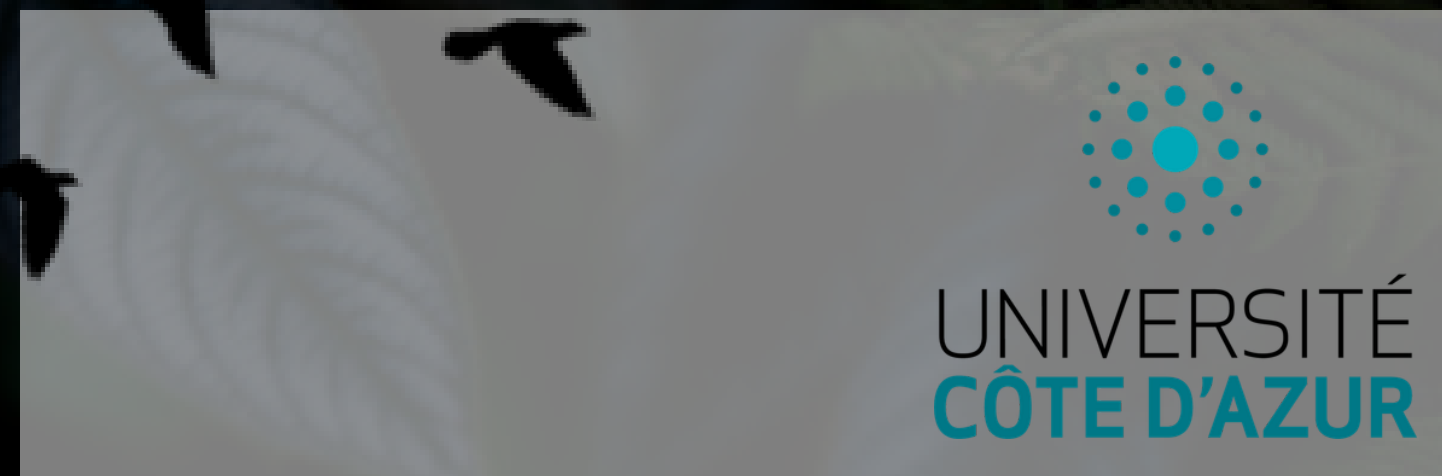
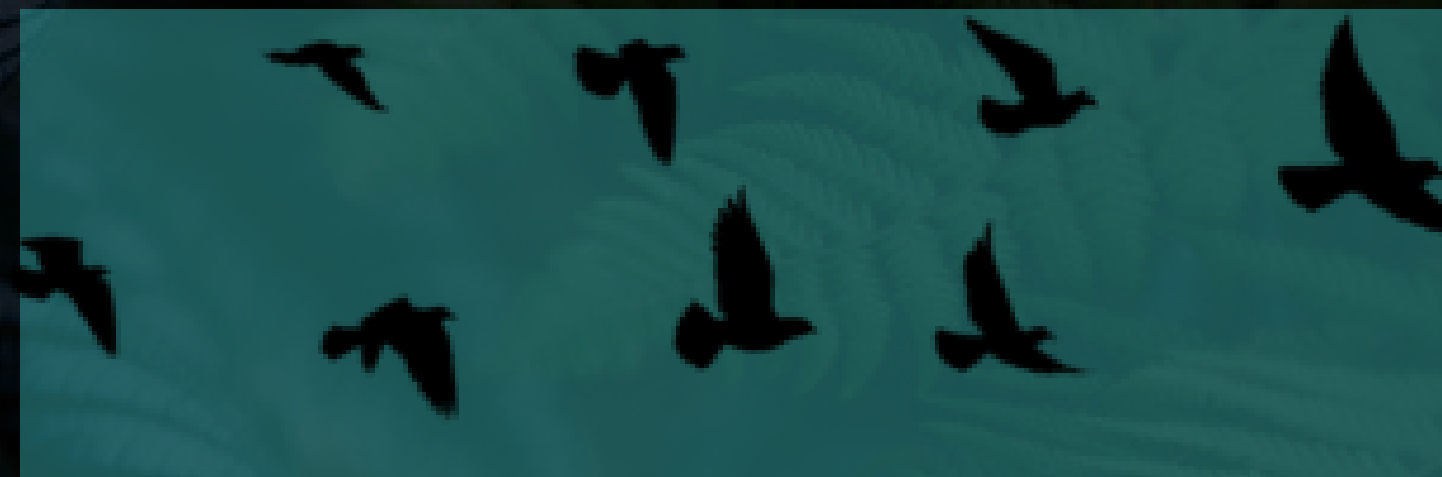
