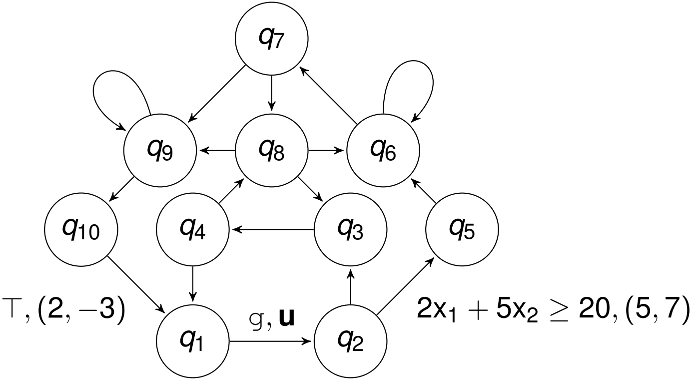
Taming Past LTL & Flat Counter Systems

Stéphane Demri

NYU & CNRS – Marie Curie Fellow

Joint work with A. Dhar & A. Sangnier (LIAFA, France)

Counter Systems



$$S = (Q, \mathtt{C}_n, \triangle, \mathbf{I})$$

Guards: Boolean combinations of linear constraints
$\Sigma_i\, a_i \cdot \mathsf{x}_i \sim b$
$(a_i \in \mathbb{Z},\ b \in \mathbb{N},\ \sim \in \{=, \leq, \geq, <, >\})$

Updates: $\mathbf{u} \in \mathbb{Z}^n$.

Runs in Counter Systems

$$\langle q_0, \mathbf{v}_0 \rangle \xrightarrow{\delta_0} \langle q_1, \mathbf{v}_1 \rangle \xrightarrow{\delta_1} \langle q_2, \mathbf{v}_2 \rangle \xrightarrow{\delta_2} \langle q_3, \mathbf{v}_3 \rangle \xrightarrow{\delta_3} \cdots$$

- For all $i \in \mathbb{N}$, $\mathbf{v}_i \in \mathbb{N}^n$ represents the counter values.
- $\delta_i = \langle q_i, guard(\delta_i), update(\delta_i), q_{i+1} \rangle$.
- For all $i \in \mathbb{N}$, $\mathbf{v}_i \models guard(\delta_i)$ & $\mathbf{v}_{i+1} = \mathbf{v}_i + update(\delta_i)$.

Well-known that counter systems are Turing-complete and therefore most of verification problems are undecidable.

Past LTL with Arithmetical Constraints: $\mathrm{PLTL}[\mathrm{C}]$

- Arithmetical constraints $\approx$ guards.

- Formulae:

$$\phi ::= \mathrm{p} \mid \mathrm{g} \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathsf{X}\phi \mid \phi \mathsf{U} \phi \mid \mathsf{X}^{-1}\phi \mid \phi \mathsf{S} \phi$$

- For model-checking, X^{-1} and S do not add expressive power but help to express properties succinctly.

- Models for $\mathrm{PLTL}[\mathrm{C}]$ are runs from counter systems.

Satisfaction Relation

- Models are ω-sequences of the form

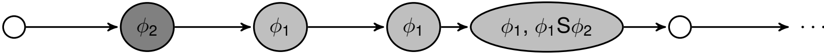
$$\sigma : \mathbb{N} \to 2^{\mathrm{AT}} \times \mathbb{N}^{\mathrm{C}}$$

- $\sigma = \langle Z_0, v_0 \rangle, \langle Z_1, v_1 \rangle, \langle Z_2, v_2 \rangle, \cdots$

- $\sigma, i \models \mathrm{p} \overset{\text{def}}{\Leftrightarrow} \mathrm{p} \in Z_i \qquad \sigma, i \models \mathsf{g} \overset{\text{def}}{\Leftrightarrow} v_i \models_{\mathrm{PA}} \mathsf{g}$

- $\sigma, i \models \mathsf{X}\phi \overset{\text{def}}{\Leftrightarrow} \sigma, i+1 \models \phi$

- $\sigma, i \models \phi_1 \mathsf{S} \phi_2 \overset{\text{def}}{\Leftrightarrow} \sigma, j \models \phi_2$ for some $0 \leq j \leq i$ such that $\sigma, k \models \phi_1$, for all $j < k \leq i$.



