Timed Control with Partial Observability

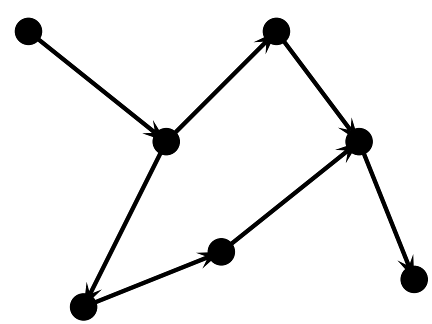
Patricia Bouyer[♣], Deepak D'Souza[♦], Madhusudan P.[⋆], Antoine Petit[♣]

[♣] LSV – CNRS & ENS de Cachan – France
[♦] Chennai Mathematical Institute – India
[⋆] University of Pennsylvania – USA

Model-checking

Does the system *satisfy* the property?

Modelling

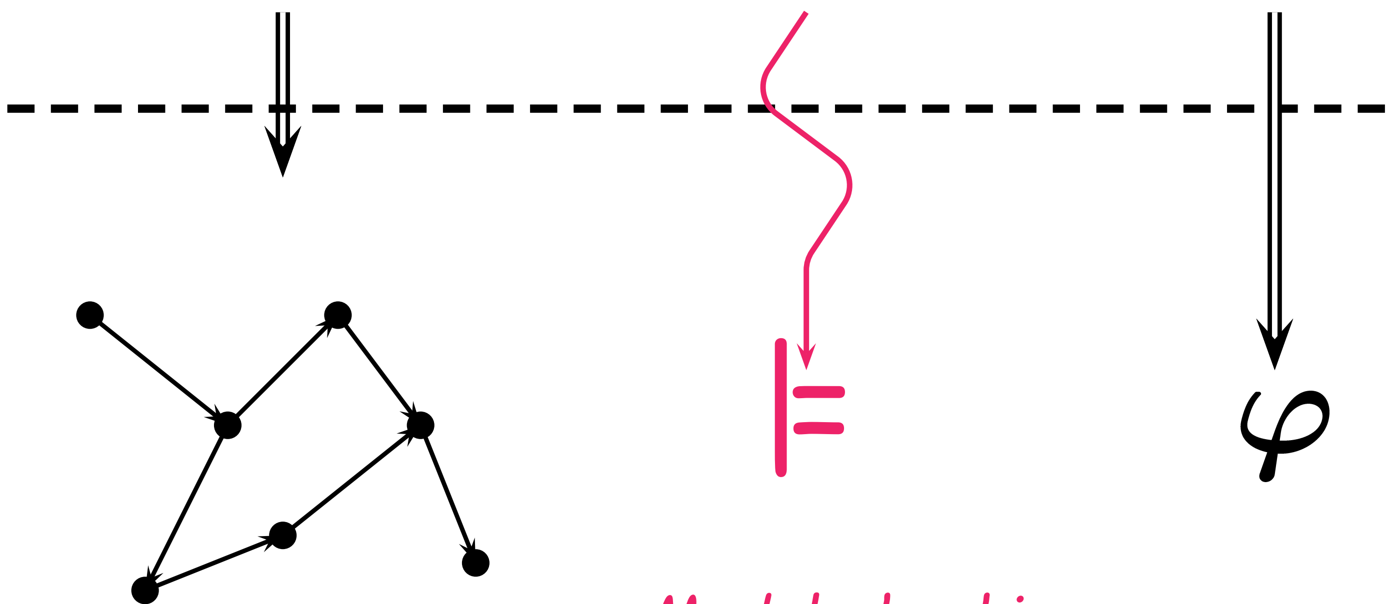


φ

Model-checking

Does the system *satisfy* the property?