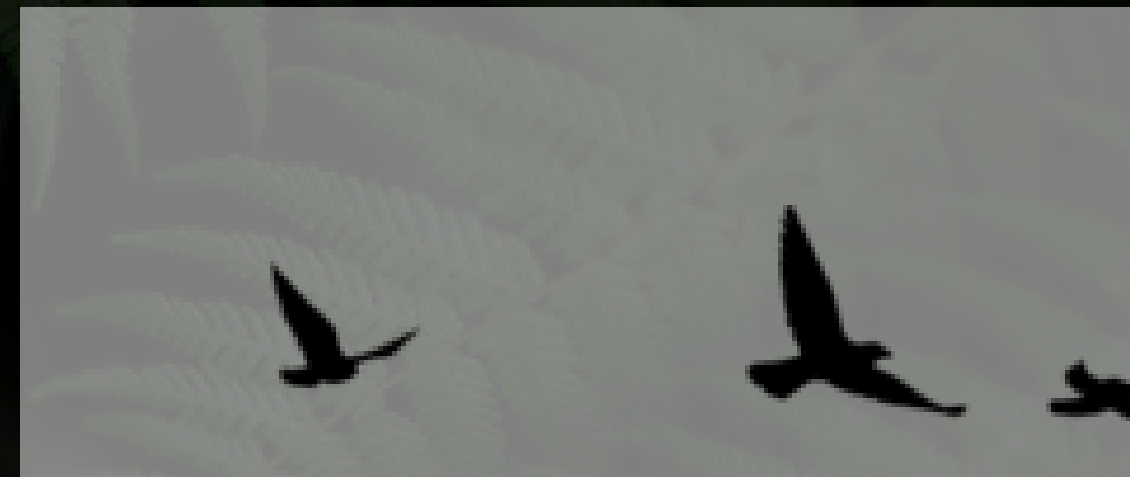


