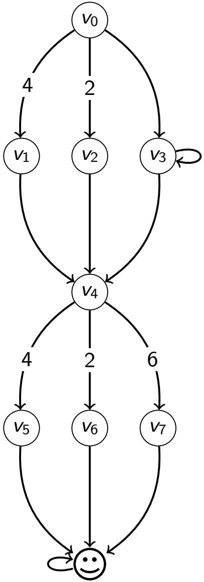
Multi-Weighted Reachability Games

Thomas Brihaye[1] and Aline Goeminne[2]

1. UMONS – Université de Mons, Belgium.
2. F.R.S.-FNRS & UMONS – Université de Mons, Belgium.

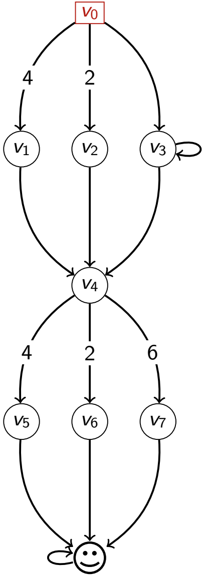
Highlights'23

Reachability Games



Is it possible to reach ☺ with cost $\leq c$?

Reachability Games

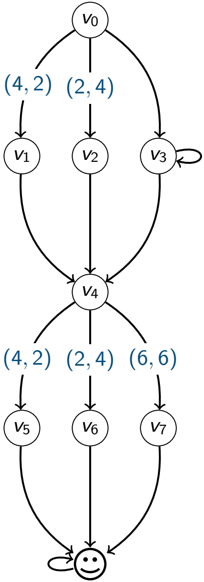


Is it possible to reach ☺ with cost $\leq c$?

↓

Is it possible to reach ☺ whatever the behavior of the **environment** with cost $\leq c$?

Reachability Games



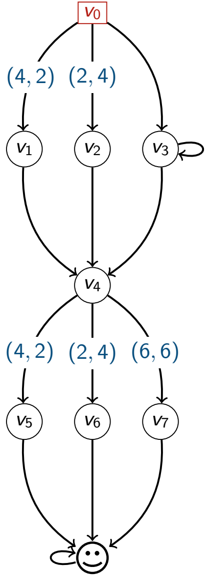
Is it possible to reach ☺ with cost $\leq c$?

↓ ↓

Is it possible to reach ☺ whatever the behavior of the **environment** with cost $\leq c$?

Is it possible to reach ☺ with cost $\leq (\mathbf{c_1}, \mathbf{c_2})$?

Reachability Games



Is it possible to reach ☺ with cost $\leq c$?

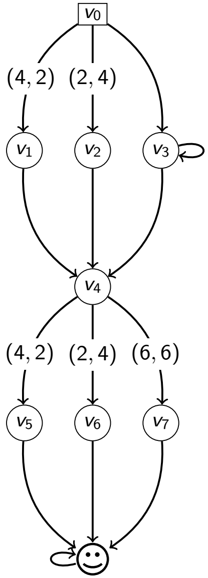
Is it possible to reach ☺ whatever the behavior of the **environment** with cost $\leq c$?

Is it possible to reach ☺ with cost $\leq (\mathbf{c_1}, \mathbf{c_2})$?

Is it possible to reach ☺ whatever the behavior of the **environment** with cost $\leq (\mathbf{c_1}, \mathbf{c_2})$?

Two-Player Multi-Weighted Reachability Games

Two-Player Multi-Weighted Reachability Games



- A d-weighted graph $G = (V, E, (w_i)_{1 \leq i \leq d})$
- Two players: Player ◯ and Player □;
- Turn-based;

Quantitative reachability objective

Given a target set $F \subseteq V$, for all **plays** (infinite paths in G) $\rho = \rho_0 \rho_1 \ldots$:

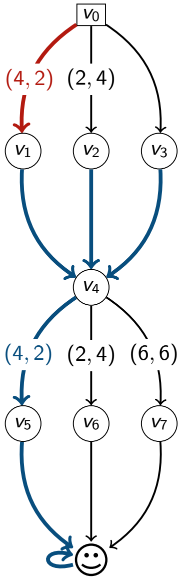
$$\mathrm{Cost}_i(\rho) = \begin{cases} \displaystyle\sum_{n=0}^{k-1} w_i(\rho_n, \rho_{n+1}) & \text{if } k \text{ is the least index st.} \rho_k \in F \\ +\infty & \text{otherwise} \end{cases}$$

Rem: **same target set** for all dimensions.

