



ALEXANDRE BONNEFOND, OLIVIER SIMONIN, ISABELLE GUÉRIN-LASSOUS

Modèles de Flocking Adaptés aux Environnements avec Obstacles et
Communications Dégradées

Volume 4, n° 2 (2023), p. 123-145.

<https://doi.org/10.5802/roia.59>

© Les auteurs, 2023.



Cet article est diffusé sous la licence
CREATIVE COMMONS ATTRIBUTION 4.0 INTERNATIONAL LICENSE.
<http://creativecommons.org/licenses/by/4.0/>



*La Revue Ouverte d'Intelligence Artificielle est membre du
Centre Mersenne pour l'édition scientifique ouverte*
www.centre-mersenne.org
e-ISSN : 2967-9672

Modèles de Flocking Adaptés aux Environnements avec Obstacles et Communications Dégradées

Alexandre Bonnefond^a, Olivier Simonin^a, Isabelle Guérin-Lassous^b

^a Univ. Lyon, Inria, INSA de Lyon, CITI & LIP Labs, 6 Av. des Arts 69621 Villeurbanne cedex (France)

E-mail : alexandre.bonnefond@inria.fr, olivier.simonin@insa-lyon.fr

^b Univ. Lyon, Université Claude Bernard Lyon 1, ENS de Lyon, CNRS, Inria, LIP, 46 allée d'Italie, Lyon (France)

E-mail : isabelle.guerin-lassous@univ-lyon1.fr.

RÉSUMÉ. — Dans ce papier, nous étudions des modèles de flocking existants et proposons des extensions afin d'améliorer leurs performances dans des environnements ayant des obstacles impactant les communications ainsi que les trajectoires des agents. En effet, les contraintes imposées par les obstacles sont généralement la cause de coupures de communication menant souvent à la séparation de la flotte en plusieurs clusters. Dans ce contexte, nous étendons deux modèles standards afin de renforcer leurs capacités à rester connectés dans des environnements avec différentes distributions d'obstacles. En tenant compte de la propagation radio, nous modélisons comment les obstacles impactent les communications dans un simulateur que nous utilisons notamment pour optimiser les paramètres du flocking. Les résultats des simulations montrent l'efficacité des modèles proposés et la façon dont ils s'adaptent à ces nouvelles contraintes environnementales.

MOTS-CLÉS. — Modèles de flocking, robotique en essaim, simulation des communications.

1. INTRODUCTION

Depuis sa formulation par Craig Reynolds [21] en 1987, le principe du flocking n'a cessé de susciter un intérêt grandissant. Il repose sur trois interactions simples entre agents, la cohésion, la séparation et l'alignement comme représenté par le figure 1.1. Cela signifie que les agents doivent se déplacer autour d'une inter-distance d'équilibre tout en ayant la même direction. Le flocking est un modèle décentralisé qui permet à un essaim de se déplacer dans divers types d'environnements. Si ce modèle a été initialement pensé comme un contrôleur fondé sur des perceptions locales, le développement des drones autonomes et leurs moyens de communication permettent d'adapter le flocking à ce type d'agents. Les drones sont capables aujourd'hui d'échanger à haute fréquence avec leurs voisins proches les informations nécessaires au flocking comme leur vitesse et leur position. Ainsi, la plupart des travaux récents [13, 17, 23, 28] considèrent des flockings exploitant les communications. Cependant, ils ne prennent pas en

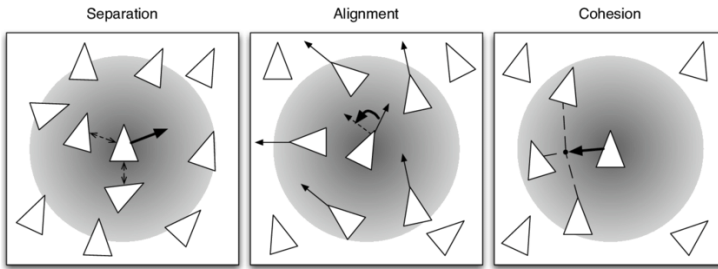


FIGURE 1.1 – Flocking : les 3 interactions définies par Reynolds (illustration de Bouraqadi et Doniec [3])

compte les caractéristiques des communications et comment elles sont impactées par la présence d’obstacles. Des hypothèses trop simples sont un frein pour la définition de modèles exploitables en conditions réelles. Les travaux de [6] donnent une bonne idée des effets de communication réalistes sur un comportement émergent. Cependant, les contrôleurs proposés sont trop simples pour refléter comment les modèles de flocking standards sont affectés.

Le but de cet article est de proposer de nouveaux modèles de flocking par extension de modèles de référence afin de mieux répondre aux différentes contraintes mentionnées plus haut (obstacles et communication). Dans ce cadre, nous proposons d’étendre deux modèles de référence. Le premier est le modèle d’Olfati-Saber [17], connu pour sa stabilité en présence d’obstacles ce qui est une propriété capitale pour le flocking. Néanmoins il fait une hypothèse de connaissance d’un objectif global incompatible avec notre objectif d’un modèle décentralisé. Nous verrons par la suite comment cette hypothèse peut être retirée tout en conservant de bonnes performances. Le deuxième modèle est celui de Vásárhelyi *et al.* [23]. Non seulement est il entièrement décentralisé et agile en environnements contraints mais il fait également la transition vers des expériences réelles en extérieur ce qui représente un atout capital. Dans notre étude nous souhaitons développer un modèle auto-propulsé sans objectif global de manière à avoir une vitesse constante tout en étant entièrement décentralisé. Les modèles sont évalués à l’aide du simulateur de flocking de Viragh *et al.* [25]. Nous étendons ce simulateur pour à la fois disposer de modèles de communication plus réalistes et d’environnements plus variés contenant des obstacles. En combinant cet outil avec un processus d’optimisation des paramètres nous montrons que les modèles de flocking proposés sont plus robustes dans les environnements très contraints.

En section 2, nous présentons les modèles de flocking existants avec un focus sur les deux modèles de référence étendus dans cet article. Ces extensions sont présentées en détail en section 3. La section 4 présente les modèles de communication réalistes introduits dans les simulations, et comment ils permettent de redéfinir la notion de voisinage. Nous comparons alors ces différents modèles en section 5 en considérant des environnements variés en termes d’obstacles. Finalement, nous concluons et discutons de la poursuite de ces travaux.

2. PRÉSENTATION DES MODÈLES DE RÉFÉRENCE DU FLOCKING

Les modèles de flocking sont généralement utilisés pour réaliser le déplacement collectif d'une flotte d'agents ou de robots. Ces modèles reposent sur trois interactions fondamentales introduites par Reynolds [21] :

- **La séparation** d'avec voisins pour éviter les collisions.
- **L'alignement** avec les voisins pour une direction/vitesse proche des voisins.
- **L'attraction** vers les voisins pour maintenir la cohésion.

Le modèle de Reynolds est basé sur une régulation en vitesse par un double intégrateur :

$$\begin{cases} \dot{\mathbf{q}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = \mathbf{u}_i \end{cases}, \quad (2.1)$$

où \mathbf{q}_i représente la position de l'agent i , \mathbf{v}_i correspond à sa vitesse et \mathbf{u}_i son entrée de contrôle :

$$\mathbf{u}_i = \mathbf{f}_i^{att} + \mathbf{f}_i^{sep} + \mathbf{f}_i^{align}, \quad (2.2)$$

avec \mathbf{f}_i^{att} , \mathbf{f}_i^{sep} et \mathbf{f}_i^{align} , les différentes forces liées respectivement à l'attraction, la séparation et l'alignement (les notations en gras représentent des vecteurs).

De plus chaque agent possède un voisinage N_i représentant l'ensemble des agents avec qui il interagit :

$$N_i = \{j \in [1 \dots N] \mid j \neq i, \|\mathbf{q}_j - \mathbf{q}_i\| \leq r\} \quad (2.3)$$

où r est la distance inter-agent maximum en deçà de laquelle deux agents sont considérés voisins et N est le nombre d'agents dans le système.

La plupart des modèles de flocking sont basés sur le modèle de Reynolds. Les principales différences se situent dans la définition des forces de l'équation (2.2). Parmi ces modèles de flocking, on trouve dans la littérature les approches leader-suiveur [7, 28], champs de potentiels [1, 8], les structures virtuelles [14] ou encore les modèles construits par apprentissage par renforcement [15, 26]. Selon l'approche choisie ce ne sont pas les mêmes caractéristiques qui sont mises en avant et le comportement du flocking peut être différent.