- etc.

Existential Model-Checking Problem

- $\mathrm{MC}(\mathrm{L}, \mathcal{C})$:

  Input: A counter system $S \in \mathcal{C}$, a configuration c_0 and a formula $\phi \in \mathrm{L}$;

  Output: Is there a run ρ from c_0 in S such that $\rho, 0 \models \phi$?

- When $\mathcal{C}$ is the full class of counter systems, the problem is known to be undecidable.

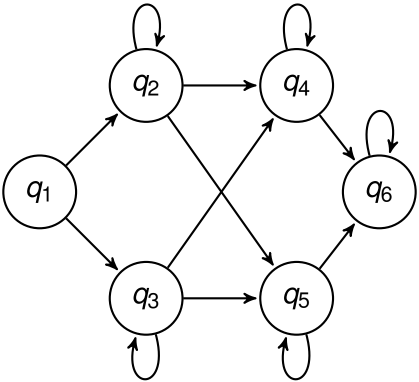
- Restrictions have been designed to regain decidability while being general enough.

- E.g., reversal-boundedness, flatness, no guard, etc.

- In this work, we study flat counter systems.

Flat Counter Systems

- Every state of the control graph belongs to at most one simple cycle.
- Simple cycles can be organized as a DAG: an edge between two cycles means that there is a path from a state of the first cycle to a state of the second cycle.



Flatness May Lead to Effective Semilinearity

- Relational flat counter systems have reachability sets that are effectively Presburger-definable [Comon & Jurski, CAV'98]. Guards/updates: conjunctions of formulae of the form either $x \sim y + c$ or $x \sim c$, with $x, y \in \{x_1, \ldots, x_n, x'_1, \ldots, x'_n\}$, $c \in \mathbb{Z}$ and $\sim \in \{\geq, \leq, =, >, <\}$.

- Flat counter systems with affine functions satisfying finite monoid property have also reachability sets that are effectively Presburger-definable [Finkel & Leroux, FTS&TCS'02]. (more general than the class of flat CS herein)

- See also generalizations in [Bozga et al., CAV'10] and the tool FLATA developed at VERIMAG.

Flattable Systems

- Flat counter systems are not always directly available.

- A relaxed version of flatness: reachability can be captured by a flat unfolding of the system.

- $\langle S, \langle q, \mathbf{x} \rangle \rangle$ is flattable whenever there is a partial unfolding of $\langle S, \langle q, \mathbf{x} \rangle \rangle$ that is flat and has the same reachability set as $\langle S, \langle q, \mathbf{x} \rangle \rangle$.

- $\Sigma = \triangle$; let L be a finite union of languages of the form

$$u_1 (v_1)^* u_2 (v_2)^* \cdots (v_k)^* u_{k+1},$$

- $\langle S, \langle q, \mathbf{x} \rangle \rangle$ is initially flattable iff there is some L of the above form such that

$$\{\langle q', \mathbf{x}' \rangle : \langle q, \mathbf{x} \rangle \xrightarrow{*} \langle q', \mathbf{x}' \rangle\} = \{\langle q', \mathbf{x}' \rangle : \langle q, \mathbf{x} \rangle \xrightarrow{u} \langle q', \mathbf{x}' \rangle, u \in \mathrm{L}\}$$

More on Flattening

- S is globally flattable $\overset{\text{def}}{\Leftrightarrow}$ there is a finite union of bounded languages L such that

$$\xrightarrow{*} = \{\langle\langle q, \mathbf{x}\rangle, \langle q', \mathbf{x}'\rangle\rangle : \langle q, \mathbf{x}\rangle \xrightarrow{u} \langle q', \mathbf{x}'\rangle, u \in \mathrm{L}\}$$

- Flattable counter systems are everywhere.

