Logique avancée

5 - Jeux d'accessibilité

Master Info Nice Sophia Antipolis
E. Lozes

6. mai 2019

Contenu de la séance d'aujourd'hui

- Jeux d'accessibilité
- Stratégies, amnésie
- Jeu d'acceptation pour les AFA
- Algorithme de recherche de stratégie

Jeu d'accessibilité

- 2 joueurs ($\diamond$ et $\square$) : la position de jeu détermine qui joue le coup suivant
  (un joueur peut a priori jouer plusieurs coups de suite).
- "un coup" $=$ un changement de **position de jeu**
- Le joueur $\diamond$ doit atteindre une position objectif parmi un ensemble de positions objectifs possibles
- Le joueur $\square$ doit l'en empêcher

Définitions

Arène

Une Arène est un tuple $\mathcal{G} = (V, V_\diamond, V_\Box, E, v_I, Z)$ tel que

- $V = V_\diamond \uplus V_\Box$ est l'ensemble des positions de jeu
- $E \subseteq V \times V$ est l'ensemble des coups
- $v_I \in V$ est la position initiale
- $F \subseteq V$ est l'ensemble des positions de but du joueur $\diamond$

Partie

Une partie dans l'arène $\mathcal{G}$ est un chemin fini ou infini dans le graph (V, E) qui commence à la position v_I.

Définitions

Arène

Une Arène est un tuple $\mathcal{G} = (V, V_\diamond, V_\Box, E, v_I, Z)$ tel que

- $V = V_\diamond \uplus V_\Box$ est l'ensemble des positions de jeu
- $E \subseteq V \times V$ est l'ensemble des coups
- $v_I \in V$ est la position initiale
- $F \subseteq V$ est l'ensemble des positions de but du joueur $\diamond$

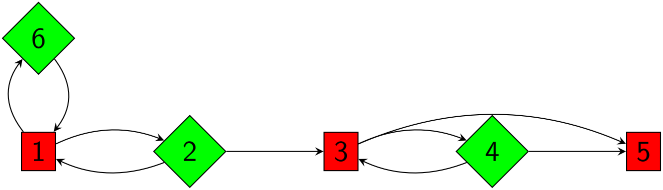
Partie

Une partie dans l'arène $\mathcal{G}$ est un chemin fini ou infini dans le graph (V, E) qui commence à la position v_I.

Gagnant

Le gagnant de la partie est le joueur $\diamond$ si soit le chemin passe par une position dans F, soit c'est un chemin fini qui termine dans une position de $V_\Box$ qui n'a pas de successeur possible (le joueur $\Box$ est « coincé »)

Exemple



- $v_I = 1,\ F = \{5, 6\}$
- une partie : ...

Stratégie de jeu

Une stratégie est une fonction de choix qui détermine le coup d'un joueur à un moment quelconque d'une partie.

Définition

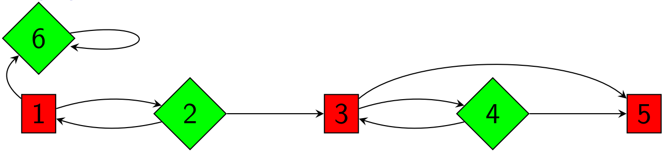
Soit $\mathcal{G} = (V, V_\diamond, V_\Box, v_I, F)$ une arène. Une stratégie pour le joueur $x \in \{\diamond, \Box\}$ est une fonction

$$\sigma : V^* V_x \to V$$

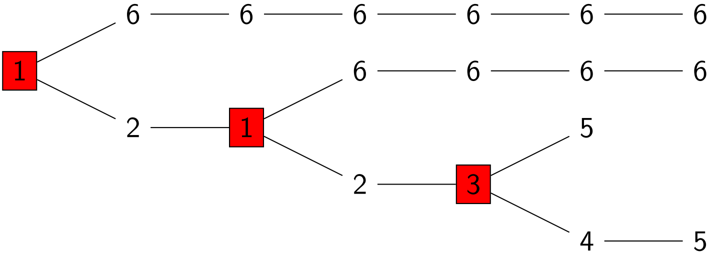
telle que pour toute partie $\pi.v \in V^* V_x$, si $v' = \sigma(\pi.v)$, alors $\pi.v.v'$ est encore une partie, i.e. $(v, v') \in E$.

Les stratégies vues comme des arbres

Arène



Stratégie



Stratégie gagnante

Partie conforme à une stratégie

Une partie π est *conforme à la stratégie* σ du joueur x si pour tout préfixe $\pi'.v \in V^* V_x$ de π, et pour $v' = \sigma(\pi.v)$, on a $\pi.v.v'$ qui est encore un préfixe de π

Stratégie gagnante

Une strategie σ du joueur x est gagnante si toute les parties conformes à σ sont gagnées par x.