Aujourd'hui le flocking est utilisé pour répondre à différentes problématiques et le modèle de Reynolds est intégré dans des contrôleurs multi-agents plus complexes comme celui d'Olfati-Saber [17] (noté O-S plus loin). Dans son travail, en 2006, l'auteur définit un cadre mathématique approprié afin d'assurer la stabilité de son modèle. Cet aspect est primordial dans la théorie du contrôle et garantit un comportement non chaotique ce qui est indispensable pour des applications basées sur l'utilisation de drones par exemple. Olfati-Saber introduit également des fonctionnalités telles que l'évitement d'obstacles ainsi qu'une « rétroaction de navigation » donnant aux agents une direction globale vers laquelle se diriger. Cette dernière caractéristique est capitale pour éviter la fragmentation de la flotte (*i.e.* la formation de multiples clusters). Pour ces raisons, ce modèle a été largement réutilisé dans la littérature [12, 15, 29] avec quelques modifications afin de servir différents objectifs.

La dynamique utilisée par Olfati-Saber est la même que celle de Reynolds (voir équation (2.1)). Cependant, les forces sont définies différemment avec notamment un terme de rétroaction :

$$\mathbf{u}_i = \mathbf{f}_i^g + \mathbf{f}_i^d + \mathbf{f}_i^y, \quad (2.4)$$

où \mathbf{f}_i^g est un terme basé sur un gradient combinant à la fois la séparation et l'attraction, \mathbf{f}_i^d est le terme d'alignement de vitesse qui agit comme une force de frottement visqueux et \mathbf{f}_i^y est la rétroaction de navigation. Ce dernier terme agit comme si il y avait un leader virtuel dont la position et la vitesse seraient connues par tous les agents. Dans ce modèle la distance d'équilibre d_{eq} est gérée par le terme \mathbf{f}_i^g qui s'annule lorsque cette distance est atteinte, il n'y a donc plus d'attraction ou de répulsion. Nous verrons dans la section 3 que nous utilisons le travail d'O-S comme première couche pour construire un nouveau modèle qui répond à nos attentes. La figure 2.1 représente l'évolution des agents utilisant le modèle d'O-S. On se rend compte de la régularité du « maillage » formé par les agents malgré les interactions avec les obstacles. Cette stabilité est typique du modèle d'O-S, ce qui en fait un modèle très utilisé.

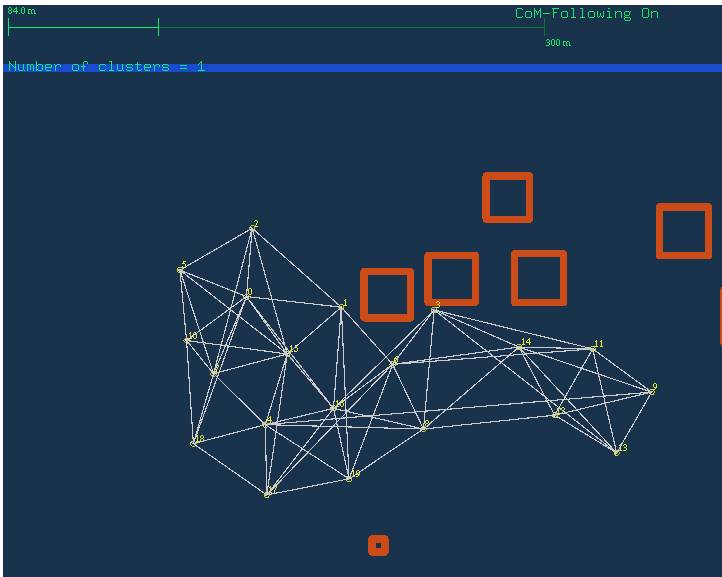


FIGURE 2.1 – Représentation des agents avec les liens de communication effectifs (utilisés pour interagir) utilisant le modèle d’Olfati-Saber (O-S), obstacles en orange

Un autre formalisme majeur du flocking a été introduit par Vicsek [24] en 1995 et considère que les agents sont « auto-propulsés » et évoluent donc à une vitesse constante. Cette caractéristique est importante pour nous car elle fait émerger un comportement dynamique sans risquer d’atteindre des zones d’équilibres où les agents

seraient quasiment immobiles. S'appuyant sur cette idée, Vásárhelyi *et al.* ont développé un framework [23] qui comprend un simulateur incluant les caractéristiques d'un environnement réaliste telles que les délais de communication, l'inertie, la fréquence de fonctionnement des capteurs, les approximations des capteurs embarqués et les bruits ambiants. De plus, l'environnement est composé d'obstacles au sein d'une arène fermée. A la différence du modèle d'O-S ou de Vicsek, le modèle de Vásárhelyi *et al.* s'inscrit dans une volonté d'effectuer des tests réels en extérieur. De ce fait, la formulation du modèle se détache du contexte purement simulateur et vient intégrer des composantes de régulation en vitesse permettant à partir d'une vitesse désirée \mathbf{v}_i^d et de la vitesse actuelle \mathbf{v}_i d'un agent i de définir une accélération \mathbf{a}_i :

$$\mathbf{a}_i = \boldsymbol{\eta}_i + \frac{\mathbf{v}_i^d - \mathbf{v}_i - \mathbf{v}_i^s}{|\mathbf{v}_i^d - \mathbf{v}_i - \mathbf{v}_i^s|} \cdot \min \left\{ \frac{\mathbf{v}_i^d - \mathbf{v}_i - \mathbf{v}_i^s}{\tau_{CTRL}}, a_{\max} \right\}, \quad (2.5)$$

où $\boldsymbol{\eta}_i$ représente le bruit extérieur (tel que l'effet du vent), \mathbf{v}_i^s correspond au bruit interne sur la mesure de vitesse (due aux approximations du GPS), τ_{CTRL} est le temps d'établissement du contrôleur en vitesse (*e.g.* PID) et a_{\max} est l'accélération maximum fixée.

C'est au sein de la vitesse désirée que les auteurs viennent intégrer les différentes composantes propres au flocking :

$$\tilde{\mathbf{v}}_i^d = \frac{\mathbf{v}_i}{|\mathbf{v}_i|} v^{flock} + \mathbf{v}_i^{rep} + \mathbf{v}_i^{frict} + \sum_s \mathbf{v}_{is}^{wall} + \sum_s \mathbf{v}_{is}^{obstacle} \quad (2.6)$$

où $\tilde{\mathbf{v}}_i^d$ est la vitesse désirée avant normalisation, \mathbf{v}_i est la vitesse réelle de l'agent i , v^{flock} est la vitesse souhaitée pour l'auto propulsion, \mathbf{v}_i^{rep} est le terme de répulsion basé sur le modèle du demi ressort, \mathbf{v}_i^{frict} est le terme d'alignement qui synchronise le mouvement au sein de la flotte, \mathbf{v}_{is}^{wall} et $\mathbf{v}_{is}^{obstacle}$ correspondent aux termes utilisés pour l'évitement d'obstacles et de murs de l'arène. Pour ce modèle la distance d'équilibre apparaît uniquement pour le terme \mathbf{v}_i^{rep} qui est positif en dessous de d_{eq} et nul sinon.

Finalement, la vitesse désirée réelle est définie de la manière suivante :

$$\mathbf{v}_i^d = \frac{\tilde{\mathbf{v}}_i^d}{|\tilde{\mathbf{v}}_i^d|} \min\{|\tilde{\mathbf{v}}_i^d|, v^{\max}\}, \quad (2.7)$$

où v^{\max} est la vitesse maximale autorisée pour chaque agent.

L'intégration de cette loi de contrôle au sein d'expériences réelles en extérieur présentées dans l'article font de ce framework une piste très intéressante pour nos travaux et c'est pour cette raison que nous allons nous en inspirer pour proposer une version étendue. La figure 2.2 représente l'évolution des agents avec le modèle de Vásárhelyi *et al.* Même si le maillage créé par les agents semble moins régulier que pour le modèle d'O-S (figure 2.1), il n'en est pas moins robuste comme le montreront les résultats de la section 5.

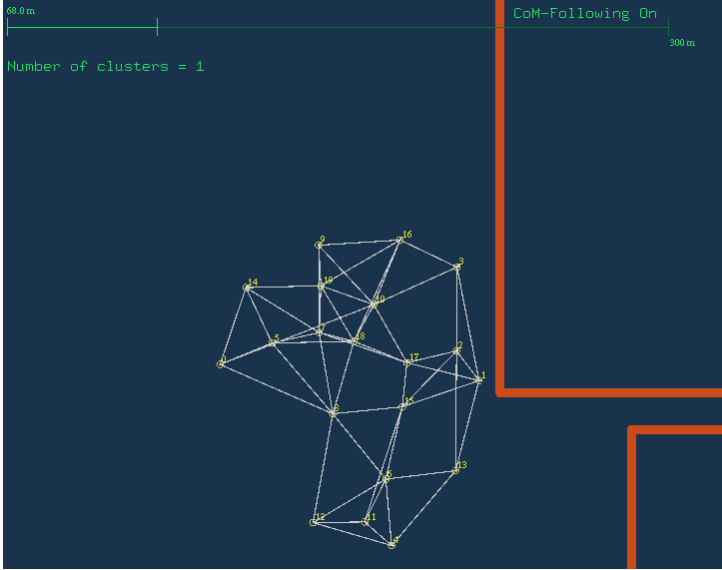


FIGURE 2.2 – Représentation des agents avec les liens de communication effectifs (utilisés pour interagir) utilisant le modèle de Vásárhelyi *et al.*

Bien que ces différents modèles atteignent de bonnes performances en simulation, en termes de stabilité et de robustesse, de nombreux problèmes surgissent quand il s’agit de passer à des expériences à grande échelle (avec des vrais drones en extérieur). En effet, comme expliqué dans l’article [23], en pratique, de nombreuses pertes de communication surviennent et c’est un inconvénient majeur lorsque l’on travaille avec des drones.

