Prompt Interval Temporal Logic

Dario Della Monica, Angelo Montanari, Aniello Murano, and Pietro Sala

Università degli Studi di Napoli Federico II

Universidad Complutense Madrid

JELIA 2016
Larnaca, Cyprus, 11/11

Outline

Outline

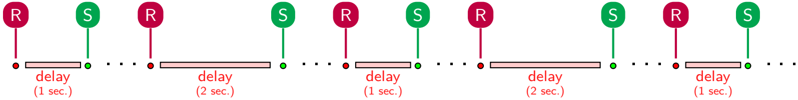
What is prompt

Intuition: to bound the delay with which a request is satisfied

What is prompt

Intuition: to bound the delay with which a request is satisfied

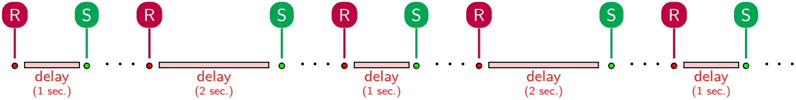
- the bound is constant ...



What is prompt

Intuition: to bound the delay with which a request is satisfied

- the bound is constant ...

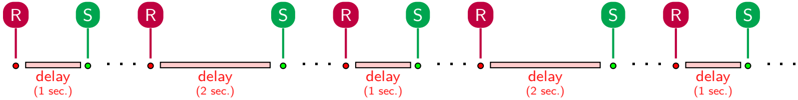


Delay sequence: 1, 2, 1, 2, 1, 2, ...

What is prompt

Intuition: to bound the delay with which a request is satisfied

- the bound is constant ...



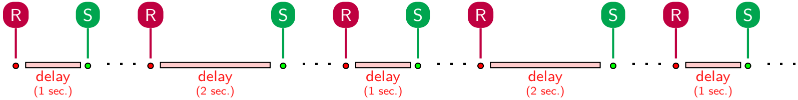
Delay sequence: 1, 2, 1, 2, 1, 2, ... Bound: 2

$[G](req \rightarrow Xsat \vee XXsat)$

What is prompt

Intuition: to bound the delay with which a request is satisfied

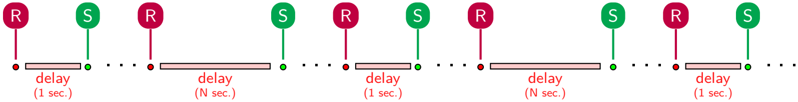
- the bound is constant ...



Delay sequence: 1, 2, 1, 2, 1, 2, . . . Bound: 2

$[G](req \rightarrow Xsat \vee XXsat)$

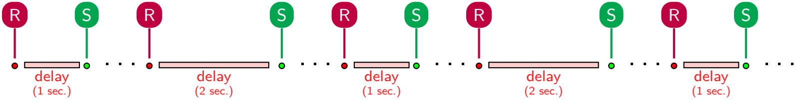
- ... but unknown (or arbitrarily large)



What is prompt

Intuition: to bound the delay with which a request is satisfied

- the bound is constant ...

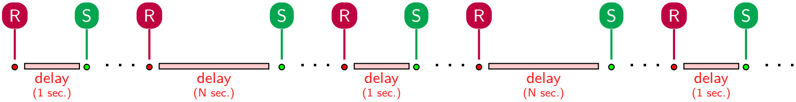


Delay sequence: 1, 2, 1, 2, 1, 2, . . .

Bound: 2

$[G](req \rightarrow Xsat \vee XXsat)$

- ... but unknown (or arbitrarily large)

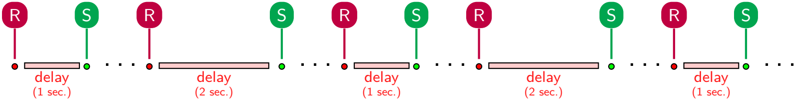


Delay sequence: 1, N, 1, N, 1, N, . . .