Ex:

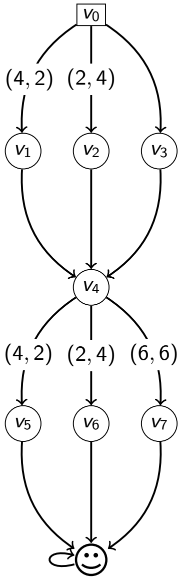
- $\mathrm{Cost}(v_0 v_3^\omega) = (\mathrm{Cost}_1(v_0 v_3^\omega), \mathrm{Cost}_2(v_0 v_3^\omega)) = (+\infty, +\infty)$
- $\mathrm{Cost}(v_0 v_1 v_4 v_5 (☺)^\omega) = (10, 6)$

Two-Player Multi-Weighted Reachability Games



- A strategy for Player ◯: $\sigma_{\bigcirc} : V^* V_{\bigcirc} \longrightarrow V$.
- Given a **strategy profile** $(\sigma_{\bigcirc}, \sigma_{\Box})$ and an initial vertex $v_0 \rightsquigarrow$ only one consistent play $\langle \sigma_{\bigcirc}, \sigma_{\Box} \rangle_{v_0}$ called the **outcome**.
  Ex: $\langle \sigma_{\bigcirc}, \sigma_{\Box} \rangle_{v_0} = v_0 v_1 v_4 v_5 (☺)^{\omega}$.

Two-Player Multi-Weighted Reachability Games



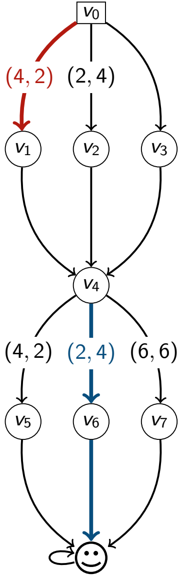
Player ◯ can **ensure** a cost profile $\mathbf{c} = (c_1, \dots, c_d)$ from v if **there exists** a strategy $\sigma_{\bigcirc}$ such that **for all strategies** $\sigma_{\Box}$ of Player □:

$$\mathrm{Cost}(\langle \sigma_{\bigcirc}, \sigma_{\Box} \rangle_v) \leq_C \mathbf{c} = (c_1, \dots, c_d)$$

Ex:

- $(8, 8) \rightsquigarrow$ **Yes.** **(with memory!)**

Two-Player Multi-Weighted Reachability Games

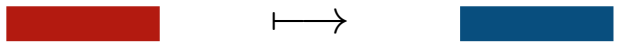


Player ◯ can **ensure** a cost profile $\mathbf{c} = (c_1, \ldots, c_d)$ from v if **there exists** a strategy $\sigma_\bigcirc$ such that **for all strategies** $\sigma_\square$ of Player □:

$$\mathrm{Cost}(\langle \sigma_\bigcirc, \sigma_\square \rangle_v) \leq_C \mathbf{c} = (c_1, \ldots, c_d)$$

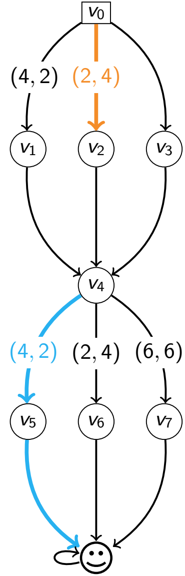
Ex:

- $(8, 8) \rightsquigarrow$ **Yes. (with memory!)**


$(8, 8)$

Player ◯ can adapt his strategy in function of the choice of Player □ $\rightsquigarrow$ finite-memory strategy!

