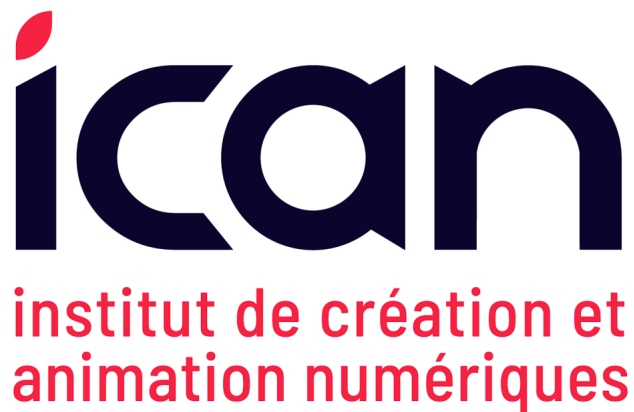

Mémoire de fin d'études

*Comportements de groupe émergents dans
les jeux vidéo*



ICAN - Mastère Game Design
Orso NEGRONI
2022 - 2023

Tables des Matières

Table des figures	2
Remerciements	3
Introduction	4
1 Intelligence artificielle et émergence de comportements	6
1.1 L'Intelligence Artificielle dans les Jeux vidéo	6
1.1.1 Qu'est ce que l'intelligence artificielle	6
1.1.2 L'intelligence artificielle dans les jeux vidéos	7
1.2 L'Intelligence Artificielle distribuée dans les Jeux vidéo	8
1.2.1 L'intelligence artificielle distribuée	8
1.2.2 Qu'en est il des jeux vidéos ?	10
1.3 L'émergence de comportement	12
1.3.1 Qu'est ce que l'émergence de comportement	12
1.3.2 Exemples de comportements émergent	13
2 Développement et émergence de comportements collectifs	15
2.1 Utilisation des comportements collectifs	15
2.1.1 Les modèles d'IA pour la simulation de groupes	15
2.1.2 L'utilisation dans les jeux	21
2.2 Développement et joueur	26
2.2.1 Pertinence dans le développement	26
2.2.2 Ressentie du joueur	27
2.3 Plan d'expérimentation	30
2.3.1 Base de l'étude	30
2.3.2 Choix du projet	32
3 Projet de fin d'étude : Pyrates : Set Sail And Discover !	34
3.1 Présentation du projet	34
3.1.1 Pyrates : Set Sail And Discover !	34
3.1.2 Lien avec la narration procédurale	35
3.2 Conceptualisation des Matelots	36
3.2.1 Modèle d'agent pour les matelots	36
3.2.2 Interaction entre le joueur et l'équipage	39
3.3 La vie sur le bateau	40
3.3.1 Simulation de la vie sur le bateau	41
3.3.2 Résultats observés	43
Conclusion	45
Bibliographie	47
Ludographie	49
Glossaire	50
Annexes	51

Table des figures

Figure 1 : Résumé des résultats expérimentaux par rapport aux contrôleurs par défaut	11
Figure 2 : Evolution du comportement des proies dans l'article de OLSON Randal	13
Figure 3 : Courbes d'apprentissage des différents scénarios	16
Figure 4 : Architecture d'un agent BDI	18
Figure 5 : Exemple du bloc, la plus petite structure stable	20
Figure 6 : cinq itérations du planeur	20
Figure 7.1: Blinky	21
Figure 7.2: Pinky	21
Figure 7.3: Inky	21
Figure 7.4: Clyde	21
Figure 8 : Exemple d'arbre de comportement	23
Figure 9.1 : Relation d'un PNJ (Cael) dans une partie de <i>Space Haven</i>	24
Figure 9.1 : Relation d'un PNJ (Shelly) dans une partie de <i>Space Haven</i>	24
Figure 10 : Modèle d'agent de smallville	24
Figure 11 : Émergence de transmission d'invitation à une fête et de nouvelle relation	25
Figure 12.1 : Structure OTCA metapixel active	28
Figure 12.2 : Jeu de la vie dans le jeu de la vie	28
Figure 13 : Système Nemesis et relation	29
Figure 14 : Modèle d'agent avec relation influençant son environnement et les autres PNJ	32
Figure 15 : Vue du jeu Pyrates : Set Sail And Discover !	35
Figure 16 : Variable d'un matelot	37
Figure 17 : Modèle d'agent adapté pour les matelots	39
Figure 18 : Conversation entre deux matelots	41
Figure 19 : Simulation de la vie sur le bateau	42
Figure 20 : Relation d'un matelot	43

Remerciements

Je tiens à remercier toutes les personnes ayant contribué à l'élaboration de ce travail.

Je remercie tout d'abord mon directeur de mémoire, M. Coussieu, pour ses conseils et ses retours durant la rédaction de ce travail. Son aide a fortement contribué à l'amélioration de ce mémoire.

Je remercie ma famille et mes amis, qui m'ont soutenu et aidé à la rédaction de ce mémoire, ainsi que pour leur soutien pour cette réorientation dans le domaine du jeu vidéo qui me fait rêver depuis longtemps.

Je remercie également l'équipe système multi-agents du laboratoire d'informatique de Paris 6, dont les enseignements lors mon premier Master m'ont inspiré pour le sujet de ce mémoire.

Je remercie aussi les intervenants et mes camarades de promotion, particulièrement mon groupe de projet, la Pirateam, qui m'ont fait apprécier cette discipline, et m'ont fait découvrir mes capacités dans ce domaine liant technologie et créativité.

Introduction

Ce mémoire explore les techniques favorisant l'intelligence en essaim dans les jeux vidéo de manière à ce que le joueur puisse percevoir adaptation et cohérence parmi les groupes de personnages non-joueurs¹ (PNJ).

L'intelligence en essaim, ou intelligence distribuée, fait référence à la propriété des systèmes de robots non intelligents présentant un comportement collectif intelligent². Ce concept venant de la nature et du comportement observé dans le règne animal, mais prend part dans l'intelligence artificielle notamment pour la simulation de groupes. L'intelligence artificielle regroupe l'ensemble des théories et des techniques mises en œuvre en vue de réaliser des machines capables de simuler l'intelligence humaine³. Dans le domaine du jeu vidéo, un ensemble de PNJ peut de manière individuelle effectuer des actions qui semblent peu utiles alors qu'en groupe celles-ci prennent un sens logique selon la situation dans laquelle ils se trouvent.

Le machine learning, ou apprentissage automatique, et les algorithmes génétiques permettent l'émergence de comportements de groupe, comme peut le montrer un grand nombre d'études sur la simulation de foule lors de situation dans des cas d'évacuation d'urgence. Ces techniques permettent, par renforcement ou par sélection naturelle, l'apparition de ces comportements. Grâce ces technologies, il est possible de voir ces émergences, cependant comme vu précédemment, l'intelligence en essaim se définit comme l'apparition de comportements collectifs de la part d'individus non intelligents. Nous pouvons nous demander s'il existe d'autres techniques plus simples permettant de voir l'émergence de ces comportements de groupe ?

De plus, dans un groupe, les relations entre les individus ont une part importante. En effet, un groupe constitué d'individus indépendants peut évoluer selon les relations entre ces derniers. Nous pouvons nous demander si la dimension sociale a une influence sur un groupe et dans le cadre du jeu vidéo, comment cela peut être perçu par le joueur ?

Si ces éléments permettent d'observer des comportements de groupe autonomes, dans le cadre du jeu vidéo, quelles en sont les conséquences pour le joueur ? Ces groupes de PNJ, en simple image de fond ont-il un intérêt pour joueur ? Ces PNJ et leur environnement forment un écosystème dans lequel le joueur évolue également. Faisant partie de cet

¹ PNJ : Personnage Non Joueur, personnage qui n'est pas contrôlé par le joueur

² BENI, Gerardo et WANG, Jing. Swarm intelligence in cellular robotic systems. In : *Robots and biological systems: towards a new bionics?*. Berlin, Heidelberg : Springer Berlin Heidelberg, 1993. p. 703-712.

³ « Encyclopédie Larousse en ligne - intelligence artificielle », Larousse.

écosystème, les actions du joueurs ont donc un impact sur celui-ci, et donc sur les comportements des PNJ. Nous pouvons donc nous demander s'il est possible de voir émerger un comportement de groupe au sein d'un ensemble de PNJ influencés par les actions du joueur ?

Dans le cadre de l'étude de l'émergence de comportements de groupe dans le jeu vidéo, nous en venons alors à nous poser la problématique suivante : Quelles conditions techniques permettent de voir émerger un comportement de groupe cohérent chez les PNJ pour qu'ils aient des interactions avec le joueur ?

Pour tenter de répondre à cette problématique, dans un premier temps les différents éléments la composant seront traités au sein d'un état de l'art. Nous allons d'abord voir l'origine de l'intelligence artificielle et son développement dans le domaine du jeu vidéo puis étudier l'introduction de l'intelligence artificielle distribuée. Cet état de l'art se termine sur l'étude des techniques d'émergence de comportements en informatique.

Après cet état de l'art, nous verrons comment développer des comportements collectifs, à savoir les modèles d'intelligence artificielle pour la simulation de ces comportements et leur utilisation dans les jeux. Puis nous nous intéresserons au développement du jeu, du point de vue de l'industrie et surtout du joueur, sur son ressenti mais également son impact avec le système. Enfin un plan d'expérimentation sera mis en place dans l'optique de l'appliquer au projet de fin d'étude.

Ce mémoire se conclut sur la mise en application de ce plan d'étude au projet de fin d'étude, Pyrates : Set Sail And Discover !

1 Intelligence artificielle et émergence de comportements

1.1 L'Intelligence Artificielle dans les Jeux vidéo

1.1.1 Qu'est ce que l'intelligence artificielle

L'intelligence artificielle est étroitement liée à l'informatique que nous connaissons aujourd'hui. En effet, en 1950, Alan Turing évoque l'intelligence artificielle dans son ouvrage sur le Test de Turing⁴. En 1956, John McCarthy, lors d'une école d'été à Dartmouth, donne un nom sur six ans d'évolution technologique, « l'Intelligence Artificielle », ou IA⁵. L'IA est donc introduite par John McCarthy, Marvin Minsky, Nathanael Rochester et Claude Shannon comme le fait qu'il est possible de décrire les différentes facultés cognitives de l'homme qui peuvent être reproduites par une machine. Cette école n'a pas regroupé uniquement des mathématiciens, mais également des chercheurs d'autres horizons, comme les sciences humaines et sociales⁶. L'IA peut donc se traduire par la retranscription de comportements intelligents, humain ou animal, par une machine.

Nous avons évoqué les pionniers et l'origine du domaine de l'IA, mais qu'est ce qu'une IA ? Une IA est un programme informatique capable d'agir selon les connaissances et les capacités dont elle dispose. L'ensemble de ses connaissances lui sont acquises via des données informatiques. Elle analyse ensuite ces données pour déterminer l'action la plus appropriée. Une IA est un algorithme s'adaptant à la situation pour fournir la réponse la plus adéquate.

Il existe un grand nombre d'intelligences artificielles plus ou moins proche de l'intelligence humaine. Selon John Searle⁷, il en existe deux types, l'IA forte et l'IA faible. La première, l'IA forte, tend à se rapprocher de celle de l'homme du fait de son intelligence mais également sa sensibilité. A contrario, l'IA faible est limitée aux actions qui lui sont mises à disposition. Il existe une multitude de modèles d'IA, se rapprochant plus ou moins de

⁴ A. M. TURING, I.—COMPUTING MACHINERY AND INTELLIGENCE, *Mind*, Volume LIX, Issue 236, October 1950, Pages 433–460

⁵ Smith, C., Huang, T., McGuire, B., & Yang, G. (2006). *The History of Artificial Intelligence*. Washington: University of Washington.

⁶ GANASCIA, Jean-Gabriel. Intelligence artificielle et épistémologie. Allers-retours indispensables. In : JECIS. 2022.

⁷ SEARLE, John. Chinese room argument. *Scholarpedia*, 2009, vol. 4, no 8, p. 3100.

l'intelligence humaine. Ceux-ci vont du simple automate exécutant l'action qui lui est demandée au « machine learning » qui apprend à partir du grand nombre de données qu'on lui donne. L'IA se divise en sous-domaines, comme le montre le panorama fait par Romero, Aloui et Heiser⁸. Ce panorama nous montre un aperçu des sept sous-domaines de l'IA et met en évidence son utilisation dans divers domaines. En effet, dans cet article, les auteurs font un éventail des différents sous-domaines de l'IA utilisés dans l'éducation. Ceci nous montre que l'IA peut être employée dans divers domaines, l'éducation comme vu précédemment, mais également la médecine. On peut donc se demander ce qu'il en est de l'IA dans les jeux vidéo.

1.1.2 L'intelligence artificielle dans les jeux vidéos

Selon Siyuan Xu⁹ les premiers jeux, sortis fin des années 50, étaient principalement joueur contre joueur. Ce n'est qu'en 1970 que les premières IA dans les jeux apparaissent sur les bornes d'arcade Atari telles que pour les jeux *Space Invaders* et *Pong*. Ces premières IA sont simples, et directement implémentées dans le code. Dans l'exemple du jeu de *Pong*, le modèle dispose d'un temps d'attente avant de prendre une décision et d'une probabilité sur l'arrivée de la balle pour donner une chance au joueur. Depuis, les IA dans les jeux vidéo n'ont cessé d'évoluer, que ce soit pour des adversaires dans les RTS ou dans la simulation d'une vie artificielle comme dans *SimCity*.

L'IA dans les jeux vidéo a continué à évoluer notamment avec le machine learning. En effet, les avancées DeepMind ont permis de voir des IA capables de surpasser l'humain. En 2017, AlphaGo¹⁰ bat le meilleur joueur du monde du jeu de go. En 2019, c'est autour d'AlphaStar¹¹ de performer dans son domaine et d'atteindre le rang de grandmaster sur *Starcraft 2*. Cependant, si ces exemples montrent l'utilisation du machine learning pour simuler un joueur humain, qu'en est-il de l'IA pour simuler un PNJ ?

En effet, en 2014 sort *Alien: Isolation* développé par Creative Assembly. Dans ce jeu d'horreur, l'adversaire invincible, l'alien, est géré par deux IA. La première est appelée director-AI ou macro, permet d'avoir une vision globale du jeu, de la position du joueur à différents moments. La seconde, alien-AI ou micro, reçoit des bribes d'informations de la part

⁸ ROMERO, Margarida, ALOUI, Hedi, HEISER, Laurent, et al. Un bref parcours sur les ressources, pratiques et acteurs en IA et éducation. 2021. Thèse de doctorat. Université Côte d'Azur.

⁹ XU, Siyuan. History of AI design in video games and its development in RTS games. 2014.