Simulation de

NUÉES D'OISEAUX

Projet réalisé par : Kevin Corona, Clément Dupouy-Paulin et Paul Arnaud

IUT Nice Côte d'Azur - Université Côte d'Azur
Département Informatique



Projet Parvada • Vol en formation & gestion énergétique

INTRODUCTION

Contexte scientifique

- Vol en « V » : capture de l'*upwash* → économie d'énergie (≈ -10 à -15%)
- Coût collectif : le leader subit la traînée maximale (rotation du rôle)

Originalité du projet

- Au-delà des règles cinématiques (Reynolds, 1987)
- Chaque agent gère un réservoir d'énergie fini → stratégies de relais & endurance

$\approx -15\%$
fréquence cardiaque

Leader : traînée ↑



PROBLÉMATIQUE & HYPOTHÈSE

Vent turbulent • pression de prédation • intelligence collective

? Question centrale

Dans un environnement stochastique,
l'organisation de la nuée (fixe vs dynamique)
améliore-t-elle la résilience globale ?

💡 Hypothèse principale

Leadership rotatif (seuil de fatigue locale) :

- ↑ distance parcourue
 - ↓ mortalité
- gain énergétique collectif > coût cognitif



Hiérarchie : fixe
↔dynamique



MODÉLISATION



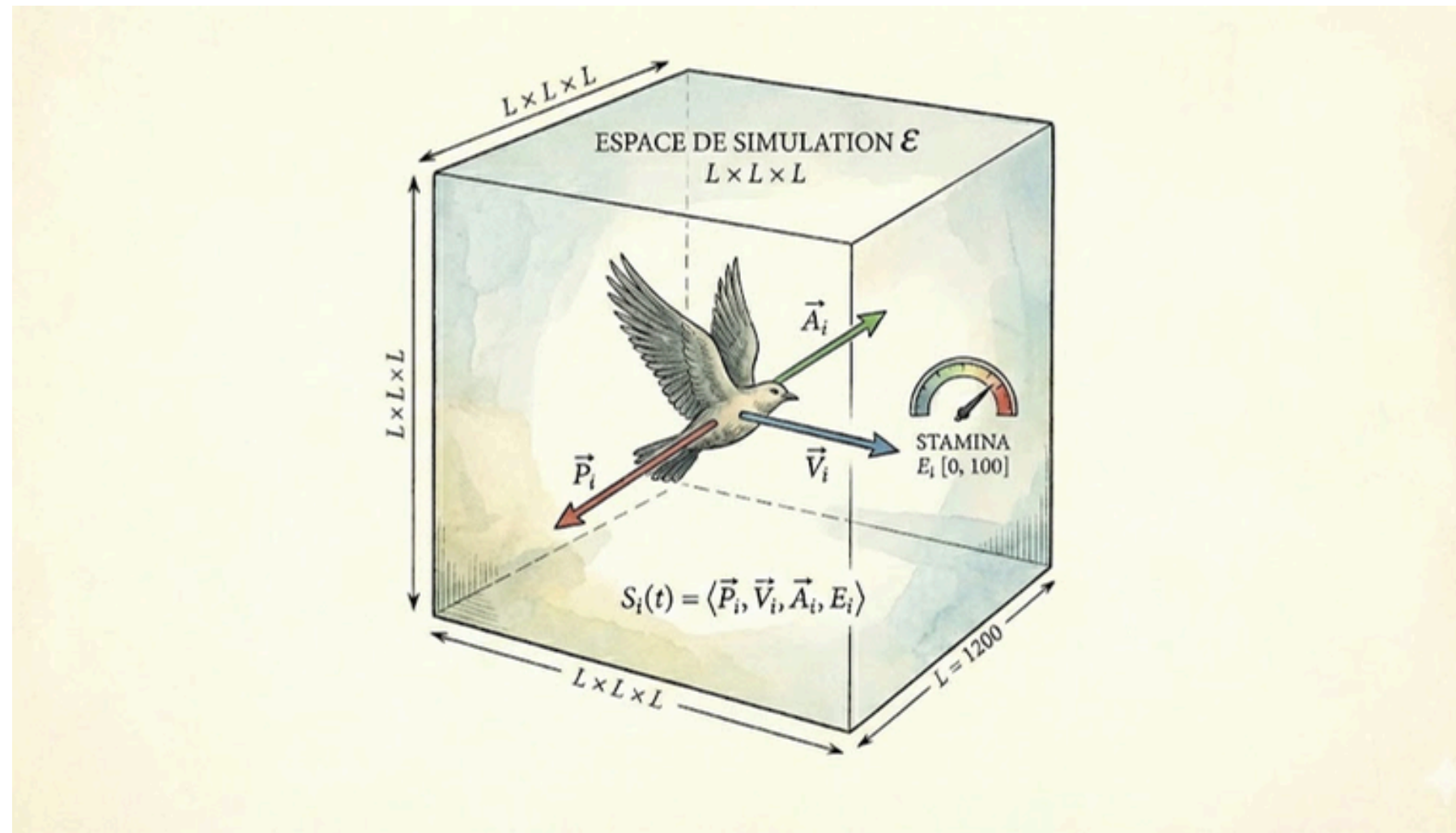
01. Architecture du système

02. Dynamique d'un oiseau isolé

03. Modèle de Leadership et Fatigue



ARCHITECTURE DU SYSTÈME



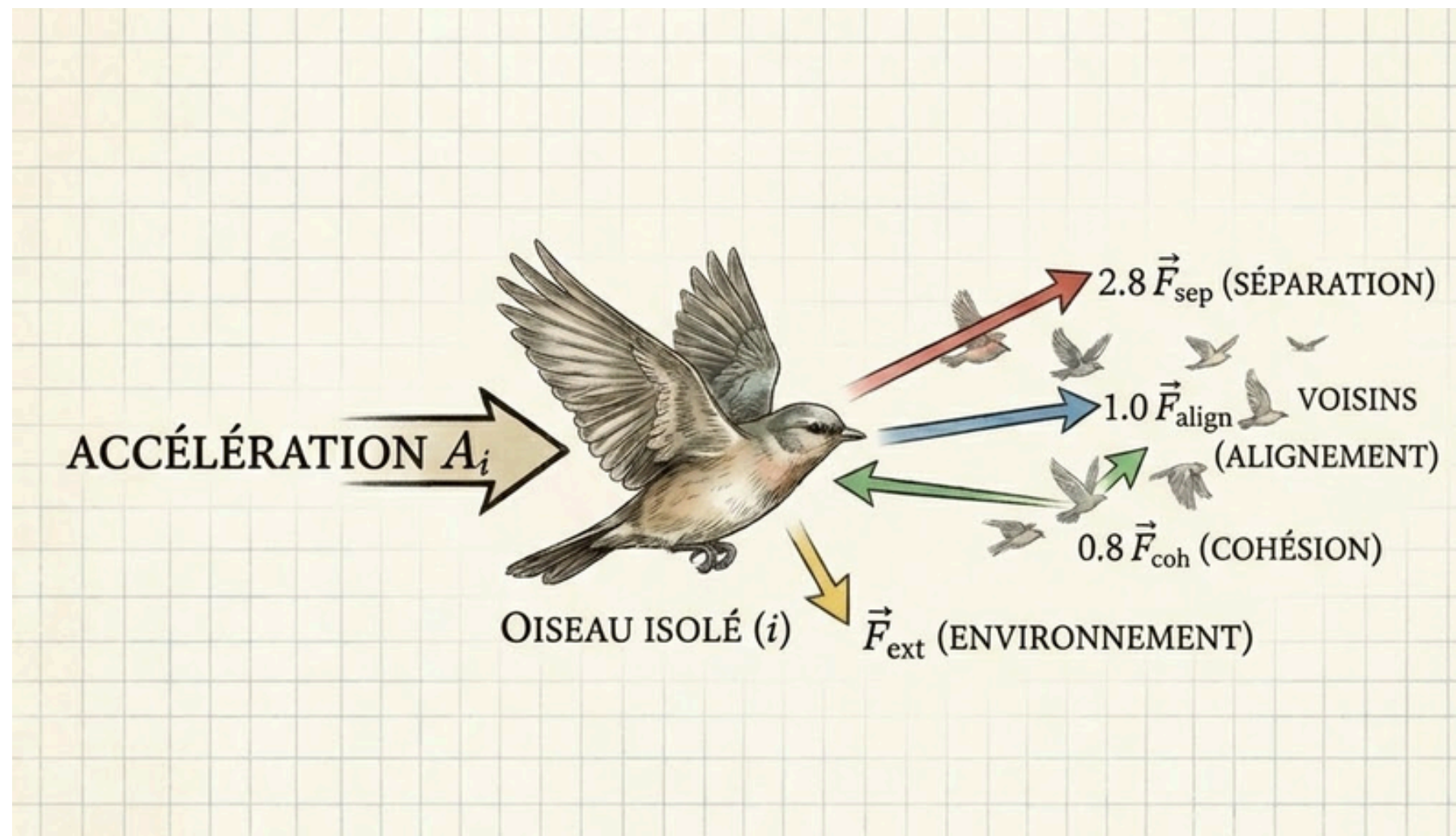
L'espace de simulation E est un cube de dimensions $L \times L \times L$ où $L = 1200$. L'état d'un agent i à l'instant t est défini par le tuple :

$$S_i(t) = \langle \vec{P}_i, \vec{V}_i, \vec{A}_i, E_i \rangle$$

Où :

- $\vec{P}_i \in \mathbb{R}^3$: Le vecteur position (x, y, z) de l'agent.
- $\vec{V}_i \in \mathbb{R}^3$: Le vecteur vitesse courant, déterminant la direction et la rapidité.
- $\vec{A}_i \in \mathbb{R}^3$: Le vecteur accélération, résultante de toutes les forces appliquées.
- $E_i \in [0, 100]$: Le niveau d'énergie (Stamina), variable scalaire.