What is prompt

Intuition: to bound the delay with which a request is satisfied

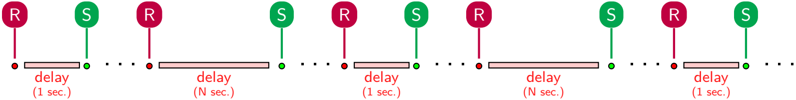
- the bound is constant ...



Delay sequence: 1, 2, 1, 2, 1, 2, . . .   Bound: 2

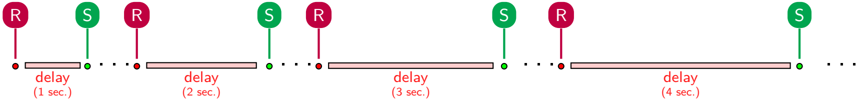
$[G](req \rightarrow Xsat \vee XXsat)$

- ... but unknown (or arbitrarily large)



Delay sequence: 1, N, 1, N, 1, N, . . .   Bound: N (a constant)

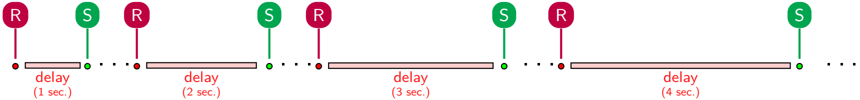
What is not prompt



Delay sequence: 1, 2, 3, 4, 5, ...

Università degli Studi di Napoli Federico II
Universidad Complutense Madrid

What is not prompt



Delay sequence: 1, 2, 3, 4, 5, . . .

Bound: ∞ (unbounded)

Prompt extensions of temporal logic

- PLTL
  [Alur-Etessami-La Torre-Peled, 2001]

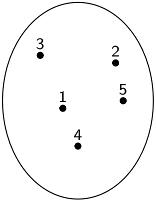
- PROMPT-LTL
  [Kupferman-Piterman-Vardi, 2009]

Outline

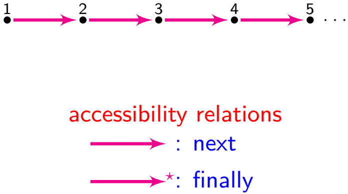
Outline

Temporal logics

- Temporal logics are (multi-)modal logics simplification

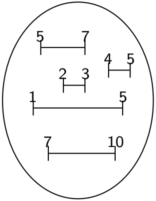


set of worlds
primitive temporal entity
time points/instants

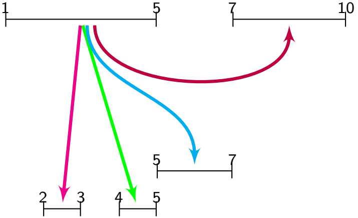


A different approach: from points to intervals

- worlds are intervals (time period — pairs of points)



set of worlds
primitive temporal entity
time intervals/periods



accessibility relations
all binary relations between pairs of intervals

The logic PNL

Syntax

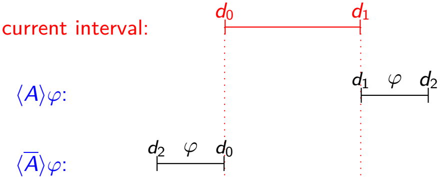
$$\varphi ::= \; p \;\mid\; \neg\varphi \;\mid\; \varphi \wedge \varphi \;\mid\; \langle A \rangle \varphi \;\mid\; \langle \overline{A} \rangle \varphi$$

Semantics

Models: $\mathbf{M} = \langle \mathbb{I}(\mathbb{D}), V \rangle$
(intervals over a linear order + atomic propositions eval.)

