



Licence de Sciences et Technologies - Mention Informatique

LU3IN025 - Intelligence Artificielle et Jeux

Cooperative Pathfinding

Rapport de projet

KHALID Ruhma - 3710442

GIGACZ Kiara - 3803025

Groupe 3

Année universitaire : 2020 - 2021

Sommaire

Sommaire	1
Introduction	2
Stratégies implémentées	2
Mise en évidence de 3 situations intéressantes	4
Comparaison des stratégies	8
Conclusion	12
Annexe	13

Introduction

« Dans un problème de *cooperative path-finding*: on dispose d'un ensemble d'agents qui doivent chacun atteindre une destination qui leur est propre. Il s'agit de trouver un ensemble de chemins, sans collision, qui permette à chaque agent d'atteindre sa destination. »

Ici nous avons décidé d'implémenter 3 stratégies : A* indépendant, path-slicing et A* coopérative que nous expliquons ci-dessous.

Ensuite, afin de mieux évaluer nos stratégies et leurs performances, nous les avons comparées dans 3 types de scénarios différents : “exchange”, “race” et “mingle”.

Enfin, nous avons pu mettre en évidence 4 situations intéressantes que nous avons observées pendant nos tests.

Stratégies implémentées

Nous avons implémenté trois stratégies au total. Les trois stratégies reposent sur l'algorithme A* pour le calcul des chemins. Nous avons légèrement modifié le A* fourni pour lui permettre de trouver un chemin où un joueur se mettrait en pause si nécessaire. L'algorithme détecte aussi le cas où il n'y a actuellement aucun chemin vers l'objectif, et propose au joueur de ne pas se déplacer pendant un tour, puis de retenter le calcul d'un chemin au tour suivant.

Stratégie 1 : A* indépendants avec recalcul de chemin régulier et en cas de blocage

Cette stratégie calcule un chemin pour le joueur en ne prenant pas en considération les autres joueurs sur la carte. Il se déplace d'une case i à une case $i+1$ seulement si cette case est libre. Dans le cas contraire, il recalcule un nouveau chemin, cette fois-ci avec connaissance de la position de tous les autres joueurs, et ainsi de suite jusqu'à arriver à son objectif.

Cette stratégie permet aussi un recalcul de chemin de façon régulière afin d'être sûr que le joueur est sur le bon chemin. Ici, nous avons décidé de recalculer le chemin toutes les 3 itérations.

Stratégie 2 : A* indépendants avec réparation locale (path-slicing)

Similaire à la stratégie précédente mais ici en cas de blocage on ne recalcule qu'une partie du chemin et non tout le chemin.

Stratégie 3 : A* coopératif

Jusqu'ici le joueur ne planifiait son chemin initial qu'en considérant la carte avec ses obstacles. Cette stratégie va permettre au joueur de pouvoir se déplacer en considérant aussi le chemin pris par les autres joueurs de son équipe afin d'éviter au maximum les blocages.