  [Leroux & Sutre, ATVA'05]

  - Reversal-bounded initialized counter automata are initially flattable.
  - Initialized gainy counter automata are initially flattable.
  - Etc.

- In the general case, flat unfolding of a counter system provides less runs but can be used as an underapproximation method.

This work

- Optimal upper bound for model-checking flat counter systems with $\mathrm{PLTL}[\mathbb{C}]$.

- Decidability known with $\mathrm{CTL}^{\star}$ (extension of $\mathrm{PLTL}[\mathbb{C}]$).

  [Demri et al., ATVA'06]

  (deciding properties richer than simple reachability)

- Complexity in 4EXPTIME by an exponential translation into satisfiability for Presburger Arithmetic.

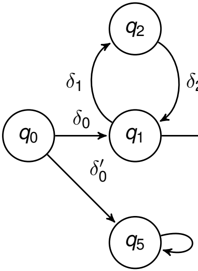
- Model-checking flat **Kripke structures** with $\mathrm{LTL}[\emptyset]$ (no past-time operators) is NP-complete.

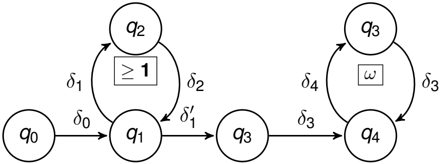
  [Khutz & Finkbeiner, CONCUR'11]

  (Kripke structure $\approx$ counter system without counters)

Path Schemas

- Path schemas: alternation of non-loop and loop segments, ending by a loop, and representing a potentially infinite set of infinite runs.





$P = (\delta_0)(\delta_1\delta_2)^+(\delta_1'\delta_3)(\delta_4\delta_3)^\omega$

- Path schema = ω-regular expression of the form $p_1 l_1^+ p_2 l_2^+ \ldots p_k l_k^\omega$ over alphabet $\triangle$.

Minimal Path Schemas

- Path schema $p_1 l_1^+ p_2 l_2^+ \dots p_k l_k^\omega$ is minimal whenever
  1. $p_1 \cdots p_k$ is either empty or a simple non-loop segment,
  2. $l_1, \dots, l_k$ are loops with disjoint sets of transitions.

- In a flat counter system, the number of minimal path schemas is bounded by $\mathrm{card}(\Delta)^{(2\times\mathrm{card}(\Delta))}$.

- Every infinite run in a flat counter system respects a minimal path schema.

- A run ρ respecting a path schema $p_1 l_1^+ p_2 l_2^+ \dots p_k l_k^\omega$ can be represented by its initial configuration and a tuple in $\mathbb{N}^{k-1}$ encoding how many times loops $l_1, \dots, l_{k-1}$ are visited.

Main Ingredients

- An algorithm in NP for $\mathrm{MC}(\mathrm{PLTL}[\mathbb{C}], \mathcal{CFS})$ may start first by guessing a minimal path schema.
- Ingredients of the proof below aim at bounding the numbers of times loops are visited.
  1. Stuttering Theorem for Past LTL.
     $\cdots$ or how to solve the problem when there is no counter.
  2. Representing the set of runs respecting a path schema by a quantifier-free Presburger formula.
     $\cdots$ or how to bound the numbers of times loops are visited when guards are conjunctions of linear constraints.
  3. Eliminating disjunctions in guards.
     $\cdots$ or how to flatten multiple loops with identical updates.
  4. Combining 1, 2 and 3.
- At least, loops may have to be visited an exponential number of times.

Stuttering Theorem for Past LTL

- Stuttering of finite words or single letters have been instrumental to show results about the expressive power of fragments of Past LTL.

- Stuttering Theorem for LTL done in [Kučera and Strejček - Acta Informatica'05].

- From [Gabbay, TLS'87], elimination of past-time operators is possible but this may cause an exponential blow-up.

- We can show that model-checking flat Kripke structures with **Past LTL** is in NP thanks to the property below.