$\langle A \rangle$: $\mathbf{M}, [d_0, d_1] \Vdash \langle A \rangle \varphi$ iff $\exists d_2$ s.t. $d_1 < d_2$ and $\mathbf{M}, [d_1, d_2] \Vdash \varphi$

$\langle \overline{A} \rangle$: $\mathbf{M}, [d_0, d_1] \Vdash \langle \overline{A} \rangle \varphi$ iff $\exists d_2$ s.t. $d_2 < d_0$ and $\mathbf{M}, [d_2, d_0] \Vdash \varphi$



Outline

The logic PROMPT-PNL

Syntax

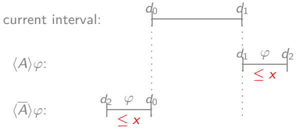
$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle A \rangle \varphi \mid \langle \overline{A} \rangle \varphi \mid \langle A_x \rangle \varphi \mid \langle \overline{A}_x \rangle \varphi$$

Semantics

Models: $\mathbf{M} = \langle \mathbb{I}(\mathbb{D}), V, \delta, \sigma \rangle$
(intervals over a linear order + atomic propositions eval. + metric + bounding variables eval.)

$\langle A_x \rangle$: $\mathbf{M}, [d_0, d_1] \Vdash \langle A_x \rangle \varphi$ iff
$\exists d_2$ s.t. $d_1 < d_2$, $\mathrm{length}_\delta([\mathrm{d}_1, \mathrm{d}_2]) \leq \sigma(x)$, and $\mathbf{M}, [d_1, d_2] \Vdash \varphi$

$\langle \overline{A}_x \rangle$: $\mathbf{M}, [d_0, d_1] \Vdash \langle \overline{A}_x \rangle \varphi$ iff
$\exists d_2$ s.t. $d_2 < d_0$, $\mathrm{length}_\delta([\mathrm{d}_2, \mathrm{d}_0]) \leq \sigma(x)$, and $\mathbf{M}, [d_2, d_0] \Vdash \varphi$



The satisfiability problem for PROMPT-PNL

Input: ▸ a PROMPT-PNL formula φ

Question: Are there

- a model $\mathbf{M} = \langle \mathbb{I}(\mathbb{D}), V, \delta, \sigma \rangle$ and
- an interval $[a, b] \in \mathbb{I}(\mathbb{D})$

that satisfy φ (i.e., $\mathbf{M}, [a, b] \Vdash \varphi$)

Outline

Undecidability of PROMPT−PNL

Theorem

The satisfiability problem for PROMPT−PNL is undecidable

Proof

By reduction from the *Finite Coloring Problem*
(aka. *Finite Tiling Problem*)

Finite Coloring Problem

Input:
- C: a set of colors
- H and V: two binary relations over colors
  (horizontal and vertical color compatibilities)
- c_i and c_f: two distinguished colors in C
  (initial and final color constraints)

Question: Are there
- naturals K and L, and
- a coloring function
  $\mathcal{C} : \{1, \ldots, K\} \times \{1, \ldots, L\} \to C$

such that that horizontal/vertical compatibilities and initial/final constraints are satisfied

Finite Coloring Problem

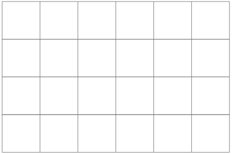
Input:

- C: a set of colors
- H and V: two binary relations over colors
  (horizontal and vertical color compatibilities)
- c_i and c_f: two distinguished colors in C
  (initial and final color constraints)

Question: Are there

- naturals K and L, and
- a coloring function
  $\mathcal{C} : \{1, \ldots, K\} \times \{1, \ldots, L\} \to C$

such that that horizontal/vertical compatibilities and initial/final constraints are satisfied

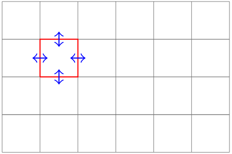


Finite Coloring Problem

Input:
- C: a set of colors
- H and V: two binary relations over colors
  (horizontal and vertical color compatibilities)
- c_i and c_f: two distinguished colors in C
  (initial and final color constraints)

Question: Are there
- naturals K and L, and
- a coloring function
  $\mathcal{C} : \{1, \dots, K\} \times \{1, \dots, L\} \rightarrow C$

such that that horizontal/vertical compatibilities and initial/final constraints are satisfied

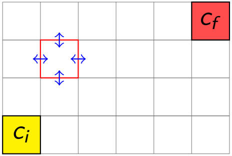


Finite Coloring Problem

Input:
- C: a set of colors
- H and V: two binary relations over colors (horizontal and vertical color compatibilities)
- c_i and c_f: two distinguished colors in C (initial and final color constraints)

