varphi \xrightarrow{①} \text{flat}(\varphi)$$

$\{\mathbf{F}, \mathbf{G}, \wedge\} \in$ NP: overview

1. Flatten formula φ
2. Convert into an almost acyclic automaton $\mathcal{A}_\varphi$

$$\varphi \xrightarrow{①} \text{flat}(\varphi) \xrightarrow{②} \mathcal{A}_\varphi$$

$\{\mathbf{F}, \mathbf{G}, \wedge\} \in$ NP: overview

1. Flatten formula φ
2. Convert into an almost acyclic automaton $\mathcal{A}_\varphi$
3. Nondeterministically guess a "path" π of $\mathcal{A}_\varphi$

$$\varphi \xrightarrow{①} \text{flat}(\varphi) \xrightarrow{②} \mathcal{A}_\varphi \xrightarrow{\quad③\quad} \pi$$

$\{\mathbf{F}, \mathbf{G}, \wedge\} \in$ NP: overview

1. Flatten formula φ
2. Convert into an almost acyclic automaton $\mathcal{A}_\varphi$
3. Nondeterministically guess a "path" π of $\mathcal{A}_\varphi$
4. Convert π into a "linear" LTL formula φ'

$$\varphi \xrightarrow{①} \mathrm{flat}(\varphi) \xrightarrow{②} \mathcal{A}_\varphi \xrightarrow{③} \pi \xrightarrow{④} \varphi'$$

$\{\mathbf{F}, \mathbf{G}, \wedge\} \in$ NP: overview

1. Flatten formula φ
2. Convert into an almost acyclic automaton $\mathcal{A}_\varphi$
3. Nondeterministically guess a “path” π of $\mathcal{A}_\varphi$
4. Convert π into a “linear” LTL formula φ'
5. Construct a first-order formula ψ s.t. $\psi(\boldsymbol{x}) \leftrightarrow \boldsymbol{x} \models_M \varphi'$

$$\varphi \xrightarrow{①} \mathrm{flat}(\varphi) \xrightarrow{②} \mathcal{A}_\varphi \xrightarrow{③} \pi \xrightarrow{④} \varphi' \xrightarrow{⑤} \psi$$

$\{\mathbf{F}, \mathbf{G}, \wedge\} \in$ NP: overview

1. Flatten formula φ
2. Convert into an almost acyclic automaton $\mathcal{A}_\varphi$
3. Nondeterministically guess a “path” π of $\mathcal{A}_\varphi$
4. Convert π into a “linear” LTL formula φ'
5. Construct a first-order formula ψ s.t. $\psi(\boldsymbol{x}) \leftrightarrow \boldsymbol{x} \models_M \varphi'$
6. Check whether $\psi(\boldsymbol{x})$ holds (in polynomial time)

$$\varphi \xrightarrow{①} \mathrm{flat}(\varphi) \xrightarrow{②} \mathcal{A}_\varphi \xrightarrow{③} \pi \xrightarrow{④} \varphi' \xrightarrow{⑤} \psi \;⑥ \begin{cases} \nearrow \text{Yes} \\ \searrow \text{No} \end{cases}$$

① Formula flattening

Definition

An LTL formula is *flat* if it can be derived from φ in

$$\varphi ::= \mathrm{goal} \mid \mathsf{G}\,\mathrm{goal} \mid \mathsf{GF}\,\mathrm{goal} \mid \mathsf{F}\,\varphi \mid \varphi \wedge \varphi$$
$$\mathrm{goal} ::= \mathit{true} \mid Z \mid \mathrm{goal} \wedge \mathrm{goal}$$

① Formula flattening

Definition

An LTL formula is *flat* if it can be derived from φ in

$$\varphi ::= \mathrm{goal} \mid \mathsf{G}\,\mathrm{goal} \mid \mathsf{GF}\,\mathrm{goal} \mid \mathsf{F}\,\varphi \mid \varphi \wedge \varphi$$

$$\mathrm{goal} ::= \mathit{true} \mid Z \mid \mathrm{goal} \wedge \mathrm{goal}$$

Intuition: only F
can nest complex goals

① Formula flattening

Definition

An LTL formula is *flat* if it can be derived from φ in

$$\varphi ::= \mathrm{goal} \mid \mathsf{G}\,\mathrm{goal} \mid \mathsf{GF}\,\mathrm{goal} \mid \mathsf{F}\,\varphi \mid \varphi \wedge \varphi$$

$$\mathrm{goal} ::= \mathit{true} \mid Z \mid \mathrm{goal} \wedge \mathrm{goal}$$

Example

Formula	Equivalent flat formula
$\mathsf{GF}(A \wedge \mathsf{G}\,B \wedge \mathsf{F}\,C)$	$\mathsf{GF}\,A \wedge \mathsf{FG}\,B \wedge \mathsf{GF}\,C$

① Formula flattening

Theorem

For every $\varphi \in$ *LTL*$(F, G, \wedge)$
There is an equivalent flat formula $\mathrm{flat}(\varphi)$ *of linear size*

① Formula flattening

Theorem

For every $\varphi \in LTL(F, G, \wedge)$
There is an equivalent flat formula $\mathrm{flat}(\varphi)$ *of linear size*

Proof.