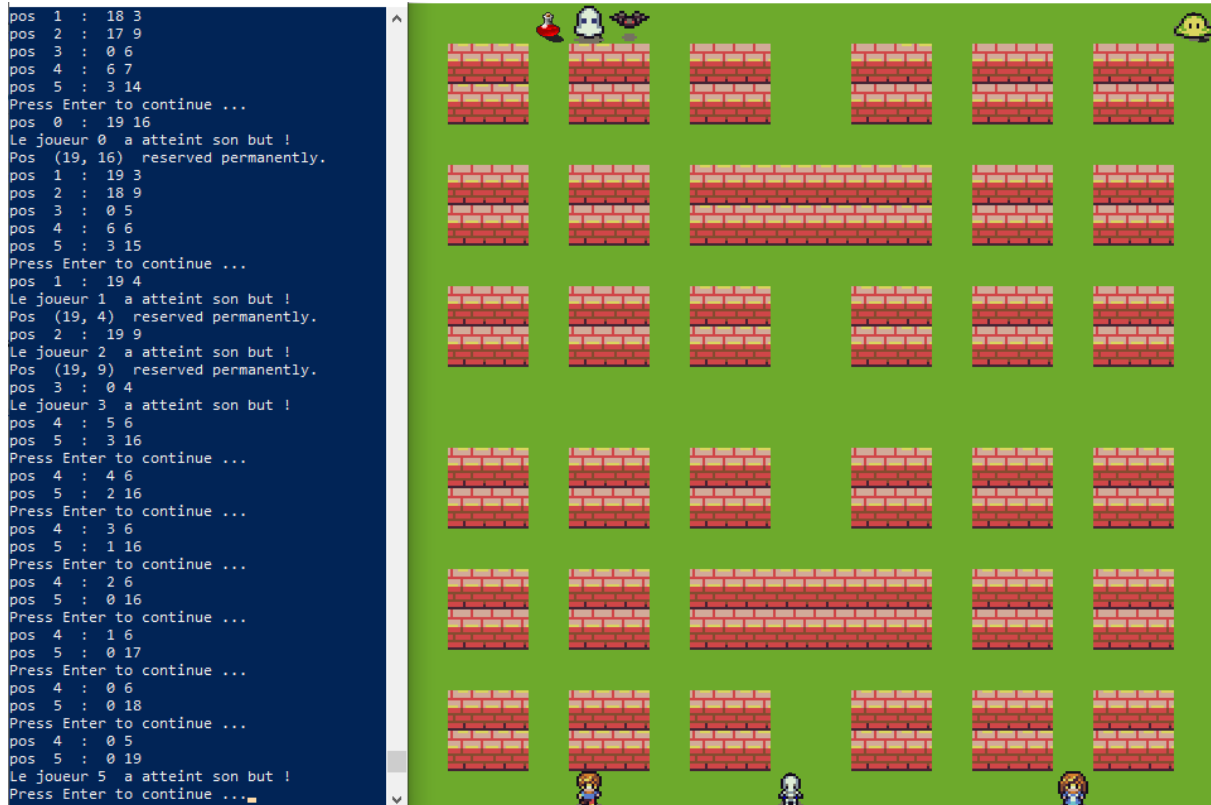
Pour cela nous avons adapté l'algorithme A^* pour considérer un tableau de réservations qui est rempli au fur et à mesure par les joueurs d'une même équipe (space-time A^*). Pour éviter que les joueurs d'une même équipe planifient des chemins où ils échangent de position, nous réservons deux cases (x, y, t) et $(x, y, t+1)$ pour chaque mouvement.

Cependant, cela n'évite pas les collisions avec les joueurs adversaires. Quand un joueur est bloqué, il recalcule donc un nouveau chemin.

Mise en évidence de 3 situations intéressantes

Situation intéressante 1 :

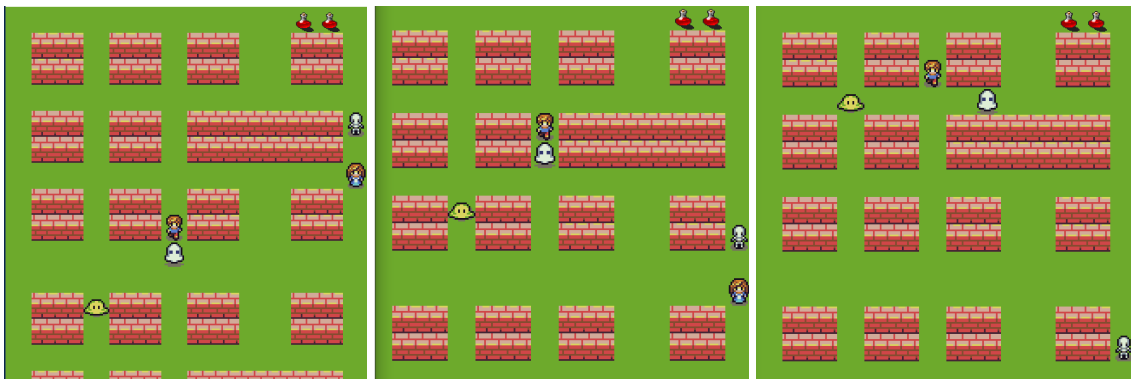
Le joueur 4 (stratégie de path-slicing) prend trop de temps pour se rendre compte que son chemin est maintenant bloqué par son équipier. C'est une situation qui aurait été évitée avec A* coop (en particulier, STA*).