¹⁰ <https://www.deepmind.com/research/highlighted-research/alphago>

¹¹ <https://www.deepmind.com/blog/alphastar-mastering-the-real-time-strategy-game-starcraft-ii>

de l'IA macro, tel que les sons que le joueur a générés ou sa position approximative. Grâce à ces informations couplées à ce que l'alien perçoit, l'alien-IA va, grâce à un arbre de comportements, décider de l'action la plus appropriée. Notons également que pour éviter la frustration d'une IA trop forte pour le joueur, l'IA possède un mode actif et un mode passif. Si elle est active, l'alien traque le joueur, si elle est active trop longtemps et redevient passive, il retourne dans les conduits durant un certain temps.

Un autre exemple de l'utilisation d'IA dans les jeux vidéo est le Système Nemesis de *La Terre du Milieu : L'Ombre du Mordor* et *La Terre du Milieu : L'Ombre de la guerre*. Dans ces jeux, l'IA gère les PNJ pour leur donner une mémoire et ainsi se souvenir des actions du joueur. Comme le montre les recherches de Shaun Downey et Darryl Charles¹², l'exemple de *La Terre du Milieu : L'Ombre du Mordor* montre que les PNJ agissent en fonction des informations de chacun et de leur communication entre eux pour connaître le joueur et le risque potentiel qu'il peut représenter. On peut voir apparaître un comportement par exemple, si un orc est vaincu plusieurs fois de la même manière, il finira par apprendre et développer des parades contre cette tactique. L'ouvrage de Shaun et Darryl nous introduit un autre modèle d'IA pour la gestion des PNJ dans un jeu. Oui en effet, les PNJ ont chacun leur connaissance et leurs champs d'actions, mais peuvent également communiquer entre eux ces informations, il forme ce qu'on appelle un système multi-agents. Les systèmes multi-agents est une approche de l'IA de manière distribuée. Ce qui nous amène à nous demander, qu'est ce que l'IA distribuée.

1.2 L'Intelligence Artificielle distribuée dans les Jeux vidéo

1.2.1 L'intelligence artificielle distribuée

L'intelligence distribuée, ou intelligence en essaim, est définie par James Kennedy dans son ouvrage *Swarm Intelligence*¹³ comme la résolution de problèmes par l'interaction entre différents agents. L'intelligence distribuée implique une multitude d'agents qui vont partager leurs connaissances pour résoudre un problème en groupe. Ces algorithmes sont

¹² DOWNEY, Shaun et CHARLES, Darryl. Distribution of artificial intelligence in digital games. *International Journal of Intelligent Information Technologies (IJIT)*, 2015, vol. 11, no 3, p. 1-14.

¹³ KENNEDY, James. *Swarm intelligence*. Springer US, 2006.

inspirés de comportements chez les animaux et les insectes et chez les humains. En effet, comme le dit James Kennedy, le principe des algorithmes d'intelligence distribuée est basé sur la capacité des humains d'apprendre à partir des connaissances des autres. En effet, les algorithmes d'intelligence distribuée consistent à développer un certain nombre d'individus qui, grâce à des règles simples et à la possibilité de communiquer entre eux, vont pouvoir résoudre des problèmes complexes en divisant les tâches.

Un exemple très connu d'intelligence en essaim est le Boids de Craig W. Reynolds¹⁴. L'algorithme Boids permet de simuler le comportement d'un essaim d'oiseaux ou d'un banc de poissons. Les individus Boids suivent trois règles, la cohésion, la séparation et l'alignement. Ces règles permettent aux différents individus de rester à une certaine distance les uns des autres et de façon à aller dans la même direction. Cet algorithme est beaucoup utilisé pour la représentation virtuelle d'essaim d'animaux dans certaines simulations ou dans certains films¹⁵.

Une approche de l'intelligence artificielle distribuée est les systèmes multi-agents. Dusan Teodorovic décrit les systèmes multi-agents comme un ensemble d'individus, physique ou virtuel, qui communiquent et collaborent entre eux et exécutent différentes tâches¹⁶. De plus, il définit les deux types d'agent que peut composer un système multi-agents, les agents "intelligents" et les agents "réactifs". Les premiers ont à leur disposition un ensemble de connaissance et d'action, ils sont capables de communiquer avec les autres et sont même capables de collaborer pour résoudre une tâche. De plus, ils ont dans la plupart des cas un objectif à accomplir. À l'inverse, les agents réactifs, comme leur nom le dit, réagissent à des événements, ils peuvent tout de même communiquer avec les autres. Les agents n'interagissent pas uniquement entre eux, mais également avec l'environnement dans lequel ils évoluent. Cet environnement est défini par son créateur et offre donc une grande liberté pour la création d'un système multi-agents. Du fait de cette liberté, les systèmes multi-agents sont une approche intéressante pour la résolution de problèmes complexes.

¹⁴ REYNOLDS, Craig W. Flocks, herds and schools: A distributed behavioral model. In : Proceedings of the 14th annual conference on Computer graphics and interactive techniques. 1987. p. 25-34.

¹⁵ <https://beforesandafters.com/2022/04/07/a-history-of-cg-bird-flocking/>

¹⁶ TEODOROVIC, Dusan. Transport modeling by multi-agent systems: a swarm intelligence approach. Transportation planning and Technology, 2003, vol. 26, no 4, p. 289-312.

1.2.2 Qu'en est il des jeux vidéos ?

Les systèmes multi-agents est donc une approche intéressante pour simuler un ensemble d'individus. Dans le cadre de notre étude, nous pouvons nous demander s'il est possible d'intégrer un système multi-agents, ou une intelligence artificielle distribuée, dans un jeu vidéo. En effet, Gustavo Recio et al. ont utilisé un framework basé sur le comportement des fourmis pour le contrôle des fantômes dans *Ms. Pac-Man*¹⁷. Le comportement des fourmis est basé sur le dépôt de phéromone, un chemin vers l'objectif va être trouvé et plus il va être emprunté, plus la concentration de phéromone va augmenter. Les fourmis vont suivre en priorité le chemin avec le plus de phéromone, mais avec une probabilité de prendre un chemin alternatif. Dans leur article, l'application sur les fantômes est faite avec deux types de fourmis, les exploratrices et les chasseuses. Les exploratrices vont partir des intersections adjacentes à la position de *Ms. Pac-Man* et vont explorer les alentours en déposant des phéromones sur leur chemin. Les chasseuses, elles vont partir des différents fantômes et vont suivre les phéromones des chasseuses et des exploratrices pour atteindre *Ms. Pac-Man*. Une fois que les fourmis ont terminé leur parcours, le fantôme sélectionne le chemin ayant le plus de phéromones. Les résultats ont montré que les ghostAnt ont fortement réduit le score des *Ms. Pac-Man*. Cette expérience montre donc que l'utilisation d'une intelligence distribuée dans un jeu est possible. La Figure 1 montre les résultats de l'expérience en faisant s'affronter des IAs contrôlant Pac-Man ou les Ghosts contre les IA par défaut du jeu. Dans le tableau, \bar{s} représente le score moyen de Pac Man et σ représente l'écarte type. On peut donc voir que les IA Pac-mAnt et ghostAnt ont un score excellent ou bloquent fortement Pac-Man.

¹⁷ RECIO, Gustavo, MARTIN, Emilio, ESTÉBANEZ, César, et al. AntBot: Ant colonies for video games. IEEE Transactions on Computational Intelligence and AI in Games, 2012, vol. 4, no 4, p. 295-308.

	RandomGhost	PincerTeam	LegacyTeam	ghostAnt
Random	$\bar{s} = 502$ $\sigma = 156$	$\bar{s} = 375$ $\sigma = 250$	$\bar{s} = 172$ $\sigma = 61$	$\bar{s} = 216$ $\sigma = 57$
Random NonReverse	$\bar{s} = 1471$ $\sigma = 411$	$\bar{s} = 1856$ $\sigma = 746$	$\bar{s} = 1522$ $\sigma = 741$	$\bar{s} = 1251$ $\sigma = 440$
SimplePillEater	$\bar{s} = 5901$ $\sigma = 2196$	$\bar{s} = 4507$ $\sigma = 1776$	$\bar{s} = 5693$ $\sigma = 1423$	$\bar{s} = 3877$ $\sigma = 840$
SmartPillEater	$\bar{s} = 3539$ $\sigma = 790$	$\bar{s} = 3899$ $\sigma = 1265$	$\bar{s} = 4704$ $\sigma = 948$	$\bar{s} = 3418$ $\sigma = 961$
pac-mAnt	$\bar{s} = 49681$ $\sigma = 2047$	$\bar{s} = 53336$ $\sigma = 2094$	$\bar{s} = 36031$ $\sigma = 11819$	$\bar{s} = 16030$ $\sigma = 7068$

Figure 1 : Résumé des résultats expérimentaux par rapport aux contrôleurs par défaut

Amund TVEIT et al.¹⁸ ont proposé une plateforme pour qu'un système multi-agents puisse gérer un ensemble de joueurs et des PNJ dans un MMOG¹⁹. Ils ont implémenté quatre types de joueur, les Achievers qui ont pour objectif d'avoir le meilleur avatar possible. Les Socializers qui cherchent à entretenir des relations sociales avec les autres joueurs. Les Killers ont pour objectif de tuer les autres joueurs et PNJ. Et enfin les Explorers qui veulent découvrir le monde dans lequel ils évoluent. Les PNJ ont été basés sur le joueur de type Killers, mais il est facilement imaginable d'avoir d'autres types de PNJ, comme des Socializers. Leurs agents, les joueurs simulés et les PNJ, sont basés sur une approche Sense-Reason-Act. Cette approche consiste en une phase de captation des connaissances de son environnement. Le choix d'un plan d'action en fonction des connaissances à disposition de l'agent et enfin une phase d'action où l'agent va effectuer le plan qu'il a choisi. Cette approche semble intéressante pour le contrôle d'un ensemble de PNJ. En effet, l'idée de se baser sur une approche Sense-Reason-Act pour gérer des PNJ permettrait de voir apparaître des interactions avec les joueurs. Ce qui nous amène à une part importante du sujet de ce mémoire, l'émergence de comportement.

¹⁸ TVEIT, Amund, REIN, Oyvind, IVERSEN, Jørgen Vinne, et al. Scalable agent-based simulation of players in massively multiplayer online games. In : Proc. the 8th Scandinavian Conference on Artificial Intelligence (SCAI2003), Bergen, Norway. 2003

¹⁹ Massive Multiplayer Online Game

1.3 L'émergence de comportement

1.3.1 Qu'est ce que l'émergence de comportement

Nous pouvons d'abord nous demander ce qu'est l'émergence. Selon le domaine dans lequel le mot est employé il peut avoir une définition ou une autre. En biologie, elle peut être définie comme l'apparition d'un organe au cours de l'évolution d'un être. Il en existe bien d'autres définitions de ce qu'est l'émergence. Dans notre cas, nous pouvons définir l'émergence comme l'apparition de comportements nouveaux et complexes qui résultent de l'interaction et de la coopération entre des éléments plus simples.

Il existe un grand nombre de définitions pour décrire l'émergence de comportement, LI, SIM, et LOW en exposent plusieurs. Une première définition partagée par un grand nombre est que l'émergence de comportements est le résultat global de la coordination d'un groupe d'agents et non d'un seul individu²⁰.

Cet article donne d'autres définitions pour l'émergence de comportements tel que celle qu'expose Dyson George dans son livre "Darwin among the Machines: The Evolution of Global Intelligence". Celle-ci dit "Le comportement émergent est celui qui ne peut être prédit par une analyse à un niveau plus simple que celui du système dans son ensemble... Le comportement émergent, par définition, est ce qui reste une fois que tout le reste a été expliqué"²¹. Comme le dit LI Zhengping, cette définition appuie sur le fait qu'il est difficile de les prédire.

LI Zhengping donne une autre définition, celle de ROLLINGS A. & ADAMS E., l'émergence est le résultat complexe qui découle d'actions simples²². Un exemple que nous avons vu plus tôt s'accorde parfaitement avec cette définition. Effectivement, les Boids de Craig W.Reynolds sont sujets à 3 règles simples et ont pour résultat la simulation d'un vol d'oiseau ou d'un banc de poisson.

LI Zhengping, grâce aux définitions vues précédemment, définit l'émergence de comportements comme l'apparition des comportements du système résultant des interactions des individus et n'étant pas propriété de ces derniers.

²⁰ LI, Zhengping, SIM, Cheng Hwee, et LOW, Malcolm Yoke Hean. A survey of emergent behavior and its impacts in agent-based systems. In : 2006 4th IEEE international conference on industrial informatics. IEEE, 2006. p. 1295-1300.

²¹ Dyson & George B. (1997). Darwin Among the Machines:The Evolution of Global Intelligence. Perseus Book Group.

²² Rollings A. & Adams E. (2003). Andrew Rollings and Ernest Adams on Game Design. New Riders Publishing.

1.3.2 Exemples de comportements émergent

Il existe différentes recherches sur l'émergence de comportement, comme nous avons pu le voir plus tôt, les Boids sont un exemple de simulation du comportement naturel d'animaux vivant en groupe. Divers algorithmes permettent de voir émerger des comportements de groupe, les algorithmes génétiques permettent cela, comme le montre OLSON Randal dans "Predator confusion is sufficient to evolve swarming behavior"²³. Dans cet article, il est question d'un proie-prédateur où les agents sont gérés par un algorithme génétiques et la question posée est de voir si les proies vont développer un comportement de groupe face à des prédateurs. La figure 2 montre l'évolution du comportement des proies au cours du temps, nous pouvons voir que les proies ont tendance à se rapprocher et former de petits groupes.

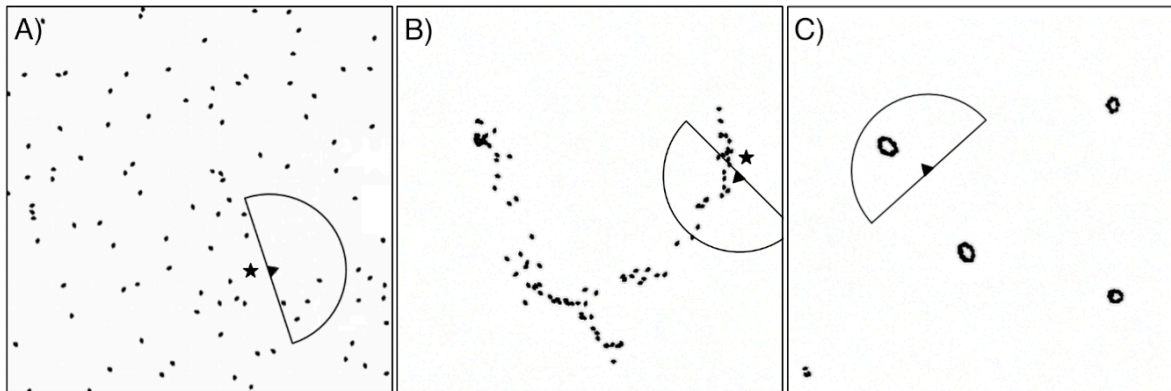


Figure 2 : Evolution du comportement des proies dans l'article de OLSON Randal

Les jeux vidéo sont un support intéressant pour l'intelligence artificielle, HONG Jin-Hyuk et CHO Sung-Bae ont utilisé un algorithme génétique sur *Robocode* un jeu de tir en vu top²⁴. L'objectif de leur recherche était de trouver diverses stratégies pour les PNJ afin que les développeurs n'aient pas à développer différents comportements complexes. Pour cela, ils ont utilisé un algorithme génétique pour voir émerger une multitude de comportements et garder les plus intéressants. Les résultats ont donné des stratégies intéressantes allant de l'agent qui charge l'adversaire au tireur d'élite en passant par le tir dans toutes les directions.

