Stochastic Strategies in Quantitative and Timed Games

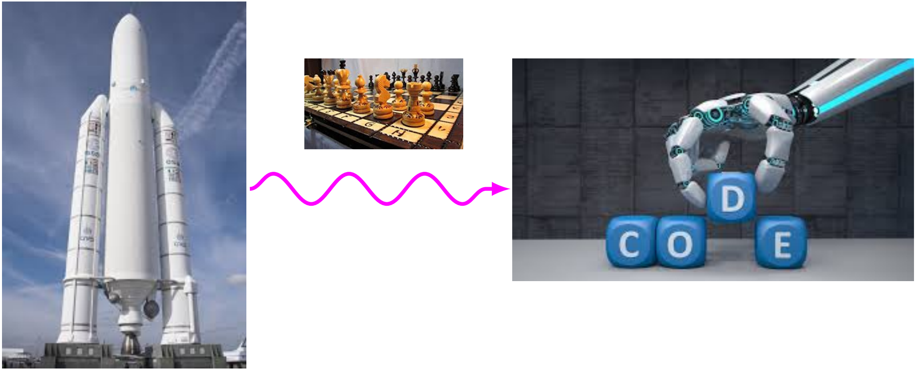
Julie PARREAUX[1]
supervised by
Benjamin MONMEGE[2], Pierre–Alain REYNIER[2]

[1]ENS Rennes

[2]LIS, Team Modelisation and Verification

October 30, 2020

Motivation : game theory for synthesis



- Check the correctness of a system
- Interaction between two antagonistic agents : environment and controller
- Correct by construction : synthesis of controller

Different sorts of games

Qualitative games

Reach or avoid some (sequences of) states

Quantitative games

- Consider quantitative parameters : energy consumption...
- Compare distinct strategies

Different sorts of games

Qualitative games

Reach or avoid some (sequences of) states

Quantitative games

- Consider quantitative parameters : energy consumption...
- Compare distinct strategies

Shortest-Path games

- Combination of a qualitative with a quantitative objective
- Reach a target with a minimum cost

Different sorts of games

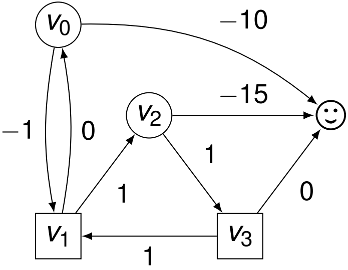
Timed games

- Consider timed issues : receive a message...
- Infinite games

Weighted Timed games

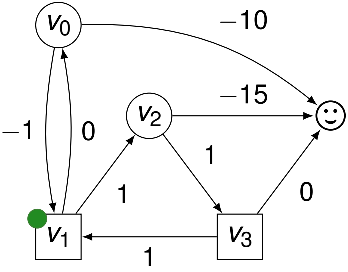
Combination of timed games and shortest path games.

Shortest Path Game



Shortest Path Game

□ Adam ○ Eve

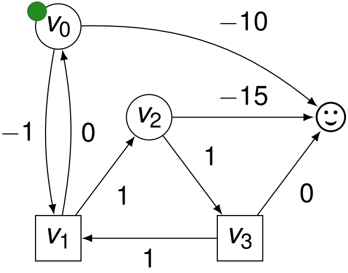


How to play?

Move a token along edge

$$\pi = v_1$$

Shortest Path Game

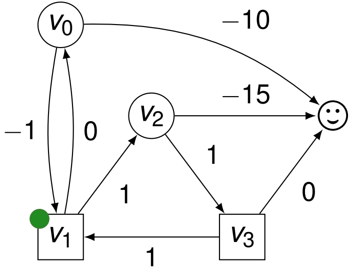


How to play?

Move a token along edge

$$\pi = v_1 v_0$$

Shortest Path Game

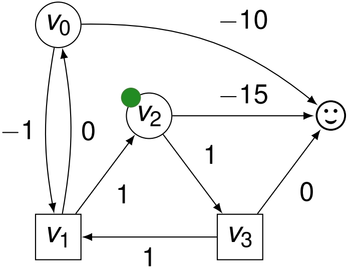


How to play?

Move a token along edge

$$\pi = v_1 v_0 v_1$$

Shortest Path Game

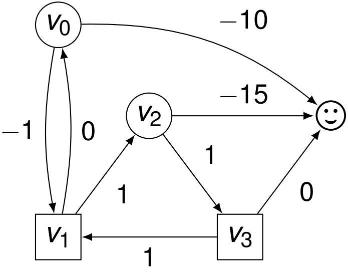


How to play?

Move a token along edge

$$\pi = v_1 v_0 v_1 v_2$$

Shortest Path Game



Play

An infinite path or reach the target

$$\pi = (v_i)_i \in V^\omega \qquad \pi = (v_i)_i ☺$$

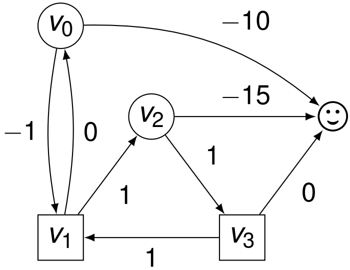
How to play?

Move a token along edge

$$\pi = v_1 v_0 v_1 v_2 v_3 ☺$$

Shortest Path Game

□ Adam ○ Eve



Play

An infinite path or reach the target

$$\pi = (v_i)_i \in V^\omega \qquad \pi = (v_i)_i ☺$$

How to play?