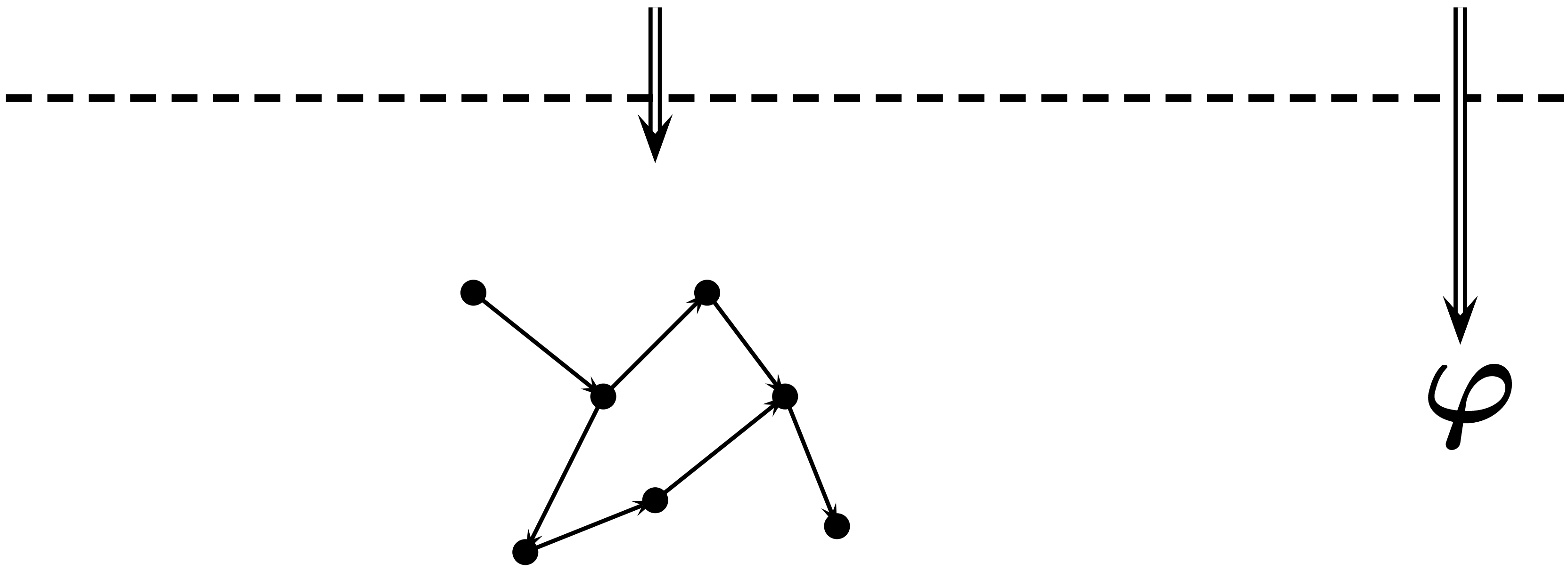
Modelling



Model-checking Algorithm

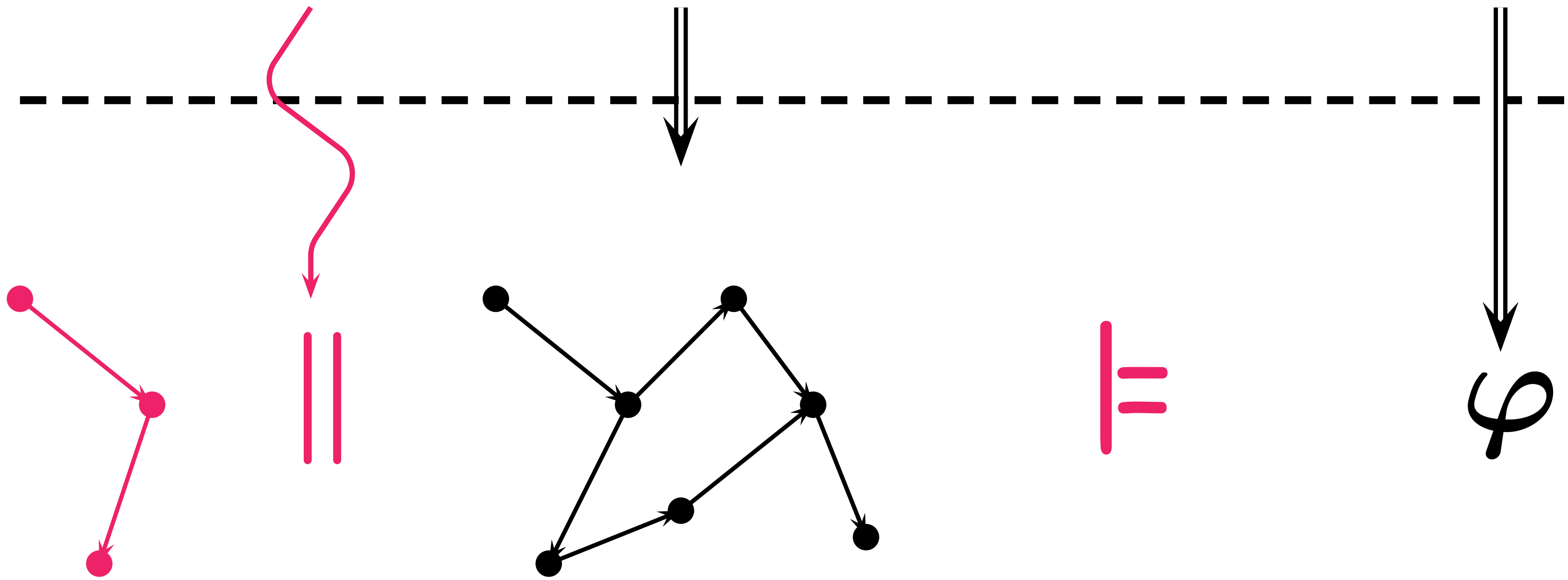
Controller synthesis

Can we force the system *satisfying* the property?



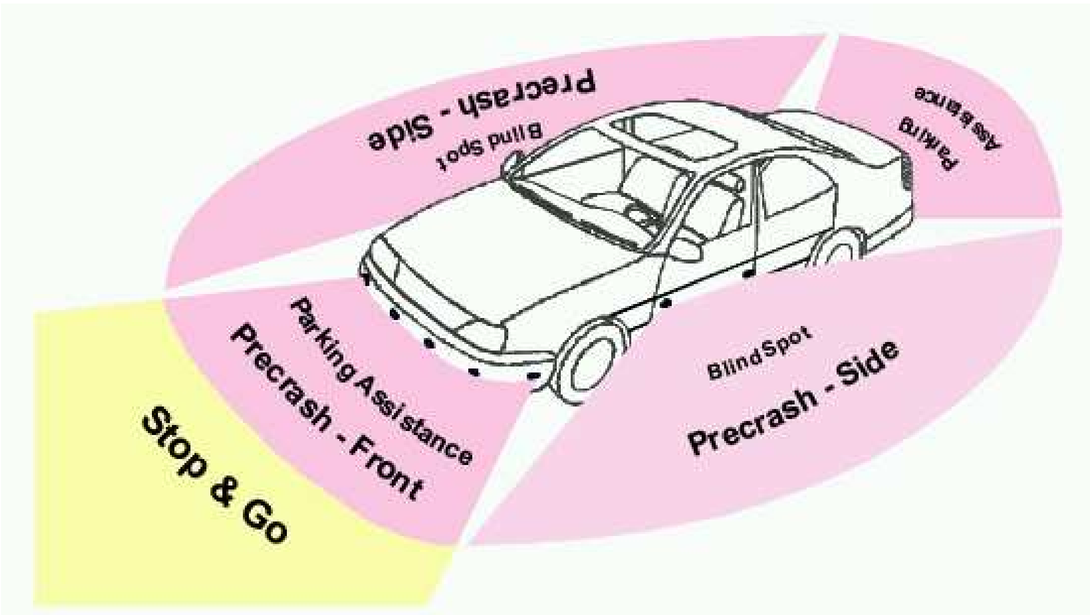
Controller synthesis

Can we force the system *satisfying* the property?



Controller synthesis

An example, the car periphery supervision



- Embedded system
- Hostile environment
- Sensors
  - distances
  - speeds

➜ controller with sampling (reaction time of the sensors) of hybrid systems

What can do the controller?

