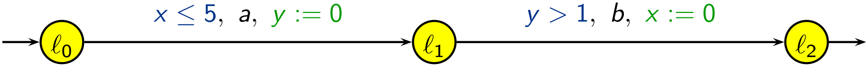
Optimal Strategies in Priced Timed Game Automata

P. Bouyer[①], F. Cassez[②], E. Fleury[③], K.G. Larsen[③]

① LSV – CNRS & ENS de Cachan – France
② IRCCyN – CNRS – France
③ CISS & BRICS – Aalborg University – Denmark

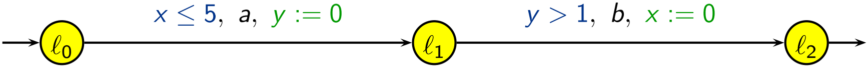
x, y : clocks



Timed Automata

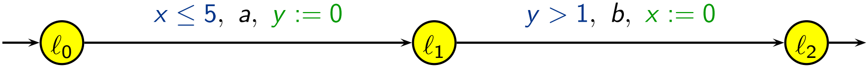
[AD 90's]

x, y : clocks



$$
\begin{array}{cccccccc}
 & \ell_0 & \xrightarrow{\delta(4.1)} & \ell_0 & \xrightarrow{a} & \ell_1 & \xrightarrow{\delta(1.4)} & \ell_1 & \xrightarrow{b} & \ell_2 \\
x & 0 & & 4.1 & & 4.1 & & 5.5 & & 0 \\
y & 0 & & 4.1 & & 0 & & 1.4 & & 1.4
\end{array}
$$

x, y : clocks

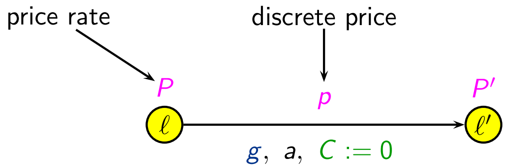


$$\begin{array}{ccccccccc} & \ell_0 & \xrightarrow{\delta(4.1)} & \ell_0 & \xrightarrow{a} & \ell_1 & \xrightarrow{\delta(1.4)} & \ell_1 & \xrightarrow{b} & \ell_2 \\ x & 0 & & 4.1 & & 4.1 & & 5.5 & & 0 \\ y & 0 & & 4.1 & & 0 & & 1.4 & & 1.4 \end{array}$$

(clock) valuation

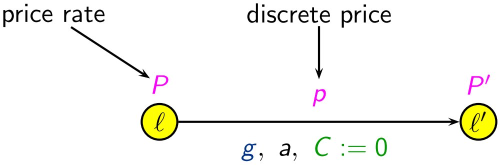
Model of Priced Timed Automata

[HSCC'01]



cost $\equiv$ price

Model of Priced Timed Automata [HSCC'01]



cost $\equiv$ price

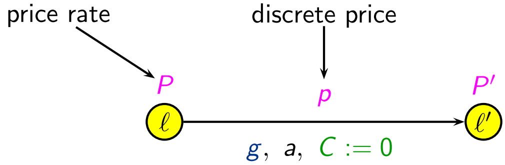
- a configuration: (ℓ, v)
- two kinds of transitions:

$$\left\{\begin{array}{l} (\ell, v) \xrightarrow{\delta(d)} (\ell, v+d) \\ (\ell, v) \xrightarrow{a} (\ell', v') \text{ where } \left\{\begin{array}{l} v \models g \\ v' = [C \leftarrow 0]v \end{array}\right. \text{ for some } \ell \xrightarrow{g, a, C:=} \ell' \end{array}\right.$$

Model of Priced Timed Automata

[HSCC'01]

cost $\equiv$ price



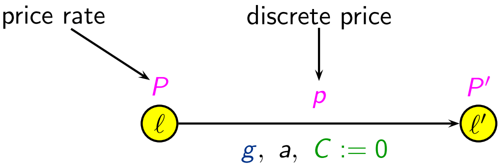
- a configuration: (ℓ, v)
- two kinds of transitions:

$$\begin{cases} (\ell, v) \xrightarrow{\delta(d)} (\ell, v+d) \\ (\ell, v) \xrightarrow{a} (\ell', v') \text{ where } \begin{cases} v \models g \\ v' = [C \leftarrow 0]v \end{cases} \text{ for some } \ell \xrightarrow{g,a,C:=} \ell' \end{cases}$$

$$\mathrm{Cost}\left((\ell, v) \xrightarrow{\delta(d)} (\ell, v+d)\right) = P.d \qquad \mathrm{Cost}\left((\ell, v) \xrightarrow{a} (\ell', v')\right) = p$$

