Real-time Model Checking

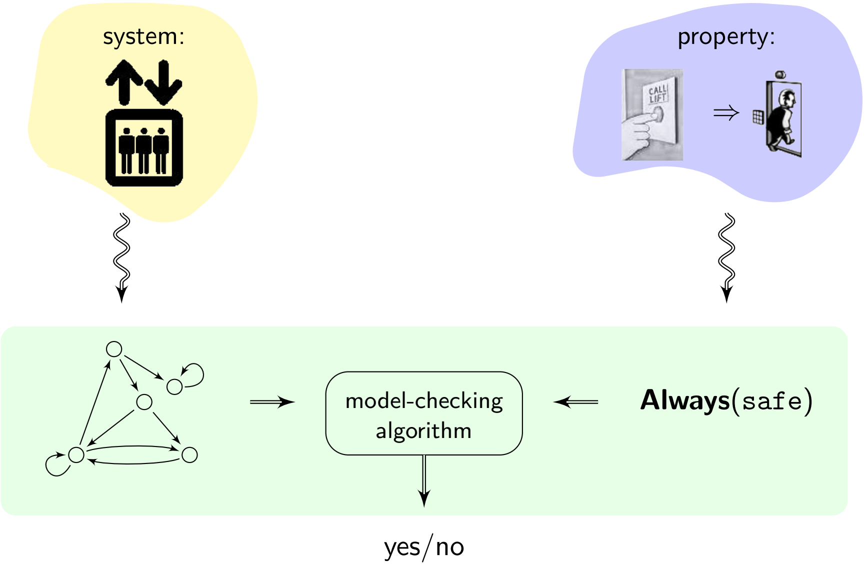
— Timed Temporal Logics —

Nicolas MARKEY

Lav. Spécification & Vérification
CNRS & ENS Cachan – France

March 3, 2010

(Quantitative) Model checking

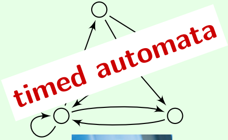


(Quantitative) Model checking

system:

property:

$\Rightarrow$



model-checking algorithm

Always(`safe`)

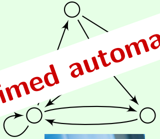
yes/no

(Quantitative) Model checking

system:

property:

⇒



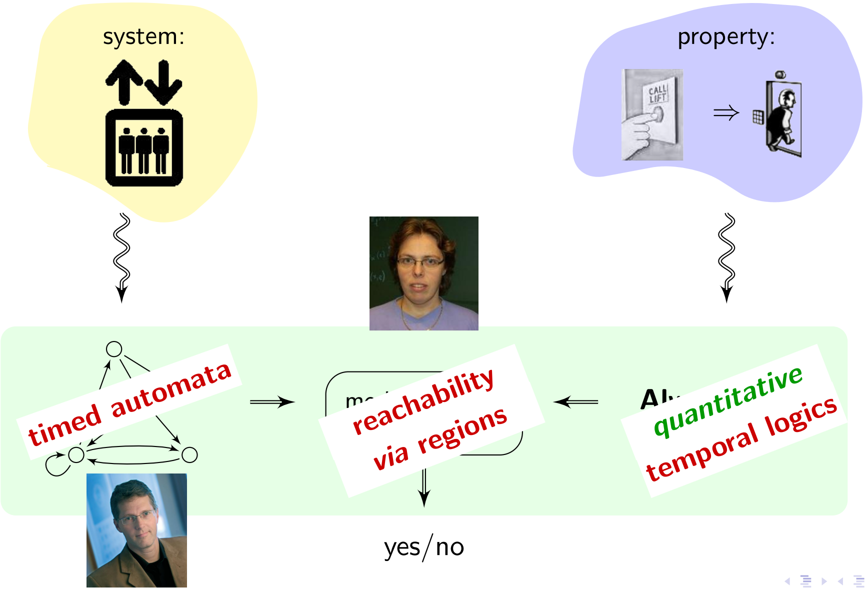
timed automata

reachability via regions

Always(`safe`)

yes/no

(Quantitative) Model checking



Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$

$$\psi ::= \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).

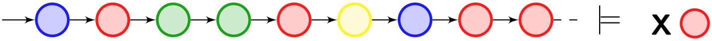
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$

$$\psi ::= \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$



Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$
$$\psi ::= \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

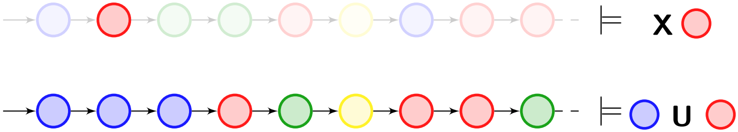
Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$

$$\psi ::= \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$



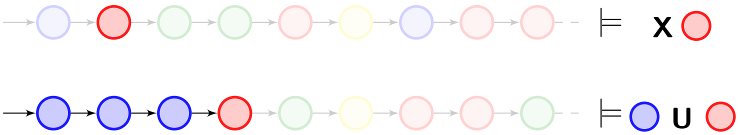
Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$

$$\psi ::= \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

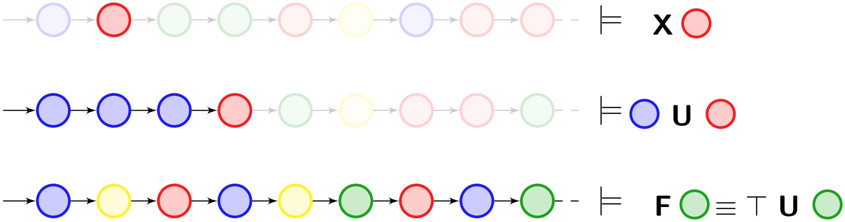


Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$
$$\psi ::= \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$



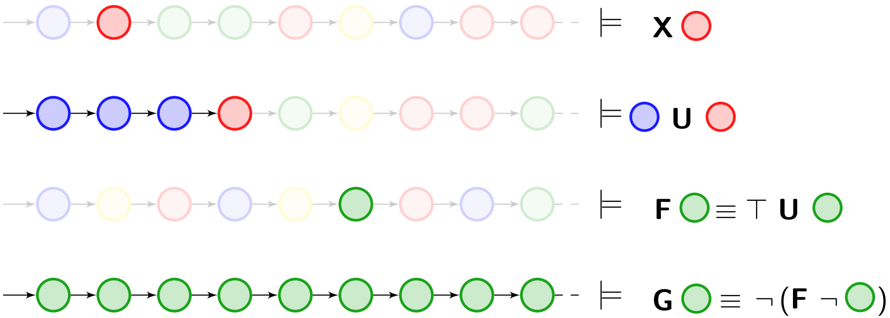
Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$

$$\psi ::= \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

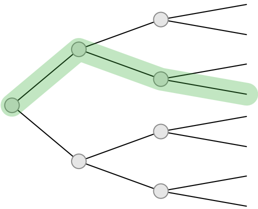


Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

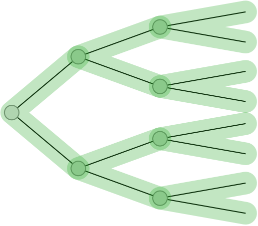
Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$
$$\psi ::= \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$



$\models \quad \mathbf{E}\varphi$



$\models \quad \mathbf{A}\varphi$

Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\text{LTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\,\mathbf{U}\,\varphi$$

$$\text{CTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$

$$\psi ::= \varphi \mid \mathbf{X}\,\varphi \mid \varphi\,\mathbf{U}\,\varphi$$

Example

- $(\bigcirc\ \mathbf{U}\ \bigcirc) \vee \mathbf{G}\,\bigcirc$: *weak* until

Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$
$$\psi ::= \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

Example

- $(\bigcirc\ \mathbf{U}\ \bigcirc) \vee \mathbf{G}\,\bigcirc$: *weak* until
- $\mathbf{G}\,\mathbf{F}\,\bigcirc$: “infinitely often”

Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$

$$\psi ::= \varphi \mid \mathbf{X}\,\varphi \mid \varphi\ \mathbf{U}\ \varphi$$

Example

- $(\bigcirc\ \mathbf{U}\ \bigcirc) \vee \mathbf{G}\,\bigcirc$: *weak* until
- $\mathbf{G}\,\mathbf{F}\,\bigcirc$: “infinitely often”
- $\mathbf{A}\,\mathbf{G}(\bigcirc \Rightarrow \mathbf{A}\,\mathbf{F}\,\bigcirc)$: response property

Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

Quick reminder on untimed temporal logics

$$\mathsf{LTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\,\varphi \mid \varphi\,\mathbf{U}\,\varphi$$

$$\mathsf{CTL} \ni \varphi ::= \bigcirc \mid \neg\,\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\psi \mid \mathbf{A}\psi$$

$$\psi ::= \varphi \mid \mathbf{X}\,\varphi \mid \varphi\,\mathbf{U}\,\varphi$$

Example

- $(\bigcirc\ \mathbf{U}\ \bigcirc) \vee \mathbf{G}\,\bigcirc$: *weak* until
- $\mathbf{G}\,\mathbf{F}\,\bigcirc$: "infinitely often"
- $\mathbf{A}\,\mathbf{G}(\bigcirc \Rightarrow \mathbf{A}\,\mathbf{F}\,\bigcirc)$: response property
- $\mathbf{A}(\mathbf{G}\,\mathbf{F}\,\bigcirc \Rightarrow \mathbf{G}\,\bigcirc)$: fair runs are safe (not a CTL formula)

Refs: [1] Pnueli. *The Temporal Logic of Programs* (1977).
[2] Emerson, Clarke. *Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons* (1982).

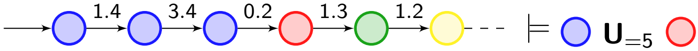
Outline of the talk

1. Introduction
2. Extending temporal logics with real-time constraints
   - Continuous and pointwise semantics
   - Expressiveness issues
3. Model checking timed linear-time logics
   - Undecidability of MTL and TPTL
   - Decidable fragments
4. Model checking timed branching-time logics
5. Conclusions and open problems

Outline of the talk

Extending temporal modalities with time

- decorating modalities with timing constraints:



Refs: [1] Alur, Henzinger. *A Really Temporal Logic* (1989).
[2] Koymans. *Specifying Real-Time Properties with Metric Temporal Logic* (1990).

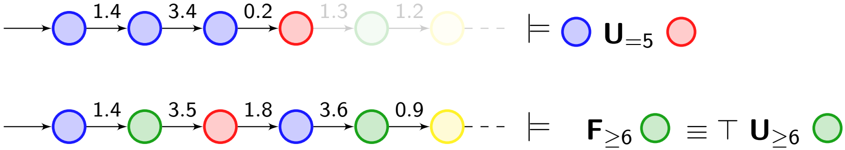
Extending temporal modalities with time

- decorating modalities with timing constraints:

Refs: [1] Alur, Henzinger. *A Really Temporal Logic* (1989).
[2] Koymans. *Specifying Real-Time Properties with Metric Temporal Logic* (1990).

Extending temporal modalities with time

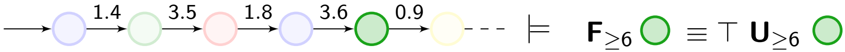
- decorating modalities with timing constraints:



Refs: [1] Alur, Henzinger. *A Really Temporal Logic* (1989).
[2] Koymans. *Specifying Real-Time Properties with Metric Temporal Logic* (1990).

Extending temporal modalities with time

- decorating modalities with timing constraints:

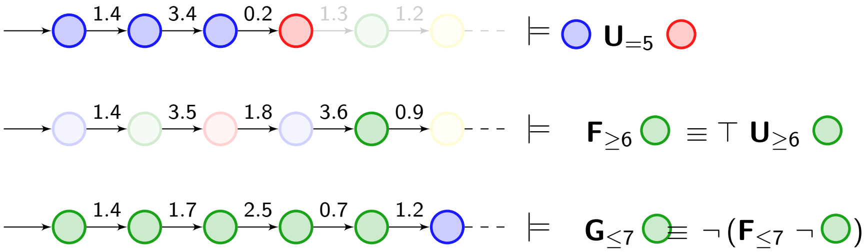


Refs: [1] Alur, Henzinger. *A Really Temporal Logic* (1989).

[2] Koymans. *Specifying Real-Time Properties with Metric Temporal Logic* (1990).

Extending temporal modalities with time

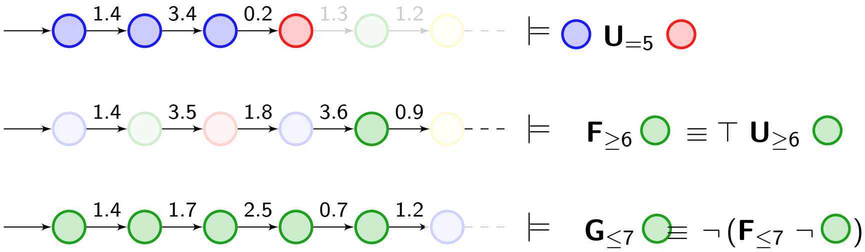
- decorating modalities with timing constraints:



Refs: [1] Alur, Henzinger. *A Really Temporal Logic* (1989).
[2] Koymans. *Specifying Real-Time Properties with Metric Temporal Logic* (1990).

Extending temporal modalities with time

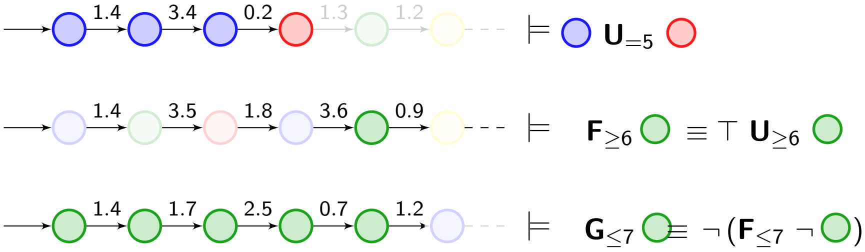
- decorating modalities with timing constraints:



Refs: [1] Alur, Henzinger. *A Really Temporal Logic* (1989).
[2] Koymans. *Specifying Real-Time Properties with Metric Temporal Logic* (1990).

Extending temporal modalities with time

- decorating modalities with timing constraints:

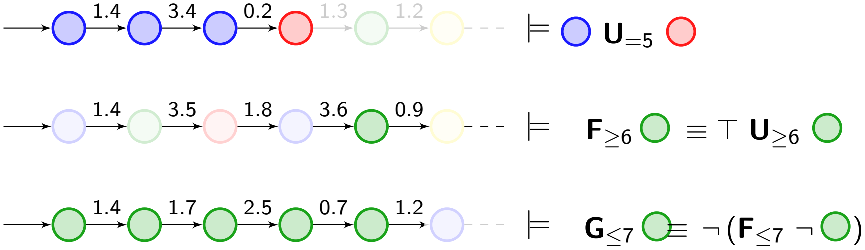


- using formula clocks

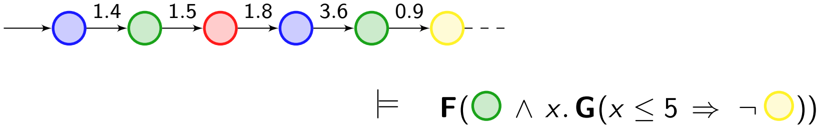
Refs: [1] Alur, Henzinger. *A Really Temporal Logic* (1989).
[2] Koymans. *Specifying Real-Time Properties with Metric Temporal Logic* (1990).