Plant = System + Environment

- What can it observe/control? **Not everything!**
  - some actions are non-controllable
  - some non-controllable actions are even non-observable

    [Partial observability]

- How powerful can it be? **Not too much!**
  - **non-blocking**: does not add new deadlocks
  - **non-restricting**: can not restrict environment actions

Untimed framework

- **Global observability:**
related to two-player games and to tree automata theory [Thomas'02]

  Player 1: plays with controllable actions
  Player 2: plays with uncontrollable actions

  « Player 1 has a winning strategy iff there exists a controller »

- **Partial observability (or incomplete information):** [KV97]

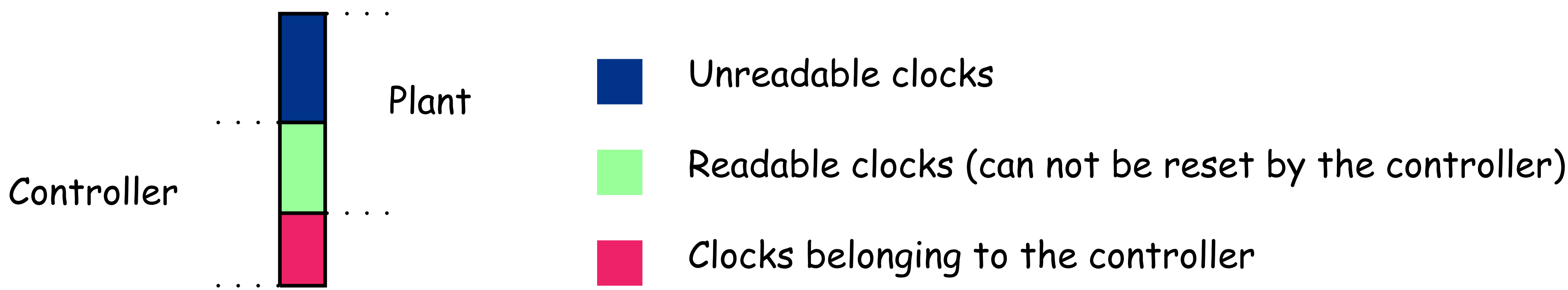
	LTL	CTL*	CTL
Partial obs.	2EXP-comp.	2EXP-comp.	EXP-comp.
Global obs.	2EXP-comp.	2EXP-comp.	EXP-comp.

  Two-player games with incomplete information [Reif84,AVW03]

Partial observability: not more complicated than global observability

Timed framework: what's specific?

- Clocks of the plant can be readable or unreadable (for the controller)



- Which constants clocks can be compared with? $\rightsquigarrow$ (X, m, max)

$$x \sim c \quad \Longrightarrow \quad c \in \frac{\mathbb{Z}}{m} \ \text{ and } \ |c| \leq max$$

→ Two different problems

- fixing the resources, does there exist a controller s.t. ...?
- do there exist resources s.t. there exists a controller s.t. ...?

Notations

- **Plant:** $\mathcal{P}$ ➜ integrates both real system and environment
- **Controller:** $\mathcal{C}$
- **Controlled system:** $\mathcal{P} \parallel \mathcal{C}$

➜ « $\parallel$ » takes into account observability

L(·) timed language associated with a timed automaton

Specifications

- **Internal specifications**: condition on the states of the plant
- **External specifications**: given e.g. by a timed automaton $\mathcal{S}$
  - **Desired behaviors:** $L(\mathcal{P} \parallel \mathcal{C}) \subseteq L(\mathcal{S})$
  - **Undesired behaviors:** $L(\mathcal{P} \parallel \mathcal{C}) \cap L(\mathcal{S}) = \emptyset$

Particular case:

$$\text{external det. spec.} \quad \equiv \quad \text{internal spec.}$$

State of the art

Full observability hypothesis			
Resources	Det. Spec. (Internal/External)	External Non-deterministic Spec.	
		Desired behaviors	Undesired behaviors
Fixed	Decidable [WTH91,AMPS98]	Undecidable [DM02]	Decidable [DM02]
Non-fixed	Decidable [DM02]	Undecidable [DM02]	Undecidable [DM02]