Move a token along edge

$$\pi = v_1 v_0 v_1 v_2 v_3 ☺$$

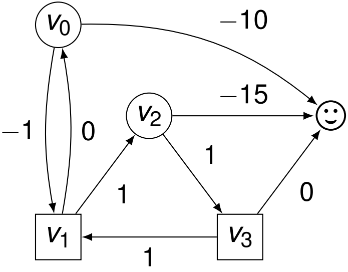
Shortest Path payoff

Let π be a run.

$$\mathbf{SP}(\pi) = \begin{cases} \sum_{i=0}^{n-1} w((\pi_i, \pi_{i+1})) & \text{if } n \text{ is the smallest index s.t. } \pi_n \in T \\ +\infty & \text{if } \pi \text{ does not reach } T \end{cases}$$

Shortest Path Game

□ Adam ○ Eve



Play

An infinite path or reach the target

$$\pi = (v_i)_i \in V^\omega \qquad \pi = (v_i)_i ☺$$

How to play?

Move a token along edge

$$\pi = v_1 v_0 v_1 v_2 v_3 ☺$$

$$\mathbf{SP}(\pi) = 0 + (-1) + 1 + 1 + 0 = 1$$

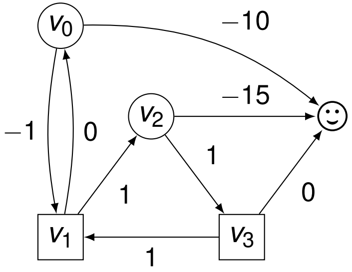
Shortest Path payoff

Let π be a run.

$$\mathbf{SP}(\pi) = \begin{cases} \sum_{i=0}^{n-1} w((\pi_i, \pi_{i+1})) & \text{if } n \text{ is the smallest index s.t. } \pi_n \in T \\ +\infty & \text{if } \pi \text{ does not reach } T \end{cases}$$

Shortest Path Game

□ Adam ○ Eve



Objectives

Eve maximise the payoff

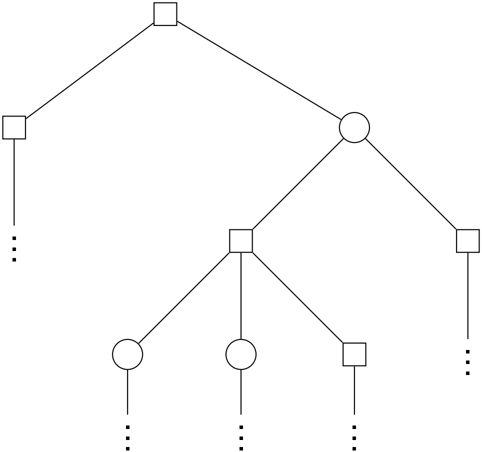
Adam minimise the payoff

Shortest Path payoff

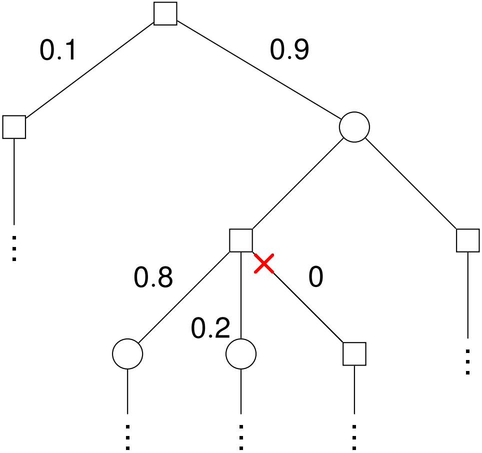
Let π be a run.

$$\mathbf{SP}(\pi) = \begin{cases} \sum_{i=0}^{n-1} w((\pi_i, \pi_{i+1})) & \text{if } n \text{ is the smallest index s.t. } \pi_n \in T \\ +\infty & \text{if } \pi \text{ does not reach } T \end{cases}$$

Strategies for Adam



Strategies for Adam



A strategy [1]

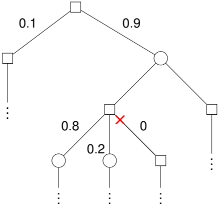
$\sigma : V^* V_{Adam} \to \Delta(V)$.

[1] *Trading Memory for Randomness*, K. Chatterjee, L. Alfaro and T. Henzinger, 2004, QEST

Strategies for Adam

Infinite memory

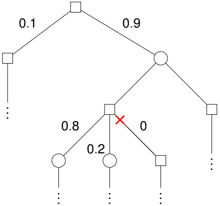
$\sigma : V^* V_{Adam} \to \Delta(V)$



Strategies for Adam

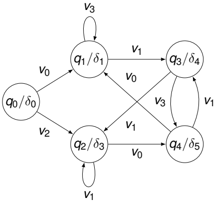
Infinite memory

$\sigma : V^* V_{Adam} \to \Delta(V)$



Finite memory

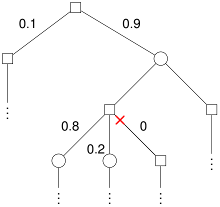
Moore machine



Strategies for Adam

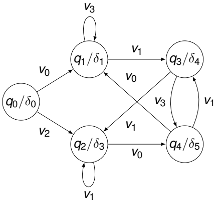
Infinite memory

$\sigma : V^* V_{Adam} \to \Delta(V)$



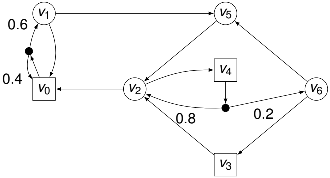
Finite memory

Moore machine



Memoryless

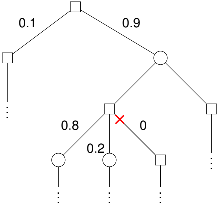
$\sigma : V_{Adam} \to \Delta(V)$



Strategies for Adam

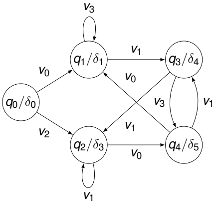
Infinite memory

$\sigma : V^* V_{Adam} \to \Delta(V)$



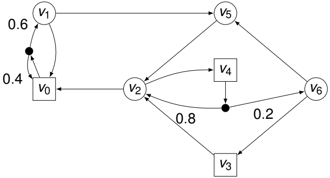
Finite memory

Moore machine



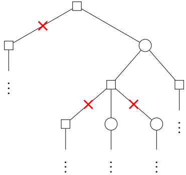
Memoryless

$\sigma : V_{Adam} \to \Delta(V)$



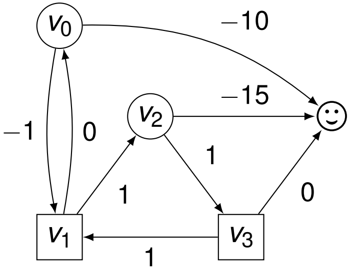
Deterministic

$\sigma : V^* V_{Adam} \to V$



Deterministic Strategies [1]