Extending temporal modalities with time

- decorating modalities with timing constraints:



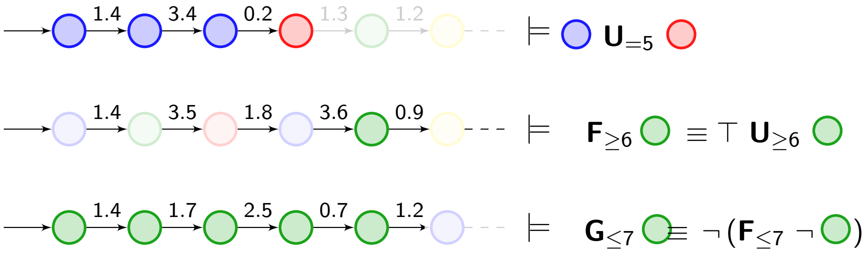
- using formula clocks



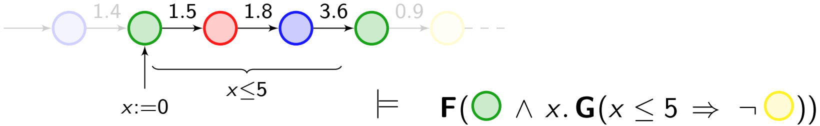
Refs: [1] Alur, Henzinger. *A Really Temporal Logic* (1989).
[2] Koymans. *Specifying Real-Time Properties with Metric Temporal Logic* (1990).

Extending temporal modalities with time

- decorating modalities with timing constraints:



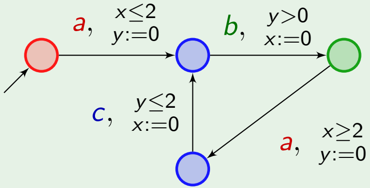
- using formula clocks



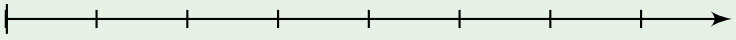
Refs: [1] Alur, Henzinger. *A Really Temporal Logic* (1989).
[2] Koymans. *Specifying Real-Time Properties with Metric Temporal Logic* (1990).

Timed words *vs.* timed state sequences

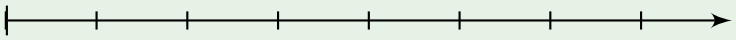
Example



continuous semantics

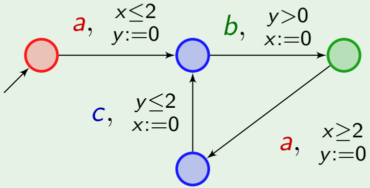


pointwise semantics

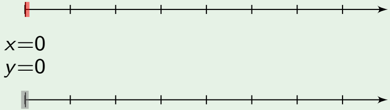


Timed words *vs.* timed state sequences

Example



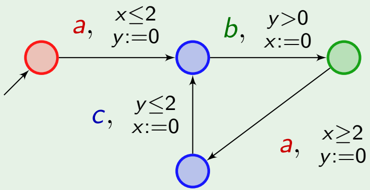
continuous semantics



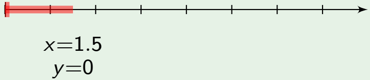
pointwise semantics

Timed words *vs.* timed state sequences

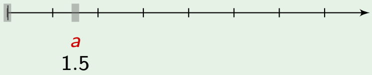
Example



continuous semantics

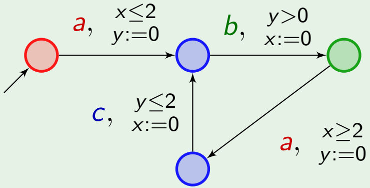


pointwise semantics

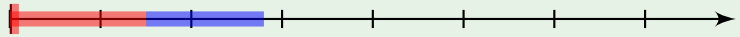


Timed words *vs.* timed state sequences

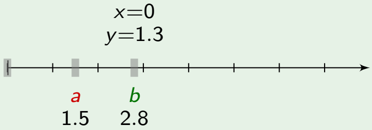
Example



continuous semantics

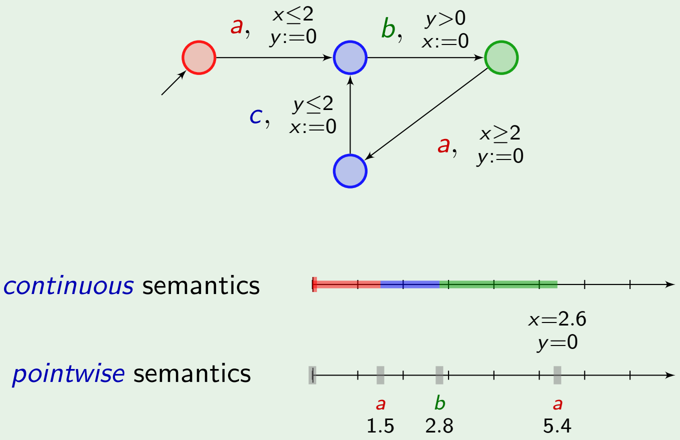


pointwise semantics



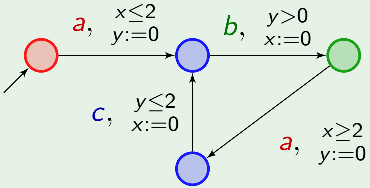
Timed words *vs.* timed state sequences

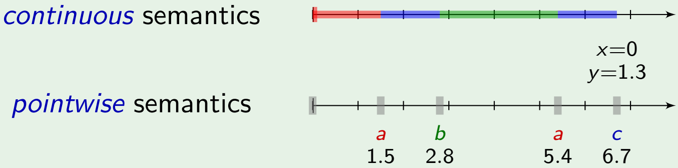
Example



Timed words *vs.* timed state sequences

Example





Timed logics in the pointwise framework

Definition

$$\mathsf{MTL} \ni \varphi ::= $$  $$ \mid \neg \varphi \mid \varphi \vee \varphi \mid \varphi \, \mathbf{U}_I \, \varphi$$

where  ranges over {, , ...} and I is an interval with bounds in $\mathbb{Q}^+ \cup \{+\infty\}$.

Timed logics in the pointwise framework

Definition

Pointwise semantics of MTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$:

- $\pi, i \models \varphi \, \mathbf{U}_I \, \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j \models \psi$,
  - $\pi, i + k \models \varphi$ for all $0 < k < j$,
  - $t_{i+j} - t_i \in I$.

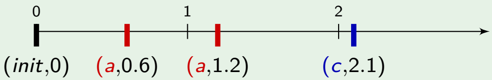
Timed logics in the pointwise framework

Definition

Pointwise semantics of MTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$:

- $\pi, i \models \varphi \; \mathbf{U}_I \; \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j \models \psi$,
  - $\pi, i + k \models \varphi$ for all $0 < k < j$,
  - $t_{i+j} - t_i \in I$.

Example



$a \; \mathbf{U}_{[2,3]} \; c$

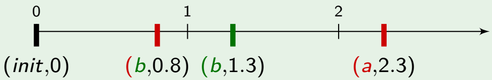
Timed logics in the pointwise framework

Definition

Pointwise semantics of MTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$:

- $\pi, i \models \varphi \; \mathbf{U}_I \; \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j \models \psi$,
  - $\pi, i + k \models \varphi$ for all $0 < k < j$,
  - $t_{i+j} - t_i \in I$.

Example



$\mathbf{F}(b \wedge \bot \; \mathbf{U}_{[1,1]} \; a)$

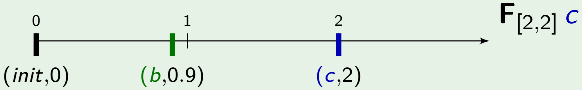
Timed logics in the pointwise framework

Definition

Pointwise semantics of MTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$:

- $\pi, i \models \varphi \, \mathbf{U}_I \, \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j \models \psi$,
  - $\pi, i + k \models \varphi$ for all $0 < k < j$,
  - $t_{i+j} - t_i \in I$.