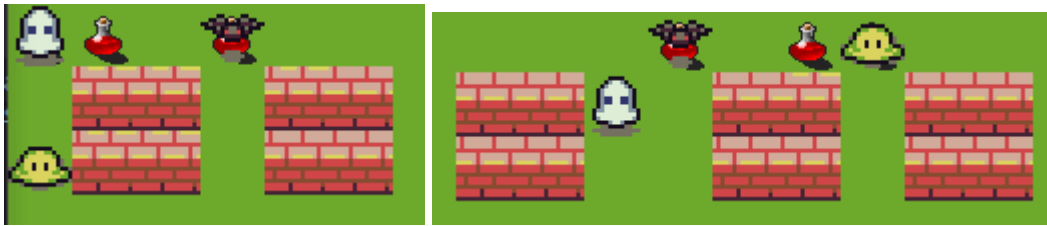
Situation intéressante 2 :

Le joueur 1 (bonhomme dans les images, stratégie A* coopératif) se retrouve totalement coincé par sa stratégie, la configuration de la carte et le fait qu'il joue avant le joueur 3 (fantôme, path-slicing). Avant de sortir d'un couloir, il se retrouve bloqué par J3. STA* lui calcul un nouveau chemin où il recule juste d'un pas pour après retenter d'avancer dans la direction initiale. Malheureusement J3 continue d'avancer dans le sens exactement opposé à lui, et n'a pas l'occasion de recalculer son chemin car J1 lui laisse la place à chaque tour. Cela fait que J1 se retrouve à reculer jusqu'à la ligne 1 avant de pouvoir avancer de nouveau.

```
Press Enter to continue ...
pos 0 : 6 13
pos 1 : 8 6
pos 2 : 4 13
pos 3 : 9 6
pos 4 : 10 18
pos 5 : 11 3
Press Enter to continue ...
pos 0 : 7 13
Le joueur 1 est bloqué par le joueur 3
astar : [(7, 6), (8, 6), (9, 6), (10, 6), (11, 6), (12, 6), (13, 6), (14, 6), (15, 6), (16, 6), (17, 6), (18, 6), (19, 6), (19, 5)]
Chemin trouvé pour joueur 1 : [(0, 7), (0, 6), (1, 6), (2, 6), (3, 6), (4, 6), (5, 6), (6, 6), (7, 6), (8, 6), (7, 6), (8, 6), (9, 6), (10, 6), (11, 6), (12, 6), (13, 6), (14, 6), (15, 6), (16, 6), (17, 6), (18, 6), (19, 6), (19, 5)]
pos 1 : 7 6
pos 2 : 5 13
pos 3 : 8 6
pos 4 : 10 17
pos 5 : 10 3
Press Enter to continue ...
pos 0 : 8 13
Le joueur 1 est bloqué par le joueur 3
astar : [(6, 6), (7, 6), (8, 6), (9, 6), (10, 6), (11, 6), (12, 6), (13, 6), (14, 6), (15, 6), (16, 6), (17, 6), (18, 6), (19, 6), (19, 5)]
Chemin trouvé pour joueur 1 : [(0, 7), (0, 6), (1, 6), (2, 6), (3, 6), (4, 6), (5, 6), (6, 6), (7, 6), (8, 6), (7, 6), (6, 6), (7, 6), (8, 6), (9, 6), (10, 6), (11, 6), (12, 6), (13, 6), (14, 6), (15, 6), (16, 6), (17, 6), (18, 6), (19, 6), (19, 5)]
pos 1 : 6 6
pos 2 : 6 13
pos 3 : 7 6
pos 4 : 10 16
pos 5 : 9 3
Press Enter to continue ...
pos 0 : 9 13
Le joueur 1 est bloqué par le joueur 3
astar : [(5, 6), (6, 6), (7, 6), (8, 6), (9, 6), (10, 6), (11, 6), (12, 6), (13, 6), (14, 6), (15, 6), (16, 6), (17, 6), (18, 6), (19, 6), (19, 5)]
Chemin trouvé pour joueur 1 : [(0, 7), (0, 6), (1, 6), (2, 6), (3, 6), (4, 6), (5, 6), (6, 6), (7, 6), (8, 6), (7, 6), (6, 6), (5, 6), (6, 6), (7, 6), (8, 6), (9, 6), (10, 6), (11, 6), (12, 6), (13, 6), (14, 6), (15, 6), (16, 6), (17, 6), (18, 6), (19, 6), (19, 5)]
pos 1 : 5 6
pos 2 : 7 13
pos 3 : 6 6
pos 4 : 9 16
pos 5 : 8 3
Press Enter to continue ...
```