3. EXTENSION DES MODÈLES DE FLOCKING

Afin d’éviter les problèmes rencontrés par les modèles présentés dans la section 2 quand ils sont mis dans un environnement plus réaliste, nous proposons de nouveaux modèles de flocking. Ces derniers exploitent les interactions fondamentales des modèles précédents [23] et [17] avec de nouvelles fonctionnalités visant à plus de robustesse en présence d’obstacles.

3.1. MODÈLE VAT : VÁSÁRHELYI + ATTRACTION

De par sa stabilité, le contrôleur en vitesse de Vásárhelyi *et al.* défini en (2.6) ne contient pas de terme d’attraction dans sa forme initiale. Cependant, une fois mis dans un environnement fortement contraint (incluant des délais et des erreurs dans les communications, approximation de position et présence de multiples obstacles) nous nous sommes rendus compte que la fréquence des coupures de communication augmentait rapidement menant à un accroissement du nombre de fragmentations de la

flotte. Nous avons donc décidé d'introduire un terme d'attraction basé sur le modèle du demi-ressort afin d'améliorer la cohésion de la flotte dès que possible :

$$v_{ij}^{att} = \begin{cases} p^{att}(r_0^{att} - q_{ij}) \frac{\mathbf{q}_i - \mathbf{q}_j}{q_{ij}}, & \text{si } q_{ij} > r_0^{att} \\ 0, & \text{sinon} \end{cases} \quad (3.1)$$

où p^{att} est un gain linéaire, $q_{ij} = |\mathbf{q}_i - \mathbf{q}_j|$ est la distance entre les agents i et j et r_0^{att} est la portée d'attraction en-dessous de laquelle les agents arrêtent de s'attirer pour éviter les collisions. Ainsi l'attraction globale d'un agent est $v_i^{att} = \sum_j v_{ij}^{att}$. Ajouter le terme v_i^{att} à la loi de contrôle doit être fait en accord avec le terme de répulsion afin d'éviter de créer un phénomène oscillatoire instable. Par conséquent, r_0^{att} doit être légèrement supérieure à la distance d'équilibre pour que les agents aient une marge de déplacement entre l'attraction et la répulsion. Notre première proposition consiste donc en un contrôleur basé vitesse dénoté VAT (ajout de \mathbf{v}_i^{att} à l'éq. (2.6)) :

$$\tilde{\mathbf{v}}_i^d = \frac{\mathbf{v}_i}{|\mathbf{v}_i|} v^{flock} + \mathbf{v}_i^{rep} + \mathbf{v}_i^{att} + \mathbf{v}_i^{frict} + \sum_s \mathbf{v}_{is}^{wall,obst} \quad (3.2)$$

Les interactions avec les murs et les obstacles sont concaténées dans le même terme $\sum_s \mathbf{v}_{is}^{wall,obst}$ et l'équation (2.7) reste valide. Cette vitesse désirée englobe 12 paramètres agissant sur chaque terme de cette loi et produisant des comportements différents. Nous verrons dans la section 5.2 comment ces paramètres sont optimisés afin d'atteindre un comportement performant.

3.2. MODÈLE VOS : VÁSÁRHELYI + OLFATI-SABER

Le second modèle que nous proposons, dénoté VOS, est basé sur une combinaison du modèle d'Olfati-Saber [17] pour sa stabilité ainsi que certaines parties du modèle de Vásárhelyi *et al.* [23] afin d'avoir l'auto propulsion et l'évitement d'obstacles. Pour cela, nous avons développé une structure de contrôle imbriquée alliant les deux contrôleurs :

$$\begin{cases} \mathbf{u}_i = \alpha \mathbf{f}_i^g + \beta \mathbf{f}_i^d \\ \tilde{\mathbf{v}}_i^d = \frac{\mathbf{v}_i}{|\mathbf{v}_i|} v^{flock} + \sum_s \mathbf{v}_{is}^{wall,obst} + \int_{T_s} \mathbf{u}_i dt \end{cases} \quad (3.3)$$

\mathbf{u}_i correspond à l'équation (2.4) d'O-S sans la rétroaction de navigation \mathbf{f}_i^γ . En effet son calcul centralisé était introduit pour éviter la fragmentation. Or nous voulons développer un système de flocking totalement décentralisé. Dans 3.3, α et β sont des coefficients multiplicateurs qui viennent pondérer les termes d'attraction/répulsion et d'alignement. $T_s = 1/f_s$ avec f_s la fréquence nominale du contrôleur. La deuxième équation du système est semblable à l'équation (2.6) dans laquelle la séparation, l'attraction et l'alignement sont gérés par le modèle de O-S. L'idée est ici de profiter de la stabilité du modèle d'O-S tout en paliant au problème de fragmentation avec l'auto propulsion. Les équations qui en découlent englobent 10 paramètres qui nécessiteront une optimisation (voir la section 5.2).

4. COMMUNICATIONS ET VOISINAGE EN PRÉSENCE D'OBSTACLES

Comme mentionné dans l'introduction, la plupart des simulations de flocking considèrent des communications idéales et ne prennent pas en compte l'impact de l'environnement sur la qualité des communications, provoquant un écart conséquent avec les conditions d'expérimentation réelles. nous proposons un modèle de communication fondé sur l'estimation de la propagation radio, intégrant la traversée des obstacles. Cela nous permet de raffiner la définition du voisinage et de mieux estimer avec qui chaque agent interagit.

Généralement, les simulations de flocking utilisent un seuil de distance en-dessous duquel la communication est possible (paramètre r de l'équation (2.3)). Avec ce type d'approche, les communications sont parfaites [12, 17], ou plus réalistes avec la prise en compte de délais et pertes de paquets aléatoires comme dans [23, 25]. Dans [13], les auteurs utilisent le BER (Bit Error Rate) combiné au SNR (Signal to Noise Ratio) afin de caractériser la qualité de communication et intègrent cela dans un index de performance exprimant une balance entre la distance inter-agent et la qualité de communication. Cependant, il n'y a aucun obstacle dans les environnements simulés par les auteurs. Dans [27], les auteurs développent un système hybride de propagation combinant une étape de prédiction et une étape d'apprentissage de l'affaiblissement de propagation. Ils utilisent l'outil SPLAT!⁽¹⁾ afin de prédire les fluctuations du champ de propagation en fonction de la topologie d'une zone. Bien que cette méthode donne une bonne estimation des mesures radio, elle ne se soucie pas de l'adaptation du système multi-agent face à ces variations. Sur le même sujet, dans [6], l'auteur explique comment un comportement émergent, tel que le flocking, est impacté lorsque l'on utilise un modèle de propagation relativement réaliste et que l'on modifie les pertes dues aux effets tels que le shadowing, le SNR, le jamming et autres spécificités de la couche physique de la technologie de communication radio utilisée. Cependant l'étude sur la dynamique du flocking mis en place est peu approfondie dans cet article.

Dans la suite, nous introduisons une étape d'estimation de la propagation radio en utilisant le modèle *log distance path loss* (noté LDPL) à la fois dans le vide mais aussi au sein des obstacles et nous regardons comment des environnements hétérogènes affectent la qualité de communication et donc la stratégie de flocking.