Two-Player Multi-Weighted Reachability Games



Player ◯ can **ensure** a cost profile $\mathbf{c} = (c_1, \ldots, c_d)$ from v if **there exists** a strategy $\sigma_{\bigcirc}$ such that **for all strategies** $\sigma_{\Box}$ of Player □:

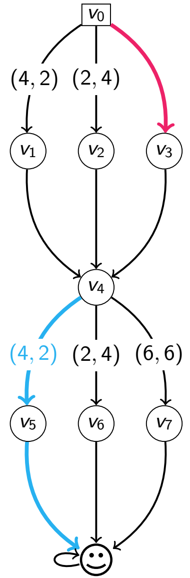
$$\text{Cost}(\langle \sigma_{\bigcirc}, \sigma_{\Box} \rangle_v) \leq_C \mathbf{c} = (c_1, \ldots, c_d)$$

Ex:

- $(8, 8) \rightsquigarrow$ **Yes. (with memory!)**

Player ◯ can adapt his strategy in function of the choice of Player □ $\rightsquigarrow$ finite-memory strategy!

Two-Player Multi-Weighted Reachability Games

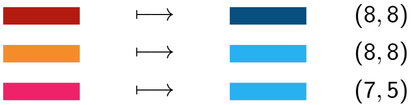


Player ◯ can **ensure** a cost profile $\mathbf{c} = (c_1, \dots, c_d)$ from v if **there exists** a strategy $\sigma_{\bigcirc}$ such that **for all strategies** $\sigma_{\square}$ of Player □:

$$\mathrm{Cost}(\langle \sigma_{\bigcirc}, \sigma_{\square} \rangle_v) \leq_C \mathbf{c} = (c_1, \dots, c_d)$$

Ex:

- $(8, 8) \rightsquigarrow$ **Yes. (with memory!)**



Player ◯ can adapt his strategy in function of the choice of Player □ $\rightsquigarrow$ finite-memory strategy!

$\mathrm{Ensure}(v) = \{\mathbf{c} \in \overline{\mathbb{N}}^d \mid \exists \sigma_{\bigcirc} \text{ st. } \forall \sigma_{\Box}, \mathrm{Cost}(\langle \sigma_{\bigcirc}, \sigma_{\Box} \rangle_v) \leq_C \mathbf{c}\}$.

$\mathrm{minimal}(\mathrm{Ensure}(v)) = \mathrm{Pareto}(v) \rightsquigarrow$ **Pareto frontier** from v.

For $\mathbf{c} = (c_1, \ldots, c_d) \in \mathrm{Pareto}(v)$, a strategy $\sigma_{\bigcirc}$ is **c-Pareto-optimal** if $\sigma_{\bigcirc}$ ensures **c** from v.

Studied problems

Studied problems

1. Compute the Pareto frontier and Pareto-optimal strategies.
2. Decide the **constrained existence problem**.

Constrained existence problem (CEP)

Given a game, a vertex $v \in V$ and $\mathbf{c} \in \mathbb{N}^d$,
does there exist a strategy $\sigma_{\bigcirc}$ of Player $\bigcirc$ such that for all strategies of Player $\square$, we have:

$$\mathrm{Cost}(\langle \sigma_{\bigcirc}, \sigma_{\square} \rangle_v) \leq_{\mathrm{C}} \mathbf{c}$$

Computing Pareto(v)

$$\text{Ensure}^k(v) = \{\mathbf{c} \in \overline{\mathbb{N}}^d \mid \exists \sigma_{\bigcirc} \text{ st. } \forall \sigma_{\Box}, \text{Cost}(\langle \sigma_{\bigcirc}, \sigma_{\Box} \rangle_v) \leq_C \mathbf{c} \wedge |\langle \sigma_{\bigcirc}, \sigma_{\Box} \rangle_v|_F \leq k\}.$$

The algorithm computes, step by step, the sets $I^k(v)$ for all $v \in V$.

For all $k \in \mathbb{N}$ and all $v \in V$, $I^k(v) = \text{minimal}(\text{Ensure}^k(v))$

There exists $k^* \in \mathbb{N}$ such that for all $v \in V$ and for all $\ell \in \mathbb{N}$, $I^{k^*}(v) = I^{k^*+\ell}(v)$.

For all $v \in V$, $I^{k^*}(v) = \text{Pareto}(v)$.

