À la recherche du meilleur coup au jeu d'Awalé

Nils Lelorieux

numéro de candidat : 24296

Introduction

Figure: Une partie d'Awalé

Figure: Photographie de Wole Soyinka

Un jeu ancien et important

Figure: Un ancien plateau de jeu découvert au Kenya

Vue d'ensemble

Problématique

L'étude théorique du jeu d'Awalé permet-elle de toujours jouer le meilleur coup, ou l'utilisation d'algorithmes est-elle indispensable ?

1. Présentation des règles du jeu et motivations
2. Échec de l'approche théorique et des premiers algorithmes
3. Des résultats concluents grâce à deux nouvelles techniques : la recherche arborescente de Monte-Carlo et l'analyse rétrograde

Règles du jeu

- Deux joueurs s'affrontent.
- 4 graines dans les 12 trous divisé en 2 rangée de 6.
- Le premier joueur choisi un puit et sème les graines.
- Si le dernier puit semé a 2 ou 3 pierres, il les récupère et fait la même chose dans le puit précédent, ...
- Le gagnant est le joueur qui a le plus de pierres

	11	10	9	8	7	6	
0	4	4	4	4	4	4	0
	4	4	4	4	4	4	
	0	1	2	3	4	5	

Figure: L'état du plateau au début de la partie

Déroulé d’une partie

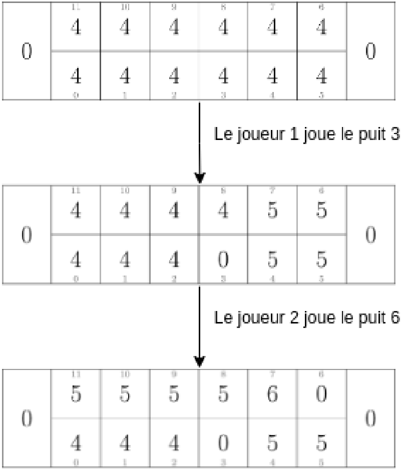

Figure: 2 coups joués par les 2 joueurs en début de partie

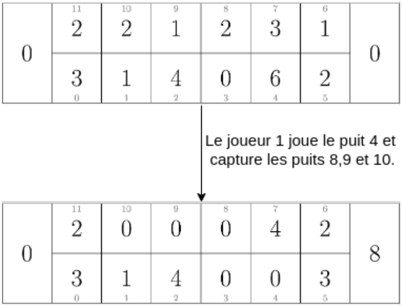

Figure: Position où le joueur 1 capture les pierres du joueur 2

Motivations

- Le facteur d'embrachement est assez faible.
- Possibilité d'utiliser des algorithmes classiques de la théorie des jeux pour trouver le coup optimal
- Il y a moins de 10^{14} états possibles.