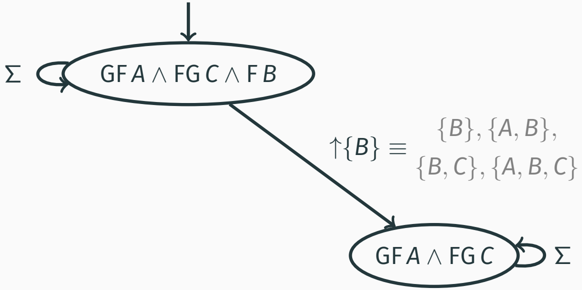
Follows by simple recursive rewriting rules □

② From flat formulas to almost acyclic automata

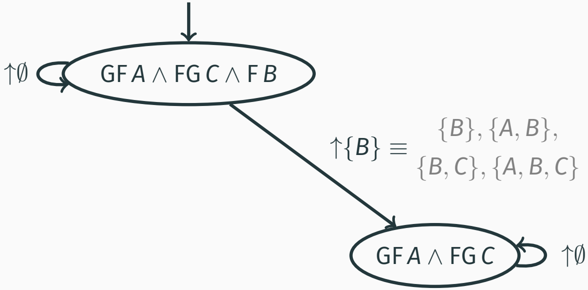
$$\mathsf{GF}\,A \wedge \mathsf{FG}\,C \wedge \mathsf{F}\,B$$

② From flat formulas to almost acyclic automata

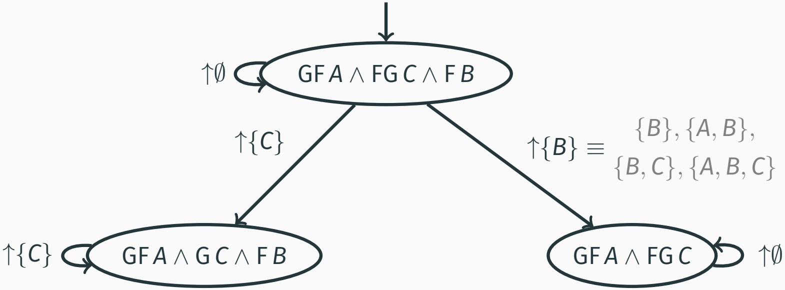
② From flat formulas to almost acyclic automata



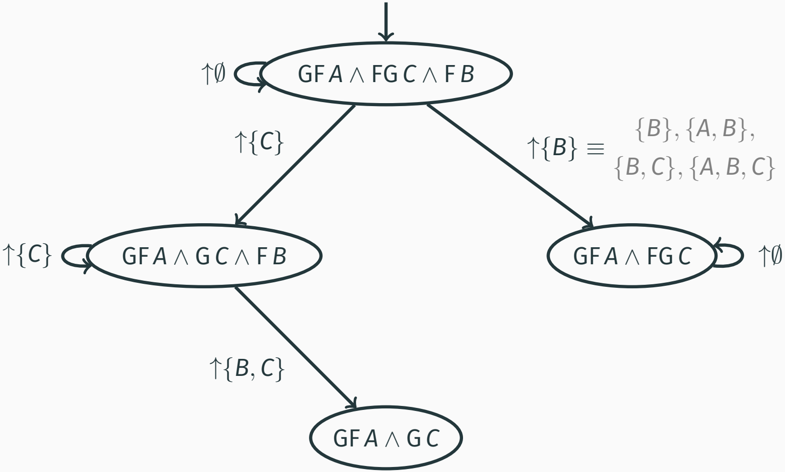
② From flat formulas to almost acyclic automata



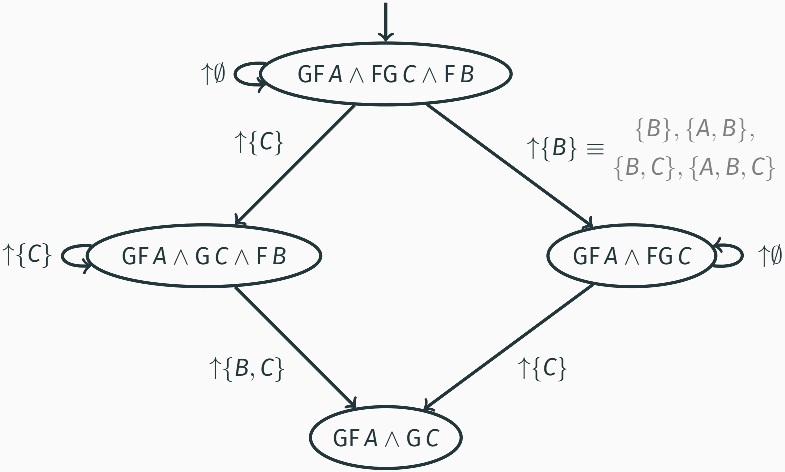
② From flat formulas to almost acyclic automata



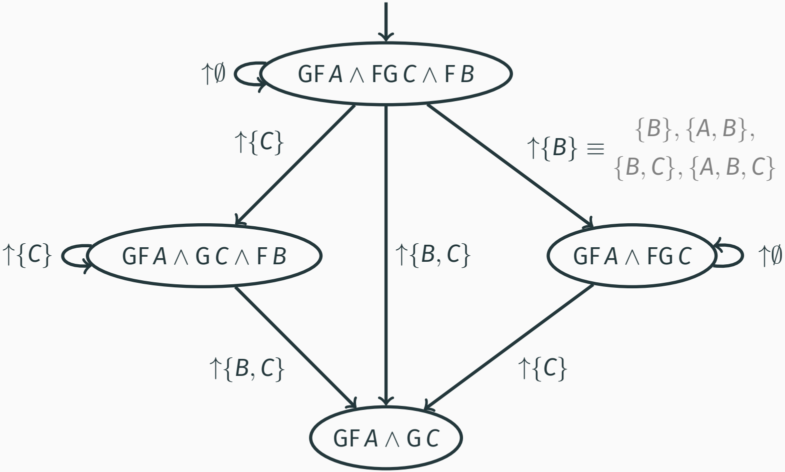
② From flat formulas to almost acyclic automata