4.1. MODÈLE DE PERTE *LOG DISTANCE* EN ENVIRONNEMENTS AVEC OBSTACLES

Au lieu d'utiliser un modèle de communication basé uniquement sur la distance inter-agent, nous souhaitons modéliser de manière plus précise la manière dont évoluent les ondes radio avec l'environnement et par conséquent savoir comment sont affectées les interactions pour un voisinage donné. Pour ce faire, nous considérons le modèle LDPL qui prédit la perte de puissance qu'un signal subit en fonction de

⁽¹⁾Informations pour SPLAT! : <https://www.qsl.net/kd2bd/splat.html>

l'environnement [18]. Il se formule de la façon suivante :

$$PL = P_{T_{x_{dBm}}} - P_{R_{x_{dBm}}}, \quad (4.1)$$

$$PL = \begin{cases} PL_0 + 10\gamma \log_{10} \frac{d}{d_0} + X_g, & \text{si } d \geq d_0 \\ PL_0, & \text{sinon} \end{cases} \quad (4.2)$$

où PL est l'affaiblissement du signal entre l'émetteur et le récepteur considérés et séparés d'une distance d mesurée en décibels (dB), $P_{T_{x_{dBm}}}$ est la puissance du signal émis par l'émetteur en dBm, $P_{R_{x_{dBm}}}$ est la puissance du signal reçu par le récepteur en dBm, PL_0 est l'affaiblissement du signal sur une distance de référence d_0 calculé en utilisant le modèle de Friis d'affaiblissement en espace libre sans obstacle (une valeur de 6 m a été choisie pour d_0), γ est le coefficient de l'affaiblissement qui dépend de l'environnement et X_g est un bruit qui représente les variations de l'affaiblissement causées par les obstacles et les chemins multiples. Cette variable suit une distribution normale centrée en zero et un écart type σ dB. Ce modèle sera appelé **regular** par la suite. La courbe verte de la figure 4.1 représente l'évolution de $P_{R_{x_{dBm}}}$ en fonction de la distance entre le récepteur et l'émetteur, dans un environnement où il n'y a pas d'obstacles et lorsque $P_{T_{x_{dBm}}} = 20$ dBm. Nous pouvons voir qu'il faut environ 300 mètres pour atteindre le seuil de puissance $P_{\min} = -70$ dBm correspondant au niveau minimum de puissance requis pour établir une communication entre les agents.

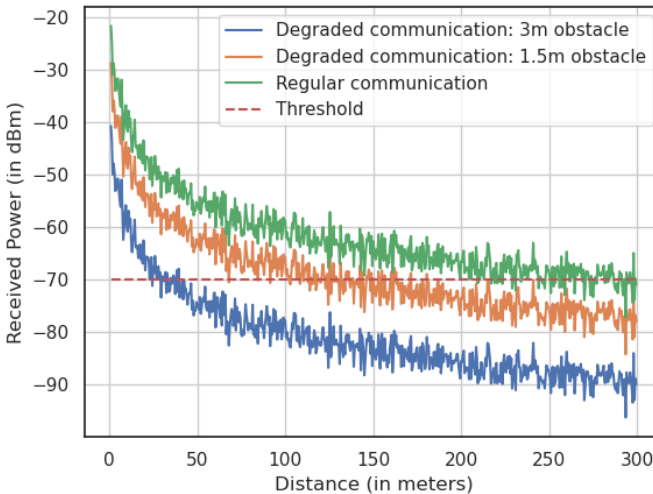


FIGURE 4.1 – Évolution de la puissance reçue en fonction de la distance inter-agent. La courbe verte représente un chemin sans obstacle, la courbe orange représente un chemin qui intègre un obstacle de 1,5 mètres et la courbe bleue intègre un obstacle de 3 mètres de long.

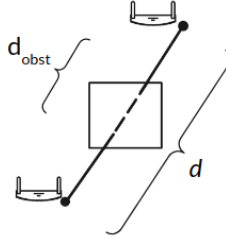


FIGURE 4.2 – Représentation du chemin parcouru par un signal entre deux nœuds et traversant un obstacle.

Dans la formule précédente, l'environnement est modélisé par, entre autres, les paramètres γ et X_g . Afin de prendre en compte l'impact des différents obstacles sur la qualité de communication, nous ajoutons un modèle de dissipation au sein des obstacles créant des pertes majeures en termes de puissance reçue. L'affaiblissement du signal résultant s'exprime de la façon suivante :

$$PL = \begin{cases} PL_0 + 10\gamma \log_{10} \left(\frac{d-d_{obst}}{d_0} \right) + X_g + PL_{obst}(d_{obst}), & \text{si } d - d_{obst} \geq d_0 \\ PL_0 + PL_{obst}(d_{obst}), & \text{sinon} \end{cases} \quad (4.3)$$

où, d_{obst} est la distance parcourue par le signal au sein d'un obstacle (voir figure 4.2) et PL_{obst} est l'affaiblissement du signal en dB dû à l'obstacle. PL_{obst} est défini de la manière suivante :

$$PL_{obst}(d_{obst}) = 10\gamma_{obst} \log_{10}(d_{obst}) + K \quad (4.4)$$

où γ_{obst} est le coefficient d'affaiblissement pour un obstacle et K est une constante. Nous supposons que $d_{obst} \geq 1$ m. Nous appellerons ce modèle **degraded**. Les courbes orange et bleu dans la figure 4.1 représentent l'évolution de $P_{R_{x\text{dBm}}}$ en fonction de la distance entre le récepteur et l'émetteur sur un chemin comprenant un obstacle tel que $d_{obst} = 1,5$ m et $d_{obst} = 3$ m. Dans cette configuration, nous avons $\gamma_{obst} = 4$, $P_{T_{x\text{dBm}}} = 20$ dBm et $K = 0$. Nous pouvons voir que ce modèle de communication a un impact très négatif sur la qualité de communication puisqu'il ne faut plus que 30 mètres pour perdre la communication entre deux agents qui ont un obstacle de 3 mètres entre eux.

Ce modèle de communication permet donc une approche plus réaliste dans le sens où les propriétés de l'environnement impactent directement la diffusion de l'information au sein de la flotte. De plus il permet de modéliser différentes technologies (e.g. WiFi, Bluetooth, UWB, etc) en modifiant les paramètres comme P_{\min} ou encore PL_0 qui dépendent de la fréquence d'émission et de la directivité des antennes.

4.2. REDÉFINITION DU VOISINAGE

En utilisant le nouveau modèle de propagation proposé à la section précédente, nous pouvons redéfinir le voisinage d'interaction N_i pour chaque agent i :

$$N_i = \{j \in \llbracket 1, \dots, N \rrbracket; j \neq i : PL_{ij} \leq P_{threshold}\} \quad (4.5)$$

où PL_{ij} est l'affaiblissement du signal (équation (4.3)) mesuré entre j (émetteur) et i (récepteur) et $P_{threshold}$ ($\neq P_{min}$) est le seuil de puissance en-dessous duquel la communication n'est plus possible. Aussi, nous introduisons un autre seuil N_{max} , ajustable, qui agit sur la cardinalité de N_i :

$$N_i = \begin{cases} N_i, & \text{if } |N_i| \leq N_{max} \\ N_i^{filtered}, & \text{sinon} \end{cases} \quad (4.6)$$

où $N_i^{filtered}$ correspond à un sous-ensemble de N_i où les agents correspondant à la puissance reçue la plus faible sont retirés jusqu'à ce que $|N_i| = N_{max}$. Ce seuil est très important car il permet de limiter le voisinage et donc le coût de calcul au niveau de chaque agent, tout en conservant les voisins les mieux connectés.

Avec cette nouvelle définition du voisinage, basée sur un modèle de communication plus réaliste, nous pouvons étudier l'évolution d'une flotte d'agents dans des environnements fortement contraints par de nombreux obstacles.

5. COMPARAISON DES MODÈLES

5.1. OUTIL DE SIMULATION

Le simulateur que nous utilisons est une extension de l'outil⁽²⁾ développé par Viragh *et al.* [25] illustré par la figure 5.1. C'est un simulateur multi-agent qui introduit des variables aléatoires pour modéliser les différents bruits ajoutés sur des variables telles que la position GPS ou encore les communications. Cela rajoute une complexité tant pour la résolution des équations différentielles stochastiques (EDS), en utilisant la méthode d'*Euler-Marayuma* [25], mais aussi pour l'optimisation des paramètres (voir section 5.2). Nous avons décidé d'utiliser ce simulateur plutôt que les simulateurs plus courants tels que ARGOS ou GAZEBO pour deux raisons principales. La première vient du fait que le modèle initial de Vásárhelyi *et al.* est déjà implémenté et nous offre donc une comparaison fidèle avec nos modèles. La deuxième est due au fait que nous utilisons ce simulateur principalement sur des serveurs de calculs pour réaliser l'optimisation des paramètres de flocking et donc sans visualisation. Ce simulateur est adapté à ce type d'optimisation, la visualisation n'étant utilisée que pour observer un comportement pour un jeu de paramètres donné.