²³ OLSON, Randal S., HINTZE, Arend, DYER, Fred C., *et al.* Predator confusion is sufficient to evolve swarming behaviour. *Journal of The Royal Society Interface*, 2013, vol. 10, no 85, p. 20130305.

²⁴ HONG, Jin-Hyuk et CHO, Sung-Bae. Evolution of emergent behaviors for shooting game characters in robocode. In : *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*. IEEE, 2004. p. 634-638.

Cet article montre l'émergence de comportements d'un seul individu, nous pouvons donc nous demander s'il est possible de voir émerger des comportements de groupe comme dans l'article d'OLSON Randal dans jeu et de voir quelles seront les interactions de ces agents avec le joueur.

2 Développement et émergence de comportements collectifs

Cette partie du mémoire a pour objectif de présenter les modèles d'intelligence artificielle et les techniques employés pour la simulation de comportements collectifs, ainsi que leur utilisation dans les jeux vidéo. Ensuite, nous étudierons cet aspect d'un point de vue développement et de l'impact sur le ressenti du joueur. Enfin, nous terminerons ce chapitre sur la mise en place d'un plan d'expérimentation afin de faire une première approche sur les techniques à employer ainsi que la mise en place d'un projet de fin d'études qui servira de support pour répondre à la problématique du mémoire.

2.1 Utilisation des comportements collectifs

2.1.1 Les modèles d'IA pour la simulation de groupes

Dans un premier temps, nous pouvons nous demander si un modèle de machine learning est un outil intéressant pour la simulation d'un ensemble d'agents dans le cadre d'un jeu vidéo. En effet, le machine learning est utilisé dans le domaine de la recherche pour la simulation de foules, notamment pour des questions d'urbanisme. Par exemple, LEE, Jaedong, WON, Jungdam, et LEE, Jehee propose une simulation de foule avec un algorithme d'apprentissage par renforcement sur différents scénarios²⁵.

²⁵ LEE, Jaedong, WON, Jungdam, et LEE, Jehee. Crowd simulation by deep reinforcement learning. In : *Proceedings of the 11th ACM SIGGRAPH Conference on Motion, Interaction and Games*. 2018. p. 1-7.

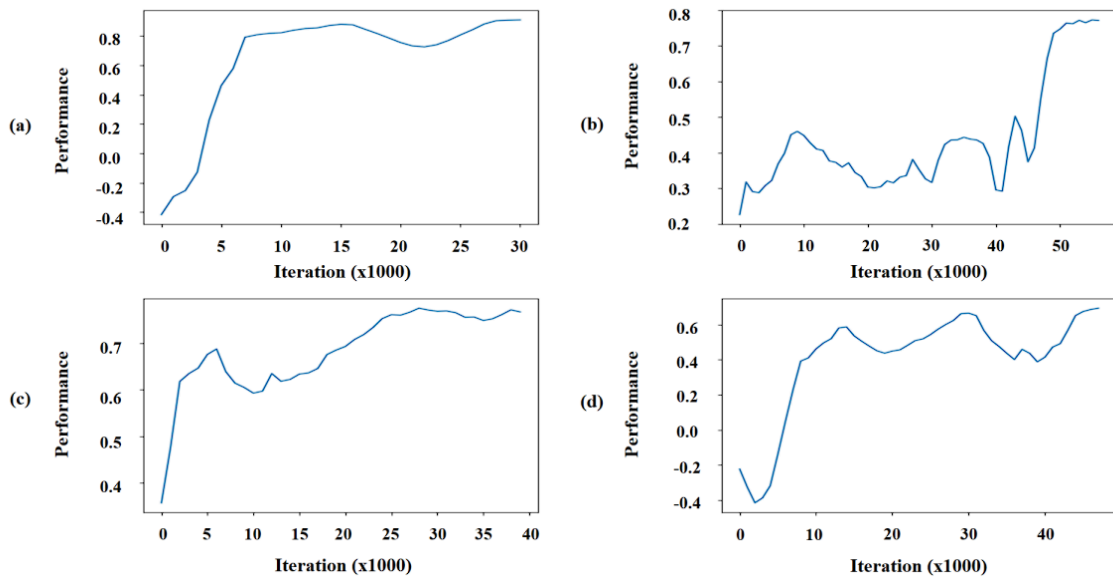


Figure 3 : Courbes d'apprentissage des différents scénarios

Comme le montre la figure 3, à force d'itérations les agents sont de plus en plus performants dans leurs déplacements. Cependant, cette méthode implique un grand nombre de simulations afin de voir un résultat efficace pour la simulation de groupe. Dans le cadre d'un jeu vidéo le machine learning permet un travail en amont pour le renforcement de comportements réaliste, mais également l'apparition de comportements inattendu qui peut être gardé comme l'ont montré les travaux de HONG Jin-Hyuk et CHO Sung-Bae²⁶. Comme nous l'avons vu dans la section 1.3.2, leur expérience consistait à l'émergence de comportements sur le *Robocode* en utilisant un algorithme génétique. Les résultats présentés montrent, après une grande série d'itérations, l'émergence de divers comportements, ceci, vus en détails dans la section 1.3.2, peuvent aller du comportement "Rambo", le personnage tire dans toutes les directions, au "Master shot", celui qui vise pour toujours faire mouche.

L'exemple de HONG Jin-Hyuk et CHO Sung-Ba nous montre la pertinence de l'utilisation du machine learning pour le développement de comportements en amont du jeu. Cependant, si l'adaptation est une force de certains modèles de machine learning, on peut donc se demander si l'emploi du machine learning est il possible pour une simulation en temps réel d'un groupe de PNJ ?

²⁶ HONG, Jin-Hyuk et CHO, Sung-Bae. Evolution of emergent behaviors for shooting game characters in robocode. In : *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*. IEEE, 2004. p. 634-638.

Nous venons voir que les algorithmes génétiques permettent l'émergence de comportements chez un individu avec les travaux de HONG Jin-Hyuk et CHO Sung-Bae. Mais nous avons aussi pu voir dans la partie 1.3.2, l'expérience du proie-prédateur de Randal OLSON²⁷. En effet, dans cet exemple, les proies, sous la pression des prédateurs, ont développé un comportement en essaim en se regroupant. De cette manière, les gènes permettant ce regroupement, se partagent et se renforcent au fil des itérations. Cet exemple montre donc l'apparition de comportements de groupe avec l'emploi d'algorithmes de machine learning.

En conclusion, l'emploi du machine learning, selon son utilisation, permet dans une phase préparatoire le développement de comportements de groupe, mais également en temps réel, avec une adaptation d'agents en réaction aux actions d'autres agents.

Cependant, comme le machine learning peut être lourd et difficile à mettre en place, nous pouvons nous demander s'il est possible de trouver une alternative permettant également l'implémentation de comportements de groupe adaptatifs.

Effectivement, il existe d'autres méthodes pour la simulation de groupe, comme le montre l'exemple vu dans la section 1.3.1, Zhengping Li et al où les systèmes multi-agents sont un modèle pour la simulation de comportements de groupe intéressant²⁸. Selon la définition de comportements en essaim, le comportement cohérent résultant d'un groupe d'individus, les systèmes multi-agents permettent cette émergence, car chaque individu dispose de son propre comportement et l'ensemble des individus permet de réaliser une tâche plus globale.

De plus, les systèmes multi-agents sont également utilisés pour la résolution de problèmes de simulation de foule. João Emílio de Almeida et al. montrent une simulation d'évacuation d'urgence en utilisant un modèle de système multi-agents²⁹. En effet, lors d'une évacuation d'urgence, l'humain sous la panique peut agir de manière irrationnelle et ainsi compliquer l'évacuation. João Emílio de Almeida et al. proposent une solution multi-agents avec des agents BDI (Belief-Desire-Intention) (Croyance-Désir-Intention). Les agents BDI permettent la bonne représentation d'un groupe ayant des objectifs différents (Désirs) et, en

²⁷ OLSON, Randal S., HINTZE, Arend, DYER, Fred C., *et al.* Predator confusion is sufficient to evolve swarming behaviour. *Journal of The Royal Society Interface*, 2013, vol. 10, no 85, p. 20130305.

²⁸ LI, Zhengping, SIM, Cheng Hwee, et LOW, Malcolm Yoke Hean. A survey of emergent behavior and its impacts in agent-based systems. In : *2006 4th IEEE international conference on industrial informatics*. IEEE, 2006. p. 1295-1300.

²⁹ ALMEIDA, João E., ROSSETI, Rosaldo JF, et COELHO, António Leça. Crowd simulation modeling applied to emergency and evacuation simulations using multi-agent systems. *arXiv preprint arXiv:1303.4692*, 2013.

prenant en compte les connaissances qu'ils ont à disposition (Belief), agir (Intention) pour remplir leurs objectifs. De ce fait, il est possible de voir la coordination d'un groupe pour essayer d'accomplir une tâche, mais aussi la possibilité d'avoir des agents avec des objectifs personnels, par exemple certains peuvent être plus altruistes et d'autres plus égoïstes. Les agents BDI sont un bon modèle d'agent pour la simulation de groupe, nous verrons en détails leurs utilisations dans le projet de fin d'étude présenté dans le chapitre 3 de ce mémoire.

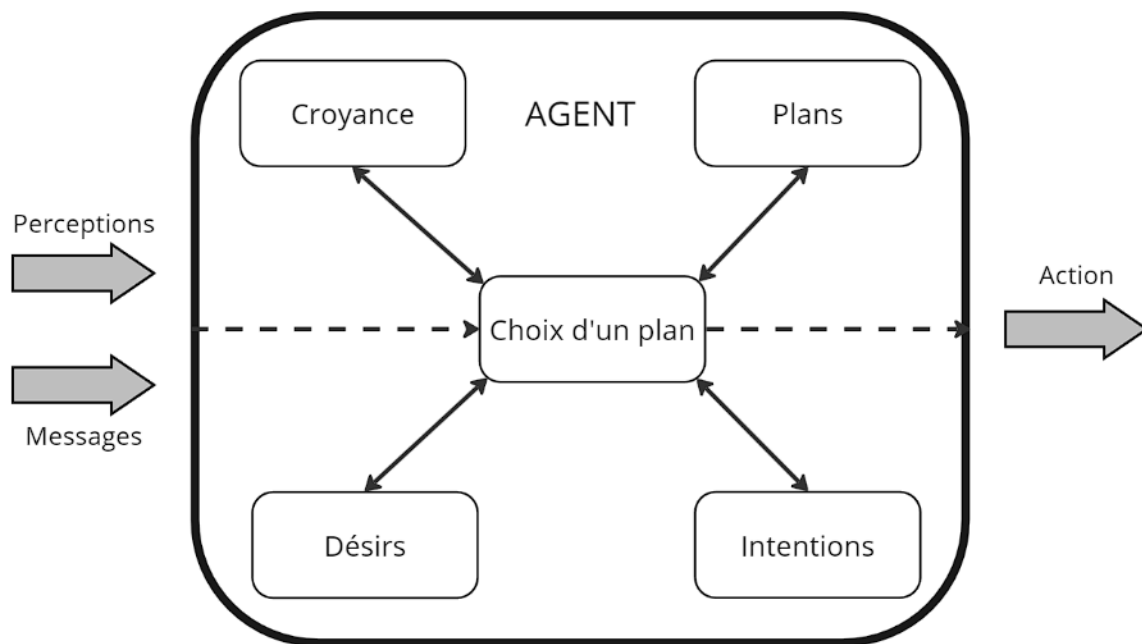


Figure 4 : Architecture d'un agent BDI

L'utilisation de systèmes multi-agents, notamment avec des agents BDI est une bonne solution pour la simulation de comportements de groupe. Toutefois, il existe d'autres méthodes de simulation de comportements de groupe.

Après avoir vu que les systèmes multi-agents seraient une bonne solution pour la simulation de groupes, est-il possible de simuler ces comportements à l'aide de règles simples ?

Nous avons pu voir dans la partie 1.2.1 les Boids de Craig W. Reynolds³⁰ seulement trois règles simples, cohésion, séparation et alignement, permettent de simuler de manière réaliste les comportements de vol d'oiseaux. Effectivement, les Boids de Craig W. Reynolds sont devenus une référence dans la simulation d'essaim et sont utilisés dans de nombreux domaines tels que le cinéma ou les jeux pour représenter de manière simple un essaim.

Les Boids s'approchent des automates cellulaires par le système d'information de voisinage. Ce qui nous amène à un autre exemple, le jeu de la vie de John Horton Conway, en 1970, qui consiste en une grille dont chaque case (cellule) peut être vivante ou morte. Puis avec deux règles simples il permet de faire vivre ou mourir une cellule, les règles sont : (ou est que, cf plus loin) si une cellule morte possède exactement trois voisins vivants alors elle naît. La seconde règle est que si une cellule est vivante et qu'elle possède deux ou trois voisins en vie alors elle reste en vie sinon elle meurt. Les voisins sont pris dans un voisinage de Moore, c'est-à-dire les cases adjacentes à l'horizontal, vertical et également en diagonale. Ces règles permettent l'émergence de certains comportements de groupe, comme les structures stables, des configurations ne changeant pas d'une itération à l'autre (cf. figure 5) ou bien les vaisseaux, des motifs se répétant au bout de quelques itérations tout en se déplaçant (cf. figure 6).

³⁰ REYNOLDS, Craig W. Flocks, herds and schools: A distributed behavioral model. In : *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*. 1987. p. 25-34.