$\boxed{\sigma}$ Adam $\bigcirc\!\!\!\!\tau$ Eve



Value

$$\overline{\text{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\text{Play}(v, \sigma, \tau))}_{\text{dVal}^{\sigma}(v)}$$

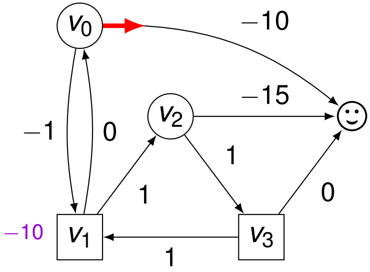
Deterministic strategy

$$\sigma : V^* \times V_{\text{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

σ Adam τ Eve



Estimate $\overline{\text{dVal}}(v_1)$

Eve chooses ☺ in v_0 : $\rightsquigarrow -10$

Value

$$\overline{\text{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \textbf{SP}(\text{Play}(v, \sigma, \tau))}_{\text{dVal}^{\sigma}(v)}$$

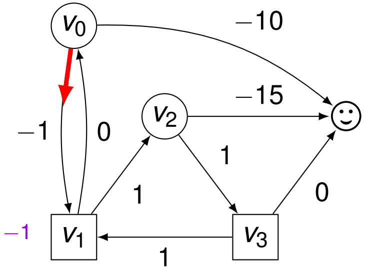
Deterministic strategy

$$\sigma : V^* \times V_{\text{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

$\boxed{\sigma}$ Adam $\quad$ (τ) Eve



Estimate $\overline{\mathsf{dVal}}(v_1)$

Eve chooses ☺ in v_0 : $\rightsquigarrow -10$

Eve chooses v_1 in v_0 : $\dashrightarrow -1$

Value

$$\overline{\mathsf{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\mathsf{Play}(v, \sigma, \tau))}_{\mathsf{dVal}^{\sigma}(v)}$$

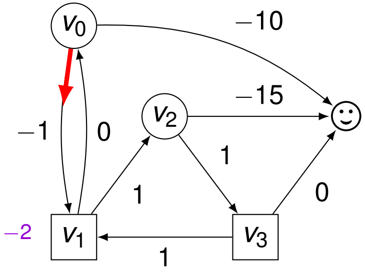
Deterministic strategy

$$\sigma : V^* \times V_{\mathsf{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

σ Adam τ Eve



Estimate $\overline{\mathsf{dVal}}(v_1)$

Eve chooses ☺ in v_0 : $\rightsquigarrow -10$

Eve chooses v_1 in v_0 : $\dashrightarrow -2$

Value

$$\overline{\mathsf{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\mathsf{Play}(v,\sigma,\tau))}_{\mathsf{dVal}^{\sigma}(v)}$$

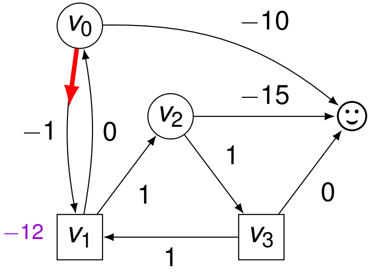
Deterministic strategy

$$\sigma : V^* \times V_{\mathsf{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

$\boxed{\sigma}$ Adam $\quad \textcircled{\tau}$ Eve



Estimate $\overline{\mathsf{dVal}}(v_1)$

Eve chooses ☺ in v_0 : $\rightsquigarrow -10$

Eve chooses v_1 in v_0 : $\dashrightarrow -12$

Value

$$\overline{\mathsf{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\mathsf{Play}(v, \sigma, \tau))}_{\mathsf{dVal}^{\sigma}(v)}$$

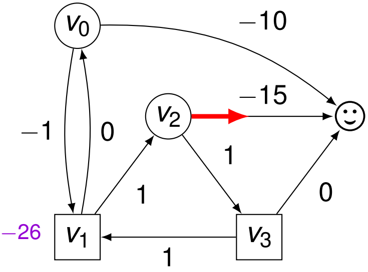
Deterministic strategy

$$\sigma : V^* \times V_{\mathsf{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

σ Adam τ Eve



Estimate $\overline{\mathrm{dVal}}(v_1)$

Eve chooses ☺ in v_0 : $\rightsquigarrow -10$

Eve chooses v_1 in v_0 : $\dashrightarrow -12$

Eve chooses ☺ in v_2 : $\rightsquigarrow -26$

Value

$$\overline{\mathrm{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\mathrm{Play}(v, \sigma, \tau))}_{\mathrm{dVal}^{\sigma}(v)}$$

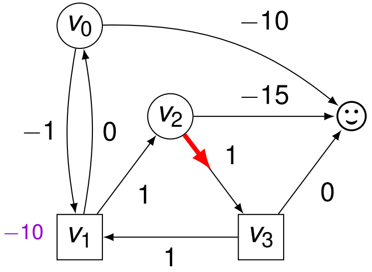
Deterministic strategy

$$\sigma : V^* \times V_{\mathrm{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

