IUT BUT3
Année 2023-24

Jeux
Technique avancées

Florent Madelaine

Contenu : variante de MINIMAX avec élagage α-β, Jeux avec hasard, arbre de jeux avec hasard, EXPECTMINIMAX.

Les feuilles de TD et d'autres documents et informations
sont disponibles sur la page du cours sur l'ENT

Exercice 1. Pub Quizz

Les affirmations suivantes concernant l'algorithme α-β sont-elles vraies ou fausses ? Dans le cas des affirmations fausses, proposez une correction.

1. L'algorithme α-β donne un meilleur résultat que l'algorithme minimax.

FAUX. L'algorithme α-β donne exactement le même résultat que l'algorithme minimax, mais il est parfois plus rapide.

2. L'algorithme α-β est basé sur un parcours en profondeur de l'arbre de jeu.

VRAI.

3. Au départ, on prend $\alpha = -\infty$ et $\beta = +\infty$.

VRAI. Puisqu'on a encore rien visité, on se place du point de vue le plus pessimiste pour le premier joueur. Potentiellement, le premier joueur pourrait perdre si il prenait la pire option possible ($\alpha = -\infty$) ; et, jusqu'à preuve du contraire, le second joueur pourrait gagner ($\beta = +\infty$).

4. À chaque instant, α représente la valeur du meilleur coup possible pour moi déjà calculé au dessus du nœud courant dans la branche en cours d'exploration, et β représente la valeur du pire coup possible pour moi déjà calculé au dessus du nœud courant dans la branche en cours d'exploration.

VRAI. On cherche donc une valeur dans l'intervalle α, β car une valeur en dehors de cet intervalle signifie que la branche n'a pas d'intérêt : l'un des deux joueurs aura mieux à faire dans une branche "cousine" déjà explorée.

5. Pendant les mises à jour, la valeur de α ne peut qu'augmenter et la valeur de β ne peut que diminuer.

VRAI

6. À chaque nœud, l'algorithme met à jour α et β.

FAUX. À un nœud max, l'algorithme met à jour α si nécessaire, mais ne change pas β et à un nœud min, il met à jour β si nécessaire, mais ne change pas α.

7. La condition qui provoque l'élagage est $\beta > \alpha$.

FAUX La condition qui provoque l'élagage est $\alpha \geq \beta$.

8. L'élagage consiste à ne pas visiter les nœuds qui sont les fils suivants du nœud courant.

VRAI

9. Un nœud max envoie à ses fils la valeur de β qu'il a reçue de son père, et un nœud min envoie à ses fils la valeur de α qu'il a reçue de son père.

VRAI

10. Un nœud max envoie à ses fils la plus grande valeur de α qui a été trouvée jusqu'à présent plus haut sur la branche et chez les fils précédents et un nœud min envoie à ses fils la plus petite valeur de β qui a été trouvée jusqu'à présent plus haut sur la branche et chez les fils précédents.

VRAI

11. Un nœud max retourne à son père la valeur de β qu'il a mise à jour si nécessaire, et nœud min retourne à son père la valeur de α qu'il a mise à jour si nécessaire.

