

# **Implantation d'une fonction d'estimation basé sur Tables pièce-carré « jeu d'échecs »**

BERKANE IBRAHIM

BENNACEUR HICHEM -SIL2-

## Table of Contents

La fonction d'estimation d'algorithme Min-Max de jeux d'echec :.....	2
Exemple concret : .....	3
Déroulement de Piece Square Table : .....	3
Remarque et Dèduction : .....	4
Remarque : .....	4
Dèduction : .....	4

## La fonction d'estimation d'algorithme Min-Max de jeux d'echec :

### Tables pièce-carré(Piece Square Table) :

un moyen simple d'attribuer des valeurs à des pièces spécifiques sur des carrés spécifiques . **Un tableau est créé pour chaque pièce de chaque couleur et des valeurs sont attribuées à chaque carré.** Ce schéma est *rapide*, car le terme d'évaluation des tables carrées des pièces peut être mis à jour de manière incrémentielle à condition que **les déplacements sont effectués et défaits dans l' arbre de recherche** . En raison de cette vitesse, les tables pièce par pièce sont d'une grande aide lors de la conduite **d'une évaluation paresseuse** .

La même technique peut être utilisée pour des termes d'évaluation plus subtils, au lieu d'une valeur fixe pour, par exemple, un pion isolé . Il est également possible d'utiliser des tables de pièces carrées pour l'ordre des déplacements , bien que **l'heuristique historique** ait tendance à mieux fonctionner. Certains programmes, comme Rebel , utilisent la combinaison des deux .

Dans une table des pièces carrées on a : « une valeur est attribuée à chaque carré du tableau pour chaque pièce. Une valeur plus élevée pour le carré signifie une meilleure position générale pour cette pièce ».

Ces valeurs sont pour le blanc, et pour le noir on utilise des valeurs en miroir. De plus, nous devons définir où commence la fin car les valeurs de la table change.

On considère la fin comme suit:

- ✓ Les deux côtés n'ont pas de reines ou
- ✓ Chaque côté qui a une reine n'a en outre pas d'autres pièces ou un maximum de pièces mineures.



[-3.0, -4.0, -4.0, -5.0, -5.0, -4.0, -4.0, -3.0],
[-3.0, -4.0, -4.0, -5.0, -5.0, -4.0, -4.0, -3.0],
[-3.0, -4.0, -4.0, -5.0, -5.0, -4.0, -4.0, -3.0],
[-3.0, -4.0, -4.0, -5.0, -5.0, -4.0, -4.0, -3.0],
[-2.0, -3.0, -3.0, -4.0, -4.0, -3.0, -3.0, -2.0],
[-1.0, -2.0, -2.0, -2.0, -2.0, -2.0, -2.0, -1.0],
[2.0, 2.0, 0.0, 0.0, 0.0, 0.0, 2.0, 2.0],
[2.0, 3.0, 1.0, 0.0, 0.0, 1.0, 3.0, 2.0]



[-2.0, -1.0, -1.0, -0.5, -0.5, -1.0, -1.0, -2.0],
[-1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -1.0],
[-1.0, 0.0, 0.5, 0.5, 0.5, 0.5, 0.0, -1.0],
[-0.5, 0.0, 0.5, 0.5, 0.5, 0.5, 0.0, -0.5],
[0.0, 0.0, 0.5, 0.5, 0.5, 0.5, 0.0, -0.5],
[-1.0, 0.5, 0.5, 0.5, 0.5, 0.5, 0.0, -1.0],
[-1.0, 0.0, 0.5, 0.0, 0.0, 0.0, 0.0, -1.0],
[-2.0, -1.0, -1.0, -0.5, -0.5, -1.0, -1.0, -2.0]



[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
[0.5, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 0.5],
[-0.5, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -0.5],
[-0.5, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -0.5],
[-0.5, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -0.5],
[-0.5, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -0.5],
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.5, 0.5, 0.0, 0.0, 0.0]