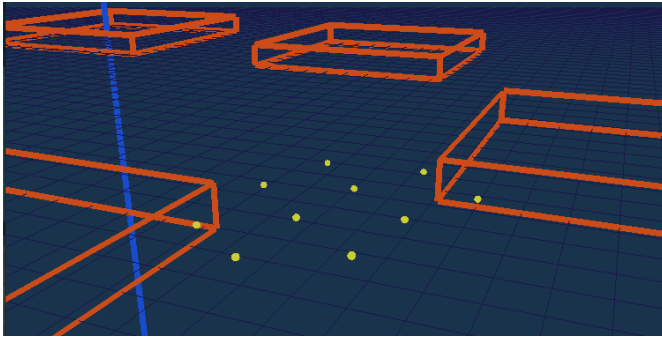
Les hyper-paramètres utilisés pour une expérience tels que sa durée T ou encore le pas de calcul de la méthode d'Euler sont réglables. La distribution des obstacles est quant à elle définie par plusieurs paramètres tels que sa densité D qui se mesure de la

⁽²⁾Accès au projet initial ici : <https://github.com/csviragh/robotsim>

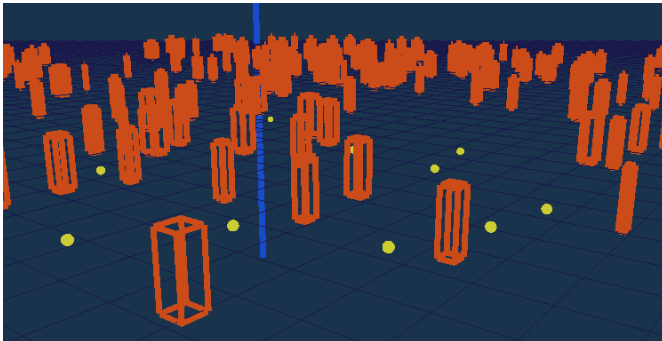
manière suivante :

$$D = 100 \times \frac{\sum_s A_s}{A_{arena}}, \quad (5.1)$$

où A_s est la surface de l'obstacle s et A_{arena} est la surface de l'arène. La figure 5.1(b) montre une vue 3D de l'environnement appelé « Forest » pour sa quantité d'obstacles.



(a) Vue 3D de l'environnement « City »



(b) Vue 3D de l'environnement « Forest »

FIGURE 5.1 – Illustration des environnements tests avec le simulateur [25]

De nombreuses fonctionnalités ont été ajoutées à ce simulateur afin de répondre à nos besoins comme la détection des clusters, en appliquant l'algorithme *Depth First Search* (DFS) sur le graphe des agents connectés, mais aussi la possibilité de changer à la volée le modèle de flocking ou encore le mode de communication (regular/degraded).

5.2. OPTIMISATION DES PARAMÈTRES

Comme nous l'avons mentionné dans la section 3, les modèles que nous étudions ont de nombreux paramètres qui doivent être optimisés afin d'obtenir le meilleur

comportement possible. En s'inspirant du travail de [23], nous avons défini plusieurs métriques permettant d'évaluer les performances de nos modèles de flocking durant une simulation de durée T :

- **Velocity** :

$$\Psi_{vel} = \frac{1}{T} \frac{1}{N} \int_0^T \sum_{i=1}^N |\mathbf{v}_i(t)| dt, \quad (5.2)$$

qui caractérise la vitesse moyenne de la flotte.

- **Collision** (entre agents) :

$$\Psi_{col} = \frac{1}{T} \frac{1}{N(N-1)} \int_0^T \sum_{i=1}^N \sum_{j \neq i} \Theta(q_{ij}(t) - r^{coll}) dt, \quad (5.3)$$

mesure le ratio de collisions, avec Θ la fonction de Heaviside et r^{coll} la distance minimale autorisée entre deux agents.

- **Connected** :

$$\Psi_{clust} = \frac{1}{T} \int_0^T |\overline{O}(t)| dt, \quad (5.4)$$

$$O(t) = \{i \in [1 \dots N] : |N_i(t)| = 0\}, \quad (5.5)$$

$\overline{O}(t)$ représente l'ensemble des agents connectés au temps t , c'est-à-dire. le complémentaire des agents isolés $O(t)$.

- **Wall** (collision) :

$$\Psi_{wall} = \frac{\int_0^T \sum_{i=1}^N \Theta(\tilde{q}_{is}(t)) \tilde{q}_{is}(t) dt}{\int_0^T \sum_{i=1}^N \Theta(\tilde{q}_{is}(t)) dt}, \quad (5.6)$$

où \tilde{q}_{is} prend des valeurs positives en-dehors de l'arène et à l'intérieur d'un obstacle, et inversement. Ψ_{wall} caractérise la distance moyenne passée à l'intérieur des obstacles.

- **Cluster Correlation** :

$$\Psi_{corr} = \frac{1}{T} \frac{1}{N} \int_0^T \sum_{i=1}^N \frac{1}{|N_i| - 1} \sum_{j \in N_i} \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{|\mathbf{v}_i| |\mathbf{v}_j|} dt, \quad (5.7)$$

qui donne une mesure de l'alignement moyen par cluster.

- **Cluster Received Power** :

$$\Psi_{pow} = \frac{1}{T} \frac{1}{N} \int_0^T \sum_{i=1}^N \frac{1}{|N_i| - 1} \sum_{j \in N_i} P_{Rx_{ij}}(t) dt, \quad (5.8)$$

qui mesure la puissance reçue moyenne par cluster.

Ces métriques permettent de définir les conditions nécessaires pour obtenir un comportement optimal : $\Psi_{vel} \rightarrow v^{flock}$, $\Psi_{col} \rightarrow 0$, $\Psi_{clust} \rightarrow 1$, $\Psi_{wall} \rightarrow 0$, $\Psi_{corr} \rightarrow 1$, $\Psi_{pow} \geq P_{min}$. Les résultats obtenus sont passés dans des fonctions non linéaires pour normaliser les valeurs entre 0 et 1 (la fonction sigmoid par exemple), 0 étant

le pire cas et 1 le meilleur, et ce afin de définir un problème d'optimisation multi-objectif.

Dans ce papier nous utilisons un algorithme génétique multi-objectif appelé *NSGA-III* [2, 4]. Ce dernier n'est cependant pas adapté aux problèmes stochastiques mais en évaluant la même solution plusieurs fois et en utilisant la variance résultante comme contrainte de l'optimisation nous avons pu pallier ce problème. Cela mène à des solutions plus robustes et par conséquent moins impactées par le caractère aléatoire d'une expérience.

La stratégie est ici d'optimiser les modèles VAT et VOS sur un environnement « générique », *i.e.* avec une densité moyenne d'obstacles et avec un modèle de communication dégradé. Le nombre d'agents utilisé lors de l'optimisation et les simulations, dont les résultats sont donnés en section 5.3, est toujours le même, $N = 10$. Nous utilisons ensuite ces versions optimisées avec différentes distributions d'obstacles afin de comparer leurs performances. L'environnement générique a $N_{obst} = 90$ obstacles uniformément distribués dans une arène carrée de côté 1 250 m. Chaque obstacle est représenté par un carré dont le coté L suit une distribution normale (Gaussienne) $L \sim \mathcal{N}(\mu, \sigma)$, centrée sur $\mu = 24$ mètres et un écart type $\sigma = 12$ mètres. La densité de cet environnement est $D = 3,3$ %.

Le processus d'optimisation est présenté en figure 5.2 où l'on peut observer l'évolution de la moyenne des métriques sur le front de Pareto à chaque génération pour les deux modèles (VOS en bleu et VAT en rouge). L'échelle de couleurs donne une information sur l'avancement de l'optimisation basée sur le nombre maximum de générations. Bien que VOS semble être bien moins performant que VAT, il existe une raison à cela. C'est notamment parce que son front de Pareto contient plus de solutions avec des variances plus grandes. La table 5.1 regroupe les valeurs moyennes des métriques après 100 évaluations des solutions optimisées dans l'environnement *generic* avec une communication *degraded*. On se rend compte ici que les deux modèles ont des performances relativement similaires sauf pour la métrique *velocity* qui est plus basse pour le modèle VOS. Ces solutions et leurs performances vont être comparées en détail dans la prochaine sous-section.

	Vel	Col	Clust	Wall	Corr	Pow
VAT	0,999	0,99	0,94	1	0,86	0,202
VOS	0,612	0,88	0,92	1	0,784	0,26

TABLE 5.1 – Métriques de VAT et VOS en env. GENERIC

	Vel	Col	Clust	Wall	Corr	Pow
VA	0,938	0,945	1	0,997	0,916	N/A

TABLE 5.2 – Performances de VA dans [23]

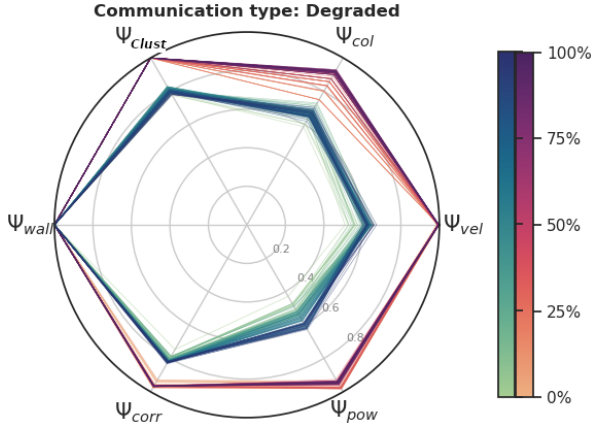


FIGURE 5.2 – Évolution des métriques durant l’optimisation en utilisant une communication dégradée dans l’environnement generic. Les courbes bleues et rouges correspondent respectivement aux modèles VOS et VAT.