- Let $\sigma = \sigma_1 s^M \sigma_2$ and $\sigma' = \sigma_1 s^{M'} \sigma_2$ with $M, M' \geq 2N + 1$ and $N \geq 2$. Then, for every Past LTL formula ϕ of temporal depth at most N, we have $\sigma, 0 \models \phi$ iff $\sigma', 0 \models \phi$.

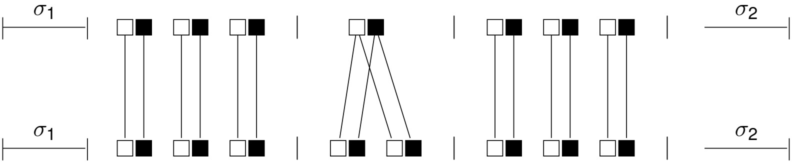
Idea of the Proof

- Binary relation between positions $\langle \sigma, i \rangle \approx_N \langle \sigma', i' \rangle$.

- $\langle \sigma, i \rangle \approx_N \langle \sigma', i' \rangle$ implies that for every ϕ of temporal depth at most N, we have $\sigma, i \models \phi$ iff $\sigma', i' \models \phi$.

- Proof by structural induction.

- $s = \square\blacksquare$, $N = 3$:



- Alternative proof can use Ehrenfeucht-Fraïssé games as defined in [Etessami & Wilke, IC 00].

On Visiting Loops a Linear Amount of Times

- Path schema $P = p_1 l_1^+ p_2 l_2^+ \dots p_k l_k^\omega$ from a flat Kripke structure and ϕ in Past LTL of temporal depth N.

- Equivalence between the propositions below:
  1. There exist $n_1, \dots, n_{k-1} \in \mathbb{N}^+$ such that $p_1 l_1^{n_1} p_2 l_2^{n_2} \dots p_{k-1} l_{k-1}^{n_{k-1}} p_k l_k^\omega \models \phi$.
  2. There exist $n_1, \dots, n_{k-1} \in [1, 2N+5]$ such that $p_1 l_1^{n_1} p_2 l_2^{n_2} \dots p_{k-1} l_{k-1}^{n_{k-1}} p_k l_k^\omega \models \phi$.

- Model-checking flat Kripke structures with Past LTL in NP. (model-checking $u \cdot v^\omega$ with Past LTL formulae ψ can be done in time in $\mathcal{O}(\mathrm{len}(uv) \times \mathrm{len}(\psi)^2)$ [Laroussinie & Markey & Schnoebelen, LICS'02])

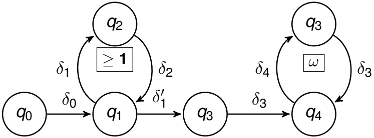
- NP-hardness is inherited from LTL.

[Khutz & Finkbeiner, CONCUR'11]

Constraint Systems

- Given $P = p_1 l_1^+ p_2 l_2^+ \ldots p_k l_k^\omega$ and configuration c_0, $iter_P(c_0) \subseteq \mathbb{N}^{k-1}$ is defined s.t. (1) and (2) are equivalent:
  1. $\langle n_1, \ldots, n_{k-1} \rangle \in iter_P(c_0)$
  2. there is a run ρ starting from c_0 respecting the ω-sequence of transitions $p_1 l_1^{n_1} p_2 l_2^{n_2} \ldots p_{k-1} l_{k-1}^{n_{k-1}} p_k l_k^\omega$
- $iter_P(c_0)$ can be characterized by a conjunction of linear constraints in which each loop has a dedicated variable when P has no disjunctions.
- $iter_P(c_0) = \emptyset$ whenever one condition below is falsified:
  1. $effect(l_k) \geq \mathbf{0}$ where $effect(\delta_1 \cdots \delta_N) \stackrel{\text{def}}{=} update(\delta_1) + \cdots + update(\delta_N)$.
  2. For each guard in l_k of the form $\sum_i a_i \mathsf{x}_i \sim b$ with $\sim \in \{\leq, <\}$, we have $\sum_i a_i \times effect(l_k)[i] \leq 0$.
  3. For each guard in l_k of the form $\sum_i a_i \mathsf{x}_i = b$, we have $\sum_i a_i \times effect(l_k)[i] = 0$.
  4. For each guard in l_k of the form $\sum_i a_i \mathsf{x}_i \sim b$ with $\sim \in \{\geq, >\}$, we have $\sum_i a_i \times effect(l_k)[i] \geq 0$.