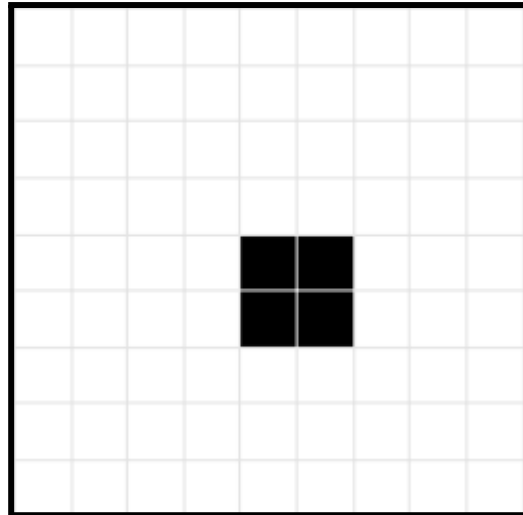


Figure 5 : Exemple du bloc, la plus petite structure stable

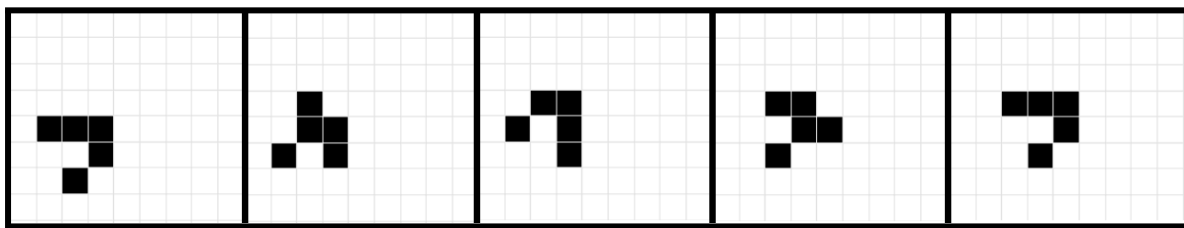


Figure 6 : cinq itérations du planeur

Bien que les règles du jeu de la vie soient simples, ce jeu est une machine de Turing et permet donc un grand nombre de calculs. En 2000, Paul Rendell a publié une machine de Turing dans le jeu de la vie³¹.

En définitive, il existe des modèles très simples pour voir émerger des comportements de groupe, au point qu'un ensemble de règles très simples peut aboutir à des comportements très complexes comme le montre le jeu de la vie.

En conclusion, nous avons pu voir qu'il est possible de simuler des comportements de groupes intéressants que ça soit avec l'utilisation du machine learning et d'algorithme génétique pour une adaptation en temps réel. Mais aussi nous avons pu voir qu'il existe d'autres moyens pour simuler des comportements de groupe, notamment avec les systèmes

³¹ RENDELL, Paul. Turing universality of the game of life. *Collision-based computing*, 2002, p. 513-539.

multi-agents ou des algorithmes bien plus simples tels que les Boids ou le jeu de la vie. Cela nous amène à nous demander leur utilisation dans les jeux vidéo.

2.1.2 L'utilisation dans les jeux

Nous venons de voir l'existence d'un grand nombre d'algorithmes permettant la simulation de mouvement et d'interaction de groupe. Cependant, qu'en est-il de leur utilisation dans les jeux vidéo ? Comment sont-ils employés dans les jeux ?

Nous avons pu voir dans la section 1.1.2, depuis 1970 l'intelligence artificielle dans les jeux vidéo est en perpétuelle évolution. Les comportements de groupe émergent dans un grand nombre de jeux, prenons l'exemple des fantômes de Pac-Man. Chacun des fantômes possède son propre comportement, ils partagent un pattern de chasse/repos. Durant les phases de repos, chacun se dirige vers son coin du tableau. Lors des phases de chasse, ils suivent les patterns suivants :

- Blinky (rouge) : suit directement Pac-Man
- Pinky (rose) : cherche à aller quelques cases devant Pac-Man
- Inky (bleu) : cherche à aller à une position devant Pac-Man à une distance équivalente à 2 fois celle entre Blinky et la cible de Pinky.
- Clyde (orange) : s'il est à plus de 8 cases de Pac-Man, il se dirige vers Pac-Man sinon il retourne dans son coin, celui en bas à gauche.

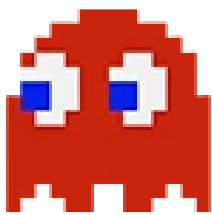


Figure 7.1: Blinky

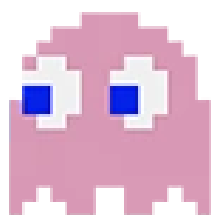


Figure 7.2: Pinky

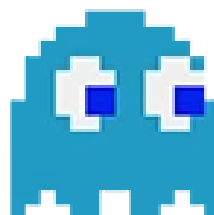


Figure 7.3: Inky

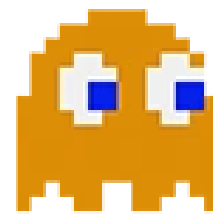


Figure 7.4: Clyde

Cet ensemble de règles simples permet d'avoir un groupe de chasseurs performant pour le jeu *Pac-Man*, l'un des jeux les plus populaires des bornes d'arcade. L'exemple des

fantômes de Pac-Man montre la possibilité de voir un comportement de groupe pertinent pour ce jeu avec des règles simples.

On peut se demander comment implémenter un comportement complexe chez un PNJ de manière simple ? Et si ces techniques permettent de faire émerger des comportements de groupe.

Outre les techniques que nous avons déjà, la technique des arbres de comportements (behavior tree) est fortement répandue dans l'implémentation d'IA pour les PNJ. Cette technique informatique, utilisée notamment en robotique et dans le jeu vidéo, permet l'exécution de tâches. À l'aide de plusieurs types de nœuds, cela permet au PNJ dans le cadre des jeux vidéo, de choisir l'action ou les actions à exécuter. Les feuilles de ces arbres sont les actions que peut réaliser le PNJ, puis il existe 2 types de nœuds, le Sélecteur qui permet d'exécuter le premier sous-arbre qui est réalisable et le Séquence qui permet d'exécuter une suite de sous-arbres. Yoones A. Sekhavat montre en détail comment fonctionnent les arbres de comportement, mais il montre également qu'ils sont devenus essentiels dans le développement des PNJ dans les jeux vidéo, de la conception à la gestion de ces derniers³². La figure 8 montre un arbre pour le comportement d'un PNJ dans un jeu de tir, le premier nœud teste si le joueur est visible, s'il ne l'est pas alors le PNJ va courir puis se mettre à couvert. Dans le cas où le joueur serait repéré alors l'arbre de comportements teste à quelle distance le joueur se trouve pour déterminer la manière de l'attaquer.

³² SEKHAVAT, Yoones A. Behavior trees for computer games. *International Journal on Artificial Intelligence Tools*, 2017, vol. 26, no 02, p. 1730001.

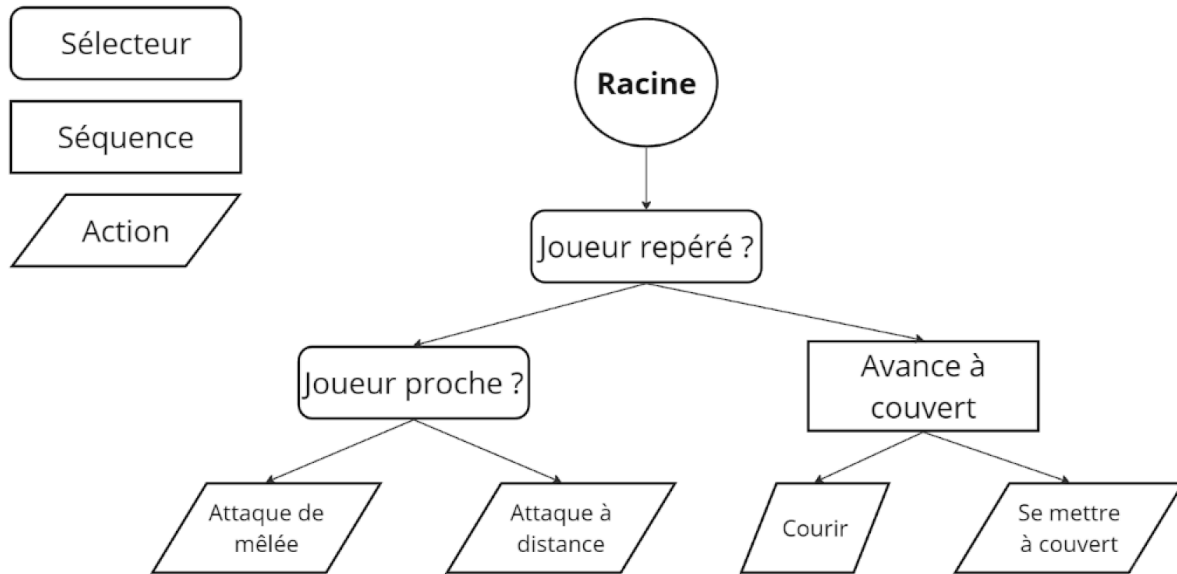


Figure 8 : Exemple d'arbre de comportement

Selon moi, il est possible d'implémenter des arbres de comportements complexes pouvant prendre en compte les autres PNJ et ainsi permettre une collaboration entre plusieurs PNJ pour réaliser une tâche et ainsi avoir un comportement de groupe.

Ces arbres semblent être une bonne solution pour l'implémentation de comportement, et dans des versions poussées, il est possible de les utiliser pour développer des comportements de groupe, par exemple avec la possibilité d'un PNJ d'aider un autre PNJ.

Les jeux de simulation de vie tels que *Les Sims* ou de gestion tels que *Space Haven* permettent-ils de voir l'apparition de comportements de groupe ?

Souvent dans ces jeux, les émotions des PNJ sont une part importante du gameplay, ainsi que les relations entre différents individus. Cela peut influencer les actions des PNJ, par exemple la possibilité de former des couples et ainsi lorsque les deux PNJ seront ensemble, il y aura un bénéfice à l'humeur de chacun. Les figures 9.1 et 9.2 montre les relations de deux PNJ dans une partie de *Space Haven*, on peut voir qu'un couple s'est formé, dans cette situation les PNJ Shelly et Cael peuvent dormir dans la même chambre sans que cela ne pose des problèmes sur leurs sommeils pour manque d'intimité.



Figure 9.1 : Relation d'un PNJ (Cael) dans une partie de Space Haven



Figure 9.1 : Relation d'un PNJ (Shelly) dans une partie de Space Haven

Les relations entre les PNJ dans *Space Haven* influent sur l'humeur de chacun, et donc influencent son comportement. De là, il est possible d'imaginer la possibilité de voir l'émergence de comportements de groupe entre différents PNJ.

Joon Sung Park et al. ont créé une simulation de la vie dans une petite ville, Smallville³³. Dans leur simulation, un ensemble d'agents vivent dans une petite ville, ceux-ci ont un modèle de mémoire, en partie servant à mettre en place un système de relations entre les agents. Ils reçoivent des informations, celles-ci seront sauvegardées et traitées afin d'élaborer des plans à long terme, mais aussi de choisir l'action à faire dans l'immédiat. Leur architecture utilise ChatGPT pour la communication des agents.

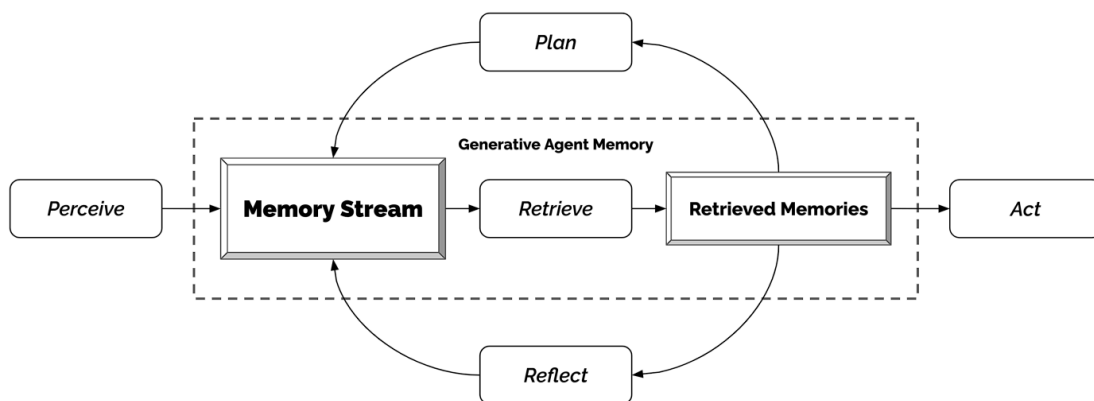


Figure 10 : Modèle d'agent de smallville

³³ PARK, Joon Sung, O'BRIEN, Joseph C., CAI, Carrie J., et al. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442*, 2023.

La simulation a pu voir émerger une évolution des relations entre les différents agents. En effet, durant la simulation, l'utilisateur a donné à un PNJ l'occasion d'organiser une fête, ce dernier a donc invité d'autres PNJ à sa fête dont certains, ont encore invité d'autres PNJ. Finalement, 12 PNJ ont reçu des invitations, mais seulement 5 sont venus à la fête, parmi eux des invités d'invités. Les PNJ n'étant pas venus avaient pour certains des raisons valables, le fait de devoir travailler, pour d'autres aucune justification. Cet exemple montre l'émergence d'un comportement de groupe, celui d'une diffusion d'information de manière autonome de la part des agents grâce aux relations qu'ils entretiennent les uns avec les autres.

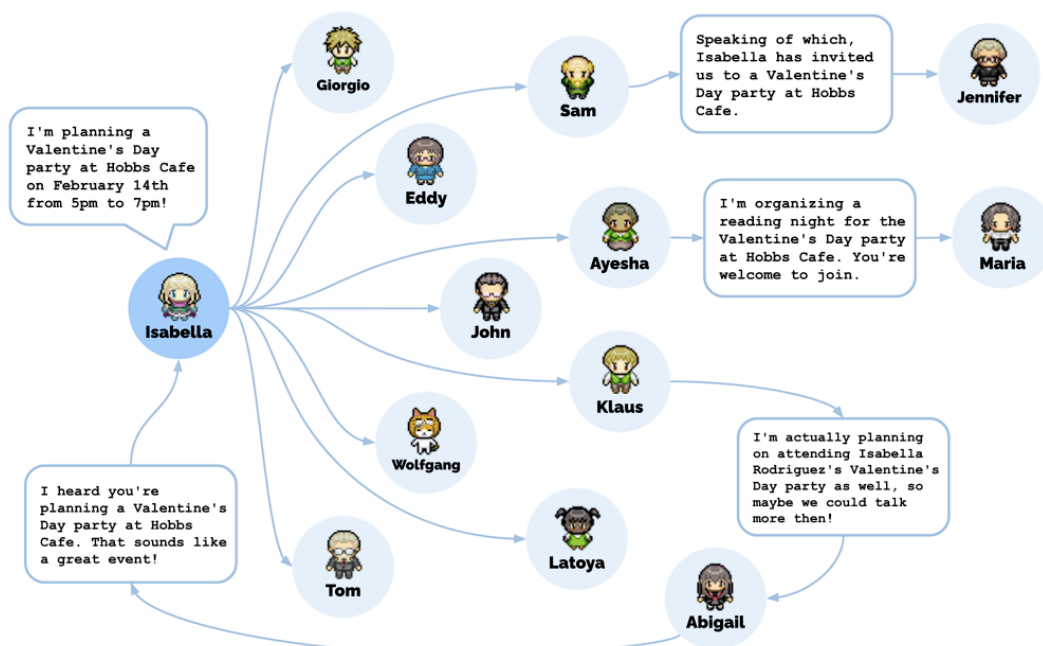


Figure 11 : Émergence de transmission d'invitation à une fête et de nouvelle relation

Les relations entre les PNJ sont un moyen intéressant pour voir l'apparition de comportements de groupe. L'exemple de smallville, le joueur a pris le contrôle d'un PNJ pour organiser un événement et cette information s'est propagée à un certain nombre de PNJ.

