Rapport du Projet d’Intelligence Artificiel

Création d’un Joueur de Tablut

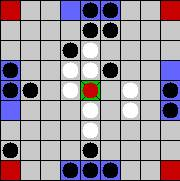


Table des matières

Description du Projet : 3

1. Modélisation du jeu 4

1.1. Modélisation du plateau 4

Description du Projet :

Le but de ce projet est la création d’un joueur artificiel capable de jouer (et même gagner) au Tablut d’Hnefetafl.

Pour ce faire, nous avons créé une structure modélisant le plateau de jeu, implémenté différentes méthodes de descente permettant de prévoir nos coups avec un maximum de tours de jeu d’avance et enfin une heuristique indiquant à la vue d’un plateau si le jeu est plutôt en notre faveur ou en la faveur de l’ennemi.

La modélisation du plateau consiste bien sûr à créer un simple tableau en deux dimensions de 9 cases sur 9. Cependant, il faut surtout prendre en compte la modélisation des pions sur le plateau. Si physiquement on considère qu’un pion est sur une case, plusieurs modélisations existent en langage objet. Est ce le pion qui appartient à la case ? Ou la case est elle simplement un attribut du pion ? Comment accéder aux pions ? Faut il parcourir tout le plateau ?

Pour les méthodes de descente, il nous a suffit de reprendre les algorithme vu en cours et de choisir celui qui nous semblait le plus adapté à la situation.

Enfin pour l’heuristique, tout a dût être créé. A quoi faut il donner de l’importance ? Faut il créer plusieurs heuristique en fonction du déroulement du jeu (début, milieu ou fin de partie) ? Faut il créer différentes heuristique en fonction de la couleur jouée (tactique plus ou moins agressive, anti jeu possible) ?

Tant de question auxquelles nous allons tenter de répondre dans ce rapport de projet IA, bien que toutes nos idées n’aient malheureusement pas pus être mise en œuvre dans ce projet.

Mais tout d’abord commençons par la modélisation du plateau.

1. Modélisation du jeu

La modélisation du jeu comprend plusieurs parties, la modélisation du plateau (la modélisation des cases, des pions…) et la gestion des mouvements sur le plateau, avec les pions d’une certaine couleur et en prenant en compte les conséquences de ces mouvements.

* 1. Modélisation du plateau

La classe *Plateau* qui comme son nom l’indique modélise le plateau a comme attribut un tableau en deux dimensions de 9 sur 9, contenant des objets *Case*.

Chaque case a pour attribut ses coordonnées (*x* pour abscisse et *y* pour ordonnée).

Le type de case dont il s’agit (*Type* a pour valeur une des valeur de l’énumération *TypeCase*. *MUR* lorsqu’il s’agit d’un des quatre coins inaccessibles pour tous les pions, *TRONE* lorsqu’il s’agit du trône du roi, la case centrale, *NORMAL* lorsque la case est accessible par tous et enfin *POUBELLE* qui est un type « virtuel » ou les pions mangés sont placés.)

Chaque case a aussi pour attribut le pion qu’elle contient (attribut *contenu* de type *Pion*), qui est nul si elle ne contient aucun pion.

Revenons en a la classe *Plateau*. Elle contient aussi deux listes de pions pour chacune des couleurs ainsi que le pion Roi (correspondant respectivement aux attributs *pionsBlanc*, *pionsNoir* et *roi*).

Intéressons nous maintenant à la classe *Pion*. Elle a tout d’abord pour attribut le booléen *mort* indiquant si le pion a été mangé ou non. Un attribut *type* dont la valeur est définie dans l’énumération *TypePion* et dont la valeur peut être *BLANC*, *NOIR* ou *ROI*. Enfin l’attribut *pere* permet au pion de connaître la case sur laquelle il se trouve.

Pour résumer, une case contient des coordonnées et éventuellement un pion. Et un pion possède aussi la case sur laquelle il se trouve.

L’ensemble des cases est contenu dans un tableau à double entrée et l’ensemble des pions est contenu dans deux listes distinctes pour chaque couleur.

Voyons à présent la gestion des mouvements dans le jeu.

* 1. Les mouvements sur le plateau

Tout d’abord il y a un élément indispensable à prendre en compte pour la suite du jeu.