Example of Conjuncts



$$P = (\delta_0)(\delta_1\delta_2)^{+}(\delta_1'\delta_3)(\delta_4\delta_3)^{\omega}$$

- Each internal loop is visited at least once: $y_1 \geq 1$.
- Counter values are non-negative.
  - $\mathbf{v_0}[i] + \mathit{effect}(\delta_0)[i] \geq 0$;
  - $\mathbf{v_0}[i] + \mathit{effect}(\delta_0)[i] + (y_1 - 1)\mathit{effect}(\delta_1\delta_2)[i] \geq 0$;
  - $\mathbf{v_0}[i] + \mathit{effect}(\delta_0)[i] + \mathit{effect}(\delta_1)[i] \geq 0$;
  - $\mathbf{v_0}[i] + \mathit{effect}(\delta_0)[i] + (y_1 - 1)\mathit{effect}(\delta_1\delta_2)[i] + \mathit{effect}(\delta_1)[i] \geq 0$;
  - etc.
- + similar constraints to guarantee that counter values satisfy the guards.
- Sufficient by convexity.

Characterization

- Path schema $P = p_1 l_1^+ p_2 l_2^+ \dots p_k l_k^\omega$ without disjunctions in guards.
- Initial configuration c_0.
- One can compute a constraint system $\mathcal{E}$ in polynomial-time such that
  - the set of solutions of $\mathcal{E}$ is equal to $iter_P(c_0)$,
  - $\mathcal{E}$ has $k - 1$ variables,
  - $\mathcal{E}$ has a polynomial number of conjuncts,
  - the size of the greatest absolute value from constants in $\mathcal{E}$ is polynomial.
  - (this can be done also symbolically by replacing counter values from c_0 by variables).

Small Run Property

- [Borosh & Treybig, AMS 76]
  - $\mathcal{M} \in [-M, M]^{U \times V}$, $\mathbf{b} \in [-M, M]^U$, where $U, V, M \in \mathbb{N}$.
  - If ($\mathbf{x} \in \mathbb{N}^V$ and $\mathcal{M}\mathbf{x} \geq \mathbf{b}$), then there is $\mathbf{y} \in [0, (\max\{V, M\})^{CU}]^V$ such that $\mathcal{M}\mathbf{y} \geq \mathbf{b}$.

- Thanks to existence of $\mathcal{E}$, existence of small runs too:
  - Path schema $P = p_1 l_1^+ p_2 l_2^+ \ldots p_k l_k^\omega$ without disjunctions.
  - Initial configuration c_0.
  - Equivalence between the propositions below:
    1. There is a run ρ from c_0 respecting P.
    2. There is a run ρ from c_0 respecting P such that each internal loop is visited at most an exponential number of times.

- Entails NP upper bound for the reachability problem in flat counter systems with no disjunction in guards.

Algorithm in NP: a First Attempt

1. Guess a minimal path schema $P = p_1 l_1^+ p_2 l_2^+ \ldots p_k l_k^\omega$.
2. Build $\mathcal{E}$ form P and c_0.
3. Guess a small solution $\langle n_1, \ldots, n_{k-1} \rangle$ for $\mathcal{E}$.
4. Check whether $\rho, 0 \models \phi$ where ρ is the unique run respecting $p_1 l_1^{n_1} p_2 l_2^{n_2} \ldots p_{k-1} l_{k-1}^{n_{k-1}} p_k l_k^\omega$.