Nous venons de constater l'existence de plusieurs moyens pour la mise en place de comportements collectifs, allant du machine learning à des règles simples comme les Boids. Nous avons vu également leur présence dans l'industrie du jeu vidéo et qu'il existe déjà plusieurs méthodes pour l'implémentation de comportement. Nous venons également de voir

qu'un système de relations entre les différents agents permet de voir émerger des comportements sociaux. De fait, nous pouvons nous demander maintenant la pertinence d'un point de vue du joueur.

2.2 Développement et joueur

2.2.1 Pertinence dans le développement

Nous avons pu voir que le machine learning est un puissant outil d'intelligence artificielle et l'utilité des algorithmes plus spécifiques, tels que les algorithmes génétiques, pour réaliser des comportements adaptatifs. Cependant, son apport dans les jeux vidéo peut être contesté.

En effet, nous avons pu voir avec les travaux de HONG Jin-Hyuk et CHO Sung-Bae³⁴, que le machine learning peut être utilisé pour élaborer de nouveaux comportements pouvant être utilisés dans la version finale du jeu. Les résultats obtenus sont intéressants d'un point de vue recherche, mais les heures de travail ne seront pas perçues par les joueurs.

De plus, comme le disent Geoff SKINNER et Toby WALMSLEY³⁵, les joueurs sont bien plus aptes à percevoir les erreurs que les prouesses. La critique d'un jeu à sa sortie impacte fortement ces ventes et il est donc important pour les industriels de miser sur des éléments visibles plutôt qu'employer une technologie, certes de plus en plus efficace, mais coûteuse.

En effet, l'exemple de Smallville de Joon Sung Park et al.³⁶ montre une émergence de comportements sociaux très intéressante, cependant les chercheurs ont admis la difficulté des résultats obtenus.

En réalité, la simulation a demandé un grand nombre de stockage de données et certains agents n'ont pas pu développer des comportements crédibles à cause d'un manque de données. De plus, le modèle d'agent possède certaines faiblesses, ils peuvent par exemple être convaincus par l'utilisateur de l'existence d'un événement n'ayant jamais eu lieu. Enfin,

³⁴ HONG, Jin-Hyuk et CHO, Sung-Bae. Evolution of emergent behaviors for shooting game characters in robocode. In : *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*. IEEE, 2004. p. 634-638.

³⁵ SKINNER, Geoff et WALMSLEY, Toby. Artificial intelligence and deep learning in video games a brief review. In : *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*. IEEE, 2019. p. 404-408.

³⁶ PARK, Joon Sung, O'BRIEN, Joseph C., CAI, Carrie J., et al. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442*, 2023.

l'expérience a coûté des milliers de dollars d'utilisation de ChatGPT pour la simulation de seulement 2 jours de jeu.

Le machine learning est un outil puissant et en perpétuelle évolution, mais reste une nouvelle technologie qui coûte cher pour des résultats qui peuvent ne pas être perçus par les utilisateurs. C'est pourquoi nous pouvons nous demander ce qu'il en est du ressenti du joueur par rapport au résultat de l'intelligence artificielle sur la modélisation de comportements de groupe.

2.2.2 Ressenti du joueur

L'utilisation d'intelligence artificielle pour simuler des comportements de groupe doit avoir un impact sur le ressenti du joueur. En effet, le joueur doit pouvoir voir les résultats de ces comportements, de plus s'il peut ressentir les conséquences de ses actions, cela renforcera son expérience ludique. Non seulement cela donnerait un certain réalisme aux PNJ, mais également une plus grande rejouabilité au jeu. Plus précisément, le joueur aura envie de voir quel serait son impact sur le système s'il agit d'une manière ou d'une autre sur ces PNJ. Par exemple, imaginons un jeu de survie, le joueur pourrait être un chasseur et attaquer les animaux, ceux-ci pourraient développer un comportement de groupe comme dans l'expérimentation présentée précédemment de Randal OLSON³⁷. À l'inverse, si le joueur ne les chasse pas, les PNJ pourraient développer des comportements amicaux avec le joueur.

La rejouabilité d'un jeu est quelque chose d'important, comme le montre le jeu de la vie de John Horton Conway présenté dans la section 2.1.1. L'émergence de comportements de groupe dans ce jeu en a fait un sujet d'étude depuis 1970 et certaines personnes continuent de jouer et repoussent les limites de ce jeu. En 2005, Brice Due met au point une construction dans le jeu de la vie, le OTCA metapixel³⁸, qui permet de créer une cellule du jeu de la vie dans le jeu de la vie³⁹. Cette cellule fait 2048*2048 cases du jeu de la vie de base et elle permet de créer un jeu de la vie dans le jeu de la vie.

³⁷ OLSON, Randal S., HINTZE, Arend, DYER, Fred C., *et al.* Predator confusion is sufficient to evolve swarming behaviour. *Journal of The Royal Society Interface*, 2013, vol. 10, no 85, p. 20130305.

³⁸ "OTCA metapixel", LifeWiki, Decembre 2022, https://conwaylife.com/wiki/OTCA_metapixel

³⁹ PROKOPENKO, Mikhail, HARRÉ, Michael, LIZIER, Joseph, *et al.* Self-referential basis of undecidable dynamics: From the Liar paradox and the halting problem to the edge of chaos. *Physics of life reviews*, 2019, vol. 31, p. 134-156.

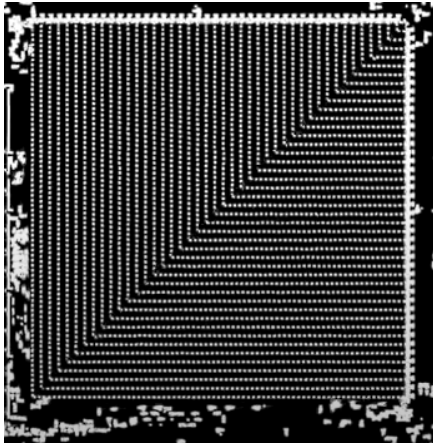


Figure 12.1 : Structure OTCA metapixel active

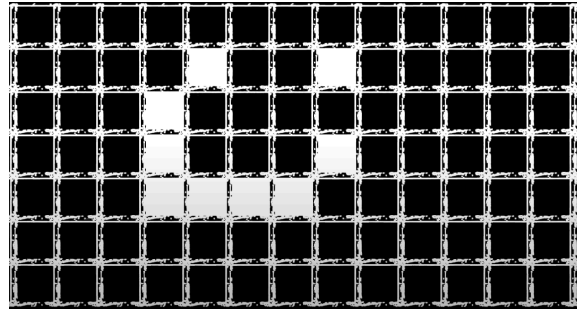


Figure 12.2 : Jeu de la vie dans le jeu de la vie

Le jeu de la vie montre un comportement de groupe et offre une rejouabilité infinie pour toujours créer de nouvelles constructions, par exemple le 26 mai 2023, Carson Cheng met au point le p78 pi-heptomino hassler⁴⁰. Cette construction est un oscillateur, basé sur le Pi-heptomino⁴¹, avec une période de 78 avant de retourner à son état initial. Cet exemple montre sa grande rejouabilité et qu'il existe encore des éléments à découvrir sur le jeu de la vie.

Un système de relations entre les PNJ permet également d'augmenter le ressenti du joueur, l'exemple des couples qui se forment dans Space Haven est une conséquence de l'organisation du joueur. Ces couples ont pu se former parce que le joueur a organisé les emplois de chacun pour qu'ils travaillent et se reposent aux mêmes horaires. Un autre exemple de système de relations impactant le ressenti du joueur est le système Nemesis de Shadow of war.

Chaque orc ayant gagné des rangs dans la hiérarchie fait partie du système Nemesis, il dispose d'un grand nombre de caractéristiques qui le rendent unique, il entretient également des relations avec les autres orcs. Ce système de relations offre, si deux orcs sont ennemis, des missions d'assassinat pour le joueur afin d'enrichir le gameplay. À l'inverse, si deux orcs ont de bonnes relations, l'un d'eux peut devenir le garde du corps de l'autre et ainsi, il viendra

⁴⁰ "Pi-heptomino hasslers - p78", LifeWiki, Juin 2023, https://conwaylife.com/wiki/Pi-heptomino_hasslers#p78

⁴¹ "Pi-heptomino", LifeWiki, Juin 2023, <https://conwaylife.com/wiki/Pi-heptomino>

en aide à son allié si le joueur l'attaque. De même, il est possible que des orcs soient frères de sang, dans ce cas si le joueur tue l'un d'eux alors l'autre tentera de le venger.

Le système Nemesis offre beaucoup de rejouabilité aux joueurs, car il fait également partie du système et peut agir dessus pour le faire évoluer comme il le désire.

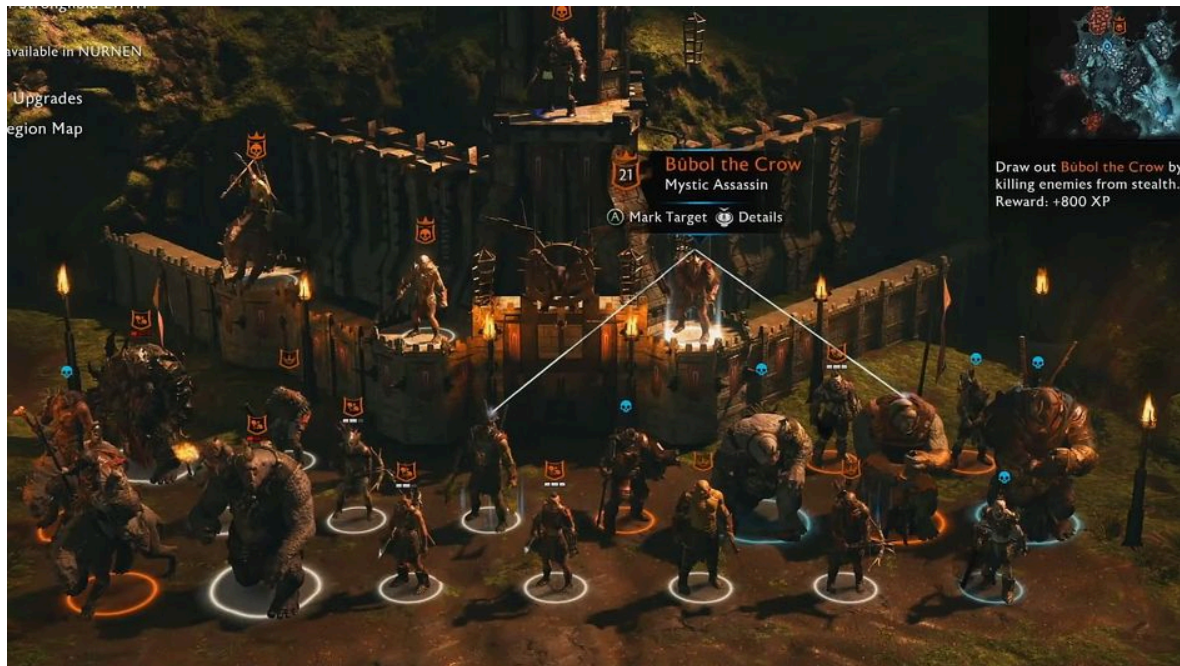


Figure 13 : Système Nemesis et relation

Nous avons donc pu voir qu'il est possible de créer des comportements réalistes et cohérents grâce à l'intelligence artificielle dans le but d'améliorer le ressenti du joueur. Cependant, nous pouvons nous demander s'il existe une limite à l'emploi d'intelligence artificielle pour la gestion des PNJ.

En 2017 AlphaGo bat l'un des meilleurs joueurs du monde de jeu de go Lee Sedol. Certes, cette victoire est une prouesse d'un point de vue technologique, mais elle est également la cause de la retraite de Lee Sedol, "Je ne suis plus au sommet, même si je deviens le numéro un. Il y a une entité qui ne peut plus être vaincue."⁴². Cet exemple montre

⁴²« « L'intelligence artificielle ne peut être vaincue » : un maître de go abandonne », lepoint.fr, Septembre 2019, https://www.lepoint.fr/high-tech-internet/l-intelligence-artificielle-ne-peut-etre-vaincue-un-maitre-de-go-abandonne-27-11-2019-2350129_47.php

que dans les jeux l'emploi du machine learning doit être modéré sinon elle peut être risqué de frustration pour le joueur.

En conclusion, l'intelligence artificielle, allant du machine learning aux algorithmes plus simples, est un outil puissant pour la représentation de groupes de PNJ pouvant être appliqués aux jeux vidéo. Bien utilisée, elle peut être un atout pour le design et le développement de comportements intelligents afin d'offrir une expérience toujours plus intéressante pour le joueur.

2.3 Plan d'expérimentation

Nous avons vu précédemment un certain nombre de modèles pouvant être utilisés pour la simulation de groupe dans un jeu. Nous nous sommes posés des questions sur l'impact du joueur sur ces groupes, mais également son ressenti, nous verrons donc dans cette partie un plan d'expérimentation afin de mettre en œuvre les fonctionnalités vues.

2.3.1 Base de l'étude

Tout d'abord, dans le cadre du master de Game Design, le projet doit être un jeu vidéo, ce qui est renforcé par le sujet de ce mémoire portant sur l'intelligence artificielle collective. De plus, les contraintes de temps imposent un modèle d'intelligence artificielle pouvant être mis en production rapidement, ainsi qu'un concept de jeu simple.

En revanche, les comportements de groupe résultant des interactions entre les PNJ doivent susciter l'intérêt du joueur. Ceci implique que le jeu doit dans un premier temps disposer d'un certain nombre de PNJ, mais aussi qu'il doit avoir une certaine richesse pour donner au joueur l'envie d'explorer les possibilités qui s'offrent à lui. Le temps restreint, ainsi que la simulation d'un grand nombre de PNJ, font que les systèmes multi-agents semblent être un modèle adapté pour la résolution de la problématique.