DYNAMIQUE D'UN OISEAU ISOLÉ



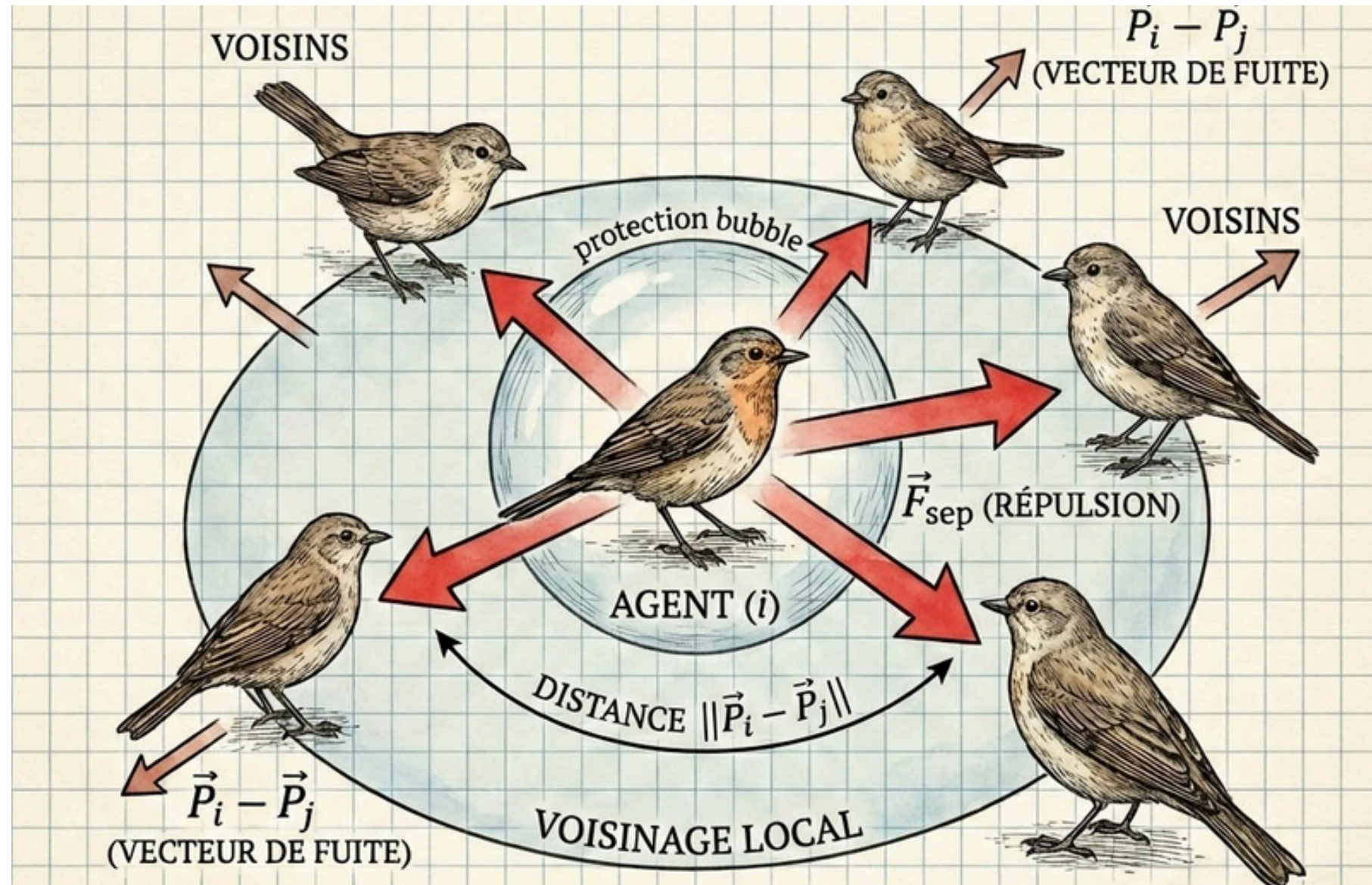
L'accélération à chaque instant est la somme pondérée de toutes les forces appliquées à l'agent.

$$\vec{A}_i = \underbrace{2.8 \vec{F}_{sep} + 1.0 \vec{F}_{align} + 0.8 \vec{F}_{coh}}_{\text{Forces Internes (Boids)}} + \underbrace{\vec{F}_{ext}}_{\text{Environnement}}$$

Principe utilisé : Modèle de Reynolds (Boids, 1986) : Comportement émergent basé sur trois règles locales simples (Séparation, Alignement, Cohésion) sans coordination centrale.