Théorème de détermination

Théorème

Soit $\mathcal{G}$ un jeu d'accessibilité.

- Ou bien le joueur $\diamond$ a une stratégie gagnante ;
- ou bien le jouer $\square$ a une stratégie gagnante.

Ce théorème est une instance d'un théorème plus général qui embrasse les jeux infinis et pose une condition topologique sur l'ensemble des parties gagnantes (théorème de Donald A. Martin, 1982)

Stratégie amnésique

Intuitivement, une stratégie amnésique est une stratégie qui dépend seulement de la position de jeu actuelle, et non de tout le début de la partie.

Contre-exemple : les échecs n'admettent pas une stratégie amnésique (règle du Pat, match nul si 3 fois même configuration)

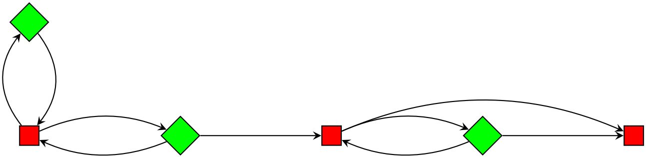
On parle aussi parfois de stratégie *positionnelle*.

Stratégie amnésique (2)

Définition

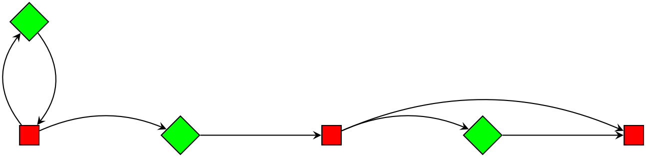
Une stratégie σ pour le joueur x est amnésique s'il existe une fonction $\sigma' : V_x \to V$ tellle que $\sigma(\pi.v) = \sigma'(v)$.

Stratégie amnésique = arène demi-déterministe



arène

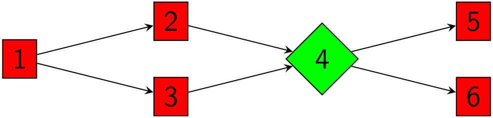
Stratégie amnésique = arène demi-déterministe



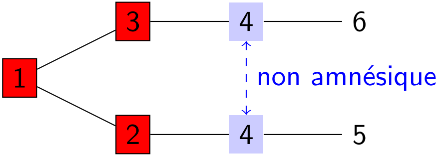
stratégie amnésique pour 

Propriété d'amnésie sur les arbres

arène

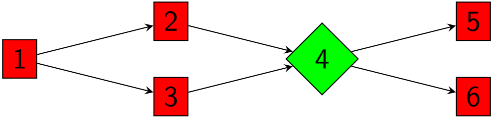


Arbre de stratégie

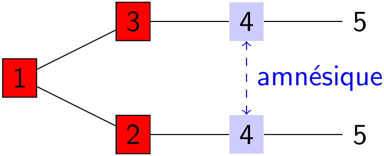


Propriété d'amnésie sur les arbres

arène

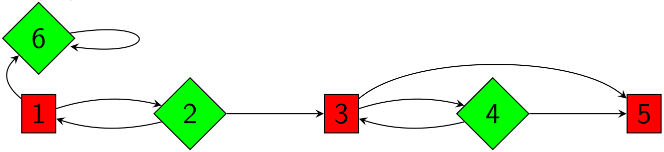


Arbre de stratégie

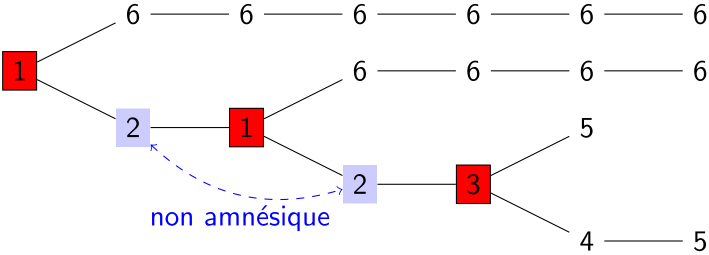


Autre exemple

Arène



Arbre de stratégie



Détermination positionelle

Théorème

Soit $\mathcal{G} = (V, V_\diamond, V_\Box, E, v_I, F)$ un jeu d'accessibilité.

- Ou bien le joueur $\diamond$ a une stratégie gagnante amnénisque ;
- ou bien le joueur $\Box$ a une stratégie gagnante amnésique.