Comme précisé dans l’énoncé, les mouvements transmis par et à l’arbitre sont quatre coordonnées. Les deux premières pour la case de départ et les deux suivantes pour la case d’arrivée. Or ces cases sont numérotées de 1 à 9 pour le plateau « physique » et dans notre modélisation, le tableau contenant le plateau a des cases « numérotées » de 0 à 8. Il est donc important qu’a chaque réception de mouvement, on retire un à chaque coordonnée et qu’à chaque envoie de coordonnées on ajoute 1.

Voyons à présent comment sont générés les mouvements.

Tout d’abord, cela dépend de la couleur à jouer. On parcourt la liste de pions de la couleur à jouer et pour chaque pion, on enregistre les mouvements possibles.

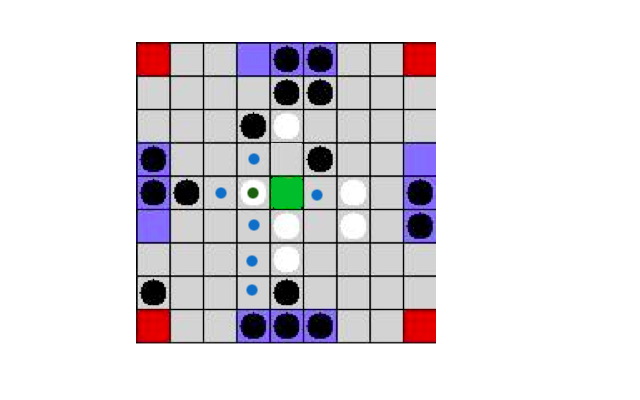
Chaque pion se déplace comme une tour aux échecs. Il a donc à chaque fois quatre directions possibles, le haut, le bas, la droite et la gauche. Dans chacune de ces directions il faudra calculer de combien de case au maximum il peut avancer. Pour ce faire, notre algorithme a été (comme précisé ultérieurement) séparé pour chacune des directions. Voyons son fonctionnement pour le haut.

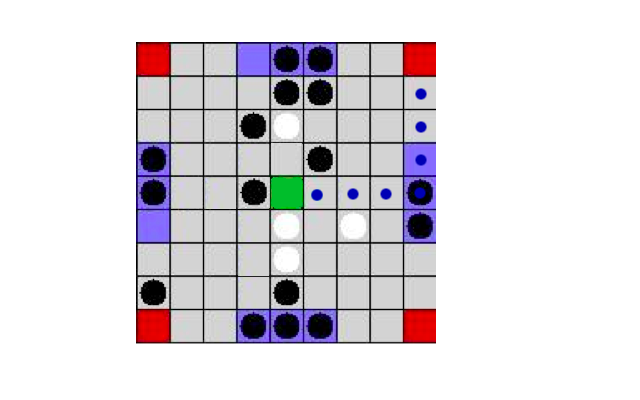
Tout d’abord, il faut partir d’une case qui contient un pion de la couleur à jouer. Puis dans notre cas, on fixe l’abscisse et on va faire évoluer l’ordonnée. Puisqu’il s’agit d’explorer la direction du haut, il va falloir incrémenter l’ordonnée.

On utilise une boucle while et à chaque itération on vérifie que la case est libre. :

* Si elle est libre et que c’est une case *NORMAL* (voir classe *Case*), alors le mouvement jusqu'à cette case est possible. On l’ajoute à la liste des mouvements possible et on passe à la case suivante.
* Si elle est libre et qu’il s’agit de la case *TRONE*, alors on peut continuer le mouvement, cependant on ne peut pas s’arrêter sur cette case. On NE l’ajoute PAS à la liste des mouvements possibles mais on test la case suivante.
* Si la case n’est pas libre, qu’elle contient un pion ou qu’il s’agit d’une case *MUR* on ne peut plus avancer, le chemin est bloqué. On sort de la boucle.
* Si on sort du plateau (dans notre cas on dépasse d’ordonnée 8), on sort automatiquement de la boucle. Le monde est trop vaste à explorer, autant se limiter au plateau !

Voyons les exemples ci dessous (exemple irréaliste puisqu’il n’y a pas de *ROI*).

Dans cet exemple ci-dessus, étudions les déplacements possibles du pion blanc marqué d’un point vert. Ici les cases en bleu sont toutes les case ou il peut se déplacer. La case *TRONE* ne lui est pas accessible, cependant il peut effectuer un déplacement vers la case derrière le trône.



Dans cet exemple ci-dessus, les déplacements possibles du pion *NOIR* marqué d’un point bleu. Le TRONE reste inaccessible ainsi que les cases derrières puisqu’un pion NOIR bloque l’accès. De même le MUR reste inaccessible. Les *MUR* sont donc les quatre cases rouges, aux coins du plateau.