Theorem

The fixpoint algorithm runs in time polynomial in W and $|V|$ and is **exponential** in d.
Where W is the maximal weight on an edge.

Computing Pareto(v) and Pareto-optimal strategies

for $v \in \mathsf{F}$ **do** $\mathsf{I}^0(v) = \{\mathbf{0}\}$
for $v \notin \mathsf{F}$ **do** $\mathsf{I}^0(v) = \{\infty\}$

repeat
  for $v \in V$ **do**
    if $v \in \mathsf{F}$ **then** $\mathsf{I}^{k+1}(v) = \{\mathbf{0}\}$

    else if $v \in V_{\bigcirc}$ **then**

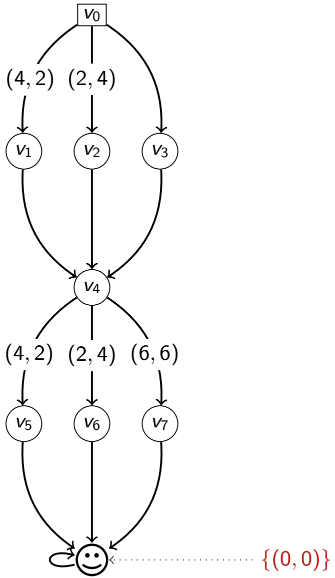
$$\mathsf{I}^{k+1}(v) = \text{minimal}\left(\bigcup_{v' \in \text{succ}(v)} \uparrow \mathsf{I}^k(v') + \mathbf{w}(v, v')\right)$$

    else if $v \in V_{\Box}$ **then**

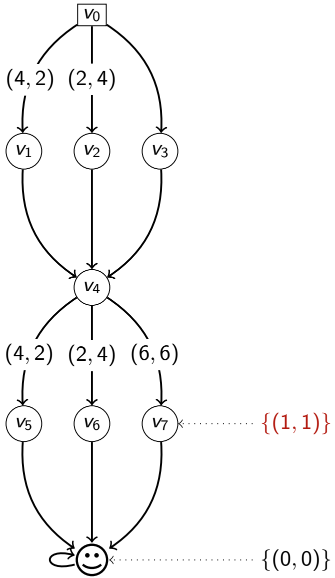
$$\mathsf{I}^{k+1}(v) = \text{minimal}\left(\bigcap_{v' \in \text{succ}(v)} \uparrow \mathsf{I}^k(v') + \mathbf{w}(v, v')\right)$$

until $\mathsf{I}^{k+1}(v) = \mathsf{I}^k(v)$ for all $v \in V$

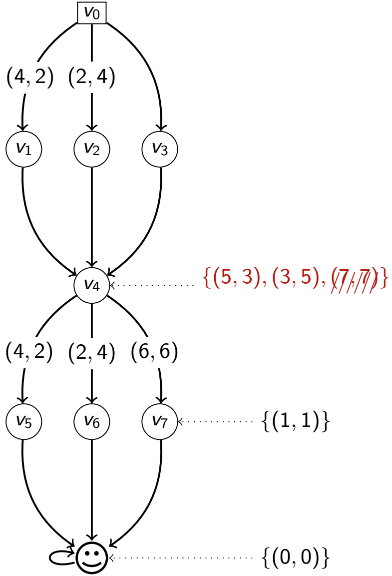
Computing Pareto(v)



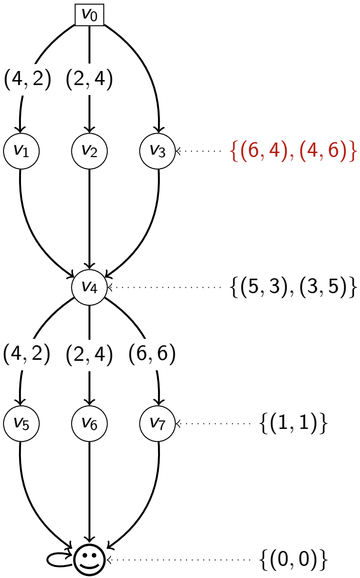
Computing Pareto(v)