Une approche théorique

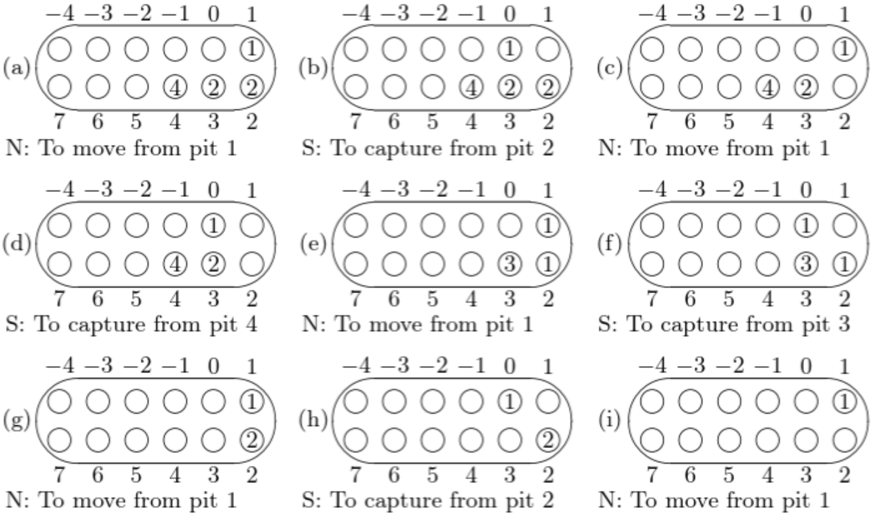

Figure: Un position déterminée

L’algorithme min-max

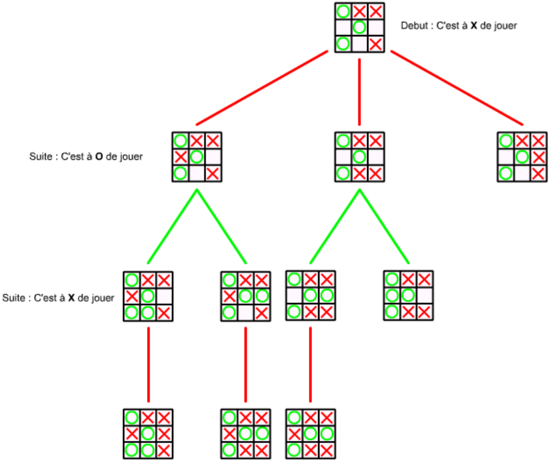

Figure: Un exemple de min max au jeu du morpion

Problème du min-max classique

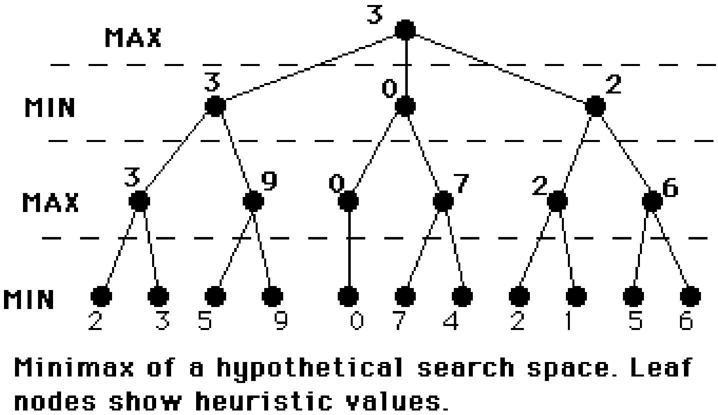

Figure: Un exemple de min max avec heuristique

Résultat min-max

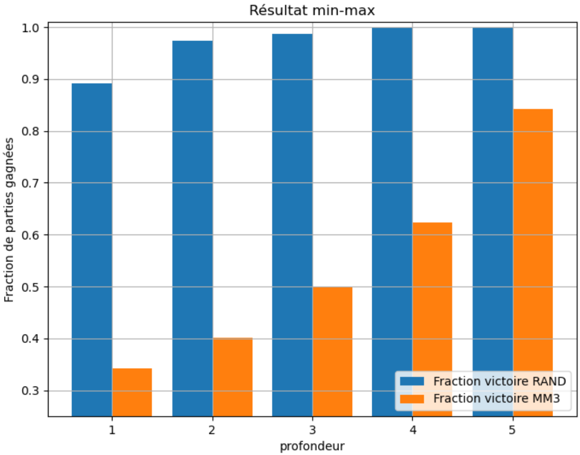

Figure: Résultats de l'algorithme min-max (1000 parties)

Un algorithme de Monte Carlo

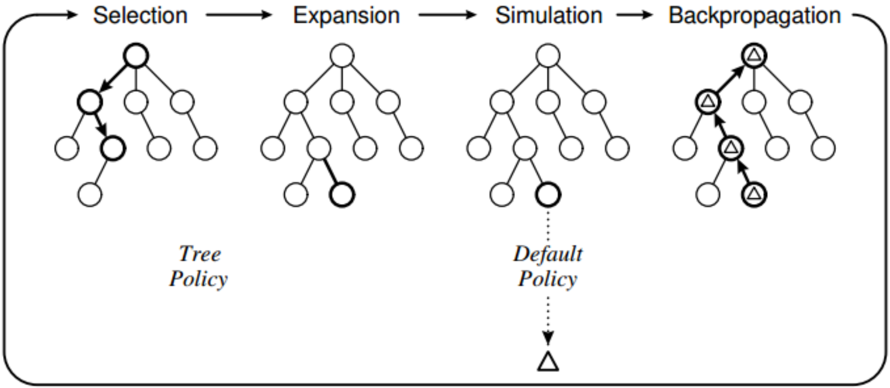

Figure: Principe de l'algorithme de Monte Carlo

Des résultats peu convaincants

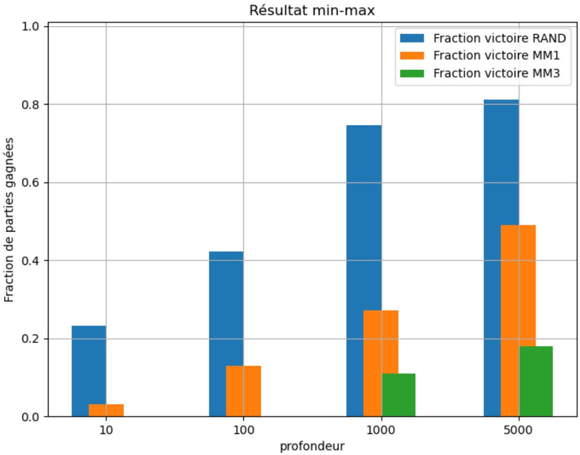

Figure: Résultat de l'algorithme flat Monte Carlo (1000 parties)

Une amélioration de l'algorithme précédent

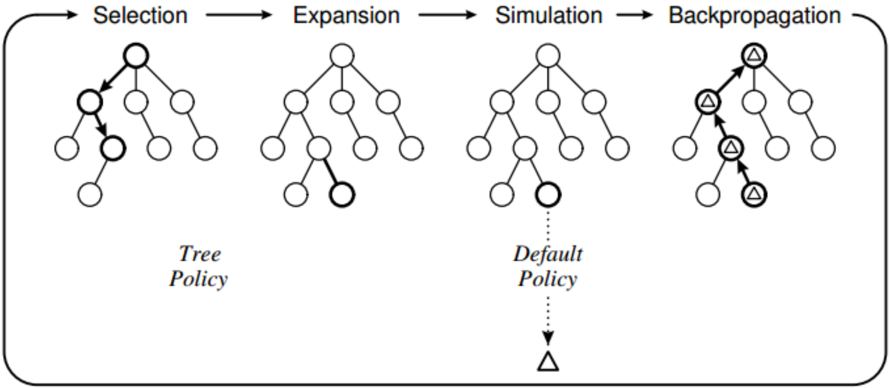

Figure: Principe de la recherche arborescente de Monte Carlo

Comment choisir le noeud à explorer ?

