

Algorithmique pour les jeux à deux joueurs

Florian Bourse

February 8, 2026

1 Jeux d'accessibilité à deux joueurs sur un graphe

On modélise les états d'une partie d'un jeu entre deux joueurs J_1 et J_2 par un graphe orienté biparti $G = (S, A)$ avec $S = S_1 \sqcup S_2$, où S_1 représente les états contrôlés par J_1 et S_2 représente les états contrôlés par J_2 , avec $A \subset S_1 \times S_2 \cup S_2 \times S_1$.

Dans ce cours, nous nous limiterons aux graphes acycliques, certains résultats utilisent cette propriété, d'autre non. Si le graphe contient un cycle, une partie peut potentiellement être infinie.

Le graphe étant orienté acyclique, il possède nécessairement des sommets sans successeurs (des puits ou états finals). Pour conclure notre modélisation, nous devons répartir ces états en états gagnants pour J_1 , que nous noterons G_1 , gagnants pour J_2 , que nous noterons G_2 , ou de match nul.

Un jeu d'accessibilité à deux joueurs est donc la donnée d'un graphe $G = (S_1, S_2, A)$ et de la répartition de ses états finals G_1 et G_2 .

Tous ces jeux sont donc équivalents au jeu suivant : un jeton est posé sur un sommet du graphe. À chaque tour, le joueur qui contrôle le sommet sur lequel est posé le pion le déplace vers un successeur de ce sommet. L'état final du jeton décide qui gagne la partie : si c'est un état de G_i , alors J_i gagne, sinon, c'est un match nul.

On se limite en MPI aux stratégies sans mémoires, c'est-à-dire qu'un coup joué ne dépend que de l'état actuel de la partie et non des coups précédents. On obtient alors la définition suivante : Une *stratégie* pour J_i est une fonction $\pi : S_i \mapsto S_{3-i}$ qui associe à chaque état s contrôlé par le joueur J_i un état s' contrôlé par le joueur J_{3-i} tel que $(s, s') \in A$.

Étant donné une stratégie π_1 pour J_1 , une stratégie π_2 pour J_2 , et un état initial s_0 (supposons sans perte de généralité que $s_0 \in S_1$), on peut alors jouer la partie et définir la suite d'états (s_i) tels que $s_{2i+1} = \pi_1(s_{2i})$ et $s_{2i+2} = \pi_2(s_{2i+1})$. Cette suite d'états est nécessairement finie car le graphe est acyclique et fini. On obtient donc un résultat à la partie : si le dernier état s_k est dans un G_i , alors J_i gagne la partie, sinon, c'est un match nul.

Si pour un sommet $s \in S$ il existe une stratégie π_i telle que pour toute stratégie π_{3-i} , la partie jouée avec les stratégies π_i et π_{3-i} commençant au sommet s se termine par une victoire du joueur J_i , alors s est une *position gagnante* pour J_i et π_i est une *stratégie gagnante* pour J_i dans cette position.

Toute position est soit gagnante pour J_1 , soit gagnante pour J_2 , soit n'est gagnante ni pour l'un ni pour l'autre. À démontrer.

Si le match nul est impossible, tout sommet est une position gagnante pour J_1 ou pour J_2 . À démontrer. *On pourra utiliser le fait que le graphe est orienté acyclique et raisonner par induction bien fondée.*

2 Calcul des attracteurs

Il s'agit de calculer l'ensemble des positions gagnantes pour J_i .

Pour simplifier les énoncés, introduisons les ensembles $\text{succ}(s) = \{t \in S \mid (s, t) \in A\}$ et $S'_i = \{s \in S_i \mid \text{succ}(s) \neq \emptyset\}$. On définit par récurrence les ensembles de sommets suivants :

$$\begin{aligned}\mathcal{X}_0 &= G_i \\ \mathcal{X}_{n+1} &= \mathcal{X}_n \cup \{s \in S'_i \mid \text{succ}(s) \cap \mathcal{X}_n \neq \emptyset\} \cup \{s \in S'_{3-i} \mid \text{succ}(s) \subset \mathcal{X}_n\}\end{aligned}$$

À démontrer :

1. La suite \mathcal{X}_n est stationnaire. (donner un n tel que $\mathcal{X}_n = \cup_{k \in \mathbb{N}} \mathcal{X}_k$)
2. Toutes les positions de \mathcal{X}_n sont gagnantes pour J_i . (*réurrence*)
3. Toutes les positions gagnantes pour J_i sont dans $\cup_{k \in \mathbb{N}} \mathcal{X}_k$.

Une fois les positions gagnantes calculées, il est facile de déterminer une stratégie gagnante :

soit $s \in S$ une position gagnante pour J_i , alors toute stratégie π telle que $\pi(s)$ est une position gagnante pour J_i si il en existe dans $\text{succ}(s)$ est une stratégie gagnante pour J_i . À démontrer.