5.3. RÉSULTATS ET ANALYSE

Notre but ici est de comparer les modèles VAT (section 3.1) et VOS (section 3.2) dans des environnements différents et avec des stratégies de communication différentes (regular et degraded). La comparaison est basée sur deux environnements différents (voir figure 5.1) :

- (1) **Forest** : $N_{obst} = 500$; $L \sim \mathcal{N}(4, 2)$; $D = 0,7 \%$.
- (2) **City** : $N_{obst} = 15$; $L \sim \mathcal{N}(200, 20)$; $D = 36,5 \%$.

Nous allons également faire une comparaison avec le modèle de Vásárhelyi *et al.*, noté VA, que nous simulerons en utilisant les paramètres optimaux fournis dans [23]. Ce modèle repose sur les équations (2.5), (2.7) et (2.6) mais est désormais soumis à la nouvelle définition du voisinage basée sur la puissance reçue. Les tables 5.3 et 5.4 regroupent les résultats de 100 évaluations stochastiques des modèles optimaux. Les valeurs moyennes des métriques normalisées entre 0 et 1 (1 étant le meilleur cas, 0 le pire) ainsi que leurs variances sont affichées. La dernière ligne correspond à l’inter-distance moyenne entre agents (en mètre). Afin d’être cohérent dans nos résultats, nous utilisons les mêmes hyper-paramètres à savoir, le temps d’expérience $T = 600$ s, la précision de la méthode d’Euler $\Delta t = 0,01$ s, le délai introduit dans les communications $t_d = 0,2$ s (représentant un délai d’accès au médium radio, le délai de transmission d’un paquet et le délai de propagation) et la fréquence de communication synchronisée avec la fréquence du GPS $f_{GPS} = 5$ Hz. Une présentation de ces modèles est proposée dans la vidéo au lien en pied de page ⁽³⁾.

⁽³⁾vidéo : https://team.inria.fr/chroma/files/2022/10/JFSMA_2021.avi

La première observation importante ici est que le modèle VA a de moins bonnes performances que dans [23] (voir table 5.2). Cela peut s'expliquer par le fait que l'auteur se base sur de petites arènes dans ses essais, menant indirectement à un effet de cohésion (d'où l'absence de force d'attraction). Afin d'éviter ce comportement avantageux, nous utilisons une arène de 1 250 m de long. Par ailleurs, la distribution des obstacles n'est pas explicitement mentionnée dans [23] bien que celle-ci ait un fort impact sur les valeurs des métriques. Aussi, le fait de considérer notre nouveau modèle de propagation ajoute une nouvelle contrainte qui n'était pas considérée dans l'article initial. Les inconvénients de ne pas ajouter un terme d'attraction sont quant à eux encore plus visibles dans l'environnement *forest* où la métrique *cluster* est relativement basse, ce qui signifie que le phénomène de fragmentation a eu lieu à maintes reprises (Clust = 0,637 dans table 5.3). Il est également intéressant de remarquer que le fait d'ajouter un terme d'attraction ne crée pas de collisions supplémentaires bien que cela soit contre-intuitif. Ceci s'explique par le fait que le terme d'attraction est actif uniquement au-dessus d'une certaine valeur de distance inter-agent et avec une inertie résultante bornée pour les agents, ce qui prévient les risques d'oscillations et donc de collisions. Le fait que le modèle VAT soit aussi conçu et optimisé dans un contexte de communication **dégradé**, et donc bénéficie du *neighbour filtering*, participe à expliquer les écarts entre VAT et VA.

	Regular			Degraded		
	VA	VAT	VOS	VA	VAT	VOS
Vel	0,975	0,946	0,032	0,975	0,955	0,055
σ	0,010	0,03	0,052	0,008	0,023	0,063
Col	0,301	1	0,793	0,286	0,961	0,799
σ	0,429	0,099	0,382	0,429	0,172	0,380
Clust	0,637	0,98	0,923	0,508	0,938	0,868
σ	0,148	0,085	0,13	0,118	0,121	0,163
Wall	1	1	1	1	1	1
σ	0	0	0	0	0	0
Corr	0,133	0,712	0,54	0,128	0,698	0,531
σ	0,068	0,106	0,12	0,06	0,12	0,112
Pow	0,033	0,53	0,517	0	0,045	0,031
σ	0,049	0,11	0,074	0	0,079	0,066
Dist	132,3	37,9	49,9	132,5	42,11	54,37

TABLE 5.3 – FOREST

5.3.1. Maintien de la connectivité

L'un de nos objectifs dans cet article est d'implémenter un modèle capable d'éviter les fragmentations à répétition. Nous pouvons voir dans les deux tables que cet objectif est globalement atteint. En effet, pour VAT et VOS, la moyenne de la métrique *cluster*

est haute, $0,868 \leq \text{Clust} \leq 0,985$. Cela démontre que même si quelques fragmentations se sont produites, les clusters résultants ont réussi à réparer les connexions détruites afin de former un seul et unique groupe. Malgré nos efforts, des fragmentations partielles restent possibles, notamment dans l'environnement *forest* où les nombreux obstacles créent de fréquentes interruptions de communication et donc d'interaction entre agents (voir vidéo).

5.3.2. Puissance reçue et collisions

La distance inter-agent est une composante résultante de toutes les interactions et est principalement modifiée lors de l'optimisation de la métrique *received power* ainsi que *collision*. Les agents doivent évoluer dans un intervalle de distance avec les autres qui est pré-défini : pas trop près afin d'éviter les collisions sans pour autant aller trop loin et perdre la connexion avec le groupe. Il est important de comprendre que dans ces tables, une valeur faible de *received power* ne veut pas dire que la connexion est perdue mais qu'elle est faible et donc sujette à des pertes de paquets et autres erreurs. Par exemple dans la table 5.4, nous avons $\text{Pow} = 0,573$ pour le modèle VOS avec une communication *degraded* mais cela est équivalent à une valeur de $P_R = -59,4$ dBm, ce qui signifie qu'en moyenne, la communication était possible. En comparaison, dans la table 5.3, pour le modèle VAT avec une communication *degraded*, $\text{Pow} = 0,045 \simeq -71,7$ dBm (-70 dBm étant la valeur seuil en deçà de laquelle la communication n'est plus possible). Cela confirme le fait que, même si la communication est en moyenne impossible, dès qu'ils le peuvent, les agents arrivent à garder une formation de flotte (la métrique de *cluster* étant haute).

	Regular			Degraded		
	VA	VAT	VOS	VA	VAT	VOS
Vel	0,999	1	0,484	0,999	1	0,501
σ	1,1e-5	0	0,095	4,8e-5	0	0,101
Col	0,71	0,999	0,918	0,654	1	0,86
σ	0,442	0,09	0,264	0,47	0	0,33
Clust	0,671	0,985	0,939	0,54	0,915	0,925
σ	0,18	0,065	0,142	0,197	0,17	0,165
Wall	1	1	1	1	1	1
σ	0	0	0	0	0	0
Corr	0,605	0,912	0,886	0,73	0,945	0,894
σ	0,165	0,067	0,082	0,112	0,04	0,073
Pow	0,512	0,797	0,684	0,335	0,777	0,573
σ	0,205	0,092	0,073	0,286	0,16	0,204
Dist	82,19	29,4	44,3	86,76	33,8	45,55

TABLE 5.4 – CITY

5.3.3. *Overfitting de l'environnement*

Comme mentionné plus haut, les modèles que nous simulons ne sont pas optimisés sur les environnements que nous utilisons pour la comparaison. Par conséquent, il y a un risque d'overfitting sur une distribution d'obstacles donnée. Nous pouvons observer cela pour le modèle VOS dans la table 5.3 où la moyenne de la métrique *velocity* est basse, $0,032 \leq \text{Vel} \leq 0,055$, ce qui équivaut à environ 60 % de la vitesse désirée v_{flock} . Bien que dans la table 5.1, $\text{Vel} = 0,612$, il y a approximativement 5 fois moins d'obstacles dans cet environnement *generic* où l'optimisation est effectuée que dans l'environnement *forest*. Dans *forest*, il y a énormément d'obstacles proches les uns des autres entraînant un « effet de cage » et empêchant la flotte de voler à la vitesse désirée. Cela crée aussi de fortes oscillations $0,531 \leq \text{Corr} \leq 0,54$ et quelques collisions $0,793 \leq \text{Col} \leq 0,799$ (voir vidéo). Le modèle VOS n'est par conséquent pas robuste à la variation des distributions d'obstacles.

5.3.4. *Communications Regular/Degraded*

Les effets résultant de la communication *degraded* peuvent s'observer dans les deux tables. De plus, la dernière ligne qui concerne la distance inter-agent montre que ce type de communication va provoquer l'augmentation de la distance moyenne entre les agents, ce qui cause la plupart du temps de la fragmentation. Cependant, le modèle VAT semble être très résilient et robuste aux pertes introduites par le mode *degraded*, ce qui en fait un modèle particulièrement performant.