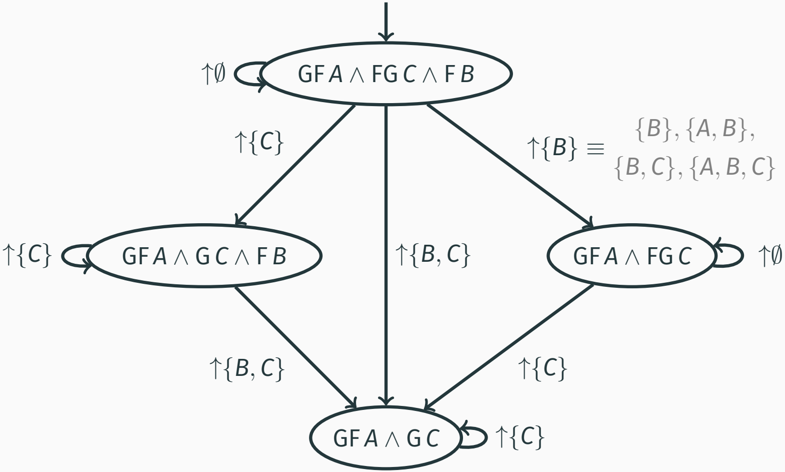
② From flat formulas to almost acyclic automata



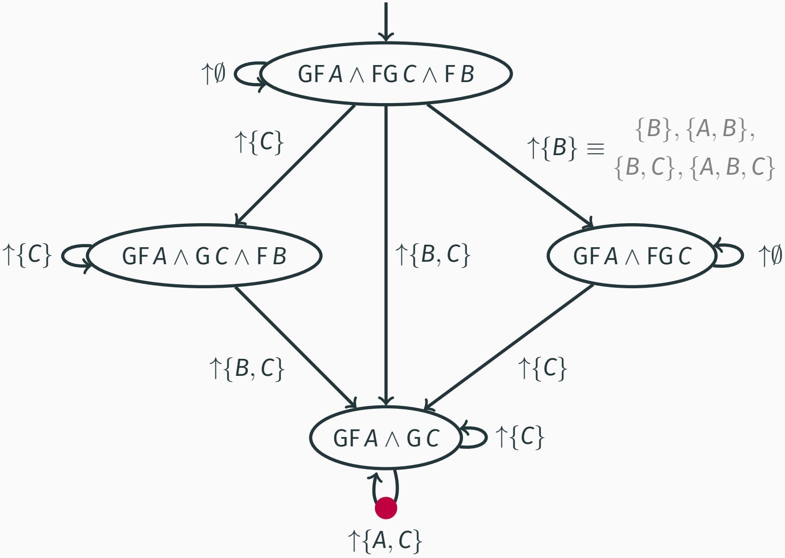
② From flat formulas to almost acyclic automata



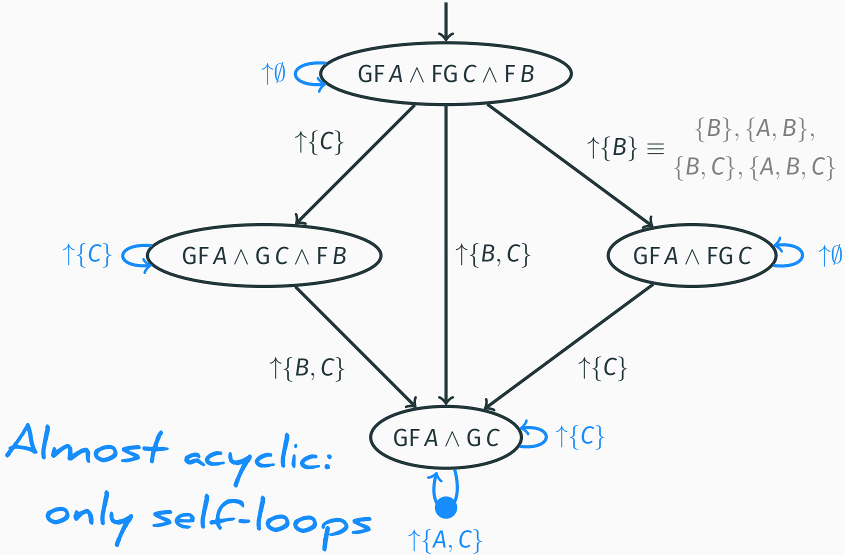
② From flat formulas to almost acyclic automata