Pb. I $\rho = \rho_1 \cdot \rho_2$, ρ_1 respects $p_1 l_1^{n_1} p_2 l_2^+ \ldots p_{k-1} l_{k-1}^{n_{k-1}} p_k$.

ρ_1 can be of exponential length.
(problem to guarantee algorithm in NP).

Pb. II How to know which arithmetical constraints hold?
(two passes on the same loop may differ w.r.t. the satisfaction of arithmetical contraints)

Towards a Solution

Idea I Replace each n_i by $\mathtt{Min}(n_i, 2td(\phi) + 5) = m_i$ and check $p_1 l_1^{m_1} p_2 l_2^{m_2} \ldots p_{k-1} l_{k-1}^{m_{k-1}} p_k l_k^{\omega}, 0 \models \phi$.

Idea II Enrich states with information about interpretation of terms $\Sigma_i \, a_i \cdot \mathrm{x}_i \sim b$ so that satisfaction of guards is easy to check.
(see e.g., regions for timed automata)

- Maybe, no run respects $p_1 l_1^{m_1} p_2 l_2^{m_2} \ldots p_{k-1} l_{k-1}^{m_{k-1}} p_k l_k^{\omega}$.
- Exponential amount of maps with profile

$$T = \{\mathtt{t}_1, \ldots, \mathtt{t}_\alpha\} \to I$$

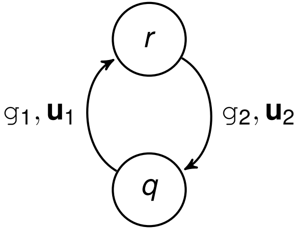
$$I = \{(-\infty, b_1 - 1], [b_1, b_1], [b_1 + 1, b_2 - 1], \cdots, [b_m, b_m], [b_m + 1, \infty)\} \setminus \{\emptyset\}$$

- We need to deal with disjunctions in guards at some stage.

A Few More Definitions

- With constants in $B = \{b_1, \ldots, b_m\}$, the set I as at most $2m + 1$ intervals.
- Term map $\mathbf{m} : T \to I$ where T is a finite set of terms.
- Footprint ft w.r.t. resource $\mathrm{R} = \langle X, T, B \rangle$ is a map $\mathrm{ft} : \mathbb{N} \to 2^X \times I^T$.
- Every run ρ can be *abstracted* by its footprint $\mathrm{ft}(\rho)$ so that $\rho, i \models \phi$ iff $\mathrm{ft}(\rho), i \models_{\mathrm{symb}} \phi$.
- For instance, $\mathrm{ft}, i \models_{\mathrm{symb}} \mathtt{t} \geq b \overset{\mathrm{def}}{\Leftrightarrow} \overbrace{\pi_2(\mathrm{ft}(i))}^{\text{term map}}(\mathtt{t}) \subseteq [b, +\infty)$.

The Case with a Single Loop

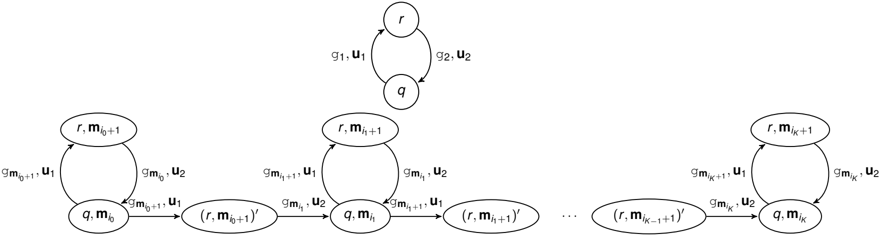


- Run $\rho = \langle q, \mathbf{v_0}\rangle\langle r, \mathbf{v_1}\rangle\langle q, \mathbf{v_2}\rangle \cdots$ with footprint $\mathrm{ft}(\rho) = \langle \mathbf{l}(q), \mathbf{m}_0\rangle\langle \mathbf{l}(r), \mathbf{m}_1\rangle\langle \mathbf{l}(q), \mathbf{m}_2\rangle\langle \mathbf{l}(r), \mathbf{m}_3\rangle \cdots$.
- $\mathcal{H}$: set of (even) positions j at q starting a new pair of term maps $\langle \mathbf{m}_j, \mathbf{m}_{j+1}\rangle$:

$$\mathcal{H} = \{0\} \cup \{2i : i \in \mathbb{N}^+,\ \langle \mathbf{m}_{2i-2}, \mathbf{m}_{2i-1}\rangle \neq \langle \mathbf{m}_{2i}, \mathbf{m}_{2i+1}\rangle\}$$