De plus, un jeu dans lequel le joueur ne dispose pas de stratégie optimale et qui est peu prévisible, renforcerait l'intérêt d'y jouer. Cela l'amènera à rejouer afin de tester un grand nombre de stratégies. James Carse définit ce genre de jeu comme infini, c'est-à-dire un jeu qui

n'a pas de fin et est peu prévisible sur ce qui peut arriver⁴³. Des événements inattendus pour mettre en difficulté le joueur, sans pour autant ruiner ses chances de succès, renforce ce côté infini du jeu et suscite encore plus l'intérêt du joueur.

Un jeu de stratégie mettant en œuvre un grand nombre de PNJ pouvant prendre leur propre initiative semble être un bon sujet d'étude pour la réalisation de comportements de groupe et leur émergence. De plus, nous avons vu dans la partie précédente qu'un système de relations est une aide pour l'émergence de comportements de groupe. Les relations entre les agents, PNJ et joueur, peuvent altérer leurs actions, par exemple un PNJ pourrait aider un autre PNJ dans une tâche si les deux entretiennent de bonnes relations. Comme nous avons pu le voir, les agents BDI disposent de croyance, en d'autres termes les éléments dont ils ont la connaissance. Cette architecture permet aussi d'avoir des objectifs personnels pour chaque agent, ce qui permet d'avoir des agents ayant des vocations différentes. Par exemple, il pourrait y avoir des agents ayant tendance à aider les autres tandis qu'à l'inverse certains pourraient être égoïstes. Le choix de l'action pourrait se faire en fonction des informations perçues, de ses relations avec les autres agents impliqués dans la situation, ainsi que de ses désirs personnels. Enfin, les arbres de comportement, natifs dans Unreal Engine 5, sont un bon outil pour l'implémentation des PNJ dans ce moteur de jeu et permettent d'implémenter facilement un agent.

⁴³ CARSE, James. *Finite and infinite games*. Simon and Schuster, 2011.

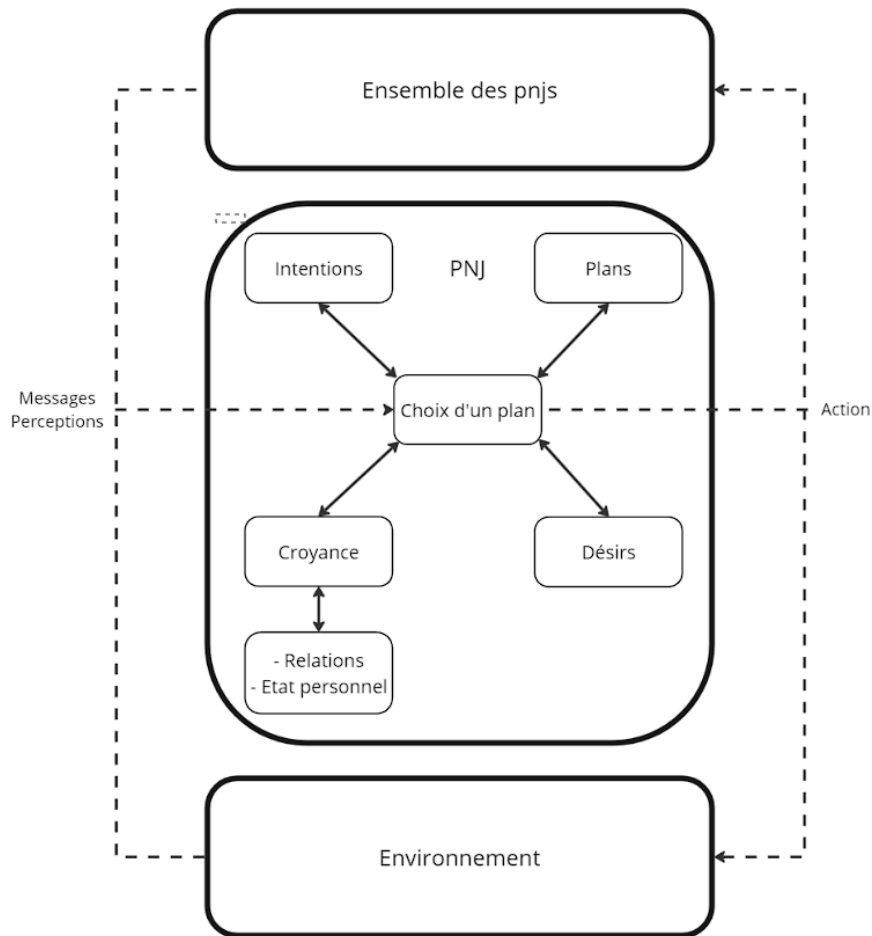


Figure 14 : Modèle d'agent avec relation influençant son environnement et les autres PNJ

Il existe un grand nombre de jeux mêlant stratégie et grand nombre d'agents permettant des comportements de groupe, cependant le projet final intègre d'autres sujets de recherche. C'est l'ensemble de ces sujets qui vont déterminer le choix du projet.

2.3.2 Choix du projet

Rappelons dans un premier temps le sujet de ce mémoire, celui-ci porte sur l'émergence des comportements de groupe chez les PNJ. Comme nous avons pu le voir précédemment, cela implique un grand nombre de PNJ pour pouvoir simuler ces comportements. En outre, nous avons pu voir qu'il existe des méthodes pour mettre en place

un système de relations entre les PNJ, mais il est également possible d'ajouter d'autres éléments pour créer un état bien défini qui pourront influencer les actions de ces derniers.

Le premier sujet de recherche porte sur la narration procédurale et son apport à l'expérience narrative d'un jeu. Ces deux premiers sujets sont fortement liés, effectivement les comportements des PNJ peuvent influencer la narration et inversement la narration peut influencer les comportements de ces derniers. Ces deux sujets confirment le nombre de PNJ, l'apparition d'événements liés aux actions des agents, mais également d'événements aléatoires pour renforcer le côté infini du jeu comme le définit James Carse.

Le second sujet de recherche pose la question de comment mettre en place un apprentissage éducatif directement par les mécaniques de jeu. Un type de jeu pouvant répondre à ces sujets serait un jeu de gestion mettant en œuvre un grand nombre de PNJ permettant ainsi la génération d'une narration procédurale. Cependant, pour répondre à ce troisième sujet, le projet doit avoir une dimension réaliste s'appuyant sur des faits réels. Pour cela, le projet sera un jeu de gestion en lien avec une période historique.

Enfin, le dernier sujet de recherche porte sur le lien entre la direction artistique des couleurs et les émotions du joueur. Ce sujet peut facilement être intégré dans un jeu de gestion historique notamment en donnant des pistes sur l'état du jeu par le biais de couleurs pour avoir un impact sur le ressenti du joueur.

Pour conclure, le projet de fin d'étude sera un jeu de gestion dans un cadre historique rappelant des faits réels. Une intention forte de ce projet portera sur la narration procédurale, mais également sur les actions des PNJ. De plus, le choix des couleurs aura une importance et permettra de donner des informations aux joueurs sur la situation du jeu. Enfin, ce jeu se positionne également dans le cadre d'un projet entrepreneurial et aura pour but d'être édité une fois terminé.

3 **Projet de fin d'étude : *Pirates : Set Sail And Discover !***

Pour répondre à la problématique de ce mémoire, Quelles conditions techniques sont nécessaires pour voir émerger des comportements de groupe chez les PNJ afin de voir s'ils vont aider le joueur ou au contraire lui faire obstacle ?, nous avons réalisé un projet de fin d'étude. Dans ce chapitre, nous présenterons, dans un premier temps, le projet que nous avons réalisé et le lien fort de ce sujet et de la narration. Nous verrons ensuite la conception de l'architecture d'agent choisi mais également la relation entre les agents et le joueur et comment ils s'influencent entre eux. Enfin, nous étudierons le système multi-agents, simulation de la vie sur un bateau pour renforcer l'immersion dans le jeu mais également l'impact du joueur sur ce système.

3.1 **Présentation du projet**

Comme nous l'avons vu dans la section 2.3.2, le projet était soumis à diverses contraintes liées aux sujets de mémoire des membres du groupe. Nous avons vu que les sujets de mémoire ont convergé vers un jeu de gestion dans un cadre historique. La partie jeu de gestion avec un grand nombre de PNJ répond aux problématiques liées aux comportements de groupe de ce mémoire et à la narration procédurale. Le cadre historique permet de répondre au sujet de mémoire portant sur le lien entre l'apprentissage éducatif et le gameplay. Enfin, la direction artistique du projet aura pour objectif de susciter des émotions aux joueurs par le biais du choix des couleurs.

3.1.1 *Pirates : Set Sail And Discover !*

Nous avons décidé de faire un jeu sur le thème des pirates, ce contexte permet d'orienter le cadre historique et la gestion d'un équipage de bateau. C'est pourquoi dans *Pirates : Set Sail And Discover !*, le joueur incarne un capitaine de navire et doit gérer la vie sur son navire. Cependant, il doit faire face aux comportements émergents des matelots et des événements inattendus et ainsi adapter sa stratégie en conséquence.

Le jeu *Pirates : Set Sail And Discover !*, se divise en plusieurs phases, la première, la phase de port, consiste à acheter des ressources pour remplir les stocks du bateau et afin de préparer la prochaine expédition en mer. Remplir les stocks est nécessaire pour le voyage, en effet la nourriture, la principale ressource, sert à la survie de l'équipage, sans cela les matelots mourront un par un et lorsqu'il n'y a plus d'équipage le joueur perd la partie. La seconde ressource principale sont les planches de bois. Il est possible que lors d'une tempête ou d'attaque d'un autre navire, la coque du bateau du joueur prenne des coups et se voit percée, le joueur aura donc besoin de planche pour bloquer les voies d'eau.

Vient ensuite la phase de navigation, le joueur a fait ses stocks pour le voyage et se lance dans l'exploration des mers. Le joueur a pour objectif de faire avancer son bateau en

gérant l'emploi du temps des matelots. Pour cela, il doit coordonner les actions des matelots pour gérer la barre, la voile et l'ancre.

Au cours de la traversée différents évènements vont venir perturber le voyage du joueur. Ceux-ci peuvent être plus ou moins importants, selon l'évènement le joueur devra donner des ordres spécifiques aux matelots pour gagner une récompense. Ces évènements peuvent être liés à la météo, par exemple, lorsque la pluie tombe, la cale se remplit lentement d'eau et le joueur devra assigner des matelots à la tâche d'écoper.

D'autres évènements plus importants mettront le jeu en pause et demanderont au joueur de prendre une décision rapidement et dans certains cas un vote sera réalisé avec les matelots.



Figure 15 : Vue du jeu Pirates : Set Sail And Discover !

Après cette présentation rapide du jeu, nous allons voir à présent le lien entre le sujet de ce mémoire est le sujet en lien avec la narration procédurale.

3.1.2 Lien avec la narration procédurale

Comme nous avons pu voir précédemment, ce projet de fin d'étude est réalisé conjointement avec d'autres sujets de mémoire. Cependant, le sujet de ce mémoire est fortement lié au sujet portant sur la narration procédurale et son apport à l'expérience narrative d'un jeu.

Le principal lien entre les deux sujets sont les matelots. Ils sont au cœur du projet, ils gèrent le bateau et forment la vie à son bord. Pour renforcer la narration ceux-ci disposent

d'une identité, de traits de caractères et d'affinité qui influencent les actions qu'ils entreprendront de leur propre chef. Ils disposent également de relations qu'ils ont avec les autres personnages du bateau, les autres matelots et le joueur. Tous ces éléments influencent les actions qu'ils font mais également leurs donnent une profondeur dans leur histoire.

En effet, l'objectif de ce mémoire étant de voir un groupe de PNJ agir de manière coordonnée et réaliste avec des possibilités d'aller à l'encontre du joueur. De ce fait, il est possible de voir certains PNJ ne pas écouter les ordres donnés par le joueur et de n'en faire qu'à leur tête. Le lien entre ces deux sujets de mémoire permet également l'apparition d'événements spécifiques, par exemple si une grande partie des matelots entretiennent de mauvaises relations avec le joueur cela peut entraîner des événements pouvant conduire à la défaite du joueur. Cela montre que le sujet de ce mémoire est au service du sujet sur la narration.

Cependant, il est possible de voir la narration influencer les comportements des PNJ. Comme nous avons pu voir, il est possible d'avoir des événements nécessitant un vote pour qu'un choix soit pris. Dans ce cas, le choix de chaque matelot sera influencé par ses affinités mais également ses relations qu'il entretient avec les autres. Par exemple, si un matelot combatif, ayant de mauvaises relations avec le joueur et si ce dernier a voté pour fuir un combat, alors ce matelot votera pour attaquer.

De plus, l'issue du vote peut influencer les caractéristiques et les relations du matelot. En effet, si le résultat du vote va à l'encontre des désirs d'un matelot alors ce dernier peut voir ces relations évoluer avec les autres matelots. En reprenant l'exemple du matelot combatif, si le résultat du vote est de fuir alors ce dernier verra ces relations diminuer avec ceux qui ont voté la fuite et à contrario les voir augmenter avec ceux qui étaient pour combattre.

Nous avons vu l'idée de ce projet de fin d'étude, ainsi que le lien fort entre le sujet de ce mémoire et celui sur la narration procédurale, chacun d'eux sert l'autre pour faire évoluer le système. Nous allons voir maintenant en détail la conceptualisation des matelots.

3.2 Conceptualisation des Matelots

Notre jeu tournant beaucoup des matelots, leur conceptualisation était un point important de la phase de préproduction. Dans cette partie nous verrons dans un premier temps la conception du matelot puis dans un second temps les interactions entre eux et le joueur.

3.2.1 Modèle d'agent pour les matelots

L'objectif de l'ensemble des matelots est de recréer la vie sur le bateau, pour représenter cela, ils ont besoin d'un grand nombre d'éléments. Nous avons divisé ces éléments en trois catégories, les paramètres vitaux, l'identité et les actions possibles.

La première de ces catégories, les paramètres vitaux, comprend l'ensemble des variables qui fluctuent au cours du temps. Parmi ces variables, les principales sont la faim et la fatigue, car elles vont gérer les actions primaires du matelot mais également les amener à vivre ou mourir. En effet, si un matelot ne peut plus manger alors il finira par en mourir, la gestion des stocks de nourriture dans le bateau avant une expédition est au cœur de notre gameplay. Un autre point important que nous voulions mettre pour les matelots est la relation qu'ils entretiennent entre eux mais également avec le joueur.