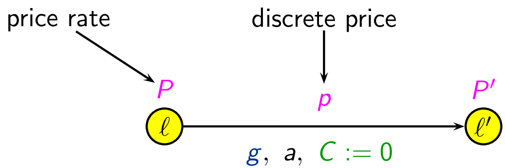
$$\mathrm{Cost}(\rho) = \text{accumulated cost along run } \rho$$

Model of Priced Timed Automata (cont.)



cost $\equiv$ price

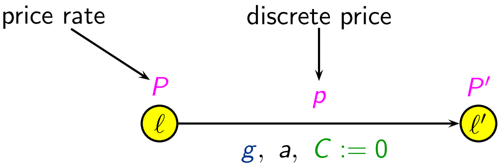
Model of Priced Timed Automata (cont.)



cost $\equiv$ price

- **one player problems:**
  - reachability with an optimization criterium on the price **[BFH+01a,BFH+01b,LBB+01,ALTP01]**
  - safety with a mean-cost optimization criterium **[BBL04]**

Model of Priced Timed Automata (cont.)

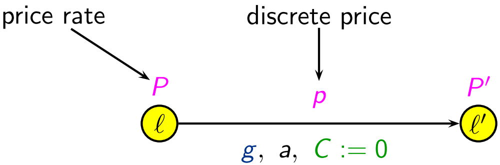


cost $\equiv$ price

- **one player problems:**
  - reachability with an optimization criterium on the price **[BFH+01a,BFH+01b,LBB+01,ALTP01]**
  - safety with a mean-cost optimization criterium **[BBL04]**

- **what if an opponent**?

Model of Priced Timed Automata (cont.)



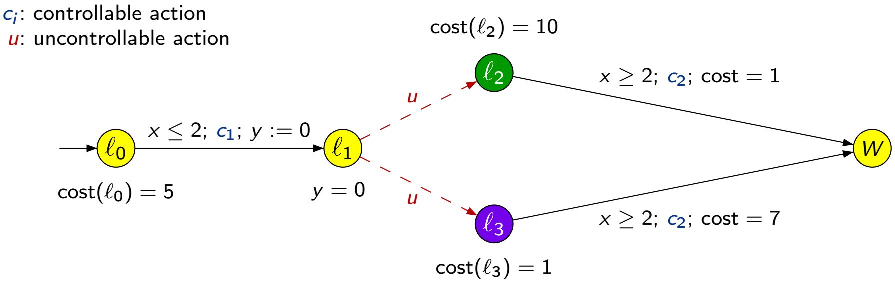
cost $\equiv$ price

- **one player problems:**
  - reachability with an optimization criterium on the price **[BFH+01a,BFH+01b,LBB+01,ALTP01]**
  - safety with a mean-cost optimization criterium **[BBL04]**

- **what if an opponent**?

  ➜ **optimal reachability timed game**

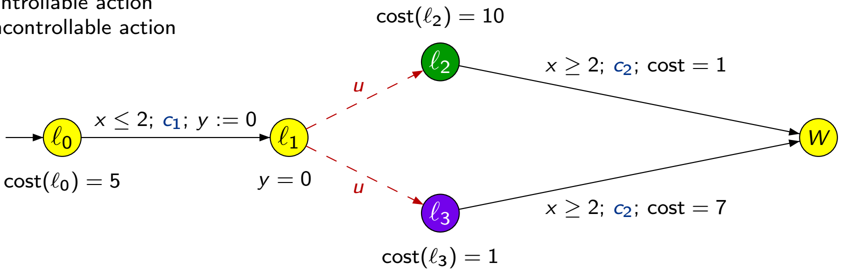
An Example



Question: what is the optimal price we can ensure in state ℓ_0?

An Example

c_i: controllable action
u: uncontrollable action

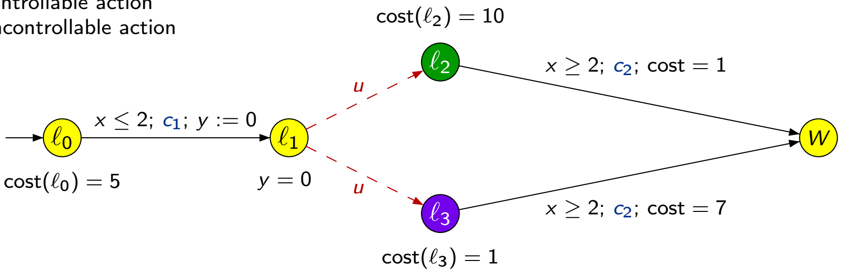


Question: what is the optimal price we can ensure in state ℓ_0?