Question: Are there
- naturals K and L, and
- a coloring function $\mathcal{C} : \{1, \ldots, K\} \times \{1, \ldots, L\} \to C$

such that that horizontal/vertical compatibilities and initial/final constraints are satisfied

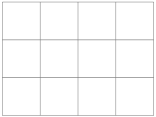


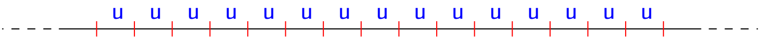
Overview of the proof

Overview of the proof

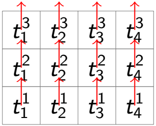
every u-interval “meets” a small u-interval

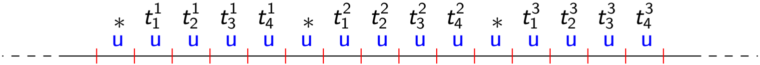
$$u \rightarrow \langle A_x \rangle u$$



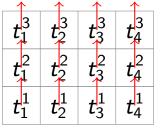


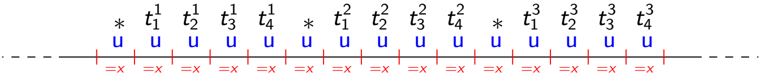
Overview of the proof





Overview of the proof

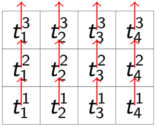


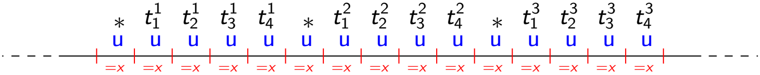


Overview of the proof

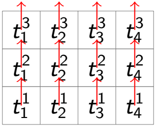
it is easy to give a length *upper bound*

$\langle A_x \rangle$u





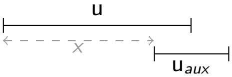
Overview of the proof

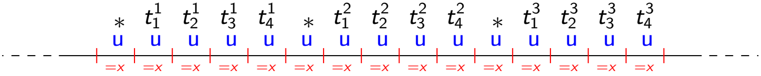


lower bound is trickier:

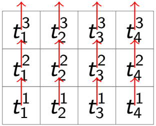
1 there is u_{aux}-interval starting at distance x from beginning of u-interval

$$\langle A\rangle u \to [A_x]\langle A\rangle u_{aux}$$





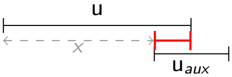
Overview of the proof

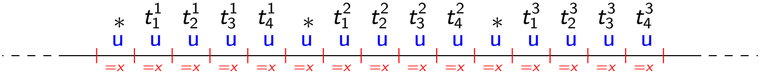


lower bound is trickier:

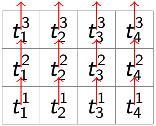
2 no small interval "meets" a u-interval while starting with a u_{aux}-interval

$$[G_x]\neg(\langle A\rangle u \wedge \langle\overline{A}\rangle\langle A\rangle u_{aux})$$





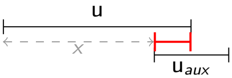
Overview of the proof

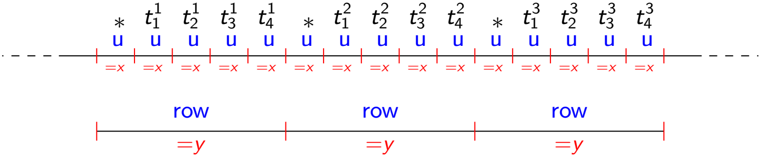


lower bound is trickier:

2 no small interval "meets" a u-interval while starting with a u_{aux}-interval

$$[G_x]\neg(\langle A\rangle u \wedge \langle \overline{A}\rangle\langle A\rangle u_{aux})$$





SAT is undecidable for PROMPT-PNL

Theorem.

