Out: May 26 | Due: June 2

Submit your solutions until Friday, June 2, 14:00, in the box next to office 343.

Warning! I have changed the definition of being a trap *for* a player in the lecture notes. The definition now fits the intuition: If X is a trap for player $\star$, then $\star$ can be trapped inside X by the opponent $\overline{\star}$ (instead of the other way around).

Definition: Trap
We call a set $X \subseteq V$ a **trap** for player $\star \in \{\square, \bigcirc\}$ if

- for all positions $x \in X$ owned by player $\star$, all successors are in X, and
- all positions $x \in X$ owned by the opponent $\overline{\star}$ have at least one successor in X.

Exercise 1: It's a trap!

a) Formally prove Part a) of Lemma 8.8 from the lecture notes:

Let $Y \subseteq V$ and $\star \in \{\square, \bigcirc\}$. The complement of the attractor $V \setminus \mathrm{Attr}_{\star}(Y)$ is a trap for player $\star$.

Is the attractor $\mathrm{Attr}_{\star}(Y)$ a trap for any of the players?

b) Formally prove Lemma 8.11 from the lecture notes:

Let $X \subseteq V$ be a trap for player $\star$ in $\mathcal{G}$ and let $s_{\overline{\star}}$ be a strategy for the opponent $\overline{\star}$ that is winning from some vertex $x \in X$ in the subgame $\mathcal{G}_{\restriction X}$. Then $s_{\overline{\star}}$ is also winning from x in the original game $\mathcal{G}$.

The proof for the positional determinacy of parity games gives rise to the following algorithm.

Algorithm: Zielonka's recursive algorithm
Input: parity game $\mathcal{G}$ given by $G = (V_\square, V_\bigcirc, R)$ and Ω.
Input: winning regions $W_\square$ and $W_\bigcirc$.

Procedure *solve*($\mathcal{G}$)

$n = \max_{x \in V} \Omega(x)$
if $n = 0$ **then**
  return $W_\bigcirc = V, W_\square = \varnothing$
else
  $N = \{x \in V \mid \Omega(x) = n\}$
  if n even **then**
    $\star = \bigcirc, \overline{\star} = \square$
  else
    $\star = \square, \overline{\star} = \bigcirc$
  end if
  $A = \mathrm{Attr}_\star(N)$
  $W'_\bigcirc, W'_\square = solve(\mathcal{G}_{\restriction V \setminus A})$
  if $W'_\star = V \setminus A$ **then**
    return $W_\star = V, W_{\overline{\star}} = \varnothing$
  else
    $B = \mathrm{Attr}_{\overline{\star}}(W'_{\overline{\star}})$
    $W''_\square, W''_\bigcirc = solve(\mathcal{G}_{\restriction V \setminus B})$
    return $W_\star = W''_\star, W_{\overline{\star}} = W''_{\overline{\star}} \cup B$
  end if
end if

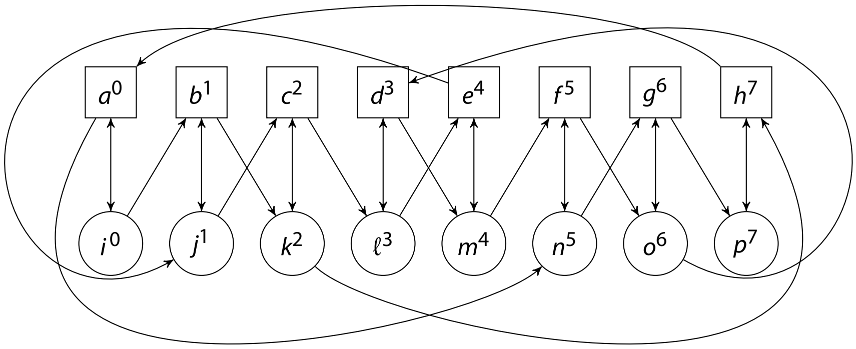
Exercise 2: Algorithmics of parity games

a) Prove Lemma 8.15 from the lecture notes:

Let $\mathcal{G}$ be a parity game, i.e. a game arena G and a priority function Ω, and let $x \in V$ be a position. Assume that $s_\star$ is a positional strategy for player $\star \in \{\square, \bigcirc\}$.

Present an algorithm that checks whether $s_\star$ is winning from x. The running time of the algorithm should be polynomial in $|G|$.

b) Use Zielonka's recursive algorithm to solve the following parity game. The notation is as in Exercise 8.20 (Exercise 2 on the last exercise sheet).



Exercise 3: Weak parity games

Let us consider **weak parity games**. Just like a parity game, a weak parity game is given by a game arena $G = (V_\square \uplus V_\bigcirc, R)$ and a priority function Ω. Instead of considering the highest priority that *occurs infinitely often* to determine the winner of a play, we consider the highest priority that *occurs at all*.

Formally, for an infinite sequence $p \subseteq A^\omega$, we define the **occurrence set**

$$\mathrm{Occ}(p) = \left\{ a \in A \;\middle|\; \exists i \in \mathbb{N}\colon p_i = a \right\} .$$

The winner of the weak parity game given by G and Ω is determined by the **weak parity winning condition**:

$$\begin{aligned} win \;:\; Plays_{max} &\to \{\square, \bigcirc\} \\ p &\mapsto \begin{cases} \bigcirc & \text{, if } \max \mathrm{Occ}\big(\Omega(p)\big) \text{ is even,} \\ \square & \text{, else, i.e. if } \max \mathrm{Occ}\big(\Omega(p)\big) \text{ is odd.} \end{cases} \end{aligned}$$

a) Present an algorithm that, given a weak parity game on a finite, deadlock-free game arena, computes the winning regions of both players.

   Briefly argue that your algorithm is correct.

   Hint: Attractors!

b) Is the winning condition of weak parity games prefix-independent, i.e. does Lemma 8.4 hold?

   Do uniform positional winning strategies exist?