$$5t + 10(2 - t) + 1$$

An Example

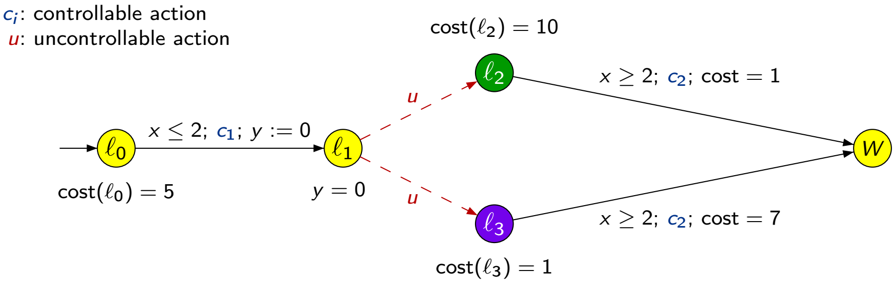
c_i: controllable action
u: uncontrollable action



Question: what is the optimal price we can ensure in state ℓ_0?

$$5t + 10(2 - t) + 1 \ , \ 5t + (2 - t) + 7$$

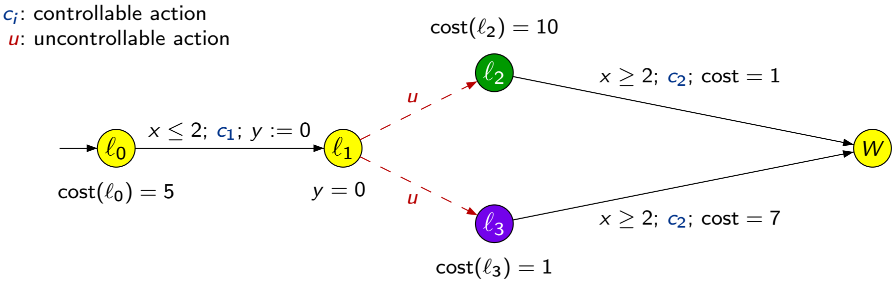
An Example



Question: what is the optimal price we can ensure in state ℓ_0?

$$\max\ (\ 5t + 10(2-t) + 1\ ,\ 5t + (2-t) + 7\ )$$

An Example

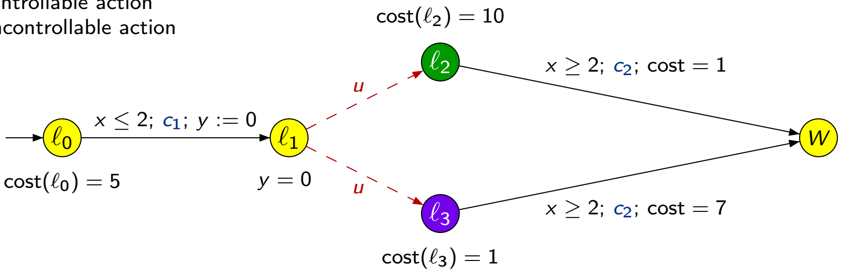


Question: what is the optimal price we can ensure in state ℓ_0?

$$\inf_{0 \leq t \leq 2} \max\left(5t + 10(2-t) + 1 \,,\; 5t + (2-t) + 7 \right) = 14 + \frac{1}{3}$$

An Example

c_i: controllable action
u: uncontrollable action

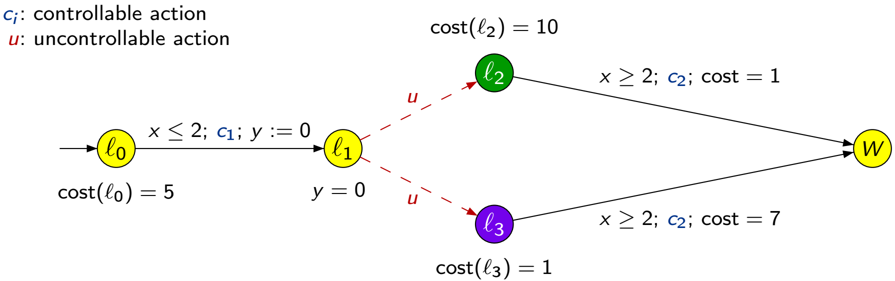


Question: what is the optimal price we can ensure in state ℓ_0?

$$\inf_{0 \leq t \leq 2} \max\left(5t + 10(2-t) + 1 \,,\; 5t + (2-t) + 7 \right) = 14 + \frac{1}{3}$$