The satisfiability problem for the future fragment of PROMPT-PNL is undecidable

Outline

The culprit for undecidability

- using bound x both in existential and universal modalities
- this gives the ability of expressing lower and upper bound for the length of intervals
- thus we can define special chains of intervals
- ... and we can use such special chains as a ruler to suitably encode vertical color compatibility relation

The culprit for undecidability

- using bound x both in existential and universal modalities
- this gives the ability of expressing lower and upper bound for the length of intervals
- thus we can define special chains of intervals
- ... and we can use such special chains as a ruler to suitably encode vertical color compatibility relation

Recipe for decidability

1. Remove the culprit for undecidability: get $\texttt{PROMPT}^{d}_{-}\texttt{PNL}$
   - split X into two sets $X_{\Diamond}$ (existential modalities) and $X_{\Box}$ (universal modalities)

Recipe for decidability

1. Remove the culprit for undecidability: get $\mathtt{PROMPT}^{d}\text{-}\mathtt{PNL}$
   - split X into two sets $X_\Diamond$ (existential modalities) and $X_\Box$ (universal modalities)
2. Realize that *now* prompt modalities are monotone
   - if $\langle A_x \rangle \varphi$ is true when x evaluates to k
     then $\langle A_x \rangle \varphi$ is true when x evaluates to $k' > k$

Recipe for decidability

1. Remove the culprit for undecidability: get $\mathtt{PROMPT}^{\underline{d}}\mathtt{PNL}$
   - split X into two sets $X_\Diamond$ (existential modalities) and $X_\Box$ (universal modalities)
2. Realize that *now* prompt modalities are monotone
   - if $\langle A_x \rangle \varphi$ is true when x evaluates to k
     then $\langle A_x \rangle \varphi$ is true when x evaluates to $k' > k$
3. Realize that *now* one can reduces to the 2-variable case
   - $X_\Diamond = x, X_\Box = y$ thanks to monotonicity

Recipe for decidability

1. Remove the culprit for undecidability: get $\texttt{PROMPT}^{d}\texttt{-PNL}$
   - split X into two sets $X_{\Diamond}$ (existential modalities) and $X_{\Box}$ (universal modalities)
2. Realize that *now* prompt modalities are monotone
   - if $\langle A_x \rangle \varphi$ is true when x evaluates to k
     then $\langle A_x \rangle \varphi$ is true when x evaluates to $k' > k$
3. Realize that *now* one can reduces to the 2-variable case
   - $X_{\Diamond} = x, X_{\Box} = y$ thanks to monotonicity
4. Solve finite satisfiability (look for finite domains)
   - trivially reduce to satisfiability for PNL (non-prompt)
     thanks to monotonicity

Recipe for decidability

1. Remove the culprit for undecidability: get $\texttt{PROMPT}^{\underline{d}}\texttt{PNL}$
   - split X into two sets $X_\Diamond$ (existential modalities) and $X_\Box$ (universal modalities)
2. Realize that *now* prompt modalities are monotone
   - if $\langle A_x \rangle \varphi$ is true when x evaluates to k
     then $\langle A_x \rangle \varphi$ is true when x evaluates to $k' > k$
3. Realize that *now* one can reduces to the 2-variable case
   - $X_\Diamond = x, X_\Box = y$ thanks to monotonicity
4. Solve finite satisfiability (look for finite domains)
   - trivially reduce to satisfiability for PNL (non-prompt)
     thanks to monotonicity
5. Solve infinite satisfiability

Infinite satisfiability

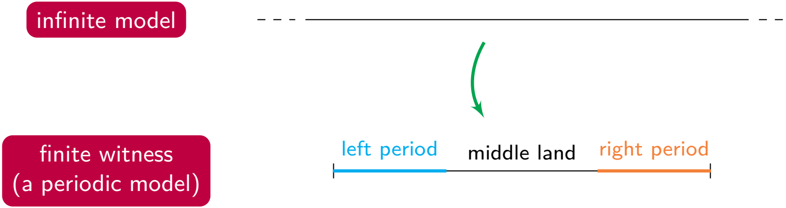
Proof via small model theorem

infinite model

finite witness
(a periodic model)