$\boxed{\sigma}$ Adam ⓣ Eve



Estimate $\overline{\text{dVal}}(v_1)$

Eve chooses ☺ in v_0 : $\rightsquigarrow -10$

Eve chooses v_1 in v_0 : $\dashrightarrow -12$

Eve chooses ☺ in v_2 : $\rightsquigarrow -26$

Eve chooses v_3 in v_2 : $\rightsquigarrow -10$

Value

$$\overline{\text{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \textbf{SP}(\text{Play}(v, \sigma, \tau))}_{\text{dVal}^{\sigma}(v)}$$

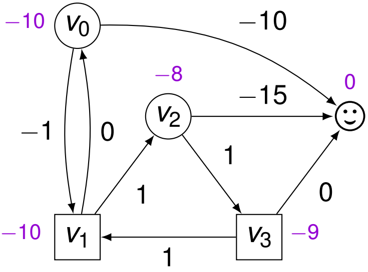
Deterministic strategy

$$\sigma : V^* \times V_{\text{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

σ Adam τ Eve



Estimate $\overline{\text{dVal}}(v_1)$

Eve chooses ☺ in v_0 : $\rightsquigarrow$ -10

Eve chooses v_1 in v_0 : $\dashrightarrow$ -12

Eve chooses ☺ in v_2 : $\rightsquigarrow$ -26

Eve chooses v_3 in v_2 : $\rightsquigarrow$ -10

Value

$$\overline{\text{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\text{Play}(v, \sigma, \tau))}_{\text{dVal}^{\sigma}(v)}$$

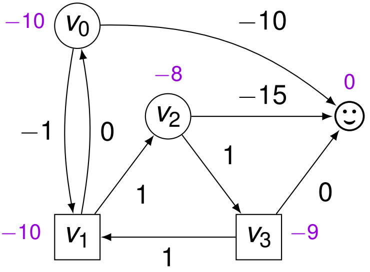
Deterministic strategy

$$\sigma : V^* \times V_{\text{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

σ Adam τ Eve



Determinacy

$$\mathrm{dVal}(v) = \overline{\mathrm{dVal}}(v) = \underline{\mathrm{dVal}}(v)$$

Value

$$\overline{\mathrm{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\mathrm{Play}(v, \sigma, \tau))}_{\mathrm{dVal}^{\sigma}(v)}$$

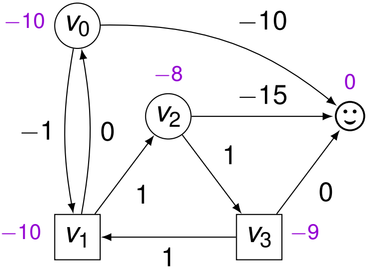
Deterministic strategy

$$\sigma : V^* \times V_{\mathrm{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

$\boxed{\sigma}$ Adam $\quad \textcircled{\tau}$ Eve



Optimal strategy

$$\mathrm{dVal}^{\sigma^*}(v) \leqslant \mathrm{dVal}(v)$$

Value

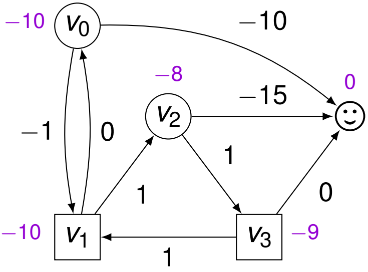
$$\overline{\mathrm{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\mathrm{Play}(v, \sigma, \tau))}_{\mathrm{dVal}^{\sigma}(v)}$$

Deterministic strategy

$$\sigma : V^* \times V_{\mathrm{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]



Optimal strategy

$\mathrm{dVal}^{\sigma^*}(v) \leqslant \mathrm{dVal}(v)$

Optimal strategy for Adam

An optimal strategy for Adam may require finite memory.

Value

$$\overline{\mathrm{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\mathrm{Play}(v, \sigma, \tau))}_{\mathrm{dVal}^{\sigma}(v)}$$

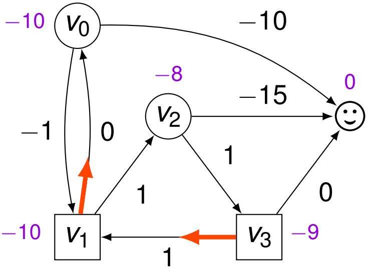
Deterministic strategy

$\sigma : V^* \times V_{\mathrm{Adam}} \rightarrow V$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

σ Adam τ Eve



Optimal strategy

$\mathrm{dVal}^{\sigma^*}(v) \leqslant \mathrm{dVal}(v)$

Optimal strategy for Adam

The switching strategy:

- σ_1 : reach negative cycle

Value

$$\overline{\mathrm{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\mathrm{Play}(v, \sigma, \tau))}_{\mathrm{dVal}^{\sigma}(v)}$$

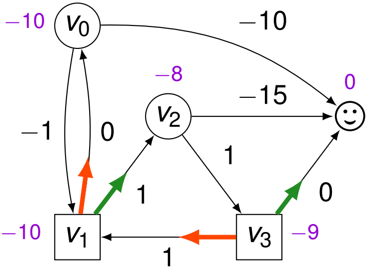
Deterministic strategy

$\sigma : V^* \times V_{\mathrm{Adam}} \to V$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

$\boxed{\sigma}$ Adam $\quad$ τ Eve



Optimal strategy

$\text{dVal}^{\sigma^*}(v) \leqslant \text{dVal}(v)$

Optimal strategy for Adam

The switching strategy:

- σ_1 : reach negative cycle
- σ_2 : reach ☺

Value

$$\overline{\text{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\text{Play}(v, \sigma, \tau))}_{\text{dVal}^{\sigma}(v)}$$

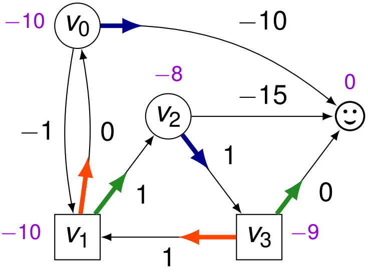
Deterministic strategy

$\sigma : V^* \times V_{\text{Adam}} \to V$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Deterministic Strategies [1]

$\boxed{\sigma}$ Adam $\quad$ $\textcircled{\tau}$ Eve



Optimal strategy

$$\mathrm{dVal}^{\sigma^*}(v) \leqslant \mathrm{dVal}(v)$$

Optimal strategy for Eve

Eve has a memoryless optimal strategy.

