



Théorie des jeux



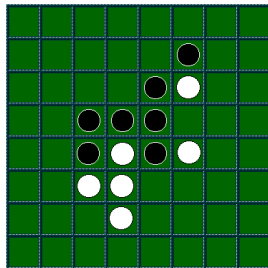
F. Bouquet

Master S&T - Mention Informatique

UNIVERSITÉ de
FRANCHE-COMTÉ

Première année







Minimax ou Négamax

- ▶ États, transitions : Arbre ou Graphe
- ▶ Deux joueurs \nRightarrow Algorithmes type A^*

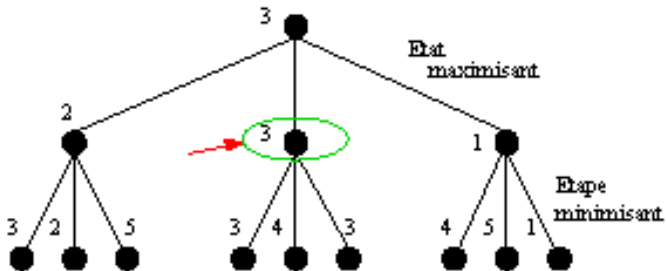
Composants :

- ▶ Fonction d'évaluation
- ▶ Profondeur de recherche
- ▶ Algorithme Minimax
 \Rightarrow Profondeur d'abord.

Exemple Minimax ou Négamax



- Vision double : Deux joueurs
- Évaluation par les feuilles



Propriétés Minimax



- ▶ *Complet* : Oui, si arbre de jeu fini
- ▶ *Optimal* : Oui, si adversaire est aussi optimal
- ▶ *Complexité en temps* : $O(b^m)$
- ▶ *Complexité en espace* : $O(bm)$

b nombre de branchements

m nombre de coups

$$\alpha - \beta$$



- ▶ Optimisation de Minimax
- ▶ Ne pas développer un nœud qui ne sera pas choisi :
 - ▶ Chaque nœud max garde la trace α équivalente à la valeur du meilleur successeur
- ⇒ on arrête l'exploration si α (courant) $\geq \beta$ (venant du parent)
- ▶ Chaque nœud min garde la trace β équivalente à la valeur du pire successeur
- ⇒ on arrête l'exploration si β (courant) $\leq \alpha$ (venant du parent)



Algorithme AlphaBeta

Algorithme 1 : Algorithme : AlphaBeta(e,p, α , β , MaxJoueur)

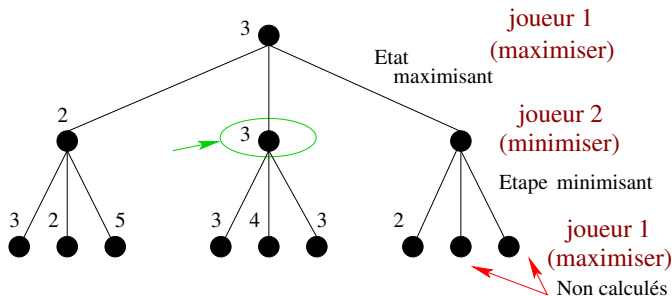
```

Input : e nœud
Input : p profondeur
Input : [ $\alpha$ ,  $\beta$ ] borne
Input : MaxJoueur Vrai si nœud type max
if (p = 0) ou (n est terminal) then return h(n);
else
    if MaxJoueur then
        f ← PremierFils(n) ;
        while (f ≠ ⊥) et ( $\alpha < \beta$ ) do
             $\alpha \leftarrow \max(\alpha, \text{AlphaBeta}(f, p - 1, \alpha, \beta, \text{faux}))$  ;
            f ← FilsSuivant(n) ;
        return  $\alpha$  ;
    else
        f ← PremierFils(n) ;
        while (f ≠ ⊥) et ( $\alpha < \beta$ ) do
             $\beta \leftarrow \min(\beta, \text{AlphaBeta}(f, p - 1, \alpha, \beta, \text{vrai}))$  ;
            f ← FilsSuivant(n) ;
        return  $\beta$  ;

```

Exemple *AlphaBeta*

- Appel : $\text{AlphaBeta}(\text{Racine}, 4, -\infty, +\infty, \text{true})$
- Au mieux parcours $2\sqrt{N}$ vs N