Infinite satisfiability

Proof via small model theorem

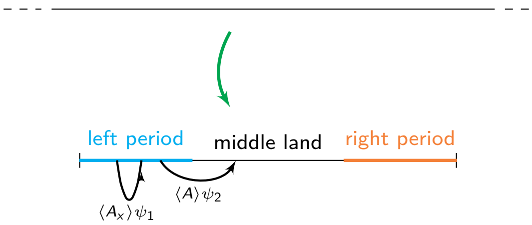


Infinite satisfiability

Proof via small model theorem

infinite model

finite witness
(a periodic model)

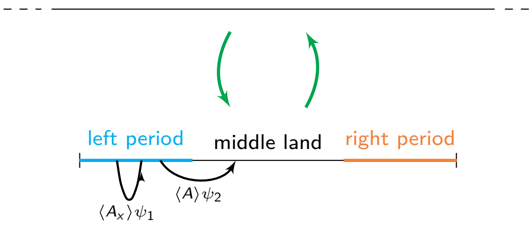


Infinite satisfiability

Proof via small model theorem

infinite model

finite witness
(a periodic model)



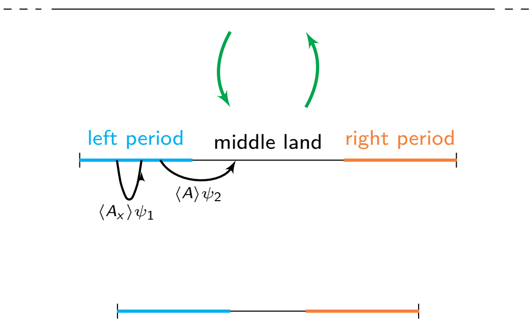
Infinite satisfiability

Proof via small model theorem

infinite model

finite witness
(a periodic model)

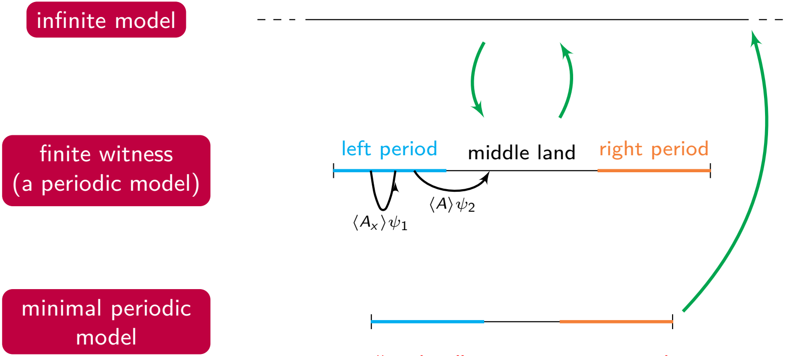
minimal periodic
model



"useless" points are removed
without loss of information

Infinite satisfiability

Proof via small model theorem



"useless" points are removed
without loss of information

SAT is decidable for $\mathtt{PROMPT}^d_{-}\mathtt{PNL}$

Theorem.

The satisfiability problem for $\mathtt{PROMPT}^d_{-}\mathtt{PNL}$ is decidable (NEXPTIME-complete)

Outline

Conclusions and future work

Conclusions

two prompt extensions of Interval Temporal Logic PNL

- full logic PROMPT-PNL is undecidable
- its syntactic restriction $\text{PROMPT}\overset{d}{-}\text{PNL}$ is decidable
  (NEXPTIME-complete)

Conclusions and future work

Conclusions

two prompt extensions of Interval Temporal Logic PNL

- full logic PROMPT−PNL is undecidable
- its syntactic restriction $\text{PROMPT}^{d}\text{-PNL}$ is decidable (NEXPTIME-complete)

Future work

- which is the minimum number of variables to make PROMPT−PNL undecidable
  - the *unrestricted* two variable fragment might be expressive and decidable
- parametric extensions of PNL
  - e.g., allowing both upper and lower bound
- comparison between PROMPT−PNL and *metric PNL*

Thank you!