

METHODES DE RECHERCHE POUR LES JEUX

1 Position du problème

La recherche dans le contexte du jeu se caractérise par le fait que, contrairement au cas courant des Systèmes de Résolution de Problèmes (SRP), le planificateur ne dispose pas de la maîtrise complète de l'enchaînement des opérateurs, puisque des décisions extérieures à lui sont prises par l'adversaire.

On peut imaginer - et il arrive en effet - que des SRP doivent élaborer un plan dans un univers "non inerte" (ex : contrôle de processus,...). Toutefois, les événements extérieurs qui peuvent survenir ne sont pas systématiquement ou délibérément hostiles à l'exécution du plan. Et s'ils le sont, il n'y a pas en général de similitude entre les événements du milieu extérieur et les actions du planificateur : celui-ci commande de façon constructive alors que le milieu extérieur évolue et atteint des états de façon destructive.

Dans le contexte du jeu, sur un terrain parfaitement formalisé, deux planificateurs antagonistes construisent l'évolution d'une situation avec des objectifs opposés. Cette symétrie fournit en fait une simplification considérable : le planificateur peut envisager les décisions de son adversaire comme identiques aux siennes, à l'inversion du point de vue près.

2 L'algorithme MINIMAX

Sauf dans les cas élémentaires, les jeux présentent une diversité de combinaisons ce qui empêche très largement toute possibilité de calcul exhaustif. Le choix doit donc être effectué en se projetant à l'horizon qu'on sait calculer (que ce soit 1,2,...,6 coups ou davantage) et en choisissant le coup qui oriente le jeu vers la meilleure des configurations envisagées.

Leur classement suppose l'existence d'une mesure : les algorithmes sont donc basés sur une *fonction d'évaluation* qui attribue à chaque position une note, caractérisant sa qualité pour le joueur : plus le résultat de la fonction est élevé, meilleure est la position ; pour les positions gagnantes, la fonction est alors à son maximum. Il est clair que pour le joueur adverse, l'interprétation des valeurs est opposée.

L'alternance des coups joués par le planificateur et par son adversaire inverse la logique de la décision à chaque niveau de l'arbre. Si le niveau du noeud de départ (symbolisant l'état courant) est numéroté à 0, à ce noeud et à tous les noeuds de rang pair, le planificateur doit choisir le coup qui maximise la fonction d'évaluation : pour cette raison, ces noeuds sont dits noeuds MAX. En revanche, à chaque étape intermédiaire, le choix du coup à jouer sera effectué par l'adversaire, donc l'intérêt est de minimiser la fonction : à ces noeuds de rang impair, le programme doit donc s'attendre à voir choisir le résultat minimum de la fonction d'évaluation sur les coups possibles (noeuds MIN).