Exemple - Othello

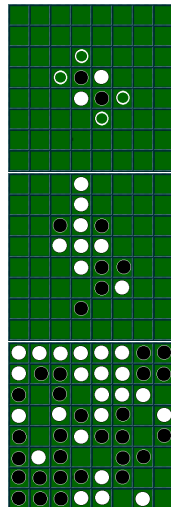
Borné à 60 demi-coups

► Phases de jeu :

- Début $\simeq 4$ demi-coups : Ouverture prés définie
- Milieu : Valeur tactique variable

500	-150	30	10	10	30	-150	500
-150	-200	0	0	0	0	-200	-150
30	0	1	2	2	1	0	30
10	0	2	16	16	2	0	10
10	0	2	16	16	2	0	10
30	0	1	2	2	1	0	30
-150	-200	0	0	0	0	-200	-150
500	-150	30	10	10	30	-150	500

- Fin $\simeq 15$ demi-coups : Calcul par différence



Exemple - Échec

10 à +100 demi-coups

► Phases de jeu :

- Début $\simeq 6$ demi-coups :

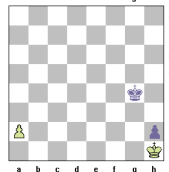
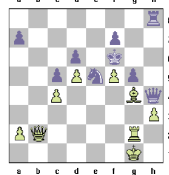
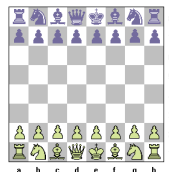
Ouverture prés définie

- Milieu :

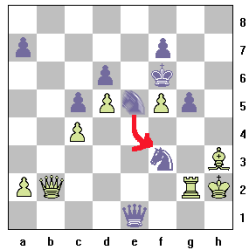
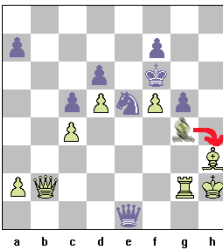
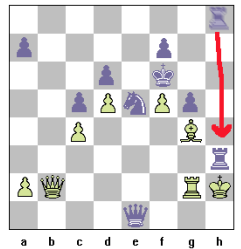
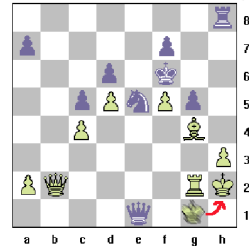
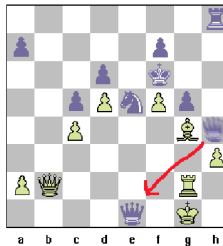
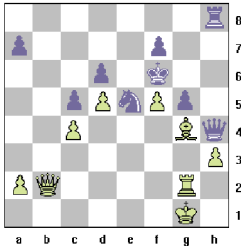
Valeur tactique

Dame	Tours	Fou	Cavalier	Pion
9	5	3	3	1

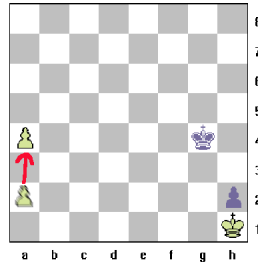
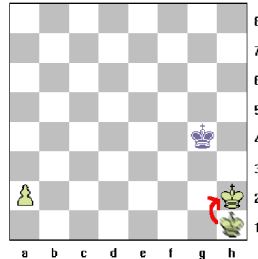
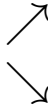
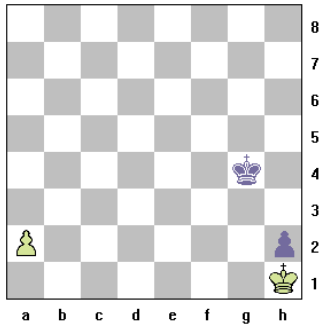
- Fin $\simeq 8$ pièces



Effet de bloc



Effet d'horizon



Contrée



	Annonce	80	90	100	110	120	130	140...
► Annonces :	Jeux	2As, V/9	2As, V&9	MA 4C	MA 3C	MA 2C	1 pli	

Ajout : +10 pour 9, +20 pour Valet, +10 par As, +10 par dix second ou filante...

► La partie :

- Règles : Si aller alors Jouer atout, Jouer ses as
- Probabilité : Déduction lors des annonces (Coefficient mis à jour pendant le jeu)