[-2.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -2.0],
[-1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -1.0],
[-1.0, 0.0, 0.5, 1.0, 1.0, 0.5, 0.0, -1.0],
[-1.0, 0.5, 0.5, 1.0, 1.0, 0.5, 0.5, -1.0],
[-1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, -1.0],
[-1.0, 0.5, 0.0, 0.0, 0.0, 0.0, 0.5, -1.0],
[-2.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -2.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]



[-5.0, -4.0, -3.0, -3.0, -3.0, -3.0, -4.0, -5.0],
[-4.0, -2.0, 0.0, 0.0, 0.0, 0.0, -2.0, -4.0],
[-3.0, 0.0, 1.0, 1.5, 1.5, 1.0, 0.0, -3.0],
[-3.0, 0.5, 1.5, 2.0, 2.0, 1.5, 0.5, -3.0],
[-3.0, 0.0, 1.5, 2.0, 2.0, 1.5, 0.0, -3.0],
[-3.0, 0.5, 1.0, 1.5, 1.5, 1.0, 0.5, -3.0],
[-4.0, -2.0, 0.0, 0.5, 0.5, 0.0, -2.0, -4.0],
[-5.0, -4.0, -3.0, -3.0, -3.0, -3.0, -4.0, -5.0]



[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
[5.0, 5.0, 5.0, 5.0, 5.0, 5.0, 5.0, 5.0],
[1.0, 1.0, 2.0, 3.0, 3.0, 2.0, 1.0, 1.0],
[0.5, 0.5, 1.0, 2.5, 2.5, 1.0, 0.5, 0.5],
[0.0, 0.0, 0.0, 2.0, 2.0, 0.0, 0.0, 0.0],
[0.5, -0.5, -1.0, 0.0, 0.0, -1.0, -0.5, 0.5],
[0.5, 1.0, 1.0, -2.0, -2.0, 1.0, 1.0, 0.5],
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]