➜ **strategy:** wait in ℓ_0, and when $t = \frac{4}{3}$, go to ℓ_1

An Example



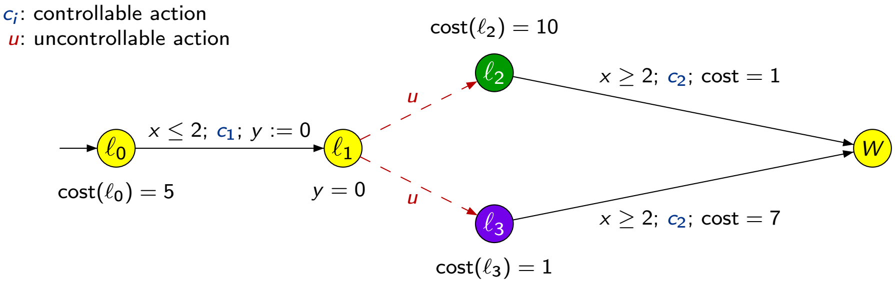
Question: what is the optimal price we can ensure in state ℓ_0?

$$\inf_{0 \leq t \leq 2} \max \left(5t + 10(2-t) + 1 \,,\, 5t + (2-t) + 7 \right) = 14 + \frac{1}{3}$$

➜ **strategy:** wait in ℓ_0, and when $t = \frac{4}{3}$, go to ℓ_1

- How to automatically compute such optimal prices?

An Example



Question: what is the optimal price we can ensure in state ℓ_0?

$$\inf_{0 \leq t \leq 2} \max \left(5t + 10(2-t) + 1 \,,\, 5t + (2-t) + 7 \right) = 14 + \frac{1}{3}$$

➜ **strategy:** wait in ℓ_0, and when $t = \frac{4}{3}$, go to ℓ_1

- How to automatically compute such optimal prices?
- How to synthesize optimal strategies (if one exists)?

Our Formalization

➜ **a run-based definition of the problem**

Our Formalization

➜ a run-based definition of the problem

- a strategy is a partial function from finite runs to $\{c \mid c \; cont.\} \cup \{\lambda\}$

Example

$$f(\rho) = \begin{cases} \lambda & \text{if } \rho \text{ ends in } (\ell_0, x < \frac{4}{3}) \text{ or in } (\ell_2, x < 2) \text{ or in } (\ell_3, x < 2) \\ c_1 & \text{if } \rho \text{ ends in } (\ell_0, x \geq \frac{4}{3}) \\ c_2 & \text{if } \rho \text{ ends in } (\ell_2, x \geq 2) \text{ or in } (\ell_3, x \geq 2) \end{cases}$$

Our Formalization

➜ a run-based definition of the problem

- a strategy is a partial function from finite runs to $\{c \mid c \ \mathit{cont.}\} \cup \{\lambda\}$

Example

$$f(\rho) = \begin{cases} \lambda & \text{if } \rho \text{ ends in } (\ell_0, x < \frac{4}{3}) \text{ or in } (\ell_2, x < 2) \text{ or in } (\ell_3, x < 2) \\ c_1 & \text{if } \rho \text{ ends in } (\ell_0, x \geq \frac{4}{3}) \\ c_2 & \text{if } \rho \text{ ends in } (\ell_2, x \geq 2) \text{ or in } (\ell_3, x \geq 2) \end{cases}$$

- $\mathrm{Outcome}(f, (\ell, v))$: runs generated by strategy f from (ℓ, v)

Our Formalization

➜ a run-based definition of the problem

- a strategy is a partial function from finite runs to $\{c \mid c \ \mathit{cont.}\} \cup \{\lambda\}$

Example

$$f(\rho) = \begin{cases} \lambda & \text{if } \rho \text{ ends in } (\ell_0, x < \frac{4}{3}) \text{ or in } (\ell_2, x < 2) \text{ or in } (\ell_3, x < 2) \\ c_1 & \text{if } \rho \text{ ends in } (\ell_0, x \geq \frac{4}{3}) \\ c_2 & \text{if } \rho \text{ ends in } (\ell_2, x \geq 2) \text{ or in } (\ell_3, x \geq 2) \end{cases}$$

- $\mathrm{Outcome}(f, (\ell, v))$: runs generated by strategy f from (ℓ, v)
- f is a winning strategy from (ℓ, v) if all *maximal* runs in $\mathrm{Outcome}(f, (\ell, v))$ contain a winning state