<u>Preuve</u> :

Soit X l'ensemble des positions v telles que le jeu avec position initiale $v_I = v$ admet une stratégie gagnante amnésique pour le joueur $\diamond$. Soit $Y = V \setminus X$. Il s'agit de montrer que pour toute position initiale $v \in Y$, le joueur $\Box$ a une stratégie gagnante amnésique. Si $v \in V_\diamond$, alors tous les successeurs de v sont dans Y (sans quoi $\diamond$ pourrait rejoindre X et gagner) ; si $v \in V_\Box$, de même, il existe au moins un successeur de v dans Y (sans quoi $\Box$ ne pourrait pas éviter X, et ne pourrait pas avoir de stratégie gagnante). On en déduit une stratégie gagnante pour $\Box$ sur Y.

Calcul effectif

Pour un ensemble de positions X, notons $\mathrm{Pre}(X)$ l'ensemble des positions v telles que

- si $v \in V_\diamond$, il existe un successeur de v dans X
- si $v \in V_\square$, tous les successeurs de v sont dans X

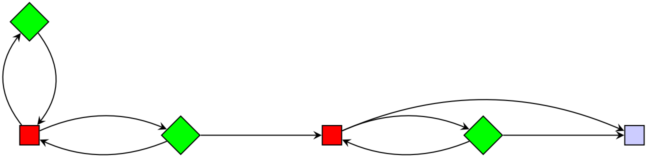
Alors l'ensemble des positions gagnantes est

$$F \cup \mathrm{Pre}(F) \cup \mathrm{Pre}(\mathrm{Pre}(F)) \cup \cdots \cup \mathrm{Pre}^i(F) \cup \ldots$$

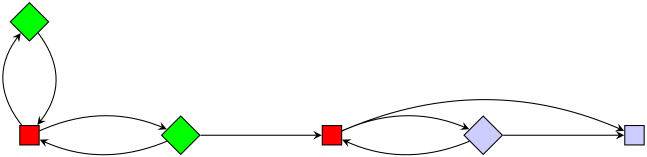
si V est fini de cardinal n, cette union infinie est égale à l'union finie des n premiers ensembles.

Cet ensemble est aussi le plus petit point fixe de la fonction $f : X \mapsto F \cup \mathrm{Pre}(X)$ au sens de l'inclusion ensembliste.

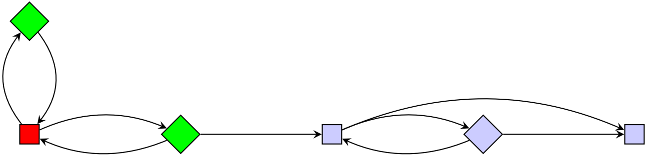
Exemple



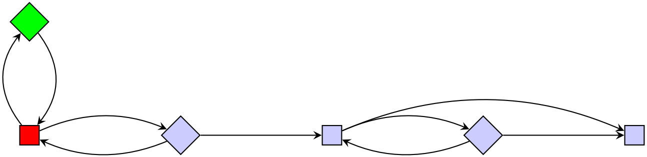
Exemple



Exemple



Exemple



Un premier algorithme

```
def pre(V, V◇, V□, succ, F) :
    res = ∅
    for v in V◇ :
        if succ(v) ∩ F ≠ ∅ :
            res.add(v)
    for v in V□ :
        if succ(v) ⊆ F
            res.add(v)
    return res

def win(V, V◇, V□, succ, F) :
    res = ∅
    new = F
    while new ≠ ∅ :
        res = res ∪ new
        new = pre(V, V◇, V□, succ, res)
    return res
```

Complexité optimale

Théorème Soit $\mathcal{G} = (V, V_\diamond, V_\Box, E, v_I, F)$ un jeu d'accessibilité. On peut calculer en temps $\mathcal{O}(|V| + |E|)$ l'ensemble des positions gagnantes.

preuve (incomplète) : Dans l'algorithme précédent, le calcul de `pre` n'est pas efficace, car on parcours à chaque fois l'ensemble des sommets. On peut gagner en efficacité si l'on associe un compteur à chaque sommet de $V_\Box$ qui indique combien de successeurs en dehors de `res` il lui reste ; ce compteur décroit chaque fois qu'un successeur est ajouté à `res`, et lorsqu'il passe à 0, le sommet est ajouté à `res`. On a besoin de la fonction inverse de succ.