Situation intéressante 3 :



Le dernier joueur ne peut pas atteindre son objectif comme les deux autres joueurs de son équipe sont arrivés et ne bougent donc plus. Pour éviter cela, il faudrait autoriser les joueurs à se déplacer pour laisser la place à leurs équipiers quand cela est possible.

Situation intéressante 4 :

```
Press Enter to continue ...
pos 0 : 9 13
pos 1 : 7 6
pos 2 : 7 13
pos 3 : 14 6
pos 4 : 11 16
pos 5 : 10 7
Press Enter to continue ...
pos 0 : 10 13
pos 1 : 8 6
pos 2 : 8 13
pos 3 : 13 6
pos 4 : 10 16
pos 5 : 10 8
Press Enter to continue ...
pos 0 : 10 14
pos 1 : 9 6
pos 2 : 9 13
pos 3 : 12 6
pos 4 : 10 15
pos 5 : 10 9
Press Enter to continue ...
Le joueur 0 est bloqué par le joueur 4
astar : [(9, 14), (10, 14), (10, 15), (10, 16), (11, 16), (12, 16), (13, 16), (14, 16), (15, 16), (16, 16), (17, 16), (18, 16), (19, 16)]
Chemin trouvé pour joueur 0 : [(0, 13), (1, 13), (2, 13), (3, 13), (4, 13), (5, 13), (6, 13), (7, 13), (8, 13), (9, 13), (10, 13), (10, 14), (9, 14), (10, 14), (10, 15), (10, 16), (11, 16), (12, 16), (13, 16), (14, 16), (15, 16), (16, 16), (17, 16), (18, 16), (19, 16)]
pos 0 : 9 14
pos 1 : 10 6
pos 2 : 10 13
pos 3 : 11 6
pos 4 : 10 14
pos 5 : 10 10
Press Enter to continue ...
```



Dans le scénario ci-dessus, J0 (bonhomme, A* coopératif) aurait pu anticiper la collision avec J4 (fantôme, path-slicing) en observant que J4 venait d'entrer dans le couloir par (10,16) et qu'il se trouverait donc très probablement à (10,15) quand J0 comptait passer. Dans ce scénario particulier, dû à la configuration de la carte, une anticipation n'aurait pas amélioré la performance du J0, mais il se peut que dans certaines cartes cela pourrait être un avantage. Observation intéressante : dans ce scénario, le joueur qui joue en premier (J0) est désavantagé car il doit recalculer son chemin et en prendre un plus long, alors que J4 n'a pas eu besoin de changer de chemin.