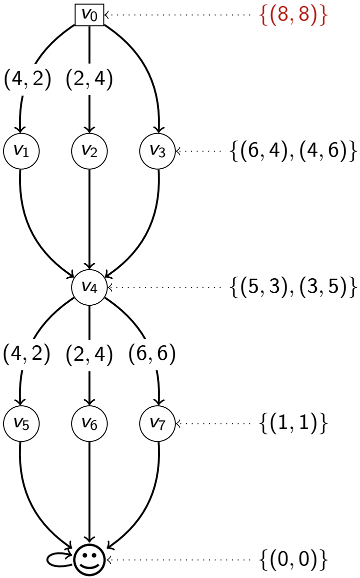
Computing Pareto(v)



Computing Pareto(v)



Computing Pareto(v)



Pareto-optimal strategies

for $v \in \mathsf{F}$ **do** $\mathsf{I}^0(v) = \{\mathbf{0}\}$
for $v \notin \mathsf{F}$ **do** $\mathsf{I}^0(v) = \{\infty\}$

repeat
  for $v \in V$ **do**
    if $v \in \mathsf{F}$ **then** $\mathsf{I}^{k+1}(v) = \{\mathbf{0}\}$

    else if $v \in V_{\bigcirc}$ **then**

$$\mathsf{I}^{k+1}(v) = \text{minimal}\left(\bigcup_{v' \in \mathsf{succ}(v)} \uparrow \mathsf{I}^k(v') + \mathbf{w}(v, v')\right)$$

      for $\mathbf{x} \in \mathsf{I}^{k+1}(v)$ **do**
        if $\mathbf{x} \in \mathsf{I}^k(v)$ **then** $f_v^{k+1}(\mathbf{x}) = f_v^k(\mathbf{x})$
        else
          $f_v^{k+1}(\mathbf{x}) = (v', \mathbf{x}')$ where v' and $\mathbf{x}'$ are such that $v' \in \mathsf{succ}(v)$, $\mathbf{x} = \mathbf{x}' + \mathbf{w}(v, v')$ and $\mathbf{x}' \in \mathsf{I}^k(v')$

    else if $v \in V_{\Box}$ **then**

$$\mathsf{I}^{k+1}(v) = \text{minimal}\left(\bigcap_{v' \in \mathsf{succ}(v)} \uparrow \mathsf{I}^k(v') + \mathbf{w}(v, v')\right)$$

until $\mathsf{I}^{k+1}(v) = \mathsf{I}^k(v)$ for all $v \in V$

Computing Pareto-optimal strategies

Given $u \in V$ and $\mathbf{c} \in \mathsf{I}^*(u) \backslash \{\infty\}$, we define a strategy $\sigma^*_{\bigcirc}$ from u such that for all $hv \in \mathrm{Hist}_{\bigcirc}(u)$, let $\mathcal{C}(hv) = \{\mathbf{x}' \in \mathsf{I}^*(v) \mid \mathbf{x}' \leq_{\mathsf{C}} \mathbf{c} - \mathrm{Cost}(hv) \wedge \mathbf{x}' \leq_{\mathsf{L}} \mathbf{c} - \mathrm{Cost}(hv)\}$,

$$\sigma^*_{\bigcirc}(hv) = \begin{cases} v' & \text{for some } v' \in \mathrm{succ}(v), \text{ if } \mathcal{C}(hv) = \emptyset \\ f^*_v(\mathbf{x})[1] & \text{where } \mathbf{x} = \min_{\leq_{\mathsf{L}}} \mathcal{C}(hv), \text{ if } \mathcal{C}(hv) \neq \emptyset \end{cases}.$$

$\sigma^*_{\bigcirc}$ is a **c**-Pareto-optimal strategy from u.

Conclusion

Conclusion

	Componentwise order	Lexicographic order
minimal(Ensure(v))	in exponential time	in polynomial time
CEP	PSPACE-complete	in P

- **uniform approach** to compute minimal(Ensure(v)) both for the componentwise order and the lexicographic order $\rightsquigarrow$ fixpoint algorithm;
- (Pareto)-optimal strategies can be synthesized thanks to the fixpoint algorithm;
- Pareto-optimal strategies may **require memory**.