Our Formalization

→ a run-based definition of the problem

- a strategy is a partial function from finite runs to $\{c \mid c\ \mathit{cont.}\} \cup \{\lambda\}$

Example

$$f(\rho) = \begin{cases} \lambda & \text{if } \rho \text{ ends in } (\ell_0, x < \frac{4}{3}) \text{ or in } (\ell_2, x < 2) \text{ or in } (\ell_3, x < 2) \\ c_1 & \text{if } \rho \text{ ends in } (\ell_0, x \geq \frac{4}{3}) \\ c_2 & \text{if } \rho \text{ ends in } (\ell_2, x \geq 2) \text{ or in } (\ell_3, x \geq 2) \end{cases}$$

- $\mathrm{Outcome}(f, (\ell, v))$: runs generated by strategy f from (ℓ, v)
- f is a winning strategy from (ℓ, v) if all *maximal* runs in $\mathrm{Outcome}(f, (\ell, v))$ contain a winning state
- the cost of a strategy f from (ℓ, v) is:

$$\mathrm{Cost}(f, (\ell, v)) = \sup \{\mathit{cost}(\rho) \mid \rho \in \mathrm{Outcome}(f, (\ell, v))\}$$

Optimal Control Problems

- **Optimal cost computation:** compute the optimal cost

$$\boldsymbol{optcost}(\ell, v) = \inf \ \{\mathrm{Cost}(f, (\ell, v)) \mid f \text{ winning strategy from } (\ell, v)\}$$

Optimal Control Problems

- **Optimal cost computation:** compute the optimal cost

$$\boxed{\textbf{\textit{optcost}}(\ell, v) = \inf\ \{\mathrm{Cost}(f,(\ell,v)) \mid f \text{ winning strategy from } (\ell,v)\}}$$

- **Optimal strategy existence:** does there exist a winning strategy f s.t.

$$\boxed{\mathrm{Cost}(f,(\ell,v)) = \textbf{\textit{optcost}}(\ell,v)}$$

Optimal Control Problems

- **Optimal cost computation:** compute the optimal cost

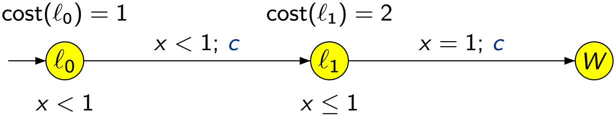
$$\boxed{\textbf{\textit{optcost}}(\ell, v) = \inf \; \{\mathrm{Cost}(f, (\ell, v)) \mid f \text{ winning strategy from } (\ell, v)\}}$$

- **Optimal strategy existence:** does there exist a winning strategy f s.t.

$$\boxed{\mathrm{Cost}(f, (\ell, v)) = \textbf{\textit{optcost}}(\ell, v)}$$

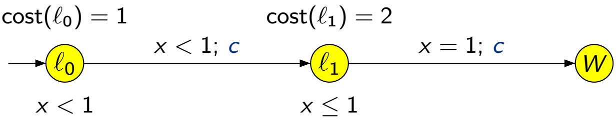
- **Optimal strategy synthesis:** in case an optimal winning strategy exists, construct one such

Do Optimal Strategies Always Exist?



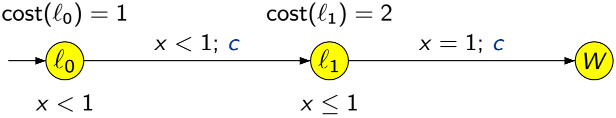
$$\begin{cases} f(\ell_0, x < 1) = \lambda \\ \\ f(\ell_1, x < 1) = \lambda \\ f(\ell_1, x = 1) = c \end{cases}$$

Do Optimal Strategies Always Exist?



$$\begin{cases} f(\ell_0, x < 1) = \lambda \\ \\ f(\ell_1, x < 1) = \lambda \\ f(\ell_1, x = 1) = c \end{cases} \quad \rightsquigarrow \quad \begin{cases} f_\varepsilon(\ell_0, x < 1 - \varepsilon) = \lambda \\ f_\varepsilon(\ell_0, 1 - \varepsilon \leq x < 1) = c \\ f_\varepsilon(\ell_1, x < 1) = \lambda \\ f_\varepsilon(\ell_1, x = 1) = c \end{cases}$$

Do Optimal Strategies Always Exist?