Comparaison des stratégies

Pour comparer nos stratégies, nous avons comparé les 3 stratégies entre elles dans 3 types de scénarios : “exchange”, “race” et “mingle”. Le scénario “exchange” consiste à placer les équipes l’une en face d’autres, et de placer les objectifs de chaque équipe au côté opposé de la carte. Le scénario “race” quant à lui consiste à placer les deux équipes sur une même ligne, et leurs objectifs sur une ligne au côté opposé. Le scénario “mingle” place les joueurs d’une équipe sur la première ligne, et ses objectifs en face, et les joueurs de l’autre équipe sur une colonne avec ses objectifs aussi en face.

Vous trouverez en annexe des exemples de ces scénarios. Nous avons répété les simulations de ces scénarios en variant les localisations des agents et des objets dans des zones définies (ex : sur une ligne ou sur une colonne).

Définition d’une victoire : L’équipe ayant le plus grand score est le vainqueur, match nul si les scores sont équivalents.

1 - A* indépendant vs Path-slicing

Tableau 1.1 : Comparaison des résultats dans le scénario “exchange”

Equipe 1	Equipe 2	Nb Itérations	Nb Tours	Nb Victoires (Eq1, Eq2)	Vainqueur
A* indépendant s	Path-slicing	100	50	(2, 0)	Equipe 1
A* indépendant s	Path-slicing	25	50	(12, 13)	Equipe 2
A* indépendant s	Path-slicing	30	50	(15, 13)	Equipe 1
A* indépendant s	Path-slicing	35	50	(22, 5)	Équipe 1
A* indépendant s	Path-slicing	40	50	(9, 2)	Équipe 1

Tableau 1.2 : Comparaison des résultats dans le scénario “race”

Equipe 1	Equipe 2	Nb	Nb Tours	Nb Victoires	Vainqueur
----------	----------	----	----------	--------------	-----------

		Itérations		(Eq1, Eq2)	
A* indépendants	Path-slicing	100	50	(0,0)	Match nul
A* indépendants	Path-slicing	30	50	(8, 21)	Équipe 2
A* indépendants	Path-slicing	35	50	(9, 20)	Équipe 2
A* indépendants	Path-slicing	40	50	(6, 4)	Équipe 1

Tableau 1.3 : Comparaison des résultats dans le scénario “mingle”

Equipe 1	Equipe 2	Nb Itérations	Nb Tours	Nb Victoires (Eq1, Eq2)	Vainqueur
A* indépendants	Path-slicing	30	50	(18,17)	Équipe 1
A* indépendants	Path-slicing	35	50	(18,15)	Équipe 1
A* indépendants	Path-slicing	40	50	(16,8)	Équipe 1

2 - A* coopérative vs Path-slicing

Tableau 2.1 : Comparaison des résultats dans le scénario “exchange”

Equipe 1	Equipe 2	Nb Itérations	Nb Tours	Nb Victoires (Eq1, Eq2)	Vainqueur
A* coopérative	Path-slicing	100	50	(0, 1)	Equipe 2
A* coopérative	Path-slicing	30	50	(24, 8)	Equipe 1
A* coopérative	Path-slicing	35	50	(26, 6)	Equipe 1
A* coopérative	Path-slicing	40	50	(10, 0)	Équipe 1

Tableau 2.2 : Comparaison des résultats dans le scénario “race”

Equipe 1	Equipe 2	Nb Itérations	Nb Tours	Nb Victoires (Eq1, Eq2)	Vainqueur
A* coopérative	Path-slicing	30	50	(21, 15)	Équipe 1
A* coopérative	Path-slicing	35	50	(19, 5)	Équipe 1
A* coopérative	Path-slicing	40	50	(11, 0)	Équipe 1

Tableau 2.3 : Comparaison des résultats dans le scénario “mingle”

Equipe 1	Equipe 2	Nb Itérations	Nb Tours	Nb Victoires (Eq1, Eq2)	Vainqueur
A* coopérative	Path-slicing	30	50	(27, 4)	Équipe 1
A* coopérative	Path-slicing	35	50	(28, 4)	Équipe 1
A* coopérative	Path-slicing	40	50	(19, 4)	Équipe 1

L'équipe 1 a remporté la presque toutes les parties donc l'algorithme A* coopérative est plus performant.