Value

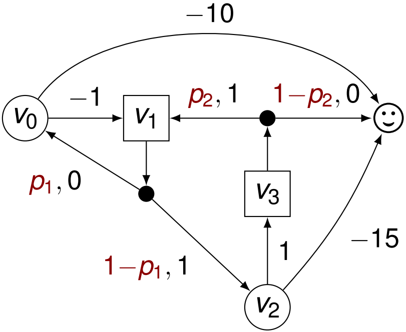
$$\overline{\mathrm{dVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbf{SP}(\mathrm{Play}(v, \sigma, \tau))}_{\mathrm{dVal}^{\sigma}(v)}$$

Deterministic strategy

$$\sigma : V^* \times V_{\mathrm{Adam}} \to V$$

[1] *Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games*, T. Brihaye, G. Geeraerts, A. Haddad and B. Monmege, 2016, Acta Informatica

Memoryless strategies

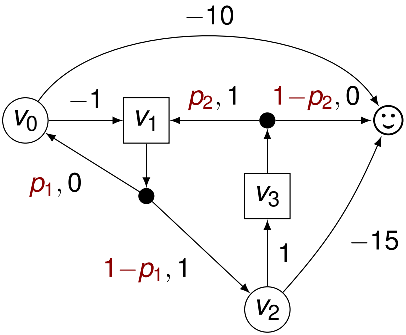


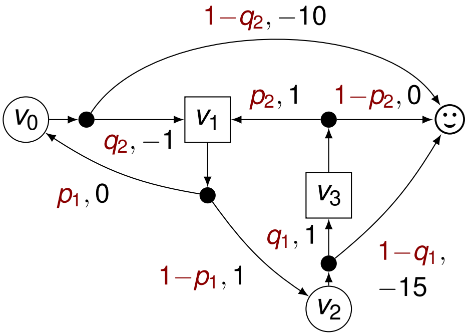
Memoryless strategy

$\sigma : V_{\text{Adam}} \to \Delta(V)$

Memoryless strategies

σ Adam τ Eve





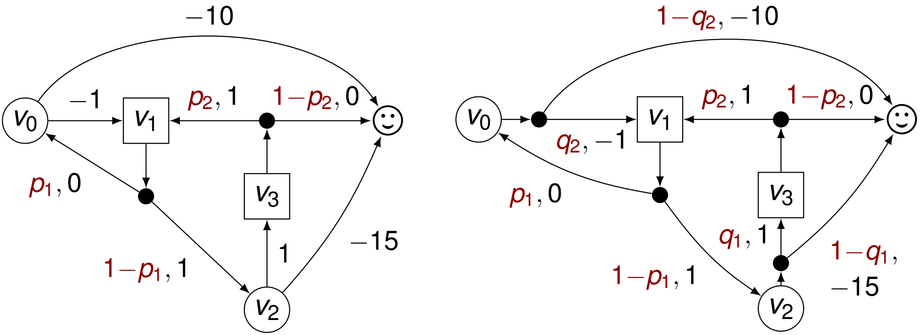
Value

$$\overline{\text{mVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbb{E}^{\sigma,\tau}(\mathbf{SP})}_{\text{mVal}^{\sigma}(v)}$$

Memoryless strategy

$$\sigma : V_{\text{Adam}} \to \Delta(V)$$

Memoryless strategies



Value

$$\overline{\mathrm{mVal}}(v) = \inf_{\sigma} \underbrace{\sup_{\tau} \mathbb{E}^{\sigma,\tau}(\mathbf{SP})}_{\mathrm{mVal}^{\sigma}(v)}$$

Memoryless strategy

$$\sigma : V_{\mathrm{Adam}} \to \Delta(V)$$

ε-optimal strategy

$$\mathrm{mVal}^{\sigma^*}(v) \leqslant \overline{\mathrm{mVal}}(v) + \varepsilon$$

Contributions (1)

Main theorem

For all shortest-path games and vertices v, $\mathrm{dVal}(v) = \overline{\mathrm{mVal}}(v)$.

Contributions (1)

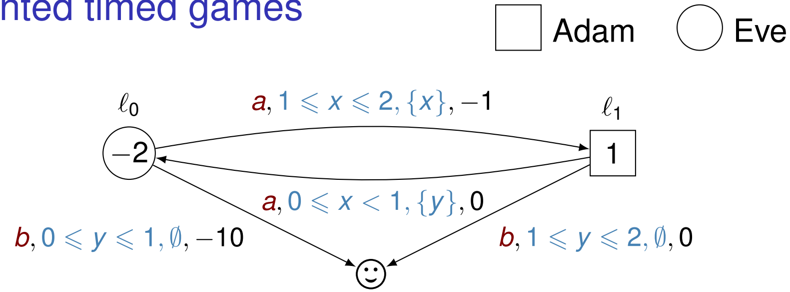
Main theorem

For all shortest-path games and vertices v, $\mathrm{dVal}(v) = \overline{\mathrm{mVal}}(v)$.

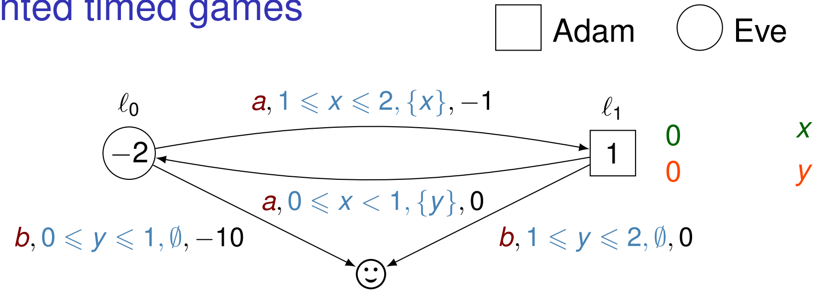
Optimality proposition

1. We can characterize and test in polynomial time the existence of an optimal memoryless strategy.
2. Adam has an optimal (randomised) memoryless strategy if and only if Adam has an optimal deterministic memoryless strategy.