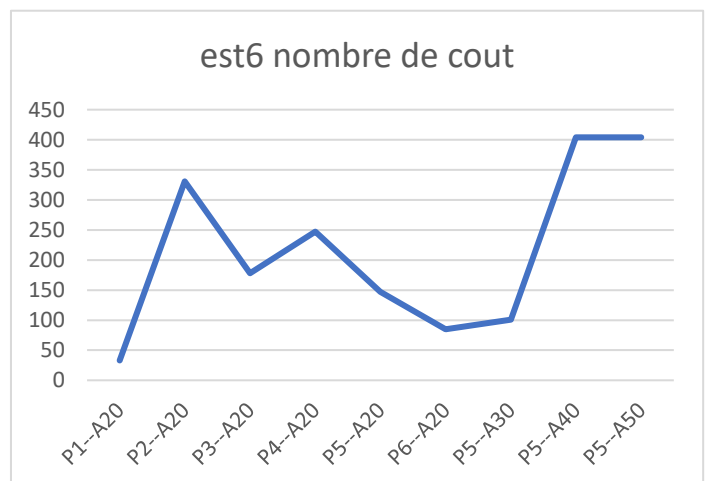
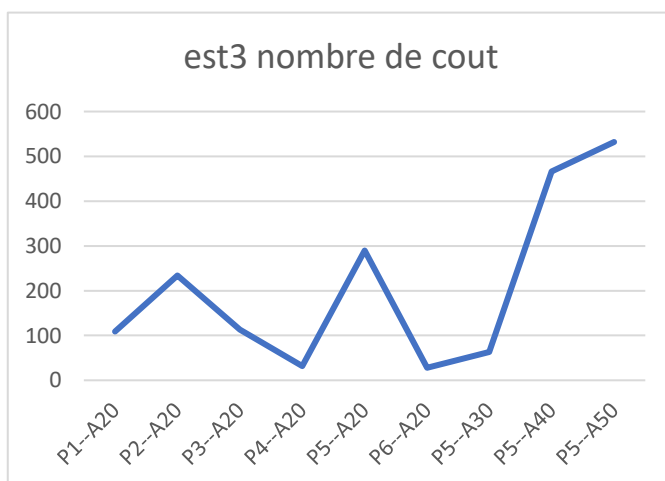
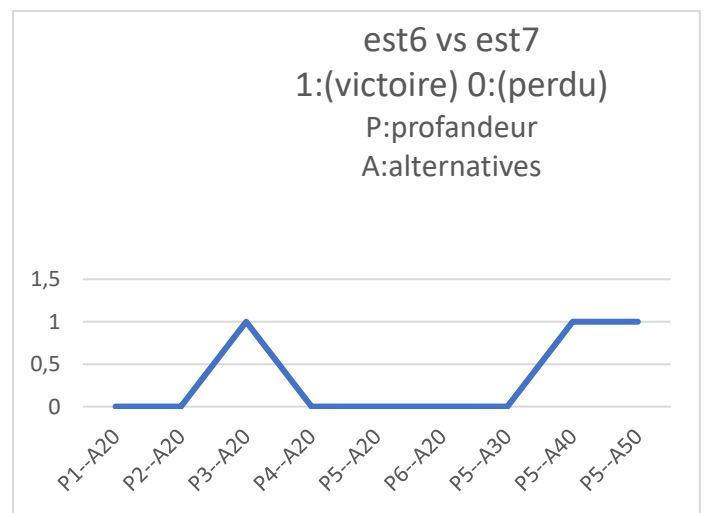
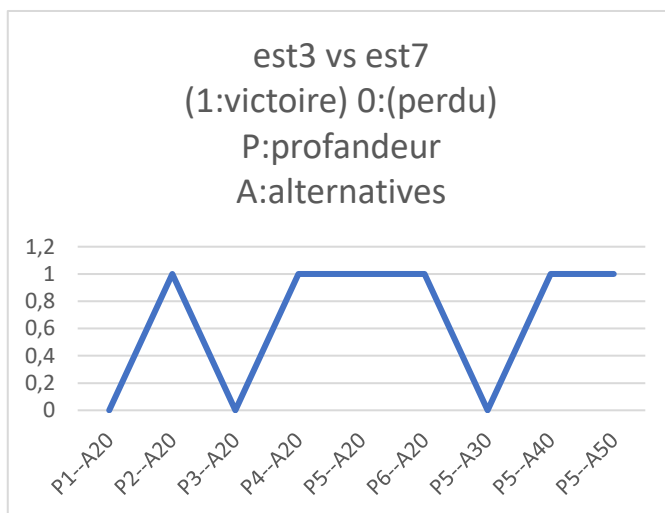
## Exemple concret :

Afin de visualiser la création des tables voici un exemple concret dans un fichier excel. Cela suppose que votre représentation du conseil commence par A1 = 0 et H8 = 63.

Voici comment utiliser la feuille de calcul Excel:

- La feuille de calcul est protégée pour éviter les écrasements accidentels. Vous pouvez le déverrouiller facilement car il n'est pas protégé par mot de passe
- Entrez les valeurs dans les carrés jaunes à gauche. Les carrés gris sont l'image miroir des carrés jaunes.
- Le tableau du milieu est ajusté pour rendre la valeur moyenne du tableau nulle. Cela signifie que si je fixe la valeur d'un évêque à 350 centi-pions et que je la place sur un carré aléatoire, la valeur renvoyée par le tableau des pièces sera en moyenne de 350 centi-pions. L'idée est de minimiser l'interaction avec d'autres termes d'évaluation.
- La colonne à l'extrême droite est le tableau formaté comme un tableau «C». Vous pouvez copier et coller ce code.

## Déroulement de Piece Square Table :



## Remarque et Dèduction :

### Remarque :

On remarque que :

- à partir du profondeur 5 et nombre alternatives 40, l'estimation 7 est meilleur que l'estimation 3 et 6, et le nombre de coût est augmentée.
- L'augmentation de la profondeur seul n'affecte pas l'estimation 7
- Afin de donner des meilleurs résultats, il faut augmenter le nombre d'alternative.

### Dèduction :

Donc pour l'estimation 7 la profondeur ne doit pas être inférieure à 5 et le nombre d'alternatives doit être supérieur ou égale à 40 pour nous donner le meilleur résultat:

*Condition d'efficacité:  $p \geq 5$  et  $A \geq 40$*