$$\begin{cases} f(\ell_0, x<1) = \lambda \\ \\ f(\ell_1, x<1) = \lambda \\ f(\ell_1, x=1) = c \end{cases} \quad \rightsquigarrow \quad \begin{cases} f_\varepsilon(\ell_0, x<1-\varepsilon) = \lambda \\ f_\varepsilon(\ell_0, 1-\varepsilon \leq x < 1) = c \\ f_\varepsilon(\ell_1, x<1) = \lambda \\ f_\varepsilon(\ell_1, x=1) = c \end{cases}$$

➜ **no optimal strategy exists**, but rather a family $(f_\varepsilon)_{\varepsilon>0}$ of ε-approximating strategies ($cost(f_\varepsilon) = 1 + \varepsilon$)

Relation with Recursive Definition

➜ **a recursive definition**

(close to that of **[LTMM02,ABM04]**)

$$O(s) = \inf_{\substack{s \xrightarrow{t,p} s' \\ t \geq 0}} \max \begin{cases} \min \left(\min_{\substack{s' \xrightarrow{c,p'} s'' \\ c \text{ cont.}}} p + p' + O(s'') \ , \ p + O(s') \right) \\ \sup_{\substack{s \xrightarrow{t',p'} s'' \\ t' \leq t}} \ \max_{\substack{s'' \xrightarrow{u,p''} s''' \\ u \text{ uncont.}}} p' + p'' + O(s''') \end{cases}$$

Relation with Recursive Definition

➜ a recursive definition
(close to that of **[LTMM02,ABM04]**)

$$O(s) = \inf_{s \xrightarrow[t \geq 0]{t,p} s'} \max \begin{cases} \min \left(\min_{\substack{s' \xrightarrow{c,p'} s'' \\ c \text{ cont.}}} p + p' + O(s'') \; , \; p + O(s') \right) \\ \sup_{s \xrightarrow[t' \leq t]{t',p'} s''} \max_{\substack{s'' \xrightarrow{u,p''} s''' \\ u \text{ uncont.}}} p' + p'' + O(s''') \end{cases}$$

Theorem

For linear hybrid games, $O(s) = \boldsymbol{optcost}(s)$.

Timed/Hybrid Games, Strategy Extraction

$$\pi(X) = \mathsf{Pred}_t(X \cup \mathsf{cPred}(X), \mathsf{uPred}(\overline{X}))$$

➜ controllable predecessors

The winning states are those computed by the least fix-point of $\lambda X.W \cup \pi(X)$.

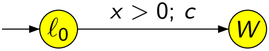
Timed/Hybrid Games, Strategy Extraction

$$\pi(X) = \mathrm{Pred}_t(X \cup \mathrm{cPred}(X), \mathrm{uPred}(\overline{X}))$$

➜ controllable predecessors

The winning states are those computed by the least fix-point of $\lambda X . W \cup \pi(X)$.

However, it does not give us directly winning strategies...



$$\begin{cases} f(\ell_0, x = 0) = \lambda \\ f(\ell_0, x > 0) = c \end{cases}$$

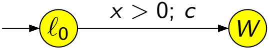
Timed/Hybrid Games, Strategy Extraction

$$\pi(X) = \mathrm{Pred}_t(X \cup \mathrm{cPred}(X), \mathrm{uPred}(\overline{X}))$$

➜ controllable predecessors

The winning states are those computed by the least fix-point of $\lambda X.W \cup \pi(X)$.

However, it does not give us directly winning strategies...



$$\begin{cases} f(\ell_0, x = 0) = \lambda \\ f(\ell_0, x > 0) = c \end{cases} \rightsquigarrow \begin{cases} f_\varepsilon(\ell_0, x < \varepsilon) = \lambda \\ f_\varepsilon(\ell_0, x \geq \varepsilon) = c \end{cases}$$

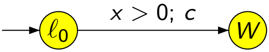
Timed/Hybrid Games, Strategy Extraction

$$\pi(X) = \mathrm{Pred}_t(X \cup \mathrm{cPred}(X), \mathrm{uPred}(\overline{X}))$$

➜ controllable predecessors

The winning states are those computed by the least fix-point of $\lambda X.W \cup \pi(X)$.

However, it does not give us directly winning strategies...



$$\begin{cases} f(\ell_0, x = 0) = \lambda \\ f(\ell_0, x > 0) = c \end{cases} \rightsquigarrow \begin{cases} f_\varepsilon(\ell_0, x < \varepsilon) = \lambda \\ f_\varepsilon(\ell_0, x \geq \varepsilon) = c \end{cases}$$

➜ a "**realizability**" problem

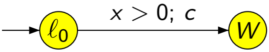
Timed/Hybrid Games, Strategy Extraction

$$\pi(X) = \mathsf{Pred}_t(X \cup \mathsf{cPred}(X), \mathsf{uPred}(\overline{X}))$$

➜ controllable predecessors

The winning states are those computed by the least fix-point of $\lambda X. W \cup \pi(X)$.