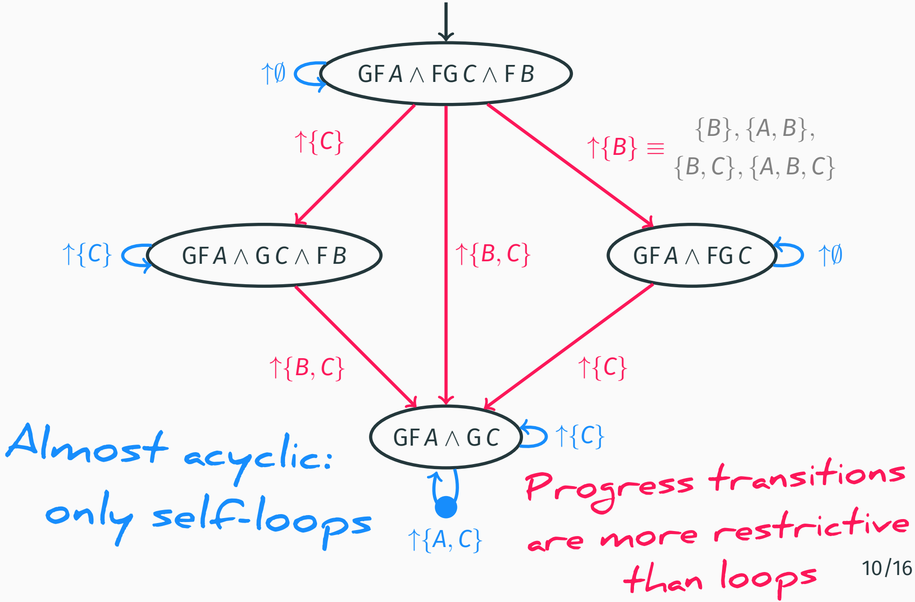
② From flat formulas to almost acyclic automata



② From flat formulas to almost acyclic automata



② From flat formulas to almost acyclic automata



② From flat formulas to almost acyclic automata

Theorem

For every formula $\varphi \in LTL(F, G, \wedge)$
There is an almost acylic automaton $\mathcal{A}_\varphi$ s.t.

- $\text{language}(\mathcal{A}_\varphi) \equiv \varphi$
- $\text{width}(\mathcal{A}_\varphi) \in \mathcal{O}(|\varphi|)$
- *transitions have "good properties"*

Proof.

Inspired by unfoldings of Křetínský and Esparza CAV'12 □

② From flat formulas to almost acyclic automata

Theorem

For every formula $\varphi \in LTL(F, G, \wedge)$
There is an almost acylic automaton $\mathcal{A}_\varphi$ *s.t.*

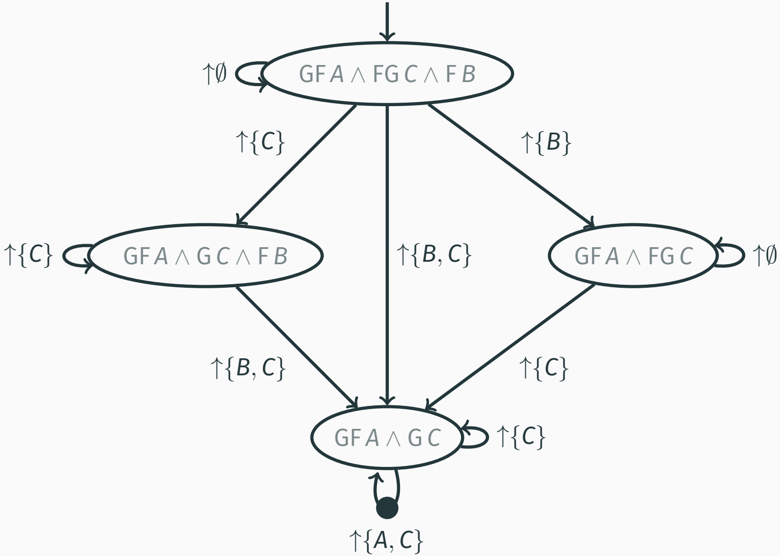
- $\text{language}(\mathcal{A}_\varphi) \equiv \varphi$
- $\text{width}(\mathcal{A}_\varphi) \in \mathcal{O}(|\varphi|)$
- *transitions have "good properties"*

(Generalized Büchi automaton with transition-acceptance)

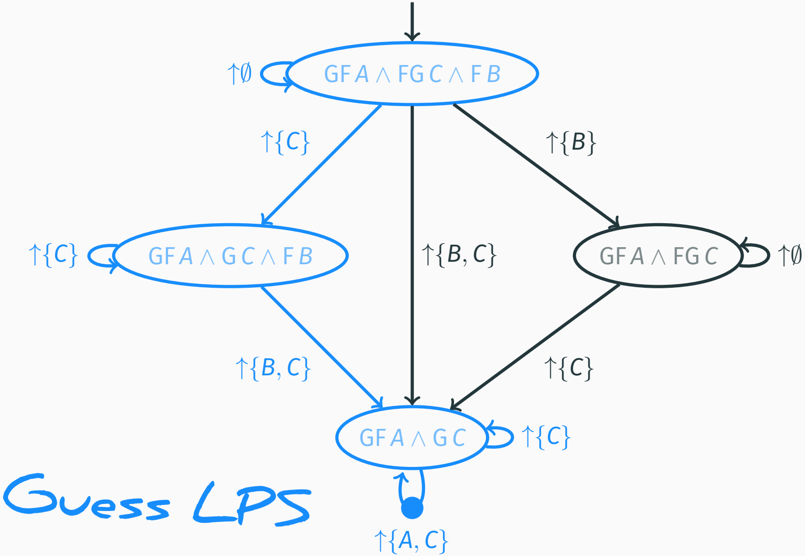
Proof.