- $\mathrm{card}(\mathcal{H}) \leq \mathrm{card}(T) \times (2\mathrm{card}(B) + 1)$.
(even though the number of term maps is exponential)

Unfolding the loop



- $g_{\mathbf{m}}$ is a conjunction of linear constraints enforcing that next interpretation of terms is compatible with **m** (depends also on the update).
- Remove of guards g_1 and g_2: their satisfaction can be checked off-line.
- For sake of simplicity, we assumed that $\mathcal{H} = i_0 < i_1 < \cdots < i_K$ and $i_k - i_{k-1} \geq 4$ for all k. (otherwise production of unfoldings with less regular alternations of non-loop segments and loops)

Elimination of Disjunctions

- Different initial counter values $\mathbf{v}_0'$ or $\mathbf{v}_0''$ may lead to different unfoldings but their length would still be polynomial.

- Y_P: set of path schemas obtained by unfolding each loop from P in all possible ways (while respecting the maximal length of each unfolding and the resources T and B).

- "Unfolding" of non-loop segments is also required but it amounts to add term maps in states.

- No path schema in Y_P has disjunction in guards.

- Footprints can be computed from the control states.

- There is a run ρ respecting P with footprint ft iff there exist P' in Y_P and run ρ' respecting P' with footprint ft.

Main Algorithm in NP

1. Guess a minimal path schema P.
2. Guess a path schema $P' = p_1 l_1^+ p_2 l_2^+ \dots p_k l_k^\omega$ from Y_P.
3. Guess a tuple $\langle n_1, \dots, n_{k-1} \rangle \in [1, 2^{P(N)}]^{k-1}$.
4. $m_i := \texttt{Min}(n_i, 2td(\phi) + 5)$ for all $i \in [1, k-1]$.
5. Check $p_1 l_1^{m_1} p_2 l_2^{m_2} \dots p_{k-1} l_{k-1}^{m_{k-1}} p_k l_k^\omega, 0 \models_{\text{symb}} \phi$.
6. Build $\mathcal{E}$ form P' and c_0.
7. Check that $\langle n_1, \dots, n_{k-1} \rangle \models \mathcal{E}$.

(polynom $P(\cdot)$ is built using [Borosh & Treybig, AMS'76])

Bounding the Number of Loops

- For a fixed $n \geq 1$, model-checking path schemas with at most n loops with Past LTL formulae can be done in polynomial-time.
  (direct consequence of the Stuttering Theorem)

- Model-checking path schemas with at most 2 loops with PLTL[$\mathbb{C}$] formulae is NP-hard.
  (requires a first-class reduction)

- Model-checking path schemas with one loop with PLTL[$\mathbb{C}$] formulae is in PTIME.
  (algorithm deterministically unfolds the single loop)

Summary

Classes of Systems	Past LTL	$\mathrm{PLTL}[\mathrm{C}]$	Reachability
$\mathcal{KPS}$	NP-complete	—	PTIME
$\mathcal{CPS}$	NP-complete	NP-complete	NP-complete
$\mathcal{KPS}(n)$	PTIME	—	PTIME
$\mathcal{CPS}(n)$, $n > 1$	??	NP-complete	??
$\mathcal{CPS}(1)$	PTIME	PTIME	PTIME
$\mathcal{KFS}$	NP-complete	—	PTIME
$\mathcal{CFS}$	NP-complete	**NP-complete**	NP-complete