Weighted timed games



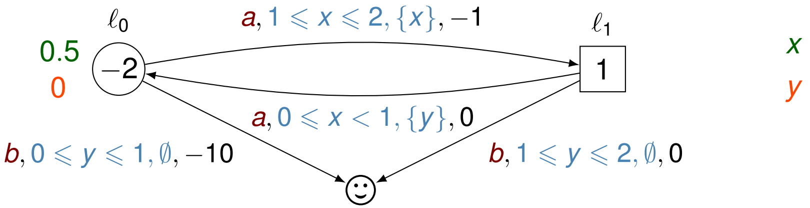
Weighted timed games



Play

$$(\ell_1, \begin{pmatrix} 0 \\ 0 \end{pmatrix})$$

Weighted timed games

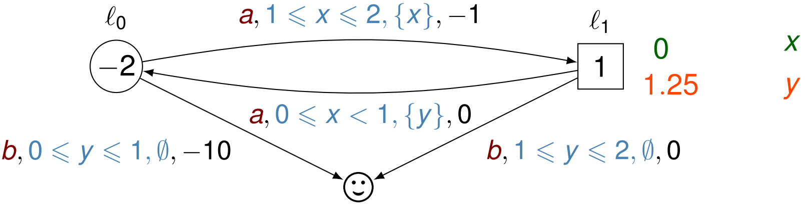


Play

$$\left(\ell_1, \begin{pmatrix} 0 \\ 0 \end{pmatrix}\right) \xrightarrow{a,\ 0.5} \left(\ell_0, \begin{pmatrix} 0.5 \\ 0 \end{pmatrix}\right)$$

$$1 \times 0.5 + 0$$

Weighted timed games



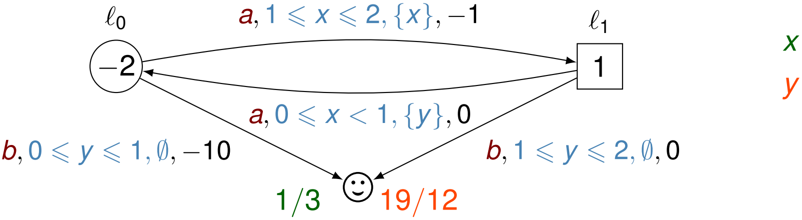
Play

$$\left(\ell_1, \begin{pmatrix}0\\0\end{pmatrix}\right) \xrightarrow{a,\ 0.5} \left(\ell_0, \begin{pmatrix}0.5\\0\end{pmatrix}\right) \xrightarrow{a,\ 1.25} \left(\ell_1, \begin{pmatrix}0\\1.25\end{pmatrix}\right)$$

$$1 \times 0.5 + 0 \qquad -2 \times 1.25 - 1$$

Weighted timed games

□ Adam ○ Eve

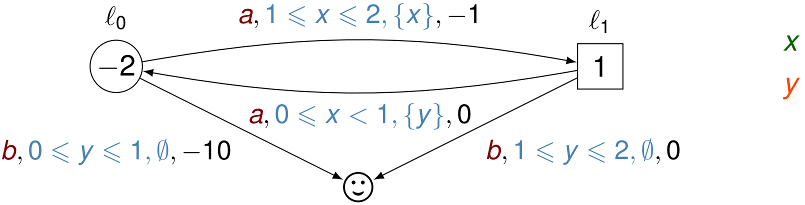


Play

$$\left(\ell_1, \begin{pmatrix}0\\0\end{pmatrix}\right) \xrightarrow{a,\ 0.5} \left(\ell_0, \begin{pmatrix}0.5\\0\end{pmatrix}\right) \xrightarrow{a,\ 1.25} \left(\ell_1, \begin{pmatrix}0\\1.25\end{pmatrix}\right) \xrightarrow{b,\ 1/3} \left(☺, \begin{pmatrix}1/3\\19/12\end{pmatrix}\right)$$

$$1 \times 0.5 + 0 \qquad -2 \times 1.25 - 1 \qquad 1 \times \tfrac{1}{3} + 0$$

Weighted timed games

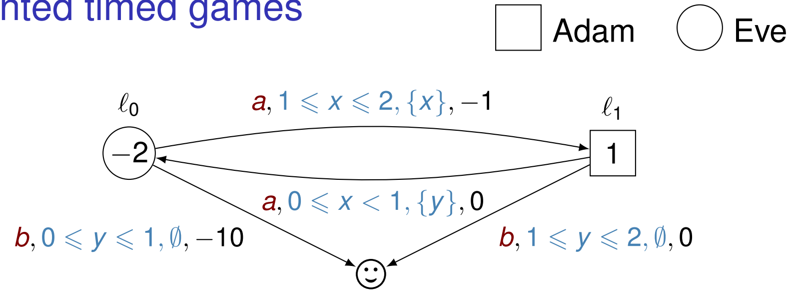


Play

$$\left(\ell_1, \begin{pmatrix}0\\0\end{pmatrix}\right) \xrightarrow{a,\ 0.5} \left(\ell_0, \begin{pmatrix}0.5\\0\end{pmatrix}\right) \xrightarrow{a,\ 1.25} \left(\ell_1, \begin{pmatrix}0\\1.25\end{pmatrix}\right) \xrightarrow{b,\ 1/3} \left(\text{☺}, \begin{pmatrix}1/3\\19/12\end{pmatrix}\right) \rightsquigarrow -\frac{8}{3}$$

$1 \times 0.5 + 0$ $\quad -2 \times 1.25 - 1$ $\quad 1 \times \frac{1}{3} + 0$

Weighted timed games



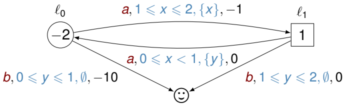
Value problem [2]

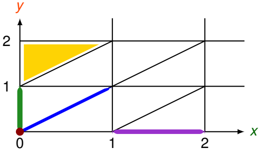
The value problem, i.e. deciding if $\mathrm{dVal}(\ell_0, \nu_0) \leqslant c$ with $c \in \mathbb{Z}$, is undecidable.