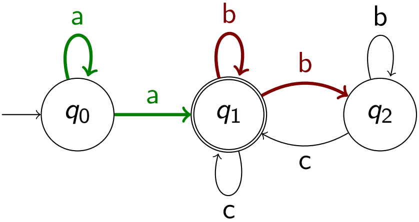
Jeu d'acceptation pour les AFA

Définition

Soit $\mathcal{A}$ un AFA et $w = a_1 \dots a_n$ un mot. On considère l'arène $\mathcal{G}_{\mathcal{A},w} = (V, V_\diamond, V_\Box, E, v_I, F)$ où

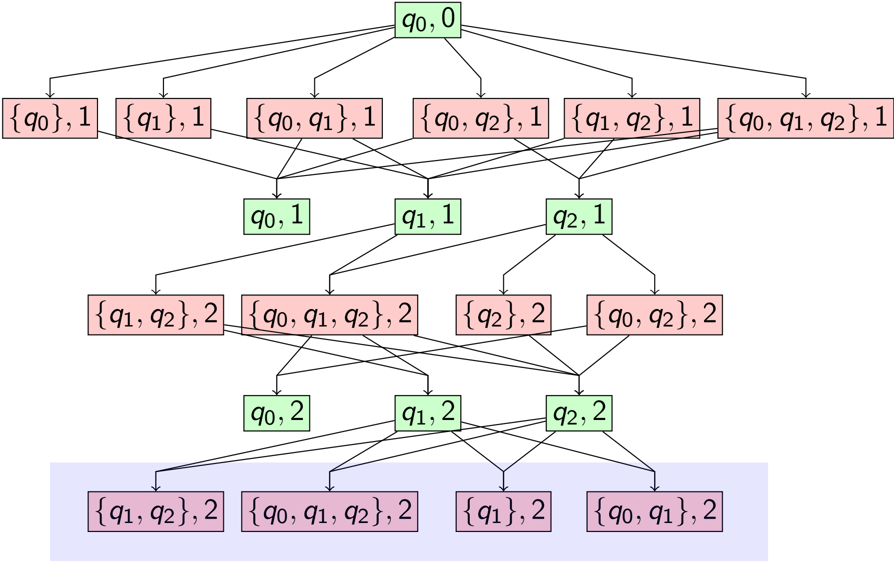
$$
\begin{aligned}
V_\diamond &\triangleq Q \times \{0, \dots, n-1\} \\
V_\Box &\triangleq 2^Q \times \{1, \dots, n-1\} \\
E &\triangleq \big\{\big((q,i),(M,i+1)\big) : M \models \delta_A(q,a_i) \text{ et } i < n\big\} \\
&\cup \big\{\big((M,i),(q,i)\big) : q \in M\big\} \\
v_I &\triangleq (q_I, 0) \\
F &\triangleq \{(M,n) : M \subseteq F_A\}
\end{aligned}
$$

Exemple



$\mathcal{A}$

$$w = abc$$


$q_0, 0$
$\{q_0\}, 1$
$\{q_1\}, 1$
$\{q_0, q_1\}, 1$
$\{q_0, q_2\}, 1$
$\{q_1, q_2\}, 1$
$\{q_0, q_1, q_2\}, 1$
$q_0, 1$
$q_1, 1$
$q_2, 1$
$\{q_1, q_2\}, 2$
$\{q_0, q_1, q_2\}, 2$
$\{q_2\}, 2$
$\{q_0, q_2\}, 2$
$q_0, 2$
$q_1, 2$
$q_2, 2$
$\{q_1, q_2\}, 2$
$\{q_0, q_1, q_2\}, 2$
$\{q_1\}, 2$
$\{q_0, q_1\}, 2$

$$F = \{\{q_1\}\}$$

Jeu d'acceptation et langage

Théorème : $w \in L(\mathcal{A})$ ssi le joueur $\diamond$ a une stratégie gagnante dans $\mathcal{G}_{\mathcal{A},w}$.

Conséquence

Le problème du mot pour un AFA avec n états est décidable en temps $\mathcal{O}(|V_\diamond| \cdot |V_\square|) = O(n \cdot 2^n \cdot |w|^2)$.

Remarque :

- AFA$\rightarrow$ DFA $= n \rightarrow 2^{2^n}$.
- il est plus efficace, de résoudre le jeu
- mais il est encore plus efficace de non-déterminiser à la volée

Exercice de programmation

À rendre par mail à elozes@i3s.unice.fr avant le 27/05

1. `gamesolver <arena.txt >winningpositions.txt` avec un algorithme plus efficace que celui vu en cours !

exemple d'entrée :

```
3
0 1
0 2
1 1
C 0
G 2
```

nombre de sommets sur la première ligne, puis les arètes, puis C i si le sommet i est « controlé », et G i si le sommet i est un objectif.

en sortie : un numéro de sommet gagnant par ligne