Inspired by unfoldings of Křetínský and Esparza CAV'12 □

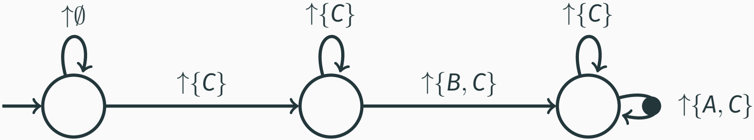
③ Linear path schemes (LPS)



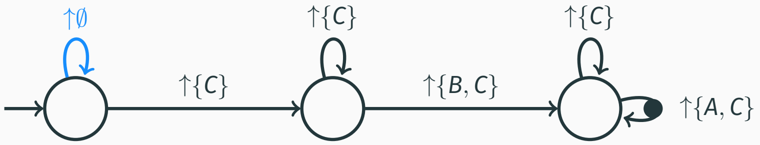
③ Linear path schemes (LPS)



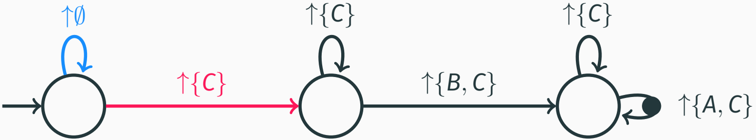
④ From LPS back to linear LTL



④ From LPS back to linear LTL

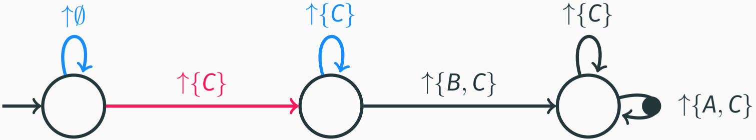


$$\mathit{true}\ \mathsf{U}\ (C \wedge (C\ \mathsf{U}\ ((B \wedge C) \wedge (\mathsf{G}\mathsf{F}\, A \wedge \mathsf{G}\, C))))$$



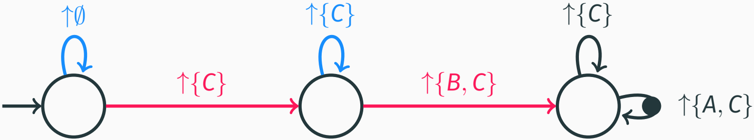
$$\mathit{true}\ \mathsf{U}\ (C \wedge (C\ \mathsf{U}\ ((B \wedge C) \wedge (\mathsf{GF}\,A \wedge \mathsf{G}\,C))))$$

④ From LPS back to linear LTL

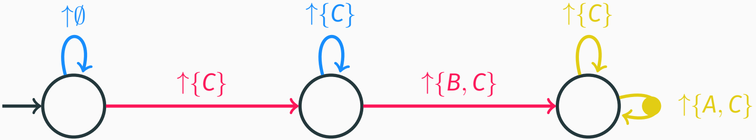


$$\mathit{true}\ \mathsf{U}\ (C \wedge (C\ \mathsf{U}\ ((B \wedge C) \wedge (\mathsf{G}\mathsf{F}\, A \wedge \mathsf{G}\, C))))$$

④ From LPS back to linear LTL



$$true \; \mathsf{U} \; (C \wedge (C \; \mathsf{U} \; ((B \wedge C) \wedge (\mathsf{GF}\, A \wedge \mathsf{G}\, C))))$$



$$\mathit{true}\ \mathrm{U}\ (C \wedge (C\ \mathrm{U}\ ((B \wedge C) \wedge (\mathrm{GF}\, A \wedge \mathrm{G}\, C))))$$

④ From LPS back to linear LTL

$$\boldsymbol{x} \models_M \mathit{true} \; \mathrm{U} \; (C \wedge (C \; \mathrm{U} \; ((B \wedge C) \wedge (\mathrm{GF}\, A \wedge \mathrm{G}\, C))))$$

④ From LPS back to linear LTL

$$\boldsymbol{x} \models_M \mathit{true}\ \mathrm{U}\ (C \wedge (C\ \mathrm{U}\ ((B \wedge C) \wedge (\mathrm{GF}\, A \wedge \mathrm{G}\, C))))$$

iff

$$\exists \boldsymbol{y} \in C,\ \boldsymbol{z} \in B \cap C : \boldsymbol{x} \rightarrow^* \boldsymbol{y} \rightarrow^*_C \boldsymbol{z} \text{ and } \boldsymbol{z} \models_M \mathrm{GF}\, A \wedge \mathrm{G}\, C$$

Convex semi-linear Horn formulas

Such a formula can be checked in polynomial time:

$$\exists \boldsymbol{x} \in \mathbb{R}^d : \bigwedge_i \boldsymbol{a}_i \boldsymbol{x} \sim_i b_i$$

with $\sim_i \in \{<, \leq, =, \geq, >\}$

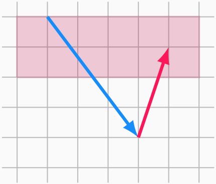
Convex semi-linear Horn formulas **(B. and Haase LICS'17)**

Such a formula can be checked in polynomial time:

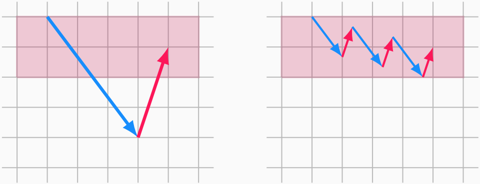
$$\exists \boldsymbol{x} \in \mathbb{R}^d, \boldsymbol{x}' \in \mathbb{R}^{d'}_{\geq 0} : \bigwedge_i \left(\boldsymbol{a}_i \boldsymbol{x} + \boldsymbol{a}'_i \boldsymbol{x}' \sim_i b_i \vee \bigvee_j \bigwedge_k \boldsymbol{x}'(k) > 0 \right)$$

with $\sim_i \in \{<, \leq, =, \geq, >\}$

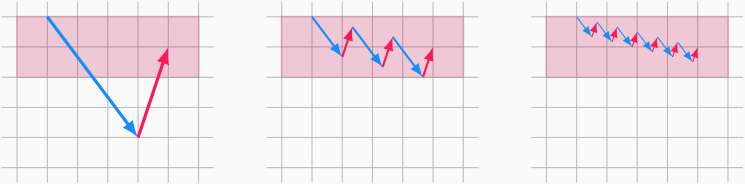
⑤ From linear LTL to FO: safe reachability