Score d'un noeud

On attribue à chaque noeud n un score Q qui correspond à la moyenne des gains remportés part toutes les parties contenant le noeud n. Étant donné un noeud n et son père p, on a :

$$Q(n) = \frac{1}{N(n)} \sum_{i=1}^{N(p)} \mathbb{I}_i(n) z_i$$

où $N(n)$ représente le nombre de visites du noeud n, z_i est le gain lors de la ième visite de noeud n et $\mathbb{I}_i(n)$ vaut 1 si le noeud n a ete choisie à la i visite par p, 0 sinon

Comment choisir le noeud à explorer (2) ?

Meilleur fils d'un noeud n

$$\underset{n' \in \text{ enfants de } n}{\arg\max} \frac{Q(n')}{N(n')} + c\sqrt{\frac{2ln(N(n))}{N(n')}}$$

Visualisation des arbres

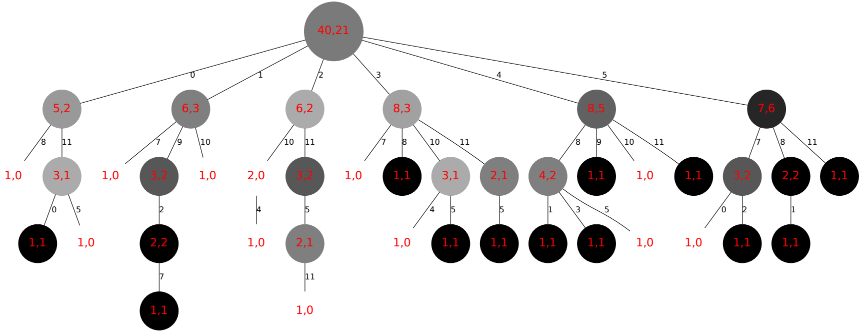

Figure: Un arbre de recherche pour le flat Monte Carlo

Visualisation des arbres

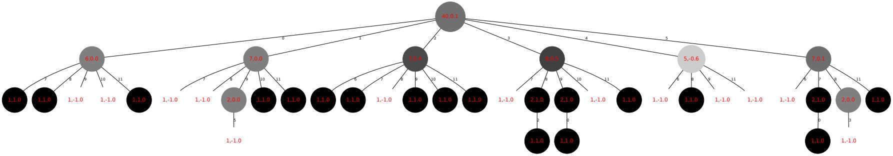

Figure: Un arbre de recherche pour la recherche arborescente de Monte Carlo

Résultats de la recherche arborescente de Monte Carlo

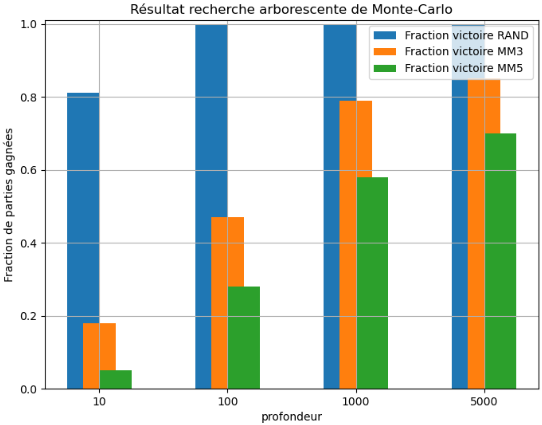

Figure: Résultats de la recherche arborescente de Monte Carlo

Un nouveau paradigme : l'analyse rétrograde

- Utilisation de la programmation dynamique

Anti-coups

Étant donné une configuration à n pierres A, on dit que A' est obtenue par anti-coup depuis la configuration A si il existe un coup jouable dans A' menant à la configuration A sans récolte

Explication de l'analyse rétrograde

L'algorithme se déroule en 3 étapes :

- Initialisation
- Convergence
- Stabilisation

Mais comment stocker toutes ses positions ?

Codage de Gödel

Vecteur $\vec{c}$

Étant donné une configuration A, on note $A(i)$ le nombre de pierre dans le puit i. On pose $\vec{c} = (c_0, ..., c_{11})$ où $c_0 = A(0)$ et $c_{k+1} = A(k+1) + 1 + c_k$

Codage de Gödel

On note,

$$enc(\vec{c}) = \sum_{i=1}^{12} \binom{c_{i-1}}{i}$$

Et on appelle nombre de Gödel de A la quantité

$$\mathcal{E}(A) = n + 25 \times enc(\vec{c})$$

où n est le nombre de pierres de A

Base de donnée construite pour $n = 29$

Conclusion