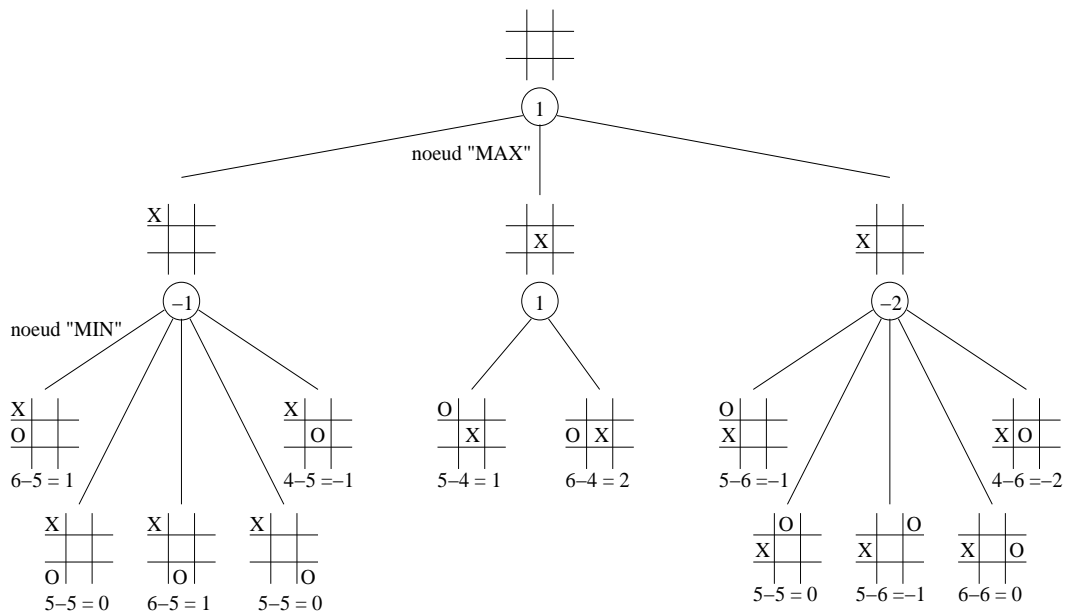


FIGURE 1 – Arbre de recherche – MINIMAX

Pour permettre une meilleure compréhension de cet algorithme, appuyons-nous sur l'exemple concret d'un cas de jeu : le *tic-tac-toe* (Fig 1). Dans le cadre de cet exemple, MOI et LUI étant nos deux joueurs (on tente de faire gagner MOI), on définit la fonction d'évaluation $e(p)$ de la position p telle que :

- si p n'est pas gagnante, $e(p) = (\text{nombre de colonnes, lignes et diagonales restant ouvertes à MOI}) - (\text{nombre de colonnes, lignes et diagonales restant ouvertes à LUI})$,
- si MOI a gagné, $e(p) = +\infty$,
- si LUI a gagné, $e(p) = -\infty$.

Le calcul commence en bas de l'arbre développé, aussi loin que possible dans le temps imparti : tous les nœuds terminaux (ou feuilles) sont évalués par application de la fonction d'évaluation.

Ensuite, si les nœuds immédiatement au-dessus des feuilles sont des nœuds correspondant à un coup joué par LUI (comme c'est le cas sur la figure 1), pour chacun d'entre eux, sa valeur est calculée non plus par application de la fonction d'évaluation, mais en remontant la plus faible (Min) des valeurs des feuilles qui lui sont rattachées (on se prépare au pire...).

A l'étape au-dessus, chacun des nœuds correspond donc à un coup joué par MOI (alternance des coups), et leur valeur est celle du plus élevé des descendants immédiats (on choisit le meilleur coup à jouer!). Et ainsi de suite jusqu'au sommet de l'arbre, où le choix est dirigé vers celui des coups possibles dont la valeur remontée est la plus forte.

3 Le formalisme NEGMAX

La symétrie soulignée précédemment peut être exploitée pour uniformiser la logique de décision, considérant que l'opposé de la fonction d'évaluation constitue la fonction d'évaluation de l'opposant. Ainsi, dans le formalisme NEGMAX, chaque niveau inverse la fonction d'évaluation e telle que :

- $E(n) = (-1)^r e(n)$, si n est une feuille et si r est son rang (compté à partir de 0 pour le nœud de départ),
- sinon, $E(n) = \max(-E(n_1), \dots, -E(n_k))$ si n_1, \dots, n_k sont les successeurs de n .

La figure 2 montre ce calcul appliqué à l'exemple de la figure 1.