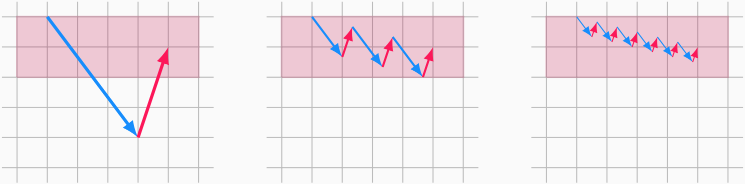
⑤ From linear LTL to FO: safe reachability



⑤ From linear LTL to FO: safe reachability



⑤ From linear LTL to FO: safe reachability



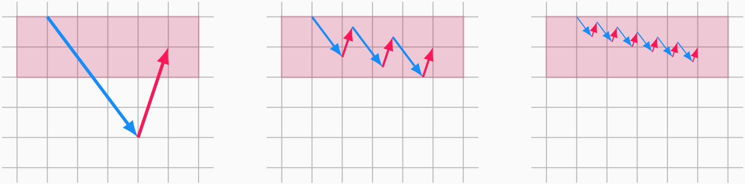
Proposition (generalization from Fraca and Haddad PN'13)

We have $\boldsymbol{x} \rightarrow_Z^* \boldsymbol{y}$ iff there exist schedules $\pi, \pi_{\text{fwd}}, \pi_{\text{bwd}}$ with

(i) $\boldsymbol{x} \rightarrow^{\pi} \boldsymbol{y}$ (ii) $\boldsymbol{x} \rightarrow_Z^{\pi_{\text{fwd}}} \cdot$ (iii) $\cdot \rightarrow_Z^{\pi_{\text{bwd}}} \boldsymbol{y}$

and $\text{supp}(\pi) = \text{supp}(\pi_{\text{fwd}}) = \text{supp}(\pi_{\text{bwd}})$

⑤ From linear LTL to FO: safe reachability



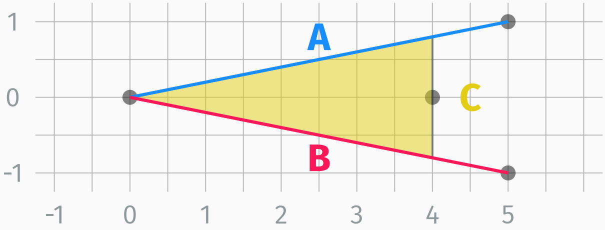
Proposition **(generalization from B. and Haase LICS'17)**

There is a convex semi-linear Horn formulas ψ_Z s.t.

$$\psi_Z(\boldsymbol{x}, \boldsymbol{y}) \iff \boldsymbol{x} \to_Z^* \boldsymbol{y}$$

⑤ From linear LTL to FO: safe repeated reachability

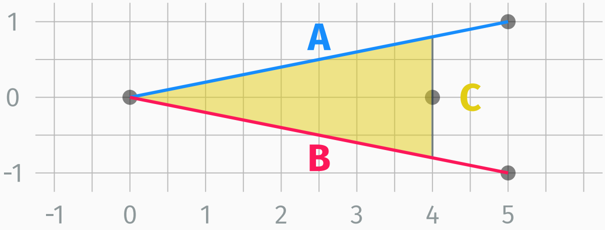
$$\mathsf{GF}\,A \wedge \mathsf{GF}\,B \wedge \mathsf{G}\,C$$



$$\boldsymbol{m}_1 = (0, 0, 1) \qquad \boldsymbol{m}_2 = (0, 0, -1)$$

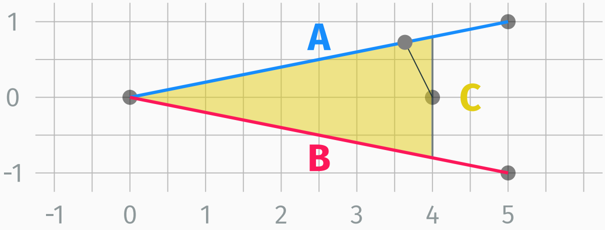
$$\boldsymbol{m}_3 = (-1, 2, 0) \qquad \boldsymbol{m}_4 = (-1, -2, 0)$$

$$\mathrm{GF}\,A \wedge \mathrm{GF}\,B \wedge \mathrm{G}\,C$$



⑤ From linear LTL to FO: safe repeated reachability

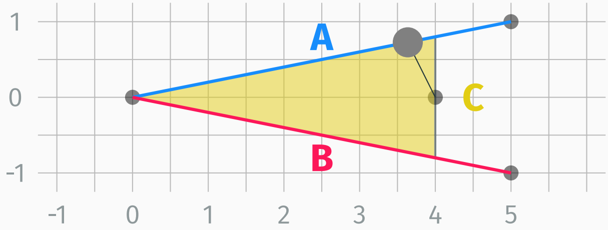
$$\mathsf{GF}\,A \wedge \mathsf{GF}\,B \wedge \mathsf{G}\,C$$