However, it does not give us directly winning strategies...



$$\begin{cases} f(\ell_0, x = 0) = \lambda \\ f(\ell_0, x > 0) = c \end{cases} \rightsquigarrow \begin{cases} f_\varepsilon(\ell_0, x < \varepsilon) = \lambda \\ f_\varepsilon(\ell_0, x \geq \varepsilon) = c \end{cases}$$

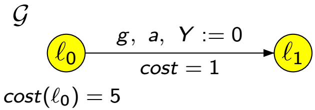
➜ a "**realizability**" problem

➜ constructive proof of **realizable** and **state-based** strategies

(see research report)

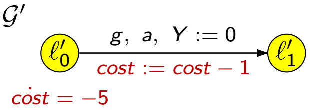
Our Solution

Idea: tranform the cost into a decreasing linear hybrid variable



Winning: W

$\Longrightarrow$



Winning: $W \wedge cost \geq 0$

Our Solution

Idea: tranform the cost into a decreasing linear hybrid variable

$\mathcal{G}$

$$\ell_0 \xrightarrow[\quad cost\,=\,1\quad]{g,\ a,\ Y:=0} \ell_1$$

$cost(\ell_0) = 5$

Winning: W

$\Longrightarrow$

$\mathcal{G}'$

$$\ell'_0 \xrightarrow[\quad cost\,:=\,cost\,-\,1\quad]{g,\ a,\ Y:=0} \ell'_1$$

$\dot{cost} = -5$

Winning: $W \wedge cost \geq 0$

Theorem

For priced timed games (under some hypotheses),

$$\left.\begin{array}{r} \exists f \text{ winning strategy in } \mathcal{G} \\ \text{s.t. } cost(f,(\ell,v)) \leq \gamma \end{array}\right\} \iff (\ell, v, cost = \gamma) \text{ winning in } \mathcal{G}'$$

+ constructive proof

Our Solution (2)

The set of winning states in $\mathcal{G}'$ is upward-closed for the cost, *i.e.* of the form

$$\bigcup_{i \in I} (P_i \wedge cost \succ_i k_i) \qquad \text{(with } \succ_i \text{ either } > \text{ or } \geq\text{)}$$

Our Solution (2)

The set of winning states in $\mathcal{G}'$ is upward-closed for the cost, *i.e.* of the form

$$\bigcup_{i \in I} (P_i \wedge \mathit{cost} \succ_i k_i) \qquad \text{(with } \succ_i \text{ either } > \text{ or } \geq\text{)}$$

Corollary

For priced timed games (under some hypotheses),

- *"reachable" optimal cost, or not (cost $\geq \gamma$ or cost $> \gamma$)*
- *existence of an optimal strategy decidable*

+ constructive proof

Our Solution (2)

The set of winning states in $\mathcal{G}'$ is upward-closed for the cost, *i.e.* of the form

$$\bigcup_{i \in I} (P_i \wedge \mathit{cost} \succ_i k_i) \qquad \text{(with } \succ_i \text{ either } > \text{ or } \geq\text{)}$$

Corollary

For priced timed games (under some hypotheses),

- *"reachable" optimal cost, or not (cost $\geq \gamma$ or cost $> \gamma$)*
- *existence of an optimal strategy decidable*

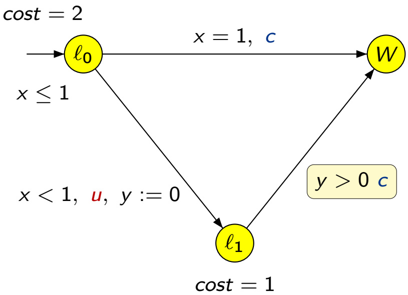
+ constructive proof

Nature of the strategy:

- state-based for the hybrid game, thus **cost-dependent** for the optimal timed game
- cost-dependence is unavoidable in general!
- cost-independent strategies for syntactical restrictions of the games

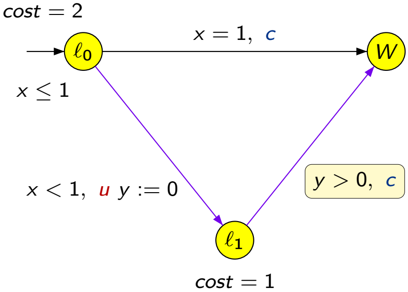
  c: large constraints, u: strict constraints