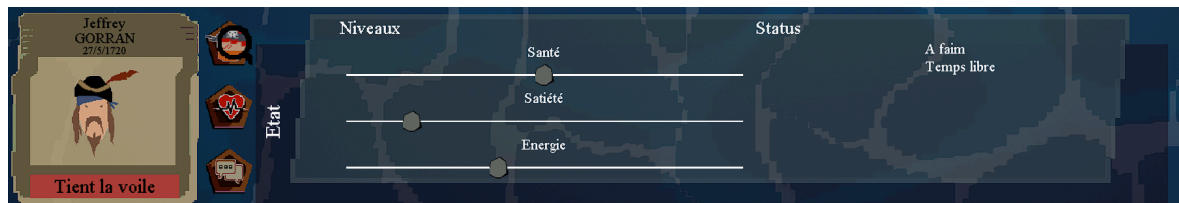


Figure 16 : Variable d'un matelot

Comme nous l'avons vu les parties 2.1.2 et 2.2.2, un système de relation permet d'apporter beaucoup de profondeur au réalisme du jeu mais également avoir un impact sur le joueur. Nous en avons donc mis un en place, chaque matelot dispose d'une variable allant de 0 à 100 pour le joueur et chaque autre matelot. Plus la valeur entre deux matelots est élevée, plus la relation est bonne entre eux. Cette relation peut influencer sur les actions qu'entreprend le matelot, par exemple plus il déteste le joueur plus il a de chance d'ignorer les ordres qu'il reçoit. Du point de vue entre matelots cela peut influencer les discussions qu'ils ont entre eux mais celles-ci peuvent également faire évoluer cette relation. Par exemple, si deux matelots ont de bonne relation, le sujet d'une conversation aura plus de chance d'être positif et ainsi renforcer cette relation. Enfin, lors des phases de votes de l'ensemble de l'équipage, le vote de tous les matelots ne se fait pas simultanément, certains voteront plus vite que d'autres ce qui influencera les plus lents. Par exemple, si l'ami d'un matelot vote pour le choix 1 alors cela pourra influencer le sien et ainsi augmenter ses chances de voter aussi pour le choix 1.

Nous avons également conceptualisé d'autres variables moins importantes, elles ne sont pas présentes dans la première version de ce jeu à cause des courts délais de réalisation. Celles-ci sont liées à l'état émotionnel, la maladie et l'alcoolémie du matelot et peuvent ainsi influencer les actions de ce dernier.

La deuxième catégorie, l'identité du matelot, offre principalement de la profondeur à l'histoire de chaque matelot. En effet, le nom, l'âge, la nationalité et l'histoire d'un matelot ne sont pas les éléments qui influencent ses actions. En revanche, les traits de caractère, ceux-ci peuvent influencer sur le choix de l'action à effectuer mais également sur la manière d'effectuer cette action. Par exemple, un matelot disposant du trait charitable aura tendance à partager sa nourriture avec ceux qui en ont le plus besoin. De même que pour les traits de caractère, les matelots disposent de métiers ou tâches qu'ils aiment faire. Ces éléments, comme les traits, influencent le choix des actions et la manière de les effectuer. Un matelot

pêcheur aura tendance à aller pêcher durant son temps libre mais aura également plus de chance de ramener du poisson.

Enfin, la dernière catégorie, les actions du matelots, celles-ci peuvent aller du plus élémentaire, manger ou dormir, à des actions plus spécifiques telles que tenir la barre ou jeter l'ancre. Certaines de ces actions peuvent être ordonnées par le joueur, comme tenir la barre, ou bien sont propres au matelot et seul lui peut décider de la faire, comme aller dormir. Les matelots disposent d'un emploi du temps, pour déterminer à quelle heure ils doivent travailler ou se reposer. De plus, les matelots peuvent avoir une tâche principale à effectuer durant leur temps de travail. Si cette action n'est pas réalisable alors il va essayer d'en trouver une autre. Par exemple, si le matelot a comme tâche principale de laver le pont et que celui-ci est déjà propre, alors il décidera d'effectuer une autre action, comme par exemple de tenir la barre si celle-ci n'est pas occupée.

Lors de son temps libre, le matelot décide par lui-même d'effectuer une tâche, celle-ci sera plutôt pour le servir lui-même plutôt que l'avancée du bateau dans le voyage. Cependant, dans certains cas, un matelot en temps libre pourra effectuer des actions de travail si la situation l'en oblige. Dans le cas où la cale est remplie d'eau, sa vie étant en danger il ira aider ces camarades pour la vider et éviter le naufrage.

Enfin, le joueur peut donner un ordre à un matelot, celui-ci arrêtera ce qu'il est entrain de faire pour effectuer cette nouvelle action. Selon la situation et l'action ordonnée par le joueur, la relation entre les deux pourra se voir modifier. De même si un matelot travaille trop, sa relation avec le joueur baissera et inversement s'il ne travaille pas.

Pour le développement des matelots nous avons opté pour un système multi-agents. Comme nous avons pu voir dans la partie 2.1.1, les agents BDI (Belief-Desire-Intention) (Croyance-Désir-Intention) sont un bon type d'agent pour notre projet. Un agent BDI possède donc des Croyances, correspondant aux informations qu'il possède sur son environnement et les autres agents, celles-ci ne sont pas forcément correctes. Une information peut être incorrecte par exemple si un matelot donne de fausses informations à un autre matelot. Les Désirs d'un agent correspondent à l'ensemble des objectifs d'un agent, pour notre projet nous aurions aimé avoir des agents avec des objectifs personnels, cependant au vu du temps limité les désirs de nos agents s'orientent principalement autour de la navigation et de la gestion du joueur mais nous verrons plus tard de leur propre survie. Enfin, les Intentions sont les actions que l'agent a décidé d'accomplir afin de réaliser un de ses Désirs.

Nous avons donc conceptualisé nos matelots comme des agents BDI, en plus des éléments présentés en début de partie, nous avons ajouté un système de croyance, celle-ci correspondant aux informations que peut avoir un agent sur l'état du bateau, sur les ressources manquantes ou sur un autre agent. En fonction des connaissances qu'ils ont sur leur état, celui du bateau et des autres agents, un matelot décide d'entreprendre une action spécifique pour subvenir à ses besoins, manger, dormir, s'occuper. La figure 17 montre le modèle d'agent utilisé pour les matelots, nous pouvons voir qu'il y a une grande part du modèle d'agent BDI auquel se greffe les ordres du joueur, les relations et l'emploi du temps.

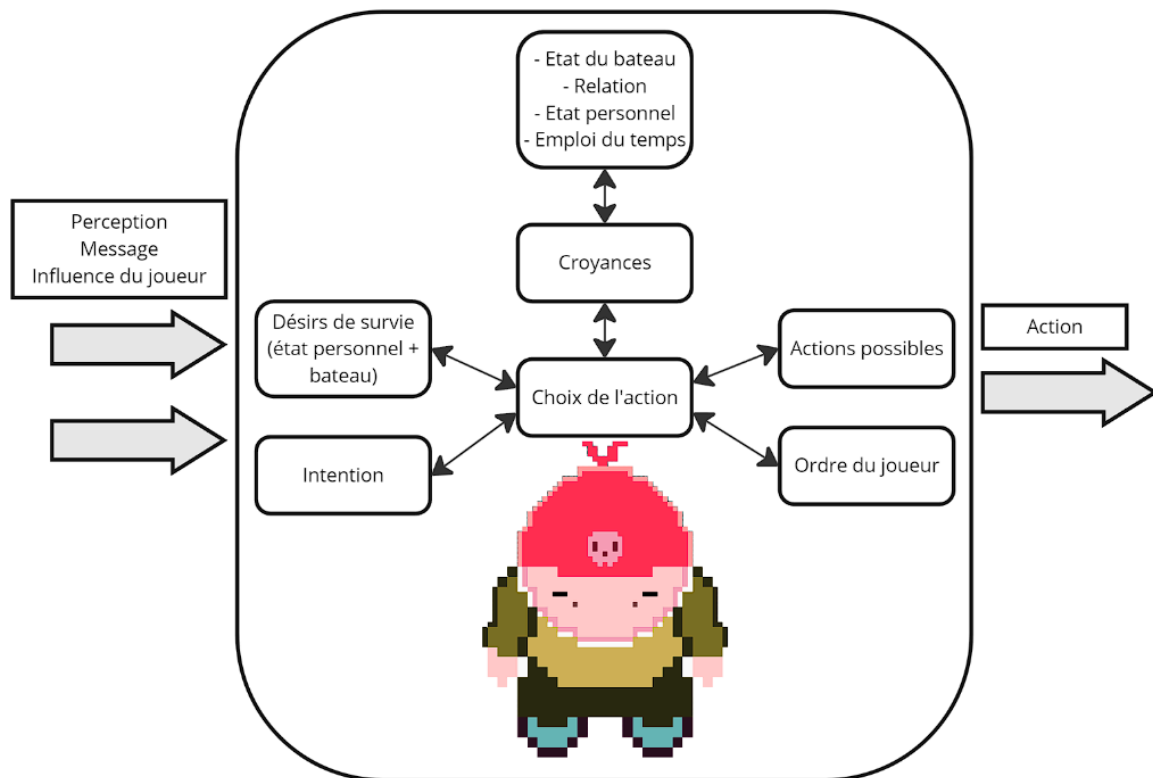


Figure 17 : Modèle d'agent adapté pour les matelots

Nous venons de voir notre adaptation d'un agent BDI pour représenter nos matelots, cependant s'ils disposent d'une autonomie complète, le jeu devient une simple simulation. C'est pourquoi nous verrons maintenant comment intervient le joueur sur les matelots.

3.2.2 Interaction entre le joueur et l'équipage

Comme nous venons de le voir, les matelots ont une certaine autonomie, lors de leur temps libre ils s'occupent par eux mêmes et pendant leurs heures de travail ils savent gérer le bateau si besoin. Cependant, le fait que les matelots savent vivre en autonomie amène au fait que le joueur n'a que peu d'implication dans le jeu. Certes, il doit gérer les stocks avant chaque voyage et l'emploi du temps des matelots afin d'avoir une équipe de nuit et une équipe de jour. Pour augmenter son implication dans le jeu, nous avons limité l'autonomie des matelots sur certains points, le joueur peut ordonner des actions aux matelots mais aussi ces derniers n'effectuent certaines tâches qu'en cas critique.

Pour impliquer le joueur d'avantage dans le jeu, le joueur doit forcer certaines actions des matelots pour avancer. Ces actions peuvent être semi automatique durant le temps de travail des matelots. Par exemple, le joueur peut assigner une tâche prioritaire aux différents matelots, ceux-ci, durant leur temps de travail, feront cette tâche en priorité, s'ils ne peuvent pas ils feront autre chose. Ces actions donnent une certaine implication du joueur et augmentent le côté gestion de l'équipage.

Durant certains événements, le joueur peut ordonner aux matelots de faire des actions spécifiques qui auront une chance de succès augmentée. Par exemple, lorsque le joueur reçoit un événement mineur indiquant un grand banc de poisson, les matelots ne changeront pas de tâche seuls mais le joueur aura intérêt de leur ordonner de pêcher pour remplir les stocks de nourriture.

Enfin d'autres actions, celles à faire durant les cas critiques ne sont pas réalisées directement par les matelots. En effet, ces actions doivent être faites par le joueur pour que la situation soit gérée dans de bonnes conditions. Cependant, comme nous l'avons vu, les matelots ont pour désir de survivre, et donc si la situation devient trop critique alors ils prendront en charge d'essayer de réparer la situation mais cela aura de fortes conséquences sur les relations avec le joueur. Ces situations sont liées aux manques de ressources, ou lorsqu'il y a trop d'eau dans la cale ou un grand nombre de voies d'eau. Tenter de trouver de la nourriture, réparer la coque ou écoper deviendront les tâches prioritaires des matelots. Si les matelots arrivent tout de même à faire survivre le bateau, et donc éviter un game over, les relations faibles avec le joueur auront de lourdes conséquences par la suite.

Comme nous l'avons précédemment, les relations joueur / matelot sont une part importante des agents. Si les relations sont bonnes, les matelots auront tendance de suivre le joueur dans ces décisions lors d'un vote. En revanche, si elles sont mauvaises, les conséquences seront plus graves. En plus du fait qu'ils ne le suivront pas dans ses décisions, ils n'écoutent plus tous les ordres que lui impose le joueur.

Enfin, si la situation se dégrade avec un grand nombre de matelots, les conséquences n'en seront que plus graves. Si la relation joueur / équipage est trop faible alors il est possible que l'équipage se mutine et si le joueur ne gère pas bien la situation alors la mutinerie aura bien lieu et implique une défaite du joueur.

Nous venons de voir les interactions qu'il peut y avoir entre le joueur et l'équipage, ainsi que l'autonomie des matelots sur le bateau et leur gestion en groupe du bateau tout en laissant une part d'implication du joueur dans le processus. L'objectif du projet est également d'avoir une dimension réaliste de la vie sur un bateau à l'époque. Pour finir ce chapitre nous verrons comment évolue la vie et l'immersion sur ce bateau.

3.3 La vie sur le bateau

Comme nous avons pu le voir, l'objectif du projet est de donner une dimension historique à notre jeu tout en ayant une grande part à la génération procédurale. Le joueur doit pouvoir se sentir directement sur le bateau, et donc pour nous de simuler la vie dans notre jeu. Nous verrons dans un premier temps quels sont nos choix pour immerger le joueur dans notre univers pirates et enfin nous parlerons des interactions des matelots avec le joueur.

3.3.1 Simulation de la vie sur le bateau

Un de nos objectifs est de représenter la vie sur le bateau et que nos matelots agissent comme bon leur semble de manière à faire vivre le jeu. Pour ce faire nous avons décidé d'utiliser différents éléments pour argumenter l'immersion du joueur dans notre jeu.

Nous avons décidé d'avoir un côté bande dessinée et 2D dans notre jeu, les actions des matelots sont représentées par des bulles ou des animations de sprite, par exemple lorsqu'un matelot dort, une série de Z apparaissent au-dessus de sa tête. Cependant, le point fort de cet élément porte sur les discussions entre les matelots. Un matelot peut aller parler à un autre matelot, le sujet de cette discussion peut être positif, négatif ou neutre et sera déterminé en fonction de la relation qu'entretiennent les deux matelots. Par exemple, un matelot allant discuter avec un ami a plus de chances d'avoir une discussion positive avec lui. Visuellement la conversation se voit avec une bulle représentant le sujet de la conversation. Ceci donne une plus grande profondeur sur les matelots qui discutent entre eux, le joueur peut voir rapidement qu'un matelot discute avec un autre matelot, sur quel sujet mais également quel sera l'impact de cette discussion sur la relation qu'ils entretiennent.