$\frac{4}{11}\boldsymbol{m}_3$

⑤ From linear LTL to FO: safe repeated reachability

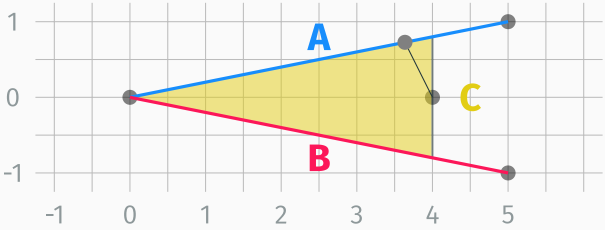
$$\mathsf{GF}\,A \wedge \mathsf{GF}\,B \wedge \mathsf{G}\,C$$



$\frac{4}{11}\boldsymbol{m}_3\ \boldsymbol{m}_1$

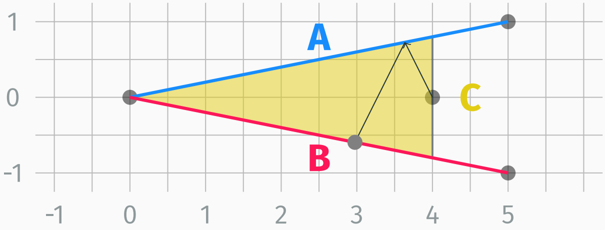
⑤ From linear LTL to FO: safe repeated reachability

$$\mathsf{GF}\,A \wedge \mathsf{GF}\,B \wedge \mathsf{G}\,C$$



$\frac{4}{11}\boldsymbol{m}_3\,\boldsymbol{m}_1\,\boldsymbol{m}_2$

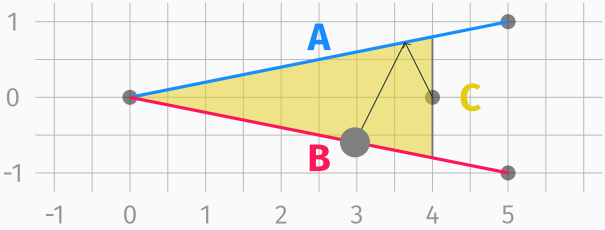
$$\mathsf{GF}\, A \wedge \mathsf{GF}\, B \wedge \mathsf{G}\, C$$



$$\frac{4}{11}\boldsymbol{m}_3\ \boldsymbol{m}_1\ \boldsymbol{m}_2\ \frac{80}{121}\boldsymbol{m}_4$$

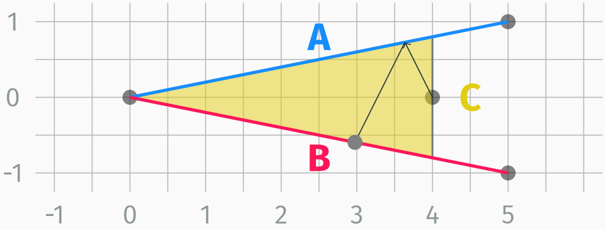
⑤ From linear LTL to FO: safe repeated reachability

$$\mathsf{GF}\,A \wedge \mathsf{GF}\,B \wedge \mathsf{G}\,C$$



$$\tfrac{4}{11}\boldsymbol{m}_3\ \boldsymbol{m}_1\ \boldsymbol{m}_2\ \tfrac{80}{121}\boldsymbol{m}_4\ \boldsymbol{m}_1$$

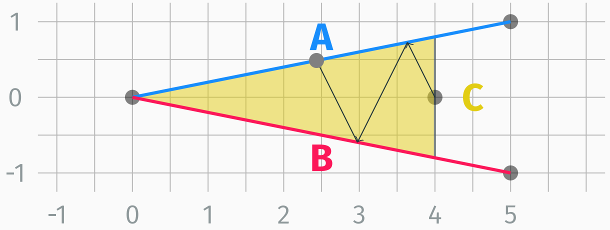
$$\mathrm{GF}\,A \wedge \mathrm{GF}\,B \wedge \mathrm{G}\,C$$



$$\tfrac{4}{11}\boldsymbol{m}_3\, \boldsymbol{m}_1\, \boldsymbol{m}_2\, \tfrac{80}{121}\boldsymbol{m}_4\, \boldsymbol{m}_1\, \boldsymbol{m}_2$$

⑤ From linear LTL to FO: safe repeated reachability

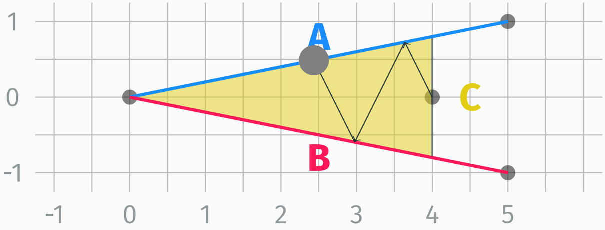
$$\mathsf{GF}\,A \wedge \mathsf{GF}\,B \wedge \mathsf{G}\,C$$



$$\tfrac{4}{11}\boldsymbol{m}_3\,\boldsymbol{m}_1\,\boldsymbol{m}_2\,\tfrac{80}{121}\boldsymbol{m}_4\,\boldsymbol{m}_1\,\boldsymbol{m}_2\,\tfrac{720}{1331}\boldsymbol{m}_3$$