Example



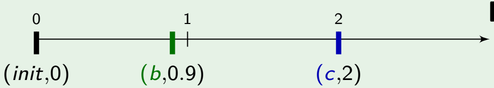
Timed logics in the pointwise framework

Definition

Pointwise semantics of MTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$:

- $\pi, i \models \varphi \, \mathbf{U}_I \, \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j \models \psi$,
  - $\pi, i + k \models \varphi$ for all $0 < k < j$,
  - $t_{i+j} - t_i \in I$.

Example



$\mathbf{F}_{[2,2]}\, c \stackrel{\text{def}}{=} \mathbf{F}_{=2}\, c$

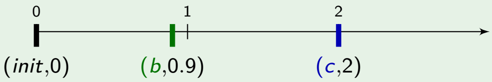
Timed logics in the pointwise framework

Definition

Pointwise semantics of MTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$:

- $\pi, i \models \varphi \, \mathbf{U}_I \, \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j \models \psi$,
  - $\pi, i + k \models \varphi$ for all $0 < k < j$,
  - $t_{i+j} - t_i \in I$.

Example



$\mathbf{F}_{[2,2]}\, c \;\not\equiv\; \mathbf{F}_{=1}\, \mathbf{F}_{=1}\, c$

Timed logics in the pointwise framework

Definition

$$\mathsf{TPTL} \ni \varphi ::= \bigcirc \mid x \sim c \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi \; \mathbf{U} \; \varphi \mid x.\ \varphi$$

where $\bigcirc$ ranges over $\{\bigcirc, \bigcirc, \ldots\}$, x ranges over a set of formula clocks, $c \in \mathbb{Q}^{+}$ and $\sim \in \{<, \leq, =, \geq, >\}$.

Timed logics in the pointwise framework

Definition

Pointwise semantics of TPTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$, under some clock valuation τ: :

- $\pi, i, \tau \models x \sim c$ iff $\tau(x) \sim c$

Timed logics in the pointwise framework

Definition

Pointwise semantics of TPTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$, under some clock valuation τ: :

- $\pi, i, \tau \models x \sim c$ iff $\tau(x) \sim c$
- $\pi, i, \tau \models x.\ \varphi$ iff $\pi, i, \tau[x \leftarrow 0] \models \varphi$

Timed logics in the pointwise framework

Definition

Pointwise semantics of TPTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$, under some clock valuation τ: :

- $\pi, i, \tau \models x \sim c$ iff $\tau(x) \sim c$
- $\pi, i, \tau \models x.\ \varphi$ iff $\pi, i, \tau[x \leftarrow 0] \models \varphi$
- $\pi, i, \tau \models \varphi\ \mathbf{U}\ \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j, \tau + t_{i+j} - t_i \models \psi$,
  - $\pi, i + k, \tau + t_{i+k} - t_i \models \varphi$ for all $0 < k < j$.

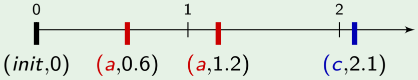
Timed logics in the pointwise framework

Definition

Pointwise semantics of TPTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$, under some clock valuation τ: :

- $\pi, i, \tau \models x \sim c$ iff $\tau(x) \sim c$
- $\pi, i, \tau \models x.\ \varphi$ iff $\pi, i, \tau[x \leftarrow 0] \models \varphi$
- $\pi, i, \tau \models \varphi\ \mathbf{U}\ \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j, \tau + t_{i+j} - t_i \models \psi$,
  - $\pi, i + k, \tau + t_{i+k} - t_i \models \varphi$ for all $0 < k < j$.

Example



$x.(a\ \mathbf{U}\ (c \wedge x \in [2,3]))$

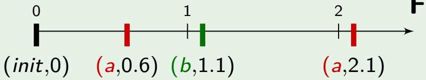
Timed logics in the pointwise framework

Definition

Pointwise semantics of TPTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$, under some clock valuation τ: :

- $\pi, i, \tau \models x \sim c$ iff $\tau(x) \sim c$
- $\pi, i, \tau \models x.\ \varphi$ iff $\pi, i, \tau[x \leftarrow 0] \models \varphi$
- $\pi, i, \tau \models \varphi\ \mathbf{U}\ \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j, \tau + t_{i+j} - t_i \models \psi$,
  - $\pi, i + k, \tau + t_{i+k} - t_i \models \varphi$ for all $0 < k < j$.

Example



$\mathbf{F}(b \wedge x.(\bot\ \mathbf{U}\ (a \wedge x = 1)))$

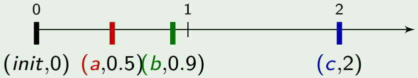
Timed logics in the pointwise framework

Definition

Pointwise semantics of TPTL: over $\pi = (w_i, t_i)_i$ with $t_0 = 0$, under some clock valuation τ: :

- $\pi, i, \tau \models x \sim c$ iff $\tau(x) \sim c$
- $\pi, i, \tau \models x.\ \varphi$ iff $\pi, i, \tau[x \leftarrow 0] \models \varphi$
- $\pi, i, \tau \models \varphi\ \mathbf{U}\ \psi$ iff there exists some $j > 0$ s.t.
  - $\pi, i + j, \tau + t_{i+j} - t_i \models \psi$,
  - $\pi, i + k, \tau + t_{i+k} - t_i \models \varphi$ for all $0 < k < j$.

Example



$$x.\,\mathbf{F}(a \wedge \mathbf{F}(b \wedge x \leq 1))$$

Timed logics in the continuous framework

Definition

Continuous semantics of MTL: over $\pi\colon \mathbb{R}^+ \to \{\bigcirc, \bigcirc, \ldots\}$:

- $\pi, t \models \varphi \ \mathbf{U}_I \ \psi$ iff there exists some $u > 0$ s.t.
  - $\pi, t + u \models \psi$,
  - $\pi, t + v \models \varphi$ for all $0 < v < u$,
  - $u \in I$.

Timed logics in the continuous framework

Definition

Continuous semantics of MTL: over $\pi \colon \mathbb{R}^+ \to \{$, , ...$\}$:

- $\pi, t \models \varphi \; \mathbf{U}_I \; \psi$ iff there exists some $u > 0$ s.t.
  - $\pi, t + u \models \psi$,
  - $\pi, t + v \models \varphi$ for all $0 < v < u$,
  - $u \in I$.
- $\pi, t \models p$ iff $p \in \pi(t)$

Timed logics in the continuous framework

Definition

Continuous semantics of MTL: over $\pi \colon \mathbb{R}^+ \to \{\bigcirc, \bigcirc, \dots\}$:

- $\pi, t \models \varphi \, \mathbf{U}_I \, \psi$ iff there exists some $u > 0$ s.t.
  - $\pi, t + u \models \psi$,
  - $\pi, t + v \models \varphi$ for all $0 < v < u$,
  - $u \in I$.
- $\pi, t \models p$ iff $p \in \pi(t)$

Example

Timed logics in the continuous framework

Definition

Continuous semantics of MTL: over $\pi\colon \mathbb{R}^+ \to \{○, ○, \ldots\}$:

- $\pi, t \models \varphi\ \mathbf{U}_I\ \psi$ iff there exists some $u > 0$ s.t.
  - $\pi, t + u \models \psi$,
  - $\pi, t + v \models \varphi$ for all $0 < v < u$,
  - $u \in I$.
- $\pi, t \models p$ iff $p \in \pi(t)$

Example

$\mathbf{F}_{=2}$ ○

Timed logics in the continuous framework

Definition

Continuous semantics of MTL: over $\pi : \mathbb{R}^+ \to \{\bigcirc, \bigcirc, \ldots\}$:

- $\pi, t \models \varphi \ \mathbf{U}_I \ \psi$ iff there exists some $u > 0$ s.t.
  - $\pi, t + u \models \psi$,
  - $\pi, t + v \models \varphi$ for all $0 < v < u$,
  - $u \in I$.
- $\pi, t \models p$ iff $p \in \pi(t)$

Example

$$\mathbf{F}_{=2} \bigcirc \ \equiv \ \mathbf{F}_{=1}(\mathbf{F}_{=1} \bigcirc)$$