Cost-Dependence is Unavoidable



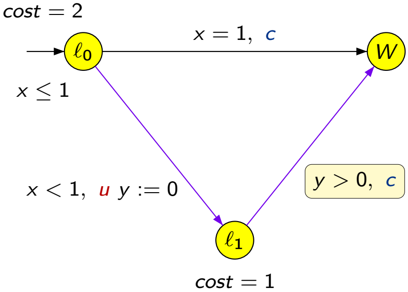
- **optimal cost:** 2
- **optimal strategy:**

Cost-Dependence is Unavoidable



- **optimal cost:** 2
- **optimal strategy:** d time before a u occurs, d' time waited in ℓ_1
  The cost of the run is $2.d + d'$.

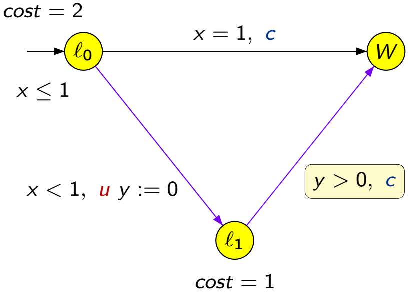
Cost-Dependence is Unavoidable



- **optimal cost:** 2
- **optimal strategy:** d time before a u occurs, d' time waited in ℓ_1
  The cost of the run is $2.d + d'$.

$$2.d + d' \leq 2$$

Cost-Dependence is Unavoidable



- **optimal cost:** 2
- **optimal strategy:** d time before a u occurs, d' time waited in ℓ_1
  The cost of the run is $2.d + d'$.

$$2.d + d' \leq 2$$

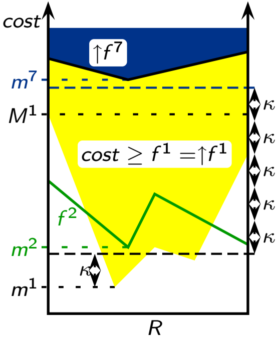
$$(\text{accumulated cost}) + d' \leq 2$$

Hypotheses for Termination

- all clocks are bounded (not restrictive)
- the cost function is *strictly non-zeno*
  - → This condition is restrictive, but is decidable

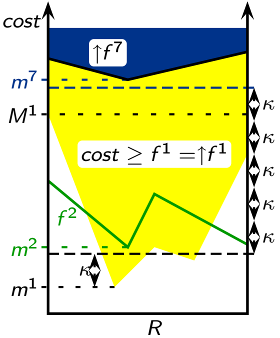
Hypotheses for Termination

- all clocks are bounded (not restrictive)
- the cost function is *strictly non-zeno*
  - → This condition is restrictive, but is decidable



Hypotheses for Termination

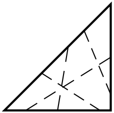
- all clocks are bounded (not restrictive)
- the cost function is *strictly non-zeno*
  - → This condition is restrictive, but is decidable



Open problem

Is this strict non-zenoness hypothesis really necessary?

Related Work



[LTMM02] and **[ABM04]**

- a recursive definition of the cost function
- a clever (exponential) bound on the number of splits needed

Our work:

- a simple and natural run-based definition of cost optimality
- decidability of the existence of an optimal strategy (under hypotheses)
- structural properties of strategies

Conclusion

Summary:

- run-based definition of cost optimality for priced timed games
- reduction to hybrid games
  - computability of optimal cost, and decidability of the existence of optimal strategies under a non-zenoness hypothesis
  - nature of optimal winning strategies
- algorithm easy to implement using HyTech
- work complementary to that of **[ABM04]**

Conclusion

Summary:

- run-based definition of cost optimality for priced timed games
- reduction to hybrid games
  - computability of optimal cost, and decidability of the existence of optimal strategies under a non-zenoness hypothesis
  - nature of optimal winning strategies
- algorithm easy to implement using HyTech
- work complementary to that of **[ABM04]**

Several remaining open questions:

- termination without the hypothesis of a strongly non-zeno cost
- when no optimal strategy exists, synthesize a family of approximating strategies

Conclusion

Summary:

- run-based definition of cost optimality for priced timed games
- reduction to hybrid games
  - computability of optimal cost, and decidability of the existence of optimal strategies under a non-zenoness hypothesis
  - nature of optimal winning strategies
- algorithm easy to implement using HyTech
- work complementary to that of **[ABM04]**

Several remaining open questions:

- termination without the hypothesis of a strongly non-zeno cost
- when no optimal strategy exists, synthesize a family of approximating strategies

Further direction:

- extension to safety games with a mean-cost optimality criterium