3 - A* indépendant vs A*coopérative

Tableau 3.1 : Comparaison des résultats dans le scénario “exchange”

Equipe 1	Equipe 2	Nb Itérations	Nb Tours	Nb Victoires (Eq1, Eq2)	Vainqueur
A* indépendant	A* coopérative	30	50	(8, 15)	Equipe 2
A* indépendant	A* coopérative	35	50	(19, 6)	Equipe 1
A* indépendant	A* coopérative	40	50	(8, 2)	Équipe 1

Tableau 3.2 : Comparaison des résultats dans le scénario “race”

Equipe 1	Equipe 2	Nb Itérations	Nb Tours	Nb Victoires (Eq1, Eq2)	Vainqueur
A* indépendant	A* coopérative	30	50	(9, 26)	Équipe 2
A* indépendant	A* coopérative	35	50	(6, 26)	Équipe 2
A* indépendant	A* coopérative	40	50	(0, 7)	Équipe 2

Tableau 3.3 : Comparaison des résultats dans le scénario “mingle”

Equipe 1	Equipe 2	Nb Itérations	Nb Tours	Nb Victoires (Eq1, Eq2)	Vainqueur
A* indépendant	A* coopérative	30	50	(11, 24)	Équipe 2
A* indépendant	A* coopérative	35	50	(10, 19)	Équipe 2
A* indépendant	A* coopérative	40	50	(9, 11)	Équipe 2

Pareillement, la stratégie A* coopérative est plus performante que la stratégie A* indépendant.

Conclusion

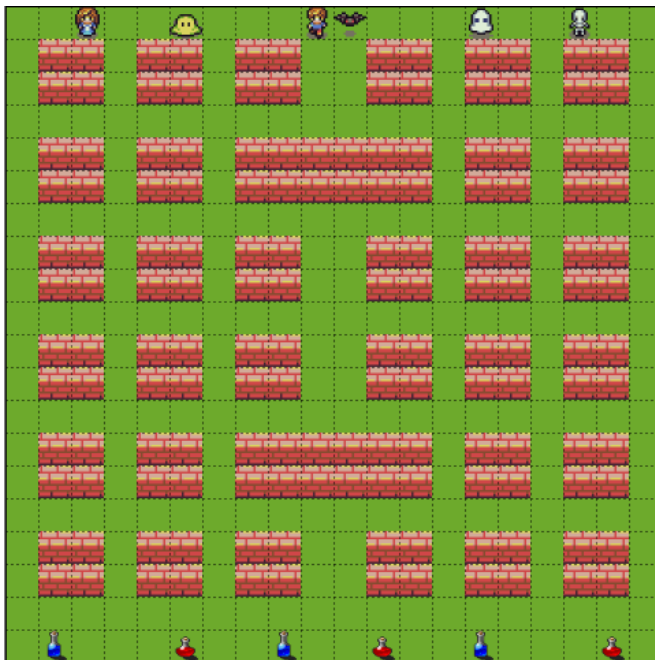
En conclusion, la stratégie A* coopérative est la plus performante ici dans la majorité des scénarios. Cela est dû à sa prévention de collision avec les joueurs d'une même équipe, qui lui permet de planifier un chemin optimal dès le début. Néanmoins, cet algorithme est très coûteux en mémoire.

Annexe

Exemple d'un scénario "exchange" :



Exemple d'un scénario "race" :



Exemple d'un scénario "mingle" :

