

Compte rendu 3I025 : IA RO Mini-Projet

ANDRES Julien : 3412470

TAYLOR Thomas : 3414621

Intro : Pour toutes ces stratégies , la recherche de chemin vers les fioles, que ce soit pour avoir la longueur du chemin , ou y aller directement , se fait en utilisant l'algorithme a-star.

Tous les tests ont été fait avec 2000 parties et tous les résultats sont disponibles dans l'annexe et dans le dossier /Resultats

IA naïves :

- **naive_pref** : Cette stratégie consiste à aller directement sur les fioles dans l'ordre de préférence (10 points puis 7 puis 2) . Cette stratégie étant vraiment simpliste , on y ajoute le fait que si le joueur passe à côté d'une fiole quelconque (dans un rayon de 1 case) pendant son trajet vers sa fiole cible, il fait un crochet pour la ramasser.
- **naive_court** : Cette stratégie consiste à aller prendre les fioles les plus proches.

IA améliorées :

- **ratio_seul** : Cette stratégie est un mixte des 2 stratégies naïves précédentes. En effet, nous nous sommes rendu compte que la stratégie **naive_court** gagne largement contre **naive_pref**. Mais elle ne prend pas du tout en compte le nombre de point que donne chaque fiole. Donc dans cette stratégie, nous attribuons à chaque fiole un ratio. C'est à dire que pour chaque fiole, nous calculons :

$$ratio = \frac{\text{Nombre de point de la fiole}}{\text{nombre de cases pour y arriver}}$$

et nous prenons la fiole qui a la meilleure valeur. Dans cette stratégie aussi, si le joueur passe à côté d'une fiole quelconque (dans un rayon de 1 case) pendant son trajet vers sa fiole cible, il fait un crochet pour la ramasser.

- **ratio_carre** : Cette stratégie est basée sur la même idée que la précédente, mais au lieu d'aller vers la fiole qui a le meilleur ratio, le joueur va se diriger vers la fiole avec la meilleure concentration de points dans un carré de 5 x 5. Pour être sûr que le nombre de points dans ce carré est bien disponible (c'est à dire que le joueur ne doit pas faire un long détour à cause d'un mur) on vérifie que le chemin le plus long pour aller de la fiole du milieu à une fiole dans le carré est de maximum 4 cases. Le ratio est donc de la forme :

$$ratio = \frac{\text{Nombre de point dans un carré de 5x5}}{\text{nombre de cases pour arriver au milieu du carré}}$$

Et comme les autres stratégies, le joueur prend les fioles proches quand il passe à côté.

IA qui « apprennent » :

- **guess_carre** : Cette stratégie prend en compte le fait qu'il y ait 4 manche : Pendant la première manche, elle va « observer » le joueur adverse et jouer comme « **ratio_carre** » (car c'est la meilleure stratégie pour le moment).

A partir de la 2nd manche, le joueur va essayer de « voler » les meilleures fioles de l'adversaire, basé sur la prédiction qu'il a fait à partir de la 1^{ère} manche :

- si une fiole à 10 points pour l'adversaire est plus proche de lui que du joueur adverse et que ce n'est pas sa pire préférence, alors il va aller la prendre.
- Sinon si une fiole à 7 points pour l'adversaire est plus proche de lui que du joueur adverse et que ce n'est pas sa pire préférence, alors il va aller la prendre.

- Sinon, il joue sa stratégie par défaut, c'est à dire « ratio_carre »

Dans cette stratégie, le joueur va modifier son chemin pour prendre les fioles proches uniquement si la différence du nombre de pas pour aller à la fiole voulu entre sa distance et celle de son adversaire est supérieure à 1. Le joueur va aussi aller vers une fiole voulu s'il est à même distance de la fiole que son adversaire, mais que c'est le joueur 0 (qui avance en premier) : il l'atteindra en effet avant son adversaire.

Les préférences supposées du joueur adverse sont actualisées en fonction des fioles ramassées par l'adversaire tout au long de la partie.

- guess_ratio_seul : Comme la stratégie précédente, cette stratégie va observer adversaire pendant la 1ere manche et jouer comme « ratio_carre ».
A partir de la 2nd manche, cette stratégie va jouer comme « ratio seul » mais a la place de prendre que ses propres points, le joueur va additionner ses propres points et ceux déduits de cette même fiole pour l'adversaire. Le ratio est donc de la forme :

$$ratio = \frac{\text{Nombre de point de la fiole} + \text{nombre de point de cette fiole pour l'adversaire}}{\text{nombre de cases pour y arriver}}$$

Et comme les stratégies précédentes, le joueur modifie son chemin s'il y a une fiole à proximité et les préférences sont mise à jour à chaque manche.

- guess_ratio_carre : Cette stratégie va jouer comme la stratégie précédente mais au lieu de jouer comme « ratio_seul », va jouer comme « ratio_carre ». C'est à dire qu'elle va additionner ses propres points et ceux de l'adversaire dans un rayon de 5x5 autour de la fiole cible, et va attribuer ce ratio à la fiole cible. :

$$ratio = \frac{\sum \text{nombre de point dans carre } 5 \times 5 + \sum \text{nombre de point pour l'adversaire dans carre } 5 \times 5}{\text{nombre de cases pour arriver à la fiole du milieu}}$$

Le joueur va ensuite aller vers la fiole avec les meilleurs ratios. Comme les stratégies précédentes, le joueur modifie son chemin s'il y a une fiole à proximité et les préférences sont mises à jour à chaque manche.

Résultats : On remarque que naive_court est une stratégie dominée, elle n'est pas du tout efficace.

On remarque aussi que le gagnant sur 2000 parties de ratio_carre VS ratio_seul est celui qui est en joueur 0. En effet, le joueur 0 a une itération d'avance sur le joueur 1 et peut donc ainsi récupérer en premier une fiole à égale distance des deux joueurs.

Or la différence de partie gagnée (sur ces deux pools de parties) est plus grande pour ratio_carre, c'est pourquoi nous avons décidé d'utiliser cette stratégie comme stratégie par défaut pour les IA suivantes.

On a donc aussi testé les deux possibilités de guess_ratio_carre VS guess_ratio_seul, et on remarque le même phénomène : le gagnant est le joueur 0, mais la différence de parties gagnées est plus grande quand guess_ratio_carre est joueur 0. On en déduit donc que guess_ratio_carre est un peu mieux que guess_ratio_seul.

Annexe :

$\frac{[X-Y]}{X} \quad Y$	Ratio_seul	Naive_court	Naive_pref	Ratio_carre	Guess_carre	Guess_ratio_seul	Guess_ratio_carre
Ratio_seul	XXX	1267-733	1445-555	1114-886	894-1106	859-1141	850-1150
Naive_court	XXX	XXX	1252-748	651-1349	794-1206	755-1245	745-1255
Naive_pref	XXX	XXX	XXX	511-1789	741-1259	592-1408	556-1444
Ratio_carre	1163-839	XXX	XXX	XXX	936-1064	884-1116	887-1113
Guess_carre	XXX	XXX	XXX	XXX	XXX	830-1170	1055 -945
Guess_ratio_seul	XXX	XXX	XXX	XXX	XXX	XXX	1081-919
Guess_ratio_carre	XXX	XXX	XXX	XXX	XXX	1224-776	XXX

	Naive_pref	Naive_court	Ratio_seul	Ratio_carre	Guess_carre	Guess_ratio_carre	Guess_ratio_seul
Nombre de victoire	0/6	1/6	3/7	3/7	5/6	5/7	6/7
Différence de points	-4894	-2140	+856	+1486	+1040	+2100	+1874