Timed logics in the continuous framework

Definition

Continuous semantics of TPTL: over $\pi\colon \mathbb{R}^+ \rightarrow \{\bigcirc, \bigcirc, \ldots\}$:

- $\pi, t, \tau \models x \sim c$ iff $\tau(x) \sim c$

Timed logics in the continuous framework

Definition

Continuous semantics of TPTL: over $\pi\colon \mathbb{R}^+ \to \{\bigcirc, \bigcirc, \ldots\}$:

- $\pi, t, \tau \models x \sim c$ iff $\tau(x) \sim c$
- $\pi, t, \tau \models x.\ \varphi$ iff $\pi, i, \tau[x \leftarrow 0] \models \varphi$

Timed logics in the continuous framework

Definition

Continuous semantics of TPTL: over $\pi\colon \mathbb{R}^+ \to \{\bigcirc, \bigcirc, \ldots\}$:

- $\pi, t, \tau \models x \sim c$ iff $\tau(x) \sim c$
- $\pi, t, \tau \models x.\ \varphi$ iff $\pi, i, \tau[x \leftarrow 0] \models \varphi$
- $\pi, t, \tau \models \varphi \ \mathbf{U}\ \psi$ iff there exists some $u > 0$ s.t.
  - $\pi, t + u, \tau + u - t \models \psi$,
  - $\pi, i + k, \tau + v - t \models \varphi$ for all $0 < v < u$.

Timed logics in the continuous framework

Definition

Continuous semantics of TPTL: over $\pi\colon \mathbb{R}^+ \rightarrow \{\bigcirc, \bigcirc, \ldots\}$:

- $\pi, t, \tau \models x \sim c$ iff $\tau(x) \sim c$
- $\pi, t, \tau \models x.\ \varphi$ iff $\pi, i, \tau[x \leftarrow 0] \models \varphi$
- $\pi, t, \tau \models \varphi \mathbf{U} \psi$ iff there exists some $u > 0$ s.t.
  - $\pi, t+u, \tau+u-t \models \psi$,
  - $\pi, i+k, \tau+v-t \models \varphi$ for all $0 < v < u$.

Example

$x.((\bigcirc \vee \bigcirc) \mathbf{U} (\bigcirc \wedge x \leq 2))$

Timed logics in the continuous framework

Definition

Continuous semantics of TPTL: over $\pi\colon \mathbb{R}^+ \rightarrow \{\bigcirc, \bigcirc, \ldots\}$:

- $\pi, t, \tau \models x \sim c$ iff $\tau(x) \sim c$
- $\pi, t, \tau \models x.\ \varphi$ iff $\pi, i, \tau[x \leftarrow 0] \models \varphi$
- $\pi, t, \tau \models \varphi\ \mathbf{U}\ \psi$ iff there exists some $u > 0$ s.t.
  - $\pi, t+u, \tau+u-t \models \psi$,
  - $\pi, i+k, \tau+v-t \models \varphi$ for all $0 < v < u$.

Example

$$x.\,\mathbf{F}(\bigcirc \wedge \mathbf{F}(\bigcirc \wedge x \leq 2))$$

Relative expressiveness of TPTL and MTL

Lemma

MTL can be translated into TPTL.

Proof.

$$\varphi \,\mathbf{U}_I\, \psi \;\equiv\; x.\, \varphi \,\mathbf{U}\, (\psi \wedge x \in I).$$

□

Relative expressiveness of TPTL and MTL

Lemma

MTL can be translated into TPTL.

Proof.

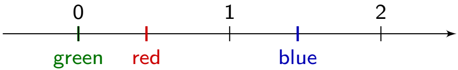
$$\varphi \, \mathbf{U}_I \, \psi \;\equiv\; x.\, \varphi \, \mathbf{U} \, (\psi \wedge x \in I).$$

□

Conversely, consider the following TPTL formula:

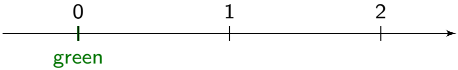
$$\mathbf{G}\left[\bigcirc \Rightarrow x.\, \mathbf{F}(\bigcirc \wedge \mathbf{F}(\bigcirc \wedge x \leq 2))\right].$$

It characterizes the following pattern:



Relative expressiveness of TPTL and MTL

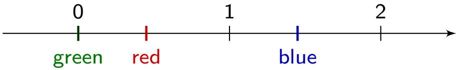
$$\mathbf{G}\left[{\color{green}\bigcirc} \Rightarrow x.\,\mathbf{F}\left({\color{red}\bigcirc} \wedge \mathbf{F}\left({\color{blue}\bigcirc} \wedge x \leq 2\right)\right)\right].$$



$$\mathbf{G} \quad {\color{green}\bigcirc} \Rightarrow \left\{ \right.$$

Relative expressiveness of TPTL and MTL

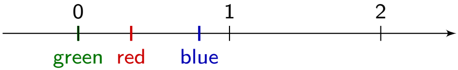
$$\mathbf{G}\left[\textcolor{green}{\bigcirc} \Rightarrow x.\,\mathbf{F}(\textcolor{red}{\bigcirc} \wedge \mathbf{F}(\textcolor{blue}{\bigcirc} \wedge x \leq 2))\right].$$



$$\mathbf{G} \quad \textcolor{green}{\bigcirc} \Rightarrow \left\{ \begin{array}{l} \mathbf{F}_{[0,1]}\, \textcolor{red}{\bigcirc} \wedge \mathbf{F}_{[1,2]}\, \textcolor{blue}{\bigcirc} \\ \\ \\ \end{array} \right.$$

Relative expressiveness of TPTL and MTL

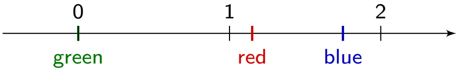
$$\mathbf{G}\left[{\color{green}\bigcirc} \Rightarrow x.\,\mathbf{F}\left({\color{red}\bigcirc} \wedge \mathbf{F}\left({\color{blue}\bigcirc} \wedge x \leq 2\right)\right)\right].$$



$$\mathbf{G}\quad {\color{green}\bigcirc} \Rightarrow \left\{ \begin{array}{l} \quad \mathbf{F}_{[0,1]}\,{\color{red}\bigcirc} \wedge \mathbf{F}_{[1,2]}\,{\color{blue}\bigcirc} \\ \vee \\ \quad \mathbf{F}_{[0,1]}\left({\color{red}\bigcirc} \wedge \mathbf{F}_{[0,1]}\,{\color{blue}\bigcirc}\right) \\ \\ \end{array} \right.$$

Relative expressiveness of TPTL and MTL

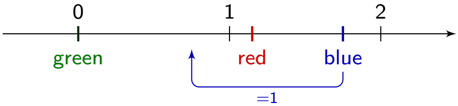
$$\mathbf{G}\,[\,\textcolor{green}{\bigcirc} \Rightarrow x.\,\mathbf{F}(\textcolor{red}{\bigcirc} \wedge \mathbf{F}(\textcolor{blue}{\bigcirc} \wedge x \leq 2))\,].$$



$$\mathbf{G}\quad \textcolor{green}{\bigcirc} \Rightarrow \left\{ \begin{array}{l} \quad \mathbf{F}_{[0,1]}\,\textcolor{red}{\bigcirc} \wedge \mathbf{F}_{[1,2]}\,\textcolor{blue}{\bigcirc} \\ \vee \\ \quad \mathbf{F}_{[0,1]}(\textcolor{red}{\bigcirc} \wedge \mathbf{F}_{[0,1]}\,\textcolor{blue}{\bigcirc}) \\ \\ \end{array} \right.$$