Partial observability hypothesis			
Resources	Det. Spec. (Internal/External)	External Non deterministic Spec.	
		Desired behaviors	Undesired behaviors
Fixed	?	Undecidable	?
Non-fixed	?	Undecidable	Undecidable

Synthesis with non-fixed resources

Full observability hypothesis			
Resources	Det. Spec. (Internal/External)	External Non-deterministic Spec.	
		Desired behaviors	Undesired behaviors
Fixed	Decidable [WTH91,AMPS98]	Undecidable [DM02]	Decidable [DM02]
Non-fixed	Decidable [DM02]	Undecidable [DM02]	Undecidable [DM02]

Partial observability hypothesis			
Resources	Det. Spec. (Internal/External)	External Non deterministic Spec.	
		Desired behaviors	Undesired behaviors
Fixed	?	Undecidable	?
Non-fixed	Undecidable	Undecidable	Undecidable

Synthesis with non-fixed resources

Theorem: Under partial observability, the controller synthesis problem with non-fixed resources is undecidable.

Take $\mathcal{A}$ a timed automaton. Construct $\mathcal{A}_s$ as follows.

$\ell \xrightarrow{g,a,C:=0} \ell'$ is replaced by $\ell \xrightarrow{(\ell,g,a,C:=0,\ell'),\,z:=0} \bullet \xrightarrow{g\wedge z=0,\,a,\,C:=0} \ell'$

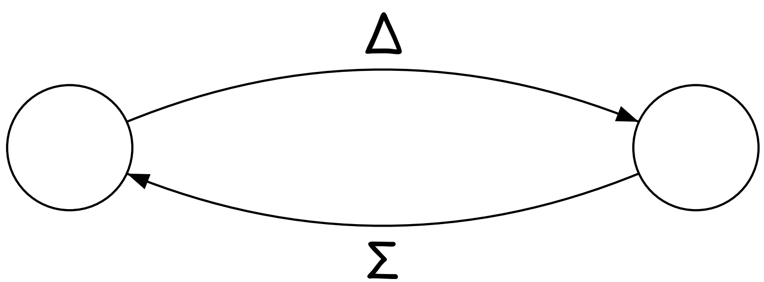
Thus,

- $\mathcal{A}_s$ is a deterministic timed automaton
- $(\delta_0,t_0)(a_0,t'_0)(\delta_1,t_1)(a_1,t'_1)\ldots$ is accepted by $\mathcal{A}_s$ iff $t_i = t'_i$ for every i and $(a_0,t_0)(a_1,t_1)\ldots$ is accepted by $\mathcal{A}$ along the path $\delta_0\delta_1\ldots$

We note $\Delta = \{(\ell,g,a,C:=0,\ell')$ transition of $\mathcal{A}\}$

cont'd

Take the plant $\mathcal{P}$ as



and make all actions from Δ non-observable.

There exists a controller $\mathcal{C}$ such that $L(\mathcal{P} \parallel \mathcal{C}) \cap L(\mathcal{A}_s) = \emptyset$
iff
$L(\mathcal{A})$ is not universal.

Indeed, for any timed word $\gamma = (a_0, t_0)(a_1, t_1)\ldots$,

$\mathcal{P} \parallel \gamma$ represents all the possible runs for γ with transitions in $\mathcal{A}$

Fixing the resources

Full observability hypothesis			
Resources	Det. Spec. (Internal/External)	External Non-deterministic Spec.	
		Desired behaviors	Undesired behaviors
Fixed	Decidable [WTH91,AMPS98]	Undecidable [DM02]	Decidable [DM02]
Non-fixed	Decidable [DM02]	Undecidable [DM02]	Undecidable [DM02]

Partial observability hypothesis			
Resources	Det. Spec. (Internal/External)	External Non deterministic Spec.	
		Desired behaviors	Undesired behaviors
Fixed	Decidable	Undecidable	Decidable
Non-fixed	Undecidable	Undecidable	Undecidable

Fixing the resources

Theorem: Under partial observability, the controller synthesis problem with fixed resources is decidable, for deterministic specifications or non deterministic specifications representing undesired behaviors.