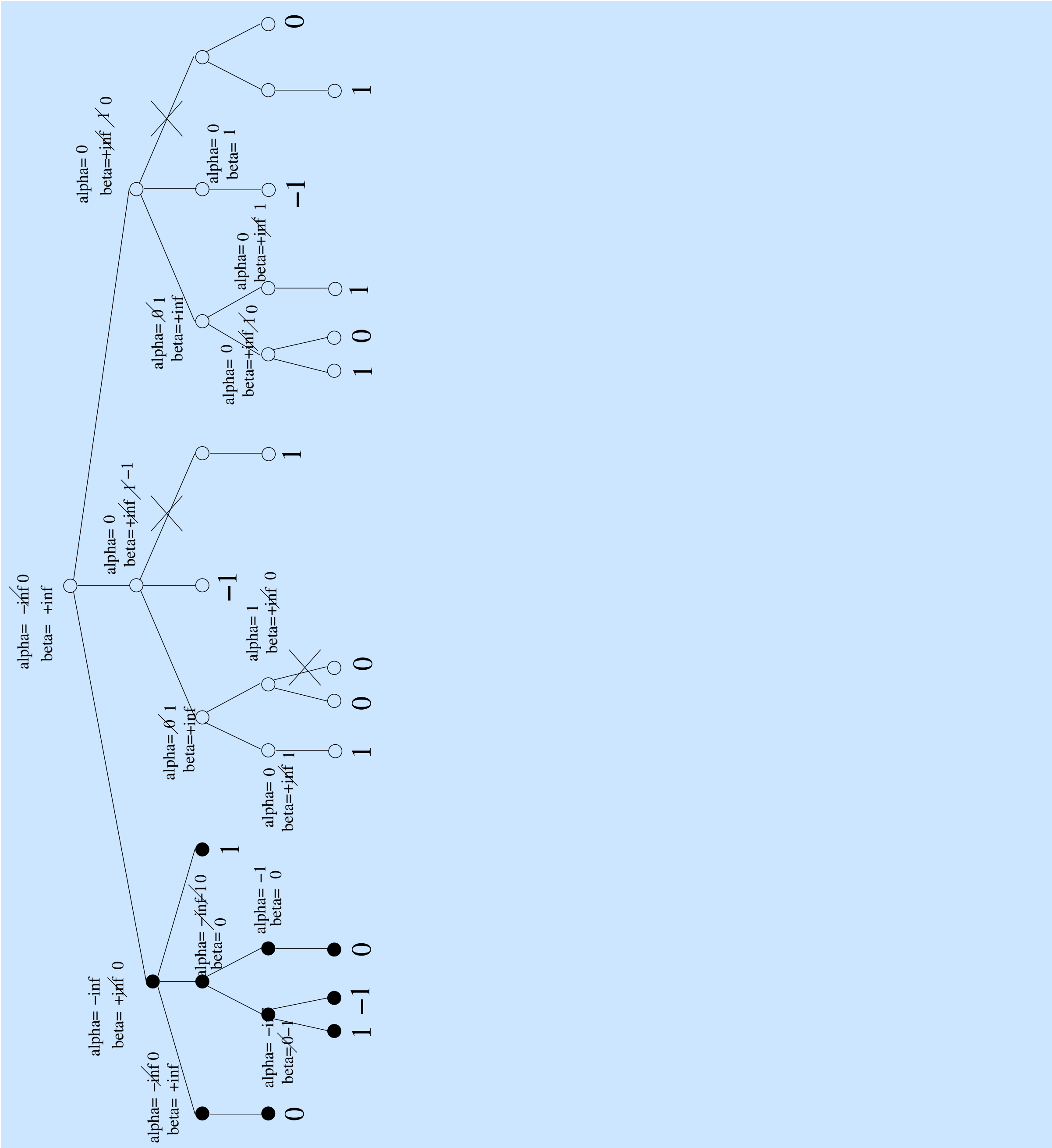
FAUX. Un nœud max retourne à son père la valeur de α qu'il a mise à jour si nécessaire, et nœud min retourne à son père la valeur de β qu'il a mise à jour si nécessaire.

Exercice 2. Mise en œuvre.

1. Poursuivez le calcul de l'algorithme α-β sur l'arbre page suivante. Le premier sous arbre (les nœuds en noirs) a été entièrement exploré, et le nœud courant est la racine. Notez avec soin les valeurs successives de α et β, ainsi que les branches élaguées, en expliquant bien le déroulement de l'algorithme.

Combien de noeuds en moins sont-ils explorés par rapport à l'algorithme minimax ?

On explore 7 nœuds en moins (voir dessin).



2. L'ordre d'exploration des noeuds joue un rôle important pour le temps de calcul de l'algorithme α-β.

   i. En théorie, quel serait l'ordre d'exploration des noeuds qui permettrait que l'algorithme α-β élague le plus de branches possibles ?

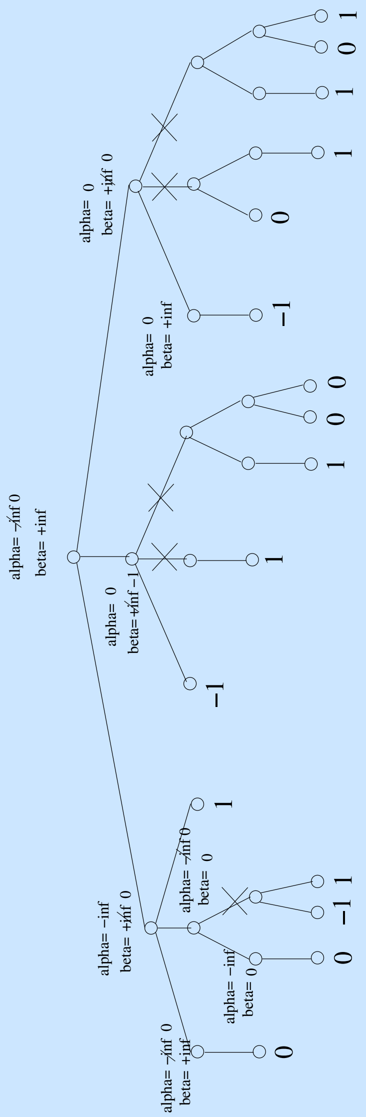
   Pour chaque nœud max de l'arbre, il faudrait que son premier fils soit celui pour lequel minimax renvoie la plus grande valeur.