⑤ From linear LTL to FO: safe repeated reachability

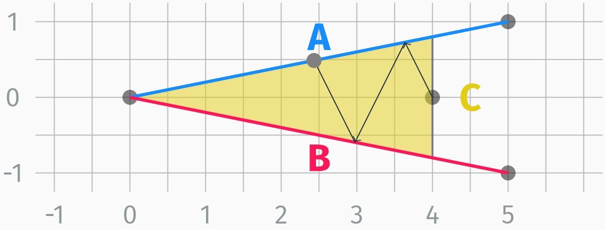
$$\mathrm{GF}\,A \wedge \mathrm{GF}\,B \wedge \mathrm{G}\,C$$



$$\tfrac{4}{11}\boldsymbol{m}_3\,\boldsymbol{m}_1\,\boldsymbol{m}_2\,\tfrac{80}{121}\boldsymbol{m}_4\,\boldsymbol{m}_1\,\boldsymbol{m}_2\,\tfrac{720}{1331}\boldsymbol{m}_3\,\boldsymbol{m}_1$$

⑤ From linear LTL to FO: safe repeated reachability

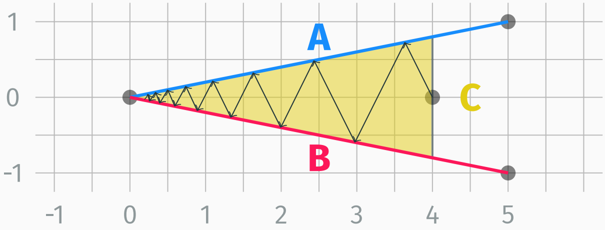
$$\mathsf{GF}\,A \wedge \mathsf{GF}\,B \wedge \mathsf{G}\,C$$



$$\tfrac{4}{11}\boldsymbol{m}_3\,\boldsymbol{m}_1\,\boldsymbol{m}_2\,\tfrac{80}{121}\boldsymbol{m}_4\,\boldsymbol{m}_1\,\boldsymbol{m}_2\,\tfrac{720}{1331}\boldsymbol{m}_3\,\boldsymbol{m}_1\,\boldsymbol{m}_2$$

⑤ From linear LTL to FO: safe repeated reachability

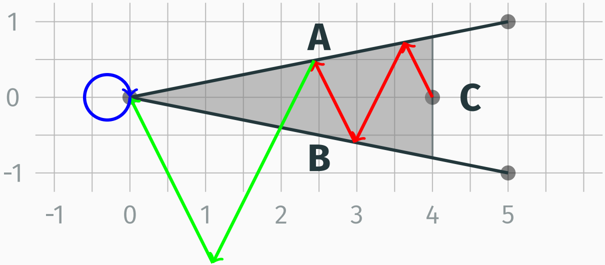
$$\mathsf{GF}\,A \wedge \mathsf{GF}\,B \wedge \mathsf{G}\,C$$



$$\tfrac{4}{11}\boldsymbol{m}_3\,\boldsymbol{m}_1\,\boldsymbol{m}_2\,\tfrac{80}{121}\boldsymbol{m}_4\,\boldsymbol{m}_1\,\boldsymbol{m}_2\,\tfrac{720}{1331}\boldsymbol{m}_3\,\boldsymbol{m}_1\,\boldsymbol{m}_2\,\cdots$$

⑤ From linear LTL to FO: safe repeated reachability

$$\mathsf{GF}\,A \wedge \mathsf{GF}\,B \wedge \mathsf{G}\,C$$



1. Safe exec. (SE) 2. Inter. exec. (IE) 3. Loop (L)

⑤ From linear LTL to FO: safe repeated reachability

Informal theorem

There is a safe exec. (SE) from $\boldsymbol{x}$, an intermediate exec. (IE) and a loop (L) s.t. $\emptyset \neq \text{supp}(L) \subseteq \text{supp}(SE) = \text{supp}(IE)$

iff

$$\boldsymbol{x} \models_M \mathsf{GF}\, A \wedge \mathsf{GF}\, B \wedge \mathsf{G}\, C$$

Proof sketch of $\Leftarrow$

(SE) follows by definition

(IE, L) follows from Farkas' lemma

⑤ From linear LTL to FO: safe repeated reachability

Informal theorem

There is a safe exec. (SE) from $\boldsymbol{x}$, an intermediate exec. (IE) and a loop (L) s.t. $\emptyset \neq \text{supp}(L) \subseteq \text{supp}(SE) = \text{supp}(IE)$

iff

$$\boldsymbol{x} \models_M \mathsf{GF}\, A \wedge \mathsf{GF}\, B \wedge \mathsf{G}\, C$$

Expressible as a
convex semi-linear Horn formula

$\{\mathbf{F}, \mathbf{G}, \wedge\} \in$ NP: recap

1. Flatten formula φ
2. Convert into an almost acyclic automaton $\mathcal{A}_\varphi$
3. Nondeterministically guess a “path” π of $\mathcal{A}_\varphi$
4. Convert π into a “linear” LTL formula φ'
5. Construct a first-order formula ψ s.t. $\psi(\boldsymbol{x}) \leftrightarrow \boldsymbol{x} \models_M \varphi'$
6. Check whether $\psi(\boldsymbol{x})$ holds (in polynomial time)

$$\varphi \xrightarrow{①} \mathrm{flat}(\varphi) \xrightarrow{②} \mathcal{A}_\varphi \xrightarrow{③} \pi \xrightarrow{④} \varphi' \xrightarrow{⑤} \psi \; ⑥ \begin{cases} \nearrow \text{Yes} \\ \searrow \text{No} \end{cases}$$

Conclusion: summary

- Introduced LTL for MMS
- Classified each syntactic fragment as P-c., NP-c. or undecidable
- Generalizes and unifies results on continuous counter systems

Conclusion: future work

- Handling richer properties
- Practical implementation
- Extension of LTL to 2-player games

Thank you!
Vielen Dank!