Relative expressiveness of TPTL and MTL

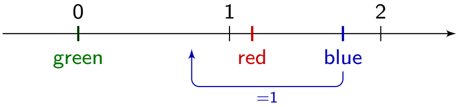
$$\mathbf{G}\left[{\color{green}\bigcirc} \Rightarrow x.\,\mathbf{F}({\color{red}\bigcirc} \wedge \mathbf{F}({\color{blue}\bigcirc} \wedge x \leq 2))\right].$$



$$\mathbf{G} \quad {\color{green}\bigcirc} \Rightarrow \begin{cases} \mathbf{F}_{[0,1]}\,{\color{red}\bigcirc} \wedge \mathbf{F}_{[1,2]}\,{\color{blue}\bigcirc} \\ \vee \\ \mathbf{F}_{[0,1]}({\color{red}\bigcirc} \wedge \mathbf{F}_{[0,1]}\,{\color{blue}\bigcirc}) \\ \vee \\ \mathbf{F}_{[0,1]}(\mathbf{F}_{(0,1)}\,{\color{red}\bigcirc} \wedge \mathbf{F}_{=1}\,{\color{blue}\bigcirc}) \end{cases}$$

Relative expressiveness of TPTL and MTL

$$\mathbf{G}\left[\textcolor{green}{\bigcirc} \Rightarrow x.\,\mathbf{F}(\textcolor{red}{\bigcirc} \wedge \mathbf{F}(\textcolor{blue}{\bigcirc} \wedge x \leq 2))\right].$$



$$\mathbf{G} \quad \textcolor{green}{\bigcirc} \Rightarrow \left\{ \begin{array}{ll} & \mathbf{F}_{[0,1]}\,\textcolor{red}{\bigcirc} \wedge \mathbf{F}_{[1,2]}\,\textcolor{blue}{\bigcirc} \\ \vee & \\ & \mathbf{F}_{[0,1]}(\textcolor{red}{\bigcirc} \wedge \mathbf{F}_{[0,1]}\,\textcolor{blue}{\bigcirc}) \\ \vee & \\ & \mathbf{F}_{[0,1]}(\mathbf{F}_{(0,1)}\,\textcolor{red}{\bigcirc} \wedge \mathbf{F}_{=1}\,\textcolor{blue}{\bigcirc}) \end{array} \right.$$

Remark

This translation is only valid in the continuous semantics

Relative expressiveness of TPTL and MTL

Theorem

TPTL is strictly more expressive than MTL.

Refs: [1] Bouyer, Chevalier, M. *On the Expressiveness of TPTL and MTL* (2005).

Relative expressiveness of TPTL and MTL

Theorem

TPTL is strictly more expressive than MTL.

Proof.

- In the pointwise semantics:

$$\mathbf{G}\left[{\color{green}\bigcirc} \Rightarrow x.\,\mathbf{F}({\color{blue}\bigcirc} \wedge \mathbf{F}({\color{blue}\bigcirc} \wedge x \leq 2))\right]$$

  cannot be expressed in MTL.

- In both semantics:

$$\varphi = x.\ \mathbf{F}({\color{green}\bigcirc} \wedge x \leq 1 \wedge \mathbf{G}(x \leq 1 \Rightarrow \neg {\color{blue}\bigcirc}))$$

  cannot be expressed in MTL.

□

Refs: [1] Bouyer, Chevalier, M. *On the Expressiveness of TPTL and MTL* (2005).

Outline of the talk

1. Introduction

2. Extending temporal logics with real-time constraints
   - Continuous and pointwise semantics
   - Expressiveness issues

3. Model checking timed linear-time logics
   - Undecidability of MTL and TPTL
   - Decidable fragments

4. Model checking timed branching-time logics

5. Conclusions and open problems

MTL model-checking

Theorem

MTL model-checking and satisfiability are undecidable under the continuous semantics.

Refs: [1] Alur, Henzinger. *Real-time logics: Complexity and expressiveness* (1990).

MTL model-checking

Theorem

MTL model-checking and satisfiability are undecidable under the continuous semantics.

Proof.

Encode the halting problem of a Turing machine:

One time-unit = one configuration of the Turing machine

Refs: [1] Alur, Henzinger. *Real-time logics: Complexity and expressiveness* (1990).

MTL model-checking

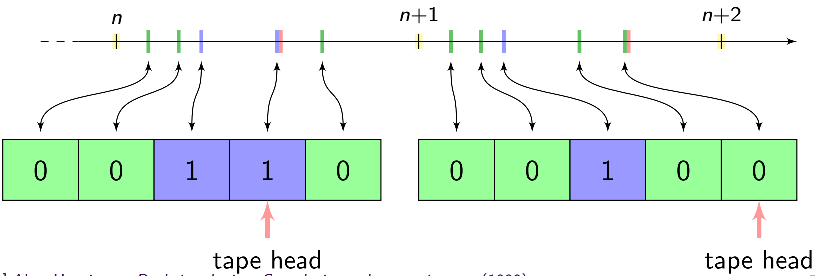
Theorem

MTL model-checking and satisfiability are undecidable under the continuous semantics.

Proof.

Encode the halting problem of a Turing machine:

One time-unit = one configuration of the Turing machine



Refs: [1] Alur, Henzinger. *Real-time logics: Complexity and expressiveness* (1990).

MTL model-checking

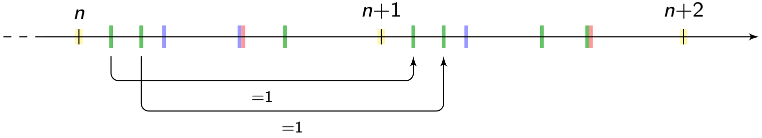
Theorem

MTL model-checking and satisfiability are undecidable under the continuous semantics.

Proof.

Encode the halting problem of a Turing machine:

One time-unit = one configuration of the Turing machine



Refs: [1] Alur, Henzinger. *Real-time logics: Complexity and expressiveness* (1990).

MTL model-checking

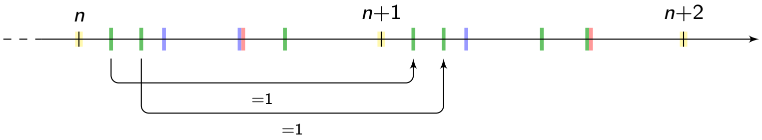
Theorem

MTL model-checking and satisfiability are undecidable under the continuous semantics.

Proof.

Encode the halting problem of a Turing machine:

One time-unit = one configuration of the Turing machine

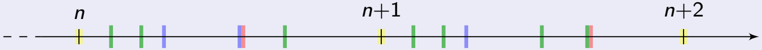


Refs: [1] Alur, Henzinger. *Real-time logics: Complexity and expressiveness* (1990).

MTL model-checking

Remark

This reduction requires continuous semantics, or the use of past-time modalities:

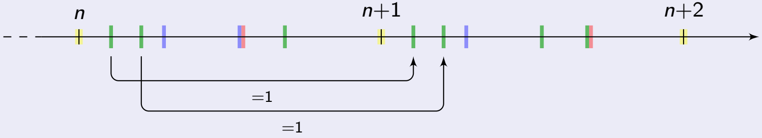


Refs: [1] Ouaknine, Worrell. *On the decidability of Metric Temporal Logic* (2005).
[2] Ouaknine, Worrell. *On Metric Temporal Logic and faulty Turing machines* (2006).

MTL model-checking

Remark

This reduction requires continuous semantics, or the use of past-time modalities:



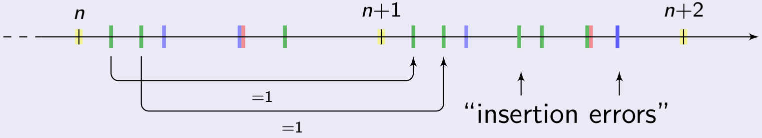
Refs: [1] Ouaknine, Worrell. *On the decidability of Metric Temporal Logic* (2005).

[2] Ouaknine, Worrell. *On Metric Temporal Logic and faulty Turing machines* (2006).