6. OPTIMISATION SOUS CONTRAINTE D'UNE DISTANCE D'ÉQUILIBRE SOUHAITÉE

6.1. CONTEXTE

Depuis le début de l'article, nous considérons la distance de vol entre drones, ou **distance d'équilibre**, comme un paramètre du flocking optimisable. En effet, durant le processus d'optimisation, différentes valeurs de distance d'équilibre peuvent être générées de manière à respecter au mieux les objectifs fixés (vitesse, corrélation, collisions, ...). La figure 6.1 illustre le résultat d'un processus de test du modèle VAT optimisé où la distance d'équilibre observée est de l'ordre de 12 m, alors que la consigne initiale était de 20 m.

Cela s'explique par la superposition des forces d'attractions générant ainsi un décalage par rapport à la position d'équilibre théorique. Afin de pallier ce comportement problématique, le processus d'optimisation va la plupart du temps générer une distance d'équilibre souhaitée relativement grande afin de prévenir les risques de collisions entre les agents. Cela peut être observé sur la dernière ligne des tableaux de la section 5.3.

En revanche, selon les besoins applicatifs, l'utilisateur est susceptible de vouloir fixer une distance d'équilibre en fonction de l'environnement dans lequel il souhaite faire évoluer les drones. Ceci nous a motivé à étudier comment le respect de la distance d'équilibre peut être un critère objectif pour l'optimisation des modèles.

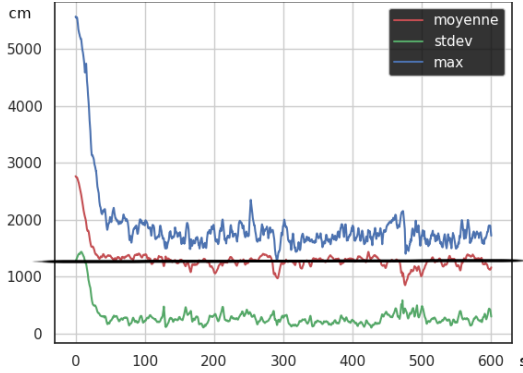


FIGURE 6.1 – Graphique représentant l’inter distance mesurée entre un agent et son plus proche voisin, avec un modèle optimisé (valeur moyenne sur la durée de l’expérience représentée par la ligne noire à 1 200 cm).

6.2. INTRODUCTION D’UNE NOUVELLE MÉTRIQUE

Dans cette section, nous optimisons notre modèle le plus performant, à savoir VAT, avec une nouvelle fonction objectif basée sur la distance d’équilibre. Cette distance correspond à l’écart entre un agent et son plus proche voisin. Nous avons par ailleurs retiré la puissance reçue en tant que critère objectif étant donnée la forte corrélation entre les deux variables (tout du moins dans les simulations). Cependant, celle-ci est tout de même prise en compte dans le processus en tant que contrainte, c’est-à-dire qu’une solution ne sera pas retenue si elle ne dépasse pas un certain niveau de qualité de communication (cf. section 5.3.2).

La métrique que nous utilisons pour optimiser la distance est définie de la manière suivante :

$$\Psi_{dist} = \frac{1}{T} \frac{1}{N} \int_0^T \sum_{i=1}^N |d_{ij}^{\min}(t)| dt, \forall j \neq i \quad (6.1)$$

où $d_{ij}^{\min}(t) = \min(d_{ij}), \forall j \neq i$, d_{ij} correspond à la distance entre un agent i et un agent j , T représente le temps de l’expérience et N le nombre d’agents.

L’objectif est ici de faire en sorte que cette métrique soit supérieure ou égale à la distance d’équilibre souhaitée d_{eq} . En effet il n’est pas question de pénaliser le modèle lorsque la distance inter-agent est supérieure à d_{eq} mais quand elle est inférieure. De ce fait les déformations que subit le système en raison d’un environnement complexe n’affecteront pas les performances de ce dernier. Comme pour les autres métriques optimisées, cette dernière est ensuite passée en entrée d’une fonction de transfert afin de la normaliser entre 0 et 1.

Nous avons également ajouté une métrique qui permet de quantifier la variabilité des performances obtenues entre plusieurs simulations utilisant le même paramétrage. Nous utilisons en effet des modèles non déterministes et il est donc important que notre

modèle soit robuste aux variabilités de chaque simulation. Cette métrique, appelée *Standard Deviation*, est donc optimale lorsqu'elle tend vers 0.

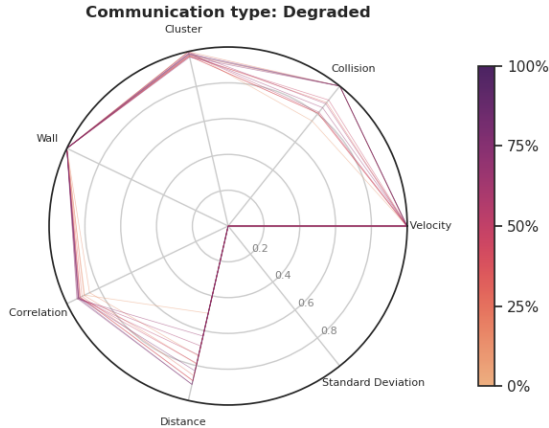


FIGURE 6.2 – Processus d’optimisation avec la distance comme fonction d’objectif.

La figure 6.2 montre la progression de l’optimisation avec la nouvelle composante en distance. On peut voir que celle-ci se rapproche de la valeur optimale 1 à chaque itération. Bien que celle-ci ne soit pas atteinte cela ne veut pas dire qu’une solution n’est pas optimale. En effet chaque courbe correspond à une moyenne de la population optimale à chaque itération, il existe donc au moins une solution ayant des performances supérieures à cette courbe.

6.3. EXPÉRIMENTATIONS

Nous avons réalisé une série de tests afin de comparer les performances de cette variante de l’optimisation avec les résultats précédents. La figure 6.3 présente la moyenne, l’écart type et la valeur maximum de l’inter-distance entre un agent et son plus proche voisin (en cm). Ce graphique correspond à la moyenne de 100 expériences avec une flotte de 16 agents pour lesquels on fixe l’inter-distance souhaitée à 20 m (2000 cm en ordonnée). On voit ici qu’après un certain temps, considéré comme état transitoire, le système atteint un équilibre en moyenne autour des 20 m, qui est la valeur désirée ce qui est bien mieux que pour la figure 6.1. La valeur exacte moyennée sur l’expérience entière est de 22,53 m, ce qui est un peu au-dessus de la valeur souhaitée mais cela s’explique par deux raisons. La première étant le biais introduit par l’état transitoire durant lequel les agents se rejoignent après avoir été positionnés initialement de manière aléatoire. La deuxième raison est liée directement à la formulation de la fonction objectif. En effet cette dernière atteint son optimum dès lors que la distance inter-agent dépasse la distance souhaitée, il est donc toujours « optimum » d’avoir une distance inter-agent supérieure à l’équilibre. On peut d’ailleurs voir que l’inter-distance maximum (en bleue) atteint des valeurs bien supérieures que celles de la figure 6.1.

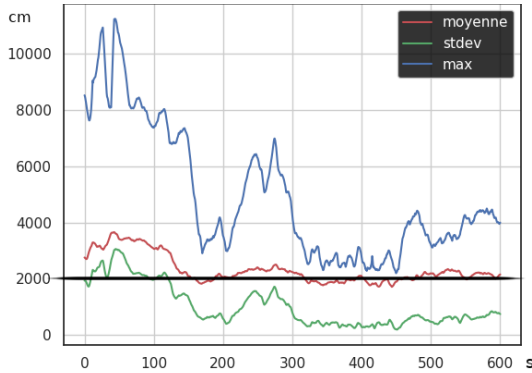


FIGURE 6.3 – Graphique représentant l’inter-distance entre un agent et son plus proche voisin (valeur moyenne sur la durée de l’expérience représentée par la ligne noire à 2 253 cm).

Finalement, même si l’on peut intégrer la distance d’équilibre comme objectif de l’optimisation, cela suppose, comme pour les autres critères, que cette optimisation soit réalisée pour chaque type d’environnement dans lequel les drones doivent évoluer.

7. CONCLUSION

Dans le cadre du flocking fondé sur la communication d’informations locales entre agents, nous avons proposé deux nouveaux modèles basés sur l’état de l’art (VAT et VOS, *i.e.* Vásárhelyi + Attraction et Vásárhelyi + Olfati-Saber) dans le but d’être robuste à des communications dégradées par la présence d’obstacles. Nous avons comparé ces deux propositions par simulation en proposant un modèle de propagation réaliste intégrant un modèle de perte (LDPL) au sein des obstacles. Cela nous a permis d’analyser comment les modèles de flocking sont affectés ou au contraire robustes à des communications dégradées par les obstacles de l’environnement. Parmi les solutions proposées, et après leur optimisation, nous pouvons affirmer que le modèle VAT donne des résultats plus encourageants que VOS ainsi que le modèle initial de Vásárhelyi *et al.* Nous avons montré plus généralement que ces modèles de flocking peuvent être optimisés en considérant un ensemble de critères en lien avec la robustesse et la qualité des comportements attendus. Nous avons aussi montré que cette optimisation peut se faire en fixant des paramètres souhaités par l’utilisateur, comme la distance d’équilibre entre agents.