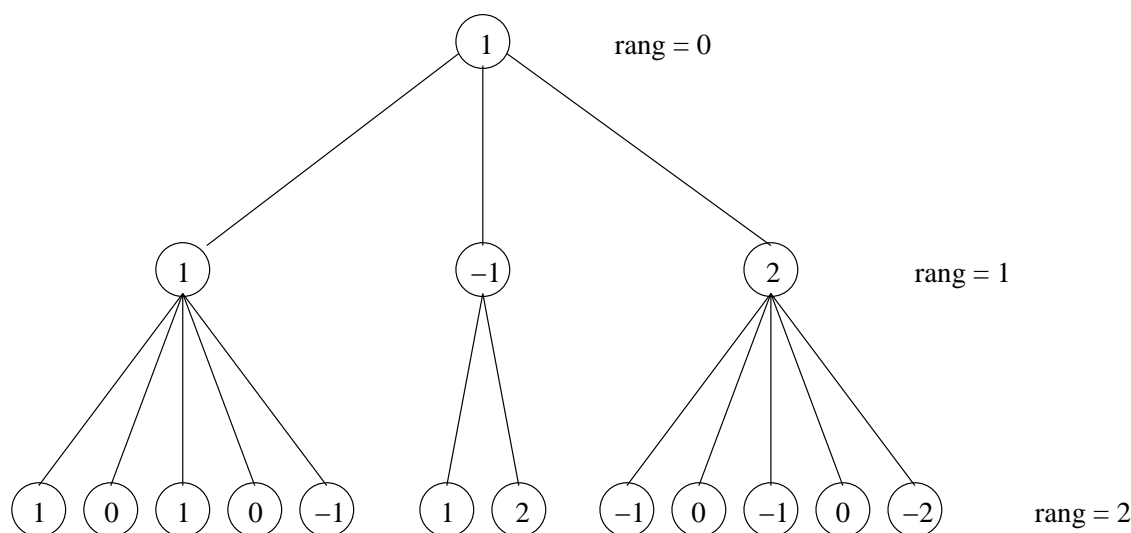


FIGURE 2 – Arbre de recherche – NEGMAX

4 Allègement de la recherche MINIMAX par le système “alpha-bêta”

Dans toute la procédure, nous avons imaginé que toutes les feuilles étaient évaluées, puis leur valeur remontée le long des branches. En réalité, un système simple permet de réduire la quantité d'évaluations, qui représente la charge principale de calcul et donc de coût de la recherche.

Ce système est connu sous le nom de “alpha-bêta” (nom historique donné aux variables dans la première réalisation). Il est basé sur le fait que, dès les premières évaluations, on possède une valeur du maximum qu'il est possible de réaliser à chaque étage de l'arbre. Cette valeur n'est pas forcément définitive, mais pour un nœud MAX par exemple, toute branche conduisant à coup sûr à une valeur inférieure peut être éliminée avant même d'être complètement exploré.

La procédure “alpha-bêta” se définit donc ainsi :

- Aux nœuds MAX est associée une variable α .
- Aux nœuds MIN est associée une variable β .
- α et β contiennent à chaque instant le meilleur résultat calculé (au sens du nœud considéré) qui peut être atteint. Par construction, α est non décroissante et β est non croissante.
- Dès que, sur un nœud MAX, α devient supérieur ou égal à β sur l’un de ses descendants MIN, la recherche peut être interrompue pour ce nœud.
- Inversement, dès que, sur un nœud MIN, β devient inférieur ou égal à α sur l’un de ses descendants MAX, la recherche peut être interrompue pour ce nœud.

Sur l’exemple du *tic-tac-toe* de la figure 1, si par chance ou par tout autre moyen légal (une heuristique par exemple), le nœud le plus intéressant avait été développé en premier, le développement aurait été celui de la figure 3, soit 4 feuilles développées sur 12.

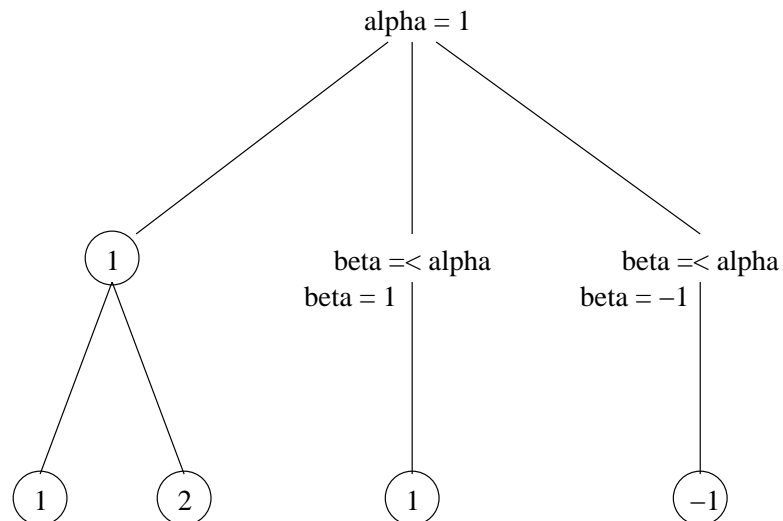


FIGURE 3 – Arbre de recherche – ALPHA-BETA

Il est important de noter que le procédé “alpha-bêta” n’est pas une heuristique : les nœuds éliminés le sont en toute rigueur, avec la certitude qu’ils n’auraient pas apporté de solutions plus intéressantes que celles qui sont déjà connues.