[2] *On the value problem in weighted timed games*, P. Bouyer, S. Jaziri, and N. Markey, 2015, CONCUR.

Weighted timed games

Adam Eve



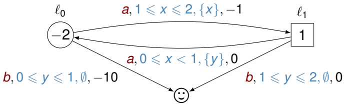


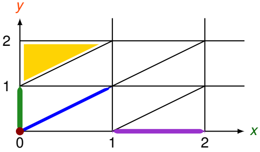
Finite abstraction

- Region : a finite abstraction of time

Weighted timed games

□ Adam ○ Eve





Finite abstraction

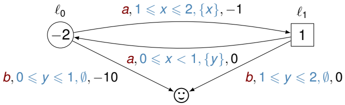
- Region : a finite abstraction of time

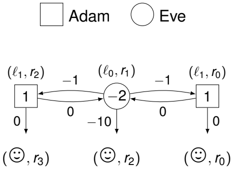
Simulate play

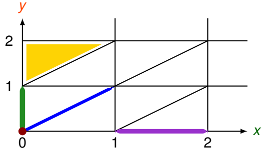
$$\left(\ell_1, \begin{pmatrix} 0 \\ 0 \end{pmatrix}\right) \xrightarrow{a,\ 0.5} \left(\ell_0, \begin{pmatrix} 0.5 \\ 0 \end{pmatrix}\right) \xrightarrow{a,\ 1.25} \left(\ell_1, \begin{pmatrix} 0 \\ 1.25 \end{pmatrix}\right) \xrightarrow{b,\ 1/3} \left(☺, \begin{pmatrix} 1/3 \\ 19/12 \end{pmatrix}\right)$$

$$(\ell_1, r_0) \xrightarrow{a,\ 0.5} (\ell_0, r_1) \xrightarrow{a,\ 1.25} (\ell_1, r_2) \xrightarrow{b,\ 1/3} (☺, r_3)$$

Weighted timed games







Finite abstraction

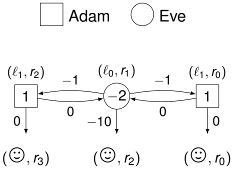
- Region : a finite abstraction of time
- Region automaton : finite abstraction of timed game

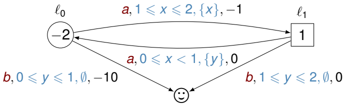
Simulate play

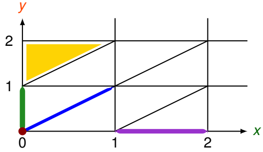
$$\left(\ell_1, \begin{pmatrix} 0 \\ 0 \end{pmatrix}\right) \xrightarrow{a,\ 0.5} \left(\ell_0, \begin{pmatrix} 0.5 \\ 0 \end{pmatrix}\right) \xrightarrow{a,\ 1.25} \left(\ell_1, \begin{pmatrix} 0 \\ 1.25 \end{pmatrix}\right) \xrightarrow{b,\ 1/3} \left(☺, \begin{pmatrix} 1/3 \\ 19/12 \end{pmatrix}\right)$$

$$(\ell_1, r_0) \xrightarrow{a,\ 0.5} (\ell_0, r_1) \xrightarrow{a,\ 1.25} (\ell_1, r_2) \xrightarrow{b,\ 1/3} (☺, r_3)$$

Weighted timed games







Finite abstraction

- Region : a finite abstraction of time
- Region automaton : finite abstraction of timed game
- Divergent weighted timed games

Simulate play

$$\left(\ell_1, \begin{pmatrix} 0 \\ 0 \end{pmatrix}\right) \xrightarrow{a,\ 0.5} \left(\ell_0, \begin{pmatrix} 0.5 \\ 0 \end{pmatrix}\right) \xrightarrow{a,\ 1.25} \left(\ell_1, \begin{pmatrix} 0 \\ 1.25 \end{pmatrix}\right) \xrightarrow{b,\ 1/3} \left(☺, \begin{pmatrix} 1/3 \\ 19/12 \end{pmatrix}\right)$$

$$(\ell_1, r_0) \xrightarrow{a,\ 0.5} (\ell_0, r_1) \xrightarrow{a,\ 1.25} (\ell_1, r_2) \xrightarrow{b,\ 1/3} (☺, r_3)$$

Contribution (2)

Conjecture

For all divergent weighted timed games and configurations (ℓ, ν),

$$\mathsf{dVal}(\ell, \nu) = \overline{\mathsf{mVal}}(\ell, \nu)$$

Contribution (2)

Conjecture

For all divergent weighted timed games and configurations (ℓ, ν),

$$\mathsf{dVal}(\ell, \nu) = \overline{\mathsf{mVal}}(\ell, \nu)$$

Steps of proof

1. Definition of randomized memoryless strategy
2. Finite abstraction and approximation successive
3. Application of main theorem in quantitative games

Memoryless simulate deterministic

Main theorem
For all shortest-path games and vertices v, $\mathrm{dVal}(v) = \overline{\mathrm{mVal}}(v)$.