Bien que l’optimisation ait été réalisée ici avec un environnement « generic », il pourrait être intéressant de systématiquement optimiser les modèles dans chacun des environnements utilisés. *A contrario*, la suite de notre travail consistera à explorer des stratégies de flocking adaptatives capables de se comporter de manière optimale dans des environnements hétérogènes, en terme de densité d’obstacles, ou avec différents types de communication. Enfin, nous préparons aussi des expérimentations pour mesurer les performances des modèles de flocking avec des drones quadricoptère réels.

REMERCIEMENTS

Ce travail a été réalisé dans le cadre du projet « DYNAFLOCK » co-financé par Inria et AID-DGA (convention N° 2019 65 0072 0072 00 470 75 01). Les expérimentations ont été réalisées avec le support de la plateforme Grid5000, supportée par un groupe d'intérêt scientifique géré par l'INRIA et incluant le CNRS, RENATER et de nombreuses universités ainsi que d'autres organisations (voir <https://www.grid5000.fr>).

BIBLIOGRAPHIE

- [1] T. BALCH & R. C. ARKIN, « Behavior-based formation control for multirobot teams », *IEEE Transactions on Robotics and Automation* **14** (1998), n° 6, p. 926-939.
- [2] J. BLANK & K. DEB, « Pymoo: Multi-Objective Optimization in Python », *IEEE Access* **8** (2020), p. 89497-89509.
- [3] N. BOURAQADI & A. DONIEC, « Flocking-Based Multi-Robot Exploration », in *4th National Conference on "Control Architectures of Robots"* (Toulouse, France), 2009.
- [4] K. DEB & H. JAIN, « An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints », *IEEE Transactions on Evolutionary Computation* **18** (2014), n° 4, p. 577-601.
- [5] K. DEB, A. PRATAP, S. AGARWAL & T. MEYARIVAN, « A fast and elitist multiobjective genetic algorithm: NSGA-II », *IEEE Transactions on Evolutionary Computation* **6** (2002), n° 2, p. 182-197.
- [6] B. FRASER, R. HUNJET & C. SZABO, « Simulating the effect of degraded wireless communications on emergent behavior », in *2017 Winter Simulation Conference (WSC)*, 2017, p. 4081-4092.
- [7] G. W. GAMAGE, G. K. I. MANN & R. G. GOSINE, « Leader follower based formation control strategies for nonholonomic mobile robots: Design, implementation and experimental validation », in *Proceedings of the 2010 American Control Conference*, 2010, p. 224-229.
- [8] S. S. GE & CHENG-HENG FUA, « Queues and artificial potential trenches for multirobot formations », *IEEE Transactions on Robotics* **21** (2005), n° 4, p. 646-656.
- [9] R. GRÜNBLATT, I. GUÉRIN-LASSOUS & O. SIMONIN, « Simulation and Performance Evaluation of the Intel Rate Adaptation Algorithm », in *MSWiM 2019 - 22nd ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems* (Miami Beach, United States), ACM, 2019, p. 27-34.
- [10] N. HANSEN, « The CMA Evolution Strategy: A Tutorial », <http://arxiv.org/abs/1604.00772>, 2016.
- [11] JIA-LIANG LU & F. VALOIS, « Performance evaluation of 802.11 WLAN in a real indoor environment », in *2006 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, 2006, p. 140-147.
- [12] H. M. LA & WEIHUA SHENG, « Flocking control of a mobile sensor network to track and observe a moving target », in *2009 IEEE International Conference on Robotics and Automation (Kobe)*, IEEE, 2009, p. 3129-3134.
- [13] H. LI, J. PENG, W. LIU, J. WANG, J. LIU & Z. HUANG, « Flocking control for multi-agent systems with communication optimization », in *Proceedings of the American Control Conference*, 2013, p. 2056-2061.
- [14] N. H. M. LI & H. H. T. LIU, « Formation UAV flight control using virtual structure and motion synchronization », in *2008 American Control Conference*, 2008, p. 1782-1787.
- [15] R. LIM & W. SHENG, « Hybrid System of Reinforcement Learning and Flocking Control in Multi-robot Domain », in *IEEE Transactions on Control Systems Technology*, 2016.
- [16] J. D. MORGAN III & W. KUTZELNIGG, « Hund's Rules, The Alternating Rule and Symmetry Holes », *The Journal of Physical Chemistry* **97** (1993), n° 10, p. 2425-2434.
- [17] R. OLFATI-SABER, « Flocking for multi-agent dynamic systems: algorithms and theory », *IEEE Trans. Automat. Contr.* **51** (2006), n° 3, p. 401-420.
- [18] T. RAPPAPORT, *Wireless Communications: Principles and Practice*, 2nd éd., Prentice Hall PTR, USA, 2001.

- [19] A. RENZAGLIA, J. S. DIBANGOYE, V. LE DOZE & O. SIMONIN, « Combining Stochastic Optimization and Frontiers for Aerial Multi-Robot Exploration of 3D Terrains », in *ROS 2019 - IEEE/RSJ International Conference on Intelligent Robots and Systems* (Macau, China), 2019, p. 4121-4126.
- [20] L. REYNAUD & I. GUÉRIN-LASSOUS, « Design of a force-based controlled mobility on aerial vehicles for pest management », *Ad Hoc Networks* **53** (2016), p. 41-52.
- [21] C. W. REYNOLDS, « Flocks, Herds and Schools: A Distributed Behavioral Model », *SIGGRAPH Comput. Graph.* **21** (1987), n° 4, p. 25-34.
- [22] H. G. TANNER, A. JADBABAIE & G. J. PAPPAS, « Flocking in Fixed and Switching Networks », *IEEE Transactions on Automatic Control* **52** (2007), n° 5, p. 863-868.
- [23] G. VÁSÁRHELYI, C. VIRÁGH, G. SOMORJAI, T. NEPU SZ, A. E. EIBEN & T. VICSEK, « Optimized flocking of autonomous drones in confined environments », *Science Robotics* **3** (2018), n° 20, article no. eaat3536.
- [24] T. VICSEK, A. CZIROK, E. BEN-JACOB, I. COHEN & O. SHOCHET, « Novel Type of Phase Transition in a System of Self-driven Particules », *Physical review letters* **75** (1995), n° 6, p. 1226-1229.
- [25] C. VIRÁGH, G. VÁSÁRHELYI, N. TARCAI, T. SZÖRÉNYI, G. SOMORJAI, T. NEPU SZ & T. VICSEK, « Flocking algorithm for autonomous flying robots », *Bioinspiration & Biomimetics* **9** (2014), n° 2, article no. 025012.
- [26] G. WANG, J. WANG, Y. SHEN & X. ZHANG, « Autonomous Navigation of UAVs in Large-Scale Complex Environments: A Deep Reinforcement Learning Approach », *Transactions on Vehicular Technology* **68** (2018), n° 3, p. 2124-2136.
- [27] S. WATZA, R. KANLAPULI & E. FREW, « Hybrid RF Propagation Model using ITM and Gaussian Processes for Communication-Aware Planning », in *RSS 2017 RCW Workshop*, 2017.
- [28] G. WEN, Z. DUAN, H. SU, G. CHEN & W. YU, « A Connectivity-preserving flocking algorithm for multi-agent dynamical systems with bounded potential function », *IET Control Theory Applications* **6** (2012), n° 6, p. 813-821.
- [29] G. WEN, Z. DUAN, H. SU, G. CHEN & W. YU, « A connectivity-preserving flocking algorithm for nonlinear multi-agent systems with bounded potential function », in *Proceedings of the 30th Chinese Control Conference, CCC 2011*, 2011, p. 6018-6024.
- [30] M. M. ZAVLANOS, M. B. EGERSTEDT & G. J. PAPPAS, « Graph-theoretic connectivity control of mobile robot networks », *Proceedings of the IEEE* **99** (2011), n° 9, p. 1525-1540.

ABSTRACT. — In this paper, we study existing flocking models and propose extensions to improve their abilities to deal with environments having obstacles impacting the communication quality as well as the trajectories of the agents. Indeed, the constraints induced by the obstacles usually lead to communication outages resulting in the separation of the flock in multiple clusters. In this context, we extend two standard models to improve their ability to stay connected while evolving in environments with different obstacles distributions. By taking into account the radio propagation, we model the obstacles' impact on communications in a simulator that we use to optimize flocking parameters. The simulation results show the efficiency of the proposed models and how they adapt to different environmental constraints.

KEYWORDS. — Flocking models, swarm robotics, simulation of communications.
