Étude des principes de l'algorithme Alpha-bêta

C'est très bien sur la structure et les renvois, ainsi que su la "pédagogie" des explications.
Certaines explications restent un peu opaques (notamment le "on voit sur la ligne min" avec les valeurs 5 et 4, j'avoue ne pas trouver évident ce qu'il faut voir

Il est **important** de connaître le fonctionnement de l'algorithme **Min Max**. Il est nécessaire de consulter la page Algorithme MinMax

---

Fonctionnement

Une coupure **Alpha-Bêta** est le procéder amélioré de l'algorithme **Min Max**. Cette amélioration consiste en l'élimination (immédiat) de solutions inutile par « déduction » afin d'améliorer la réactivité de l'IA.

L'algorithme de l'**Alpha-bêta** est exactement le même que celui du **Min Max**. Les seules différences sont :

Il n'utilise pas les variables **Min** et **Max** avec des valeurs minimales et maximales. A la place, il utilise les valeurs d'**Alpha** et de **Bêta**.

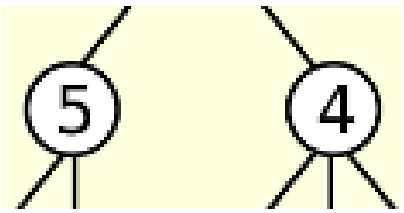
- La variable **Alpha** est mise à jour dans les nœuds max.
- La variable **Bêta** est mise à jour dans les nœuds min.
- Nous arrêtons le parcours si la variable **Alpha** se retrouve supérieure à **Bêta**.

Exemples

1er exemple

Le graphique se lit de gauche à droite

Dans cette exemple on peut voir que dans la première ligne **Min** en partant du bas, que les deux nombres choisi sont le chiffre **5** (pour un choix entre **5** et **6**) et **4** (entre **7**, **4**, **5**).

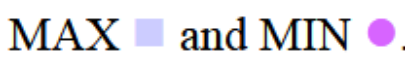


Mais pourquoi l'algorithme ne vérifie pas la valeur **5** ? Tout simplement car la valeur **4** est inférieur à

5, il est donc inutile pour l'algorithme de continuer.

L'algorithme vérifie par rapport a la branche adjacente pour voir si il a besoin de continué à chercher. Ce procédé permet de gagner du temps.

2ème exemple



Dans cette exemple ci-dessus, nous avons en chiffre de référence **6**, il faudra ici remplir la condition suivante : le résultat doit être égal ou supérieur à **6**. Si cette condition est rempli par un des chiffres alors l'algorithme n'explorera par d'avantage une branche. Par exemple si on prends la dernières branche (tout à droite) on voit que **7** est supérieur à **6**, donc la condition est rempli, il est donc futile de continuer à explorer cette branche car la condition est déjà rempli.

Il est possible d'améliorer les performances de l'élagage **Alpha-bêta** comme par exemple trier les valeurs par ordre croissant ou décroissant afin d'augmenter les nœuds qui ne sont pas examinés. Cela dépends du résultat souhaiter.

Cela permet un gain de temps. L'**Alpha-bêta** est une amélioration du **Min Max**, cette optimisation permet d'avoir de bonnes performances.

Alpha-bêta est largement utilisé par les programmes qui jouent aux jeux de réflexion.

Pour des explications plus approfondies

PDF Wikipédia