Figure 18 : Conversation entre deux matelots

En plus des effets visuels pour représenter la vie sur le bateau, les effets sonores sont très importants, ils apportent une ambiance importante sur le bateau. Le piratish et la musique intradiégétique sont deux éléments qui augmentent la simulation de la vie et l'immersion du joueur. Dans *Les Sims*, les personnages communiquent en parlant le *Simlish*, cet élément donne une bonne immersion dans le jeu, en effet, les personnages sont dotés de parole ce qui leur donne vie. Pour nos matelots, et pour leur donner vie nous avons décidé de les doter de parole, dans notre cas, le piratish n'est pas une nouvelle langue mais des bruits ou grognement. Lorsque les matelots discuteront entre eux en plus des bulles ils parlent vraiment.

Enfin, nous avons décidé que durant une partie, qu'il n'y ait aucun son extra diégétique. Nous trouvant sur un bateau, les sons de vague, cordage, bois sont importants mais également la musique jouée par les pirates. Pour cela nous avons fait le choix que la musique durant une partie est celle jouée par les matelots et du coup avoir une musique intradiégétique. Cet élément renforce l'objectif de donner la vie sur le bateau et les matelots peuvent donc organiser de petit orchestre en jouant ensemble. Disposant pour le moment de trois instruments, ils peuvent donc varier les musiques.

Tous ces ajouts aux actions que peuvent déjà entreprendre les matelots, permettent d'observer un équipage œuvrant en groupe pour faire vivre le jeu. Cet équipage, en partie géré par le joueur, dispose d'une autonomie pour agir selon bon lui semble pour faire avancer le jeu. Cependant, le joueur fait également partie de cette vie et influe sur cet équipage, mais celui-ci interagit également avec lui.



Figure 19 : Simulation de la vie sur le bateau

Les matelots peuvent donc recréer une forme de vie en communauté sur le bateau, de plus comme nous l'avons vu plus tôt, le joueur fait également partie du système et donc influe sur l'équipage.

Un des objectifs des matelots, pour donner un certain réalisme, est de survivre si la situation devient trop critique. De ce fait, les matelots réparant la coque ou écopant sans demande du joueur, agissent sur le court terme pour la survie du joueur et tentent d'éviter le game over par naufrage du navire mais pas sans impact sur les relations entre le joueur et les matelots. Cependant, cet impact peut avoir des répercussions négatives sur le long terme, en réalité, si les relations avec l'équipage deviennent trop faibles, l'apparition d'une mutinerie

arrive et si le joueur ne la gère pas correctement alors la partie se terminera par un game over par mutinerie.

De plus, les connaissances sur le manque de ressources peuvent être transmises pendant les conversations entre les matelots. Ce partage de connaissances peut avoir des répercussions aussi bien négatives que positives pour le joueur, un matelot peut apprendre le restockage d'une ressource par un matelot avant de s'en rendre compte par lui-même et inversement.

Enfin, les relations entre l'équipage et le joueur peuvent fortement influencer le résultat d'un vote. La relation entre un matelot et son capitaine peut influencer la prise de décision lors d'un vote. De plus, le joueur vote en premier mais les matelots ne votent pas simultanément, ce qui peut entraîner qu'un groupe de matelots ayant de bonnes relations puisse converger vers les mêmes votes. Sachant que ce système de vote est important dans la piraterie, historiquement beaucoup de décisions sur un bateau se faisaient démocratiquement, beaucoup de choix voulant être faits par le joueur devront être soumis au vote de l'équipage et donc entretenir de bonnes relations avec une majorité est important.

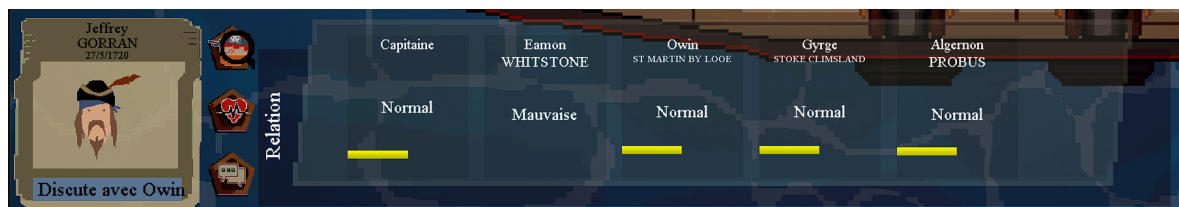


Figure 20 : Relation d'un matelot

3.3.2 Résultats observés

Dans la première version de *Pirates : Set Sail And Discover !*, les matelots sont soumis à la faim et à la fatigue, entretiennent des relations entre eux et disposent d'un certain nombre d'actions possibles. Ils disposent également d'une première version de *Croyances* sur des informations qu'ils ont pu avoir, le manque de ressources, le niveau d'eau et les trous dans la coque. Ces éléments permettent au matelot de choisir une tâche de manière simple, les situations critiques en priorité puis, selon son emploi, du temps d'autres tâches de travail ou des activités plus reposantes. Le système de discussion permet de faire évoluer les relations entre les matelots et le joueur, de plus elles peuvent transmettre des informations sur le manque d'une ressource pour renforcer le système multi-agents du projet. Malgré les contraintes de temps qui ne nous ont pas permis de mettre en place tous les éléments voulus, les agents peuvent être améliorés, par exemple on peut augmenter le nombre d'actions possibles, ou introduire de nouvelles variables en plus de la faim et de la fatigue, telle que la maladie. Cependant nous pouvons déjà observer une communauté qui se développe sur le

bateau, par leurs occupations et leurs discussions, une vie se forme. Également, lors d'un voyage, un matelot seul ne peut effectuer toutes les tâches pour faire avancer le navire, cependant en groupe ils peuvent faire avancer le bateau de manière autonome. Par souci d'intérêt du joueur, leur autonomie a été limitée et ont besoin des ordres donnés par le joueur pour agir sur certaines actions, comme monter ou larguer l'ancre.

Conclusion

Malgré les contraintes de temps, la première version de *Pirates : Set Sail And Discover !* permet déjà de voir des comportements de groupe et permet de répondre en partie à la problématique soulevée en début d'étude. Les matelots discutant entre eux montrent une évolution des relations et donc des conversations qu'ils ont. De plus, la possibilité de voir un équipage de matelots autonomes est possible mais pour susciter l'intérêt du joueur, celle-ci reste limitée. Enfin, pour donner plus de réalisme, en cas de situation critique, les matelots gagnent en autonomie pour représenter leur désir de survie. Les matelots développent donc des comportements de groupe qui sont influencés et influencent les actions du joueur.

Le système multi-agents est donc une bonne technique pour le développement de comportements de groupe. Ajouté à cela les agents BDI permettent d'introduire les croyances potentiellement vraies qui influent sur le choix des actions pour atteindre les désirs de l'agent. Cependant, le modèle des matelots peut être plus poussé en augmentant les croyances qu'ils peuvent avoir et la possibilité de les transmettre. Et aussi comme nous l'avons vu pour le machine learning, les systèmes multi-agents sont développés en amont de l'expérience ludique. En effet, cette méthode est limitée dans le cadre d'une évolution en temps réel par le fait qu'elle n'est pas évolutive, résultat qui aurait pu être obtenu par l'emploi d'un algorithme génétique mais plus complexe à mettre en place.

De plus, comme vu précédemment, l'autonomie des PNJ peut avoir des conséquences sur l'intérêt ludique. Comme l'a montré *Alpha go* de *Deep Mind*, l'intelligence artificielle peut surpasser l'humain sur les tâches pour lesquelles a été entraînée, ce qui peut être frustrant pour le joueur et ce qui entraîne une perte d'intérêt pour le jeu. Le développement de *Pirates : Set Sail And Discover !* a rapidement montré qu'un groupe de PNJ entièrement autonome effectuant les actions pour le joueur peut entraîner un manque d'intérêt de ce dernier pour le jeu, celui-ci se transformant en une simulation.

Enfin, la réponse sur l'émergence de comportements peut être améliorée en ajoutant une dimension évolutive aux agents par le biais d'algorithmes génétiques, ce qui permettrait d'améliorer le réalisme des comportements de groupe et renforcerait l'émergence de comportements inattendus. Nous pouvons nous demander : en utilisant ces deux techniques, l'emploi d'un système multi-agents apprenant en temps réel peut-il augmenter le réalisme des

comportements de groupe ? Et permet-il une émergence de ceux-ci sans pour autant perdre l'intérêt ludique du jeu ? L'intelligence artificielle permettant l'émergence de comportements peut augmenter l'attractivité du jeu, la question reste qu'il faut veiller à ce qu'elle ne prenne pas le dessus et savoir doser sa force au risque de perdre le bénéfice sur l'intérêt du joueur.

Bibliographie

- BENI, Gerardo et WANG, Jing. Swarm intelligence in cellular robotic systems. In : Robots and biological systems: towards a new bionics?. Berlin, Heidelberg : Springer Berlin Heidelberg, 1993. p. 703-712.
- A. M. TURING, I.—COMPUTING MACHINERY AND INTELLIGENCE, *Mind*, Volume LIX, Issue 236, October 1950, Pages 433–460
- Smith, C., Huang, T., McGuire, B., & Yang, G. (2006). *The History of Artificial Intelligence*. Washington: University of Washington.
- GANASCIA, Jean-Gabriel. Intelligence artificielle et épistémologie. Allers-retours indispensables. In : JECIS. 2022.
- SEARLE, John. Chinese room argument. *Scholarpedia*, 2009, vol. 4, no 8, p. 3100.
- ROMERO, Margarida, ALOUI, Hedi, HEISER, Laurent, et al. Un bref parcours sur les ressources, pratiques et acteurs en IA et éducation. 2021. Thèse de doctorat. Université Côte d'Azur.
- XU, Siyuan. History of AI design in video games and its development in RTS games. 2014.
- DOWNEY, Shaun et CHARLES, Darryl. Distribution of artificial intelligence in digital games. *International Journal of Intelligent Information Technologies (IJIIT)*, 2015, vol. 11, no 3, p. 1-14.
- KENNEDY, James. *Swarm intelligence*. Springer US, 2006.
- REYNOLDS, Craig W. Flocks, herds and schools: A distributed behavioral model. In : *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*. 1987. p. 25-34.
- TEODOROVIC, Dusan. Transport modeling by multi-agent systems: a swarm intelligence approach. *Transportation planning and Technology*, 2003, vol. 26, no 4, p. 289-312.
- RECIO, Gustavo, MARTIN, Emilio, ESTÉBANEZ, César, et al. AntBot: Ant colonies for video games. *IEEE Transactions on Computational Intelligence and AI in Games*, 2012, vol. 4, no 4, p. 295-308.
- TVEIT, Amund, REIN, Øyvind, IVERSEN, Jørgen Vinne, et al. Scalable agent-based simulation of players in massively multiplayer online games. In : *Proc. the 8th Scandinavian Conference on Artificial Intelligence (SCAI2003)*, Bergen, Norway. 2003
- LI, Zhengping, SIM, Cheng Hwee, et LOW, Malcolm Yoke Hean. A survey of emergent behavior and its impacts in agent-based systems. In : *2006 4th IEEE international conference on industrial informatics*. IEEE, 2006. p. 1295-1300.

- Dyson & George B. (1997). *Darwin Among the Machines: The Evolution of Global Intelligence*. Perseus Book Group.
- Rollings A. & Adams E. (2003). *Andrew Rollings and Ernest Adams on Game Design*. New Riders Publishing.
- OLSON, Randal S., HINTZE, Arend, DYER, Fred C., et al. Predator confusion is sufficient to evolve swarming behaviour. *Journal of The Royal Society Interface*, 2013, vol. 10, no 85, p. 20130305.
- HONG, Jin-Hyuk et CHO, Sung-Bae. Evolution of emergent behaviors for shooting game characters in robocode. In : *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*. IEEE, 2004. p. 634-638.
- LEE, Jaedong, WON, Jungdam, et LEE, Jehee. Crowd simulation by deep reinforcement learning. In : *Proceedings of the 11th ACM SIGGRAPH Conference on Motion, Interaction and Games*. 2018. p. 1-7.
- ALMEIDA, João E., ROSSETI, Rosaldo JF, et COELHO, António Leça. Crowd simulation modeling applied to emergency and evacuation simulations using multi-agent systems. *arXiv preprint arXiv:1303.4692*, 2013.
- RENDELL, Paul. Turing universality of the game of life. *Collision-based computing*, 2002, p. 513-539.
- SEKHAVAT, Yoonas A. Behavior trees for computer games. *International Journal on Artificial Intelligence Tools*, 2017, vol. 26, no 02, p. 1730001.
- PARK, Joon Sung, O'BRIEN, Joseph C., CAI, Carrie J., et al. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442*, 2023.
- SKINNER, Geoff et WALMSLEY, Toby. Artificial intelligence and deep learning in video games a brief review. In : *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*. IEEE, 2019. p. 404-408.
- PROKOPENKO, Mikhail, HARRÉ, Michael, LIZIER, Joseph, et al. Self-referential basis of undecidable dynamics: From the Liar paradox and the halting problem to the edge of chaos. *Physics of life reviews*, 2019, vol. 31, p. 134-156.
- CARSE, James. *Finite and infinite games*. Simon and Schuster, 2011.

Ludographie

Taito. *Space Invaders*. Taito, 1978.

Allan Alcorn. *Pong*. Atari Inc., 1972

Maxis. *SimCity*. Maxis, 1989

The Creative Assembly. *Alien: Isolation*. Sega, 2014

Blizzard Entertainment. *Starcraft 2*. Blizzard Entertainment, 2010

Monolith Productions. *La Terre du Milieu : L'Ombre du Mordor*. Warner Bros. Interactive Entertainment, 2014

Monolith Productions. *La Terre du Milieu : L'Ombre de la guerre*. Warner Bros. Interactive Entertainment, 2017

General Computer. *Ms. Pac-Man*. Namco, 1981

IBM. *Robocode*. IBM, 2001

Namco. *Pac-Man*. Namco, 1980

Maxis. *Les Sims*. Electronic Arts, 2000

Bugbyte. *Space Haven*. Bugbyte, 2020

Glossaire

PNJ : Personnage non-joueur

Machine Learning : Apprentissage Automatique

IA : Intelligence Artificielle

MMORPG (JDRMM) : Massively Multiplayer Online Role-Playing Game (Jeu De Rôle en ligne Massivement Multijoueur)

Agent BDI : Agent Belief-Desire-Intention (Croyances-Désirs-Intention)

Game Over : Fin de partie, souvent utilisé pour signifier une défaite du joueur

Annexes

Liens vers les documents du projet :

<https://drive.google.com/drive/folders/1c0cbpgGU7xqGOoJGPUi0THwxtTyd7BTI?usp=sharing>