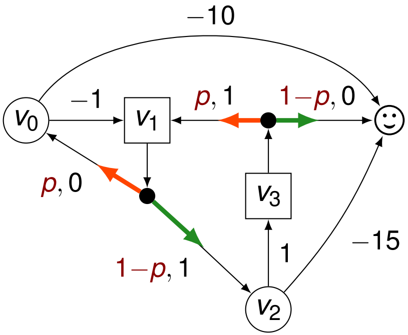
Memoryless simulate deterministic

Claim

For all v, there exists p such that $\mathrm{mVal}^{\sigma_p}(v) \leqslant \mathrm{dVal}(v)$.

Memoryless simulate deterministic

$\boxed{\sigma}$ Adam $\quad$ $\textcircled{\tau}$ Eve



Claim

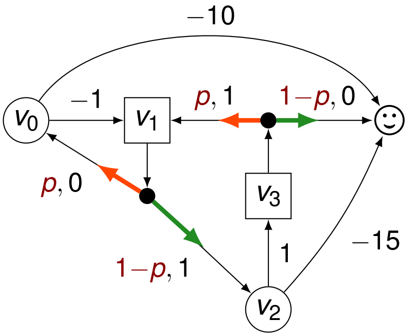
For all v, there exists p such that $\mathrm{mVal}^{\sigma_p}(v) \leqslant \mathrm{dVal}(v)$.

Strategy σ_p

Let $\langle \sigma_1, \sigma_2 \rangle$ be an optimal switching strategy, for all $p \in (0, 1)$,

$$\sigma_p = p \times \sigma_1 + (1-p) \times \sigma_2$$

Memoryless simulate deterministic



Claim

For all v, there exists p such that $\mathrm{mVal}^{\sigma_p}(v) \leqslant \mathrm{dVal}(v)$.

Properties of σ_p

- For all τ, $\mathbb{P}^{\sigma_p,\tau}(\diamond ☺) = 1$

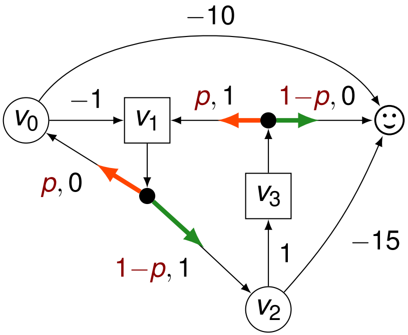
Strategy σ_p

Let $\langle \sigma_1, \sigma_2 \rangle$ be an optimal switching strategy, for all $p \in (0, 1)$,

$$\sigma_p = p \times \sigma_1 + (1 - p) \times \sigma_2$$

Memoryless simulate deterministic

σ Adam τ Eve



Claim

For all v, there exists p such that $\mathrm{mVal}^{\sigma_p}(v) \leqslant \mathrm{dVal}(v)$.

Properties of σ_p

- For all τ, $\mathbb{P}^{\sigma_p,\tau}(\diamond☺) = 1$
- Eve has an optimal memoryless deterministic strategy [2]

Strategy σ_p

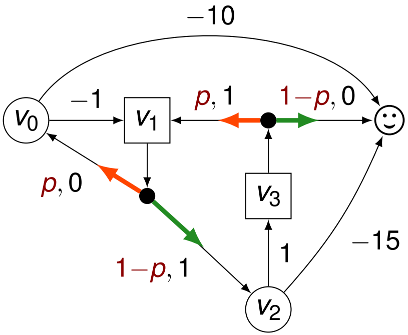
Let $\langle \sigma_1, \sigma_2 \rangle$ be an optimal switching strategy, for all $p \in (0,1)$,

$$\sigma_p = p \times \sigma_1 + (1-p) \times \sigma_2$$

[2] *An analysis of stochastic shortest path problems*, D. Bertsekas and J. Tsitsiklis, 1991, Mathematics of Operations Research.

Memoryless simulate deterministic

σ Adam τ Eve



Claim

For all v, there exists p such that $\mathrm{mVal}^{\sigma_p}(v) \leqslant \mathrm{dVal}(v)$.

Problem

Presence of non-negative cycles

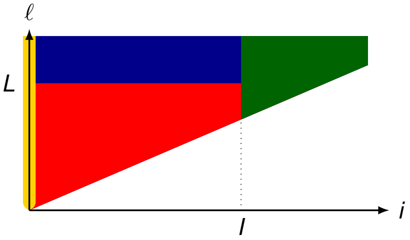
Strategy σ_p

Let $\langle \sigma_1, \sigma_2 \rangle$ be an optimal switching strategy, for all $p \in (0, 1)$,

$$\sigma_p = p \times \sigma_1 + (1-p) \times \sigma_2$$

Memoryless simulate deterministic

$\boxed{\sigma}$ Adam τ Eve



Claim

For all v, there exists p such that $\mathrm{mVal}^{\sigma_p}(v) \leqslant \mathrm{dVal}(v)$.

Problem

Presence of non-negative cycles

Tool of proof

Control them with a partition of plays

Strategy σ_p

Let $\langle \sigma_1, \sigma_2 \rangle$ be an optimal switching strategy, for all $p \in (0, 1)$,

$$\sigma_p = p \times \sigma_1 + (1-p) \times \sigma_2$$

Deterministic simulate memoryless

Claim

For all v and for all memoryless strategy ρ, there exists a deterministic strategy σ such that

$$\mathrm{dVal}^{\sigma}(v) \leqslant \mathrm{mVal}^{\rho}(v)$$

Tools of proof

- Build a switching strategy $\sigma = \langle \sigma_1, \sigma_2 \rangle$ for ρ.
- Value iteration for the fixpoint that gives the value.

Conclusion

Quantitative games

1. Adam has the same hope using memory or randomness.
2. Existence of an optimal memoryless strategy for Adam is testable in polynomial time.

Divergent weighted timed games

Claim : memoryless value is equal to deterministic value

Perspectives

Quantitative games

- A polynomial-time algorithm to compute the value
- Extension to probabilistic value (memory and randomisation)

Weighted timed games

Extension to other class of decidable game such as games restricted to a single clock