   Et pour chaque nœud min de l'arbre, il faudrait que son premier fils soit celui pour lequel minimax renvoie la plus petite valeur.

   Évidemment, ce n'est pas utilisable en pratique, parce-que si on savait à l'avance quel est le nœud qui renvoie la valeur optimale avec minimax, on n'aurait pas besoin de α-β.

   ii. Ordonnez les feuilles de l'arbre des questions précédentes de manière que l'algorithme α-β élague le plus de branches possibles. Représentez l'arbre de jeu obtenu.

Combien de noeuds en moins sont-ils explorés par rapport à l'algorithme minimax ?



On explore 21 nœuds en moins (voir dessin).

Exercice 3. Antoine Gombaud, Chevalier de Méré

On considère le jeu de dés suivant. On peut choisir de jouer l'une des deux variantes suivantes :

- (jeu à 1 dé) on lance un dé 4 fois de suite, on gagne si on obtient au moins une fois un as ; ou bien,
- (jeu à 2 dés) on lance deux dés 24 fois de suite, et on gagne si on obtient au moins une fois un double as (*snake eyes*).

En cas de victoire, on gagne un Louis d'or. En cas de défaite, on perd un Louis d'or.

1. Déterminez la meilleure stratégie à adopter.

- Probabilité de victoire au jeu à 1 dé : $1 - (\frac{5}{6})^4 \approx 0,52$.
- Pour le second jeu : $1 - (\frac{35}{36})^{24} \approx 0,49$.

2. Déterminez l'arbre de jeu et en redéduire une stratégie optimale.

cf. correction en td.

3. On change maintenant les gains, on joue pour 10 Louis d'or. Que se passe-t-il ?

ça ne change rien puisqu'on a multiplié les gains précédents par 10 (fonction linéaire positive). On s'en rend compte facilement en refaisant l'arbre de jeu.

4. On change à nouveau les gains : au jeu à 1 dé on joue pour deux Louis d'Or et au jeu à 2 dés, on perd 20 Louis d'or ou bien on en gagne 21. Quelle est la stratégie optimale dans ce cas ?

On calcule les valeurs avec ExpectMinimax.
- Le jeu à un dé vaut $(-2) \times 0,48 + (+2) \times 0,52 = +0,08$.
- Celui à deux dés vaut $(-20) \times 0,51 + (+21) \times 0,49 = +0,09$.

Il vaut mieux choisir le jeux à deux dés dans ce cas (l'espérance de gain est légèrement plus élevée).

Exercice 4. Puissance 2 : un jeu hasardeux

On considère le jeu de puissance 2 (une variante du célèbre jeu Puissance 4) : deux joueurs jouent alternativement leur pions (X ou O) en les laissant tomber le long des colonnes d'une grille. Un joueur gagne si il arrive à aligner 2 pions, soit horizontalement, soit verticalement, soit en diagonale.

1. On considère tout d'abord la grille 2×2. Déterminez si l'un des joueurs à une stratégie gagnante.

Clairement le premier joueur.

On considère une variante dans laquelle on introduit du hasard. Le joueur ne peut plus choisir sa colonne, une colonne s'ouvre au hasard (de manière équiprobable) par contre le joueur peut sélectionner l'un des deux types de pions X ou 0. Un joueur n'est plus déterminé par son pion et gagne si il aligne 2 pions identiques.

2. Déterminez la stratégie optimale du premier joueur pour la grile 2×2.

ExpectMinimax retourne -1 donc le premier joueur est sûr de perdre si le second joueur joue de manière optimale (cf td pour les détails de l'arbre de jeu).

3. Même question pour une grille à 3 colonnes et 2 lignes.

ExpectMinimax retourne $-\frac{1}{9}$. Si les deux joueurs jouent de manière optimale, alors le premier joueur va, en moyenne, perdre (cf td pour les détails de l'arbre de jeu).

Exercice 5. On considère un jeu minimax sans hasard pour lequel le second joueur a une stratégie gagnante. Sachant que vous jouez en premier et que vous savez que l'adversaire et un idiot et qu'il joue chaque coup au hasard, que pouvez vous faire pour maximiser vos chances de victoires ? Donnez un exemple pour illustrer votre démarche.

On se réduit à un jeu avec seulement 2 types de nœuds, Max et Chance, en remplaçant chaque nœud Min de l'arbre de notre jeu minimax par un nœud Chance. Si on a de la chance, la stratégie gagnante est difficile à trouver. (cf. TD pour les détails d'un exemple).