- Controller synthesis problem as a (syntactic) timed game
- Solving a timed game
  - construction of an untimed arena
  - construction of an untimed winning condition
- Apply results on untimed games

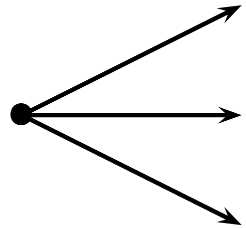
Timed games

On the same symbolic alphabet as the controller:

-

Timed games

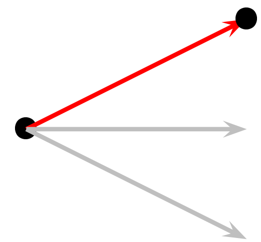
On the same symbolic alphabet as the controller:



Player C

Timed games

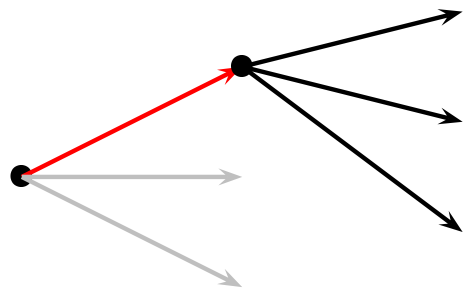
On the same symbolic alphabet as the controller:



Player E

Timed games

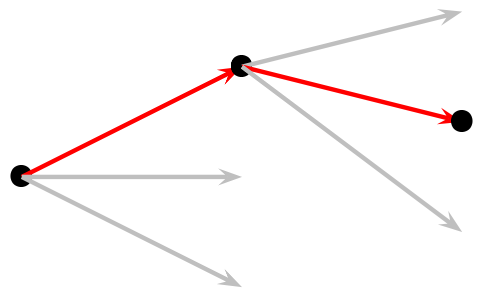
On the same symbolic alphabet as the controller:



Player C

Timed games

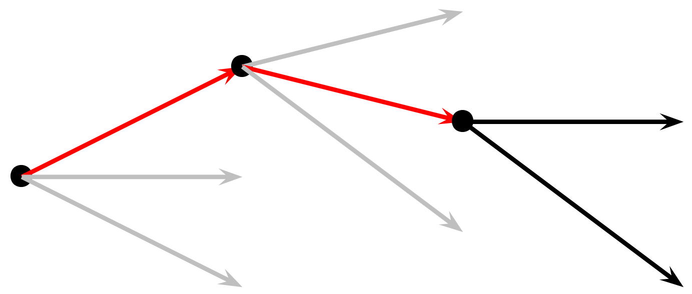
On the same symbolic alphabet as the controller:



Player E

Timed games

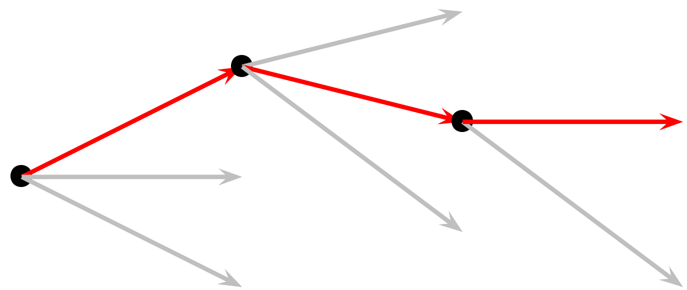
On the same symbolic alphabet as the controller:



Player C

Timed games

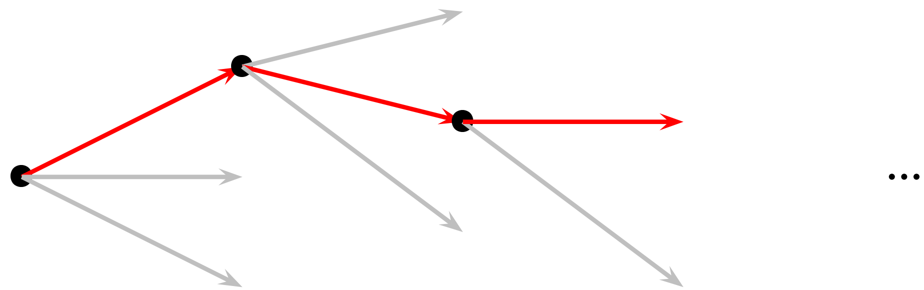
On the same symbolic alphabet as the controller:



Player E

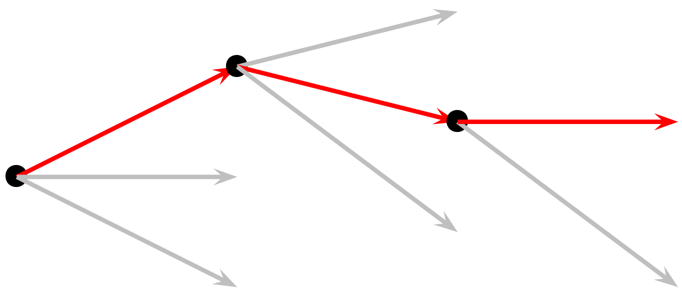
Timed games

On the same symbolic alphabet as the controller:



Timed games

On the same symbolic alphabet as the controller:



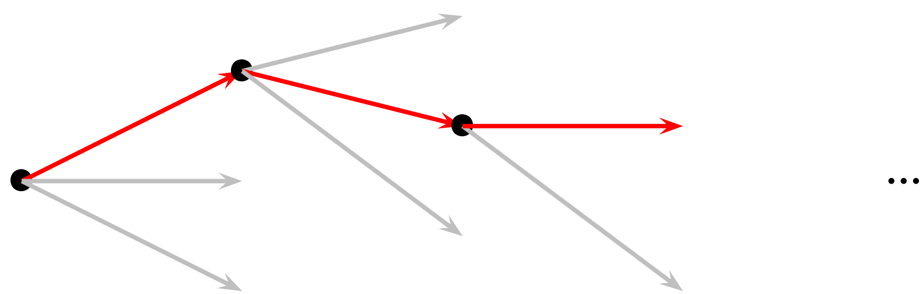
...

- whether the play γ is winning or not depends on the synchronization of the plant with γ:

$$L(\mathcal{P} \parallel \gamma) \cap L(\mathcal{S}) = \emptyset$$

Timed games

On the same symbolic alphabet as the controller:



- whether the play γ is winning or not depends on the synchronization of the plant with γ:

$$L(\mathcal{P} \parallel \gamma) \cap L(\mathcal{S}) = \emptyset$$

- usual notion of strategy, with additional hypotheses for the non-blocking and non-restricting hypotheses

Untimed games

- **Construction of the arena:** projection of the region automaton of the timed arena onto observable actions ; valid sets of actions for player C
- **Construction of the winning condition:** based on the fact that the set

$$\{\gamma \in \Gamma^\infty \mid L(\mathcal{P} \parallel \gamma) \cap L(\mathcal{S}) = \emptyset\}$$

is regular (where Γ is the symbolic alphabet for $\mathcal{C}$).

➜ classical game [Thomas'95]

Untimed games

- **Construction of the arena:** projection of the region automaton of the timed arena onto observable actions ; valid sets of actions for player C
- **Construction of the winning condition:** based on the fact that the set

$$\{\gamma \in \Gamma^\infty \mid L(\mathcal{P} \parallel \gamma) \cap L(\mathcal{S}) = \emptyset\}$$

is regular (where Γ is the symbolic alphabet for $\mathcal{C}$).

➜ classical game [Thomas'95]

Notice: unfortunately, it does not correspond to classical games where
« there is a strategy iff there is a memory-free strategy »

Conclusion and further work

- Formalization of a timed controller synthesis problem
- Which resources are really necessary?
  - can we not only fix the number of clocks?
  - can we not only fix the « sampling » of the controller?
- Algorithms for building controllers? [when decidable]