FORCE DE SÉPARATION (RÉPULSION)



Cette force assure l'évitement des collisions en générant une répulsion drastique lorsque deux agents sont trop proches.

$$\vec{F}_{sep} = \sum_{j \in N_i} \frac{\vec{P}_i - \vec{P}_j}{\|\vec{P}_i - \vec{P}_j\|^2 + \epsilon}$$

Variables :

- N_i : Voisinage local de l'agent ($r < 28.0$).
- $\vec{P}_i - \vec{P}_j$: Vecteur de fuite.
- $\|\dots\|^2$: Carré de la distance.

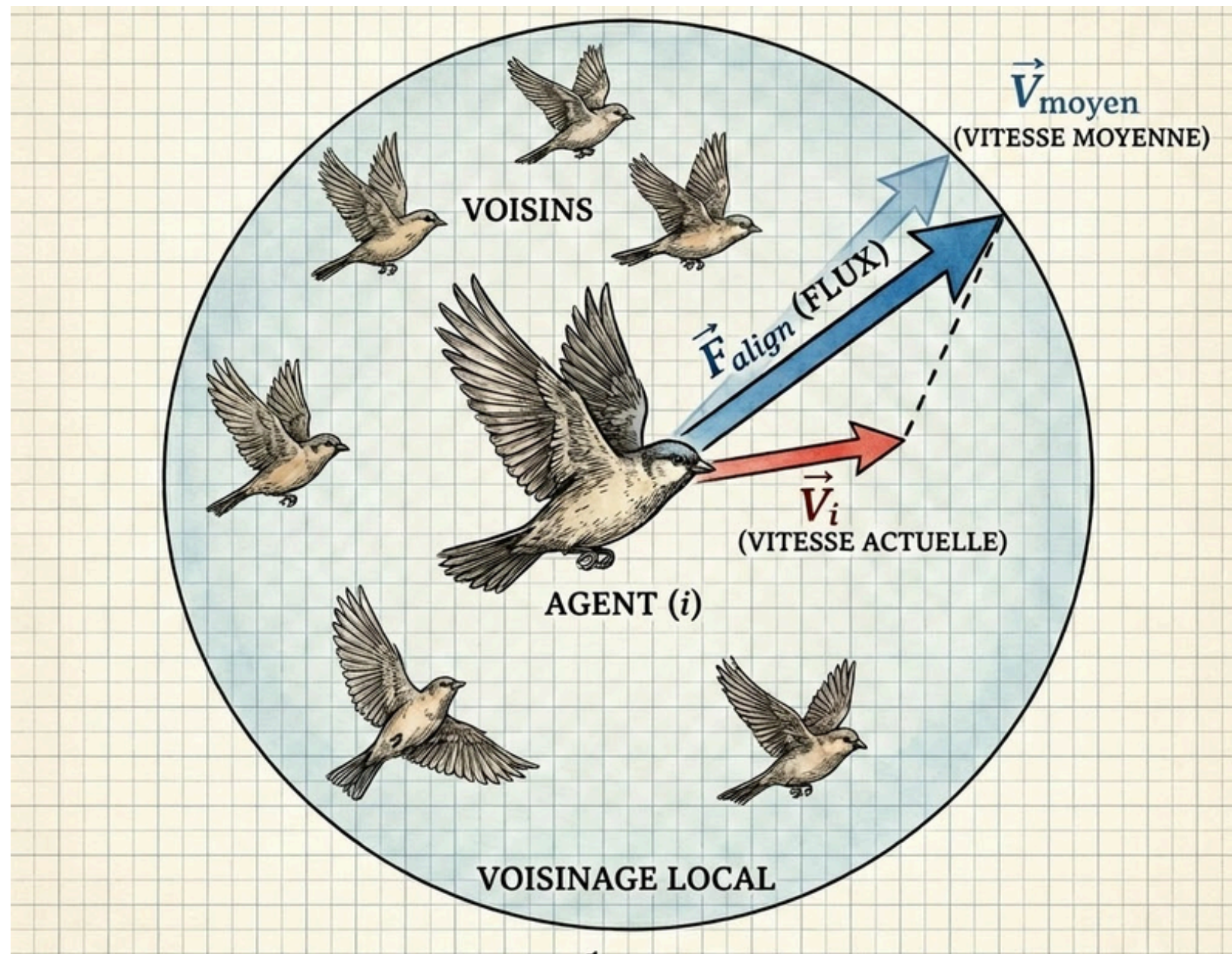
Méthode de Calcul :

- On soustrait la position du voisin \vec