

Games and controller synthesis - Reachability games

September 28, 2021

You can experiment with the open-access tool GAVS+ (Game Arena Visualization and Synthesis).

As a general rule, all game arenas in our exercises are *finite*.

Reachability games

We fix a set of target states (vertices) $F \subseteq V$. A play is won by P_0 if it visits F . Otherwise, so if the play avoids F , it is won by P_1 .

Attractors for P_0 :

$$\begin{aligned} X_0 &= F \\ X_{n+1} &= X_n \cup \\ &\quad \{v \in V_0 \mid \exists w \in X_n, (v, w) \in E\} \\ &\quad \{v \in V_1 \mid \forall w \text{ t.q. } (v, w) \in E : w \in X_n\} \end{aligned}$$

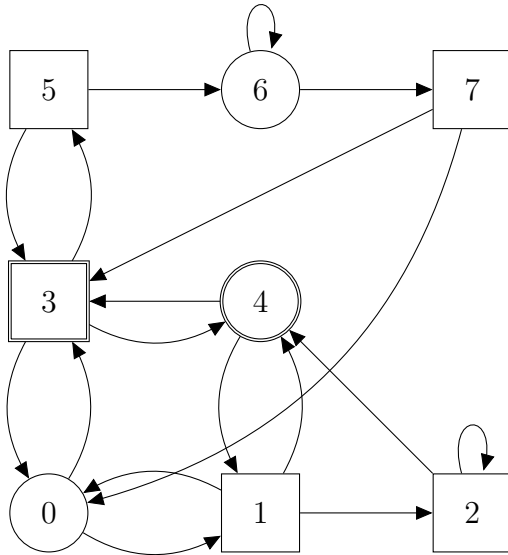
$$X_0 \subseteq X_1 \subseteq \dots \subseteq X_{|V|} =: \text{Attr}_0(F)$$

We saw that X_k is the set of vertices from which P_0 can reach F in at most k moves. The set $\text{Attr}_0(F)$ is the **winning region** of P_0 (*attractor of P_0 to F*) and its complement $V \setminus \text{Attr}_0(F)$ is the **winning region** of P_1 .

The attractor $\text{Attr}_1(\cdot)$ of P_1 to some set of nodes is defined symmetrically, exchanging P_0 and P_1 . The complement of an attractor for (say) P_0 , i.e., a set of the form $U = V \setminus \text{Attr}_0(F)$, is called *trap* pour P_0 (or 0-trap for short): P_0 cannot leave U , and P_1 has always a possible move to stay in U .

Exercise 1

Compute the winning regions and winning strategies of P_0 and P_1 in the reachability game below.



Exercise 2

Show how to compute the winning regions and the winning strategies for P_0 and P_1 in the two games below. Are the winning strategies positional? If no, which memory is required by each strategy?

1. A maximal play is won by P_0 if and only if the play is finite.
2. A maximal play is won by P_0 if and only if it does not visit any vertex twice.

What changes if the game arena is infinite?

Exercise 3

We reconsider the winning condition $\text{Reach}(F_1) \wedge \text{Reach}(F_2)$ for P_0 , where $F_1 \cap F_2 = \emptyset$. So P_0 wins a play if and only if the play visits both F_1 and F_2 .

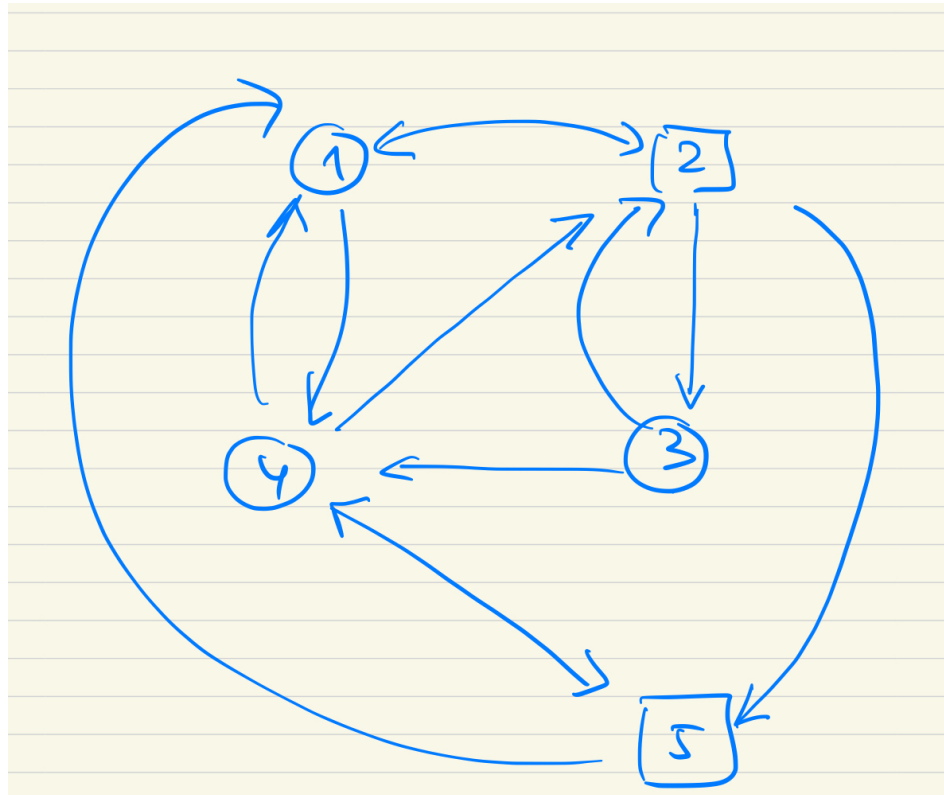
Show that the winning strategy of P_1 is memoryless, whereas P_0 needs one bit of memory in order to win.

Exercise 4

We consider the following winning condition Win for P_0 , given two disjoint sets $F, B \subseteq V$.

A play π is won by P_0 if it reaches F without visiting B before. So $\text{Win} = (V \setminus B)^* F V^\omega$.

1. Adapt the attractor construction to this winning condition.
2. Compute the winning regions on the following example, with $F = \{1\}$ and $B = \{2\}$:



Exercise 5

We consider the winning condition $\text{Win} = \text{Reach}(F) \vee \text{Avoid}(B)$ for P_0 , where $F \cap B = \emptyset$. A maximal play π is won by P_0 if and only if π visits F or avoids B . In other words, $\text{Win} = V^*FV^\omega \cup (V \setminus B)^\omega$.

We want to compute the winning regions W_0, W_1 , and winning strategies σ_0, σ_1 .

1. What is the winning condition for P_1 ?
2. First solution: transform this game into a weak parity game (on the same arena) by associating priorities to vertices in a suitable way. What do you deduce about the winning strategies?
3. Second solution: reduce this game to a safety game on a larger arena, where the players remember which sets were visited.
4. Compute the winning regions for this game on the example in Exercise 1 with $F = \{3\}$ and $B = \{2\}$.

Exercise 6

We consider the winning condition $\text{Win} = (\text{Reach}(F_1) \wedge \text{Avoid}(B_1)) \vee (\text{Reach}(F_2) \wedge \text{Avoid}(B_2))$ for P_0 , where the sets F_1, F_2, B_1, B_2 are pairwise disjoint. So a maximal play π is won by P_0 if and only if π visits F_1 and avoids B_1 , or it visits F_2 and avoids B_2 .

Compute the winning regions W_0, W_1 , and winning strategies σ_0, σ_1 .