Remarque importante, la case du pion ne compte pas parmi les cases accessibles lors du déplacement. On ne peut pas refuser de bouger pendant un tour, un mouvement est un changement de case, donc pas de fainéantise, il faut que le pion bouge s’il est considéré en mouvement.

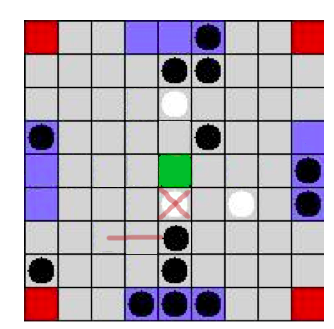
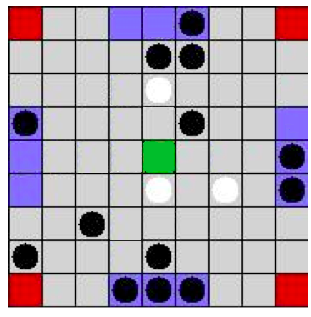
A présent que nous avons vu les mouvements, on peut aussi s’intéresser aux conséquences de ces mouvements, c’est à dire la prise d’un pion adverse. La encore plusieurs cas sont a distinguer. Un pion ne peut être mangé que lorsque :

* Il s’agit du *ROI*, il est entouré par quatre pions adverses ou par trois pions adverses et le *TRONE* (par les cases du haut, du bas, de droite et de gauche et NON par les cases en diagonales, celle-ci n’ont aucune influence).
* Il s’agit d’un pion, il est entouré par deux pions adverses ou par un pion et la case *TRONE* libre (configuration haut-bas ou gauche-droite, encore une fois, les cases en diagonales n’ont aucune influence).

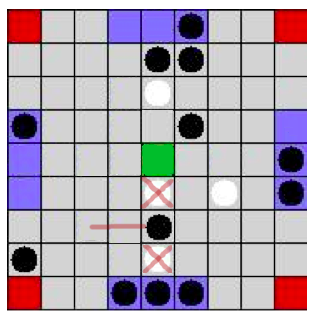
Attention !!!!

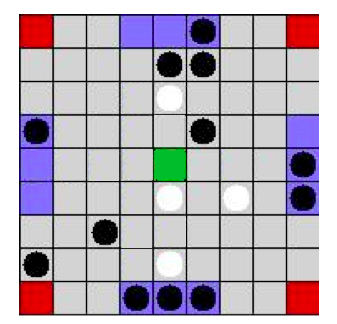
* Il n’y a prise de pion que s’il y a eu déplacement et que le déplacement entraine un des configurations expliquée ci-dessus et qui n’était pas présente avant le déplacement. Pour ces raisons, on regarde les conséquences du mouvement en observant les cases voisines de la case d’arrivée, après déplacement.
* La case *TRONE* contient le *ROI* au début de partie. Dès que ce dernier la quitte, le *TRONE* est libre et le reste jusqu'à la fin de la partie. Plus aucun pion ne pourra l’occuper, il reste cependant traversable pour les mouvements.

Voyons à présent des exemples de prises de pions.

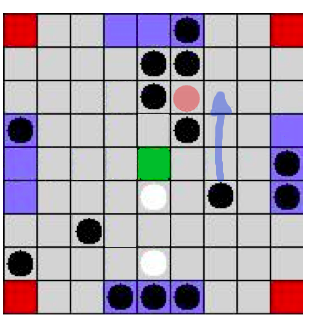


Dans ce cas ci-dessus, le déplacement de pion noir entraine la prise du pion blanc grâce au trône.

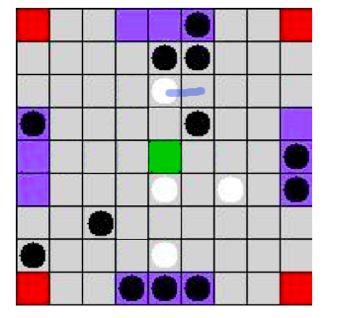




Dans le cas ci-dessus, il y a une double prise !!!



En se déplaçant ici, le pion noir mangera le roi, la partie sera perdue !



En se déplaçant ainsi, le pion blanc ne risque rien. C’est le déplacement d’un pion qui peut entrainer la prise d’un autre pion, par le premier pion. C’est au blanc de jouer, les noirs ne peuvent donc pas manger.