MTL model-checking

Remark

This reduction requires continuous semantics, or the use of past-time modalities:

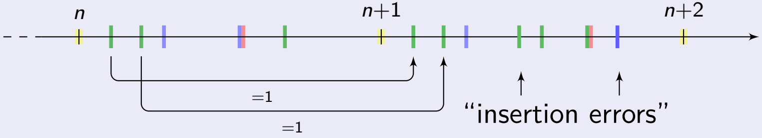


Refs: [1] Ouaknine, Worrell. *On the decidability of Metric Temporal Logic* (2005).
[2] Ouaknine, Worrell. *On Metric Temporal Logic and faulty Turing machines* (2006).

MTL model-checking

Remark

This reduction requires continuous semantics, or the use of past-time modalities:



Theorem

Under pointwise semantics, MTL model-checking and satisfiability

- *are undecidable over infinite timed words;*
- *are decidable (with non-primitive recursive complexity) over finite timed words.*

Refs: [1] Ouaknine, Worrell. *On the decidability of Metric Temporal Logic* (2005).
[2] Ouaknine, Worrell. *On Metric Temporal Logic and faulty Turing machines* (2006).

Metric Interval Temporal Logic

Definition

MITL is the fragment of MTL where punctuality is not allowed:

$$\mathsf{MITL} \ni \varphi ::= \bigcirc \mid \neg \varphi \mid \varphi \vee \varphi \mid \varphi \, \mathbf{U}_I \, \varphi$$

where $\bigcirc$ ranges over $\{\bigcirc, \bigcirc, \ldots\}$ and I is a non-punctual interval with bounds in $\mathbb{Q}^+ \cup \{+\infty\}$.

Refs: [1] Alur, Feder, Henzinger. *The benefits of relaxing punctuality* (1991).

Metric Interval Temporal Logic

Definition

MITL is the fragment of MTL where punctuality is not allowed:

$$\mathsf{MITL} \ni \varphi ::= \bigcirc \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi \, \mathbf{U}_I \, \varphi$$

where $\bigcirc$ ranges over $\{\bigcirc, \bigcirc, \ldots\}$ and I is a non-punctual interval with bounds in $\mathbb{Q}^+ \cup \{+\infty\}$.

Example

- $\mathbf{G}(\bigcirc \Rightarrow \mathbf{F}_{[1,2]} \bigcirc)$ is an MITL formula;
- $\mathbf{G}(\bigcirc \Rightarrow \mathbf{F}_{=1} \bigcirc)$ is not.

Refs: [1] Alur, Feder, Henzinger. *The benefits of relaxing punctuality* (1991).

Metric Interval Temporal Logic

Definition

MITL is the fragment of MTL where punctuality is not allowed:

$$\mathsf{MITL} \ni \varphi ::= \bigcirc \mid \neg \varphi \mid \varphi \vee \varphi \mid \varphi \, \mathbf{U}_I \, \varphi$$

where $\bigcirc$ ranges over $\{\bigcirc, \bigcirc, \dots\}$ and I is a non-punctual interval with bounds in $\mathbb{Q}^+ \cup \{+\infty\}$.

Example

- $\mathbf{G}(\bigcirc \Rightarrow \mathbf{F}_{[1,2]} \bigcirc)$ is an MITL formula;
- $\mathbf{G}(\bigcirc \Rightarrow \mathbf{F}_{=1} \bigcirc)$ is not.

Theorem

MITL model checking and satisfiability are EXPSPACE-complete.

Refs: [1] Alur, Feder, Henzinger. *The benefits of relaxing punctuality* (1991).

(Co)Flat MTL

Definition

CoFlatMTL is the fragment of MTL defined as:

$$\text{CoFlatMTL} \ni \varphi ::= \bigcirc \mid \neg \bigcirc \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid$$
$$\varphi \; \mathbf{U}_I \; \varphi \mid \varphi \; \mathbf{U}_J \; \psi \mid \varphi \; \mathbf{R}_I \; \varphi \mid \psi \; \mathbf{R}_J \; \varphi$$

where

- $\bigcirc$ ranges over $\{\bigcirc, \bigcirc, \ldots\}$,
- I ranges over *bounded* intervals with bounds in $\mathbb{Q}$,
- J ranges over intervals with bounds in $\mathbb{Q} \cup \{+\infty\}$, and
- ψ ranges over MITL.

Refs: [1] Bouyer, M., Ouaknine, Worrell. *The Cost of Punctuality* (2007).

(Co)Flat MTL

Definition

CoFlatMTL is the fragment of MTL defined as:

$$\mathrm{CoFlatMTL} \ni \varphi ::= \bigcirc \mid \neg \bigcirc \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \varphi \,\mathbf{U}_I\, \varphi \mid \varphi \,\mathbf{U}_J\, \psi \mid \varphi \,\mathbf{R}_I\, \varphi \mid \psi \,\mathbf{R}_J\, \varphi$$

Remark

CoFlatMTL is not closed under negation.

Refs: [1] Bouyer, M., Ouaknine, Worrell. *The Cost of Punctuality* (2007).

(Co)Flat MTL

Definition

CoFlatMTL is the fragment of MTL defined as:

$$\text{CoFlatMTL} \ni \varphi ::= \bigcirc \mid \neg \bigcirc \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid$$
$$\varphi \ \mathbf{U}_I \ \varphi \mid \varphi \ \mathbf{U}_J \ \psi \mid \varphi \ \mathbf{R}_I \ \varphi \mid \psi \ \mathbf{R}_J \ \varphi$$

Remark

CoFlatMTL is not closed under negation.

Example

- $\mathbf{G}(\bigcirc \Rightarrow \mathbf{F}_{=1} \bigcirc)$ is in CoFlatMTL.
- $\mathbf{F}(\bigcirc \wedge \mathbf{G}_{=1} \bigcirc)$ is in FlatMTL, but not in CoFlatMTL.

Refs: [1] Bouyer, M., Ouaknine, Worrell. *The Cost of Punctuality* (2007).

(Co)Flat MTL

Definition

CoFlatMTL is the fragment of MTL defined as:

$$\text{CoFlatMTL} \ni \varphi ::= \bigcirc \mid \neg \bigcirc \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \varphi \, \mathbf{U}_I \, \varphi \mid \varphi \, \mathbf{U}_J \, \psi \mid \varphi \, \mathbf{R}_I \, \varphi \mid \psi \, \mathbf{R}_J \, \varphi$$

Remark

CoFlatMTL is not closed under negation.

Theorem

CoFlatMTL model-checking is EXPSPACE-complete.
CoFlatMTL satisfiability is undecidable.

Refs: [1] Bouyer, M., Ouaknine, Worrell. *The Cost of Punctuality* (2007).

Outline of the talk

1. Introduction
2. Extending temporal logics with real-time constraints
   - Continuous and pointwise semantics
   - Expressiveness issues
3. Model checking timed linear-time logics
   - Undecidability of MTL and TPTL
   - Decidable fragments
4. Model checking timed branching-time logics
5. Conclusions and open problems

Branching-time logics with timing constraints – syntax

Definition

$$\mathsf{TCTL} \ni \varphi ::= \bigcirc \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\varphi\ \mathbf{U}_{\sim c}\ \varphi \mid \mathbf{A}\varphi\ \mathbf{U}_{\sim c}\ \varphi$$

where $\bigcirc \in \{\bigcirc, \bigcirc, \bigcirc, \ldots\}$, $\sim \in \{\leq, <, =, >, \geq\}$ and $c \in \mathbb{N}$.

Refs: [1] Alur, Courcoubetis, Dill. *Model-Checking in Dense Real-Time* (1993).

Branching-time logics with timing constraints – syntax

Definition

$$\mathsf{TCTL} \ni \varphi ::= {\color{gray}\bigcirc} \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\varphi\ \mathbf{U}_{\sim c}\ \varphi \mid \mathbf{A}\varphi\ \mathbf{U}_{\sim c}\ \varphi$$

where ${\color{gray}\bigcirc} \in \{{\color{red}\bigcirc}, {\color{blue}\bigcirc}, {\color{green}\bigcirc}, \ldots\}$, $\sim\, \in \{\leq, <, =, >, \geq\}$ and $c \in \mathbb{N}$.

Example

- $\mathbf{A}\,\mathbf{G}({\color{red}\bigcirc} \Rightarrow \mathbf{E}\,\mathbf{F}_{\leq 5}\,{\color{green}\bigcirc})$

Refs: [1] Alur, Courcoubetis, Dill. *Model-Checking in Dense Real-Time* (1993).

Branching-time logics with timing constraints – syntax

Definition

$$\text{TCTL} \ni \varphi ::= \bigcirc \mid \neg \varphi \mid \varphi \wedge \varphi \mid \mathbf{E}\varphi \, \mathbf{U}_{\sim c} \, \varphi \mid \mathbf{A}\varphi \, \mathbf{U}_{\sim c} \, \varphi$$

where $\bigcirc \in \{\bigcirc, \bigcirc, \bigcirc, \ldots\}$, $\sim \in \{\leq, <, =, >, \geq\}$ and $c \in \mathbb{N}$.

Example

- $\mathbf{A}\,\mathbf{G}(\bigcirc \Rightarrow \mathbf{E}\,\mathbf{F}_{\leq 5}\,\bigcirc)$
- $\mathbf{A}\,\mathbf{F}(\mathbf{A}\,\mathbf{G}_{\leq 5}\,\bigcirc)$

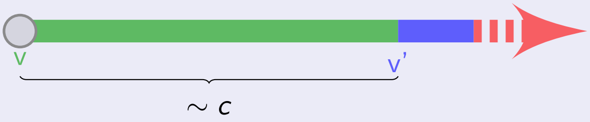
Refs: [1] Alur, Courcoubetis, Dill. *Model-Checking in Dense Real-Time* (1993).

Branching-time logics with timing constraints – semantics

Definition

The semantics of TCTL is defined as follows: let ◯ be a location and v be a clock valuation.

- ◯, $v \models \mathbf{E}($◯ $\mathbf{U}_{\sim c}$ ◯$)$ iff there is a run from (◯, v) such that



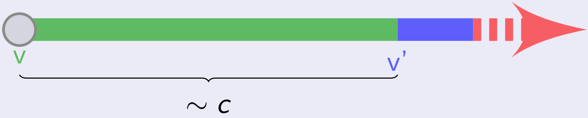
- ◯, $v \models \mathbf{A}($◯ $\mathbf{U}_{\sim c}$ ◯$)$ is defined similarly.

Branching-time logics with timing constraints – semantics

Definition

The semantics of TCTL is defined as follows: let ○ be a location and v be a clock valuation.

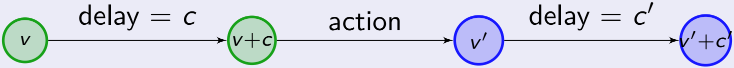
- ○, $v \models \mathbf{E}(○\ \mathbf{U}_{\sim c}\ ○)$ iff there is a run from (○, v) such that



- ○, $v \models \mathbf{A}(○\ \mathbf{U}_{\sim c}\ ○)$ is defined similarly.

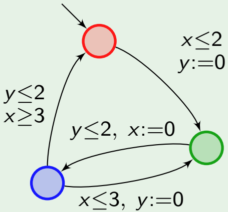
Remark

We could also define a pointwise semantics:



Branching-time logics with timing constraints – semantics

Example

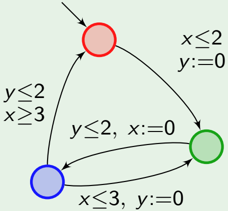


- ${\color{green}\bigcirc}, \begin{pmatrix} x{=}1.2 \\ y{=}0.4 \end{pmatrix} \models \mathbf{E}\,{\color{green}\bigcirc}\ \mathbf{U}_{\geq 1}\ {\color{blue}\bigcirc}$

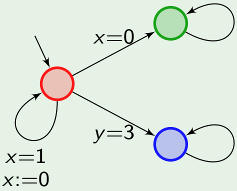
- ${\color{green}\bigcirc}, \begin{pmatrix} x{=}1.2 \\ y{=}0.4 \end{pmatrix} \models \mathbf{A}\,\mathbf{G}\ \neg\ {\color{red}\bigcirc}$

Branching-time logics with timing constraints – semantics

Example



- , $\begin{pmatrix} x{=}1.2 \\ y{=}0.4 \end{pmatrix} \models$ **E**  $\mathbf{U}_{\geq 1}$ 
- , $\begin{pmatrix} x{=}1.2 \\ y{=}0.4 \end{pmatrix} \models$ **A G** $\neg$ 



- , $\begin{pmatrix} x{=}0 \\ y{=}0 \end{pmatrix} \overset{?}{\models}$ **E**(**E F**$_{=1}$ ) $\mathbf{U}_{=3}$ 

TCTL model checking

Lemma

Let  *be a location and* φ *be a TCTL formula. For any two valuations* v *and* v' *that belong to the same region,*

 $, v \models \varphi \quad \Leftrightarrow \quad$  $, v' \models \varphi.$

Refs: [1] Alur, Courcoubetis, Dill. *Model-Checking in Dense Real-Time* (1993).

TCTL model checking

Lemma

Let $\bigcirc$ be a location and φ be a TCTL formula. For any two valuations v and v' that belong to the same region,

$$\bigcirc, v \models \varphi \quad \Leftrightarrow \quad \bigcirc, v' \models \varphi.$$

Proof.

By induction on φ. □

Refs: [1] Alur, Courcoubetis, Dill. *Model-Checking in Dense Real-Time* (1993).

TCTL model checking

Lemma

Let $\bigcirc$ *be a location and* φ *be a TCTL formula. For any two valuations* v *and* v' *that belong to the same region,*

$$\bigcirc, v \models \varphi \quad \Leftrightarrow \quad \bigcirc, v' \models \varphi.$$

Proof.

By induction on φ. □

Theorem

TCTL model-checking is PSPACE-complete.

Refs: [1] Alur, Courcoubetis, Dill. *Model-Checking in Dense Real-Time* (1993).

TCTL model checking

Lemma

Let $\bigcirc$ be a location and φ be a TCTL formula. For any two valuations v and v' that belong to the same region,

$$\bigcirc, v \models \varphi \quad \Leftrightarrow \quad \bigcirc, v' \models \varphi.$$

Proof.

By induction on φ. □

Theorem

TCTL model-checking is PSPACE-complete.

Proof.

Space-efficient CTL labelling algorithm on the region graph. □

Refs: [1] Alur, Courcoubetis, Dill. *Model-Checking in Dense Real-Time* (1993).

Outline of the talk

Conclusions and perspectives

Real-time temporal logics have been much studied:

Conclusions and perspectives

Real-time temporal logics have been much studied:

- linear-time:
  - natural extensions of LTL are undecidable;
  - several restrictions lead to decidability;
  - however, model-checking linear-time logics is hard;
  - no implementation exists.

Conclusions and perspectives

Real-time temporal logics have been much studied:

- linear-time:
  - natural extensions of LTL are undecidable;
  - several restrictions lead to decidability;
  - however, model-checking linear-time logics is hard;
  - no implementation exists.
- branching-time:
  - TCTL model-checking is in PSPACE;
  - can be made efficient in practice;
  - implemented in several tools (Uppaal, Kronos, ...)

Conclusions and perspectives

Real-time temporal logics have been much studied:

- linear-time:
  - natural extensions of LTL are undecidable;
  - several restrictions lead to decidability;
  - however, model-checking linear-time logics is hard;
  - no implementation exists.
- branching-time:
  - TCTL model-checking is in PSPACE;
  - can be made efficient in practice;
  - implemented in several tools (Uppaal, Kronos, ...)

Hot topics in real-time temporal logic model-checking:

- symbolic algorithms for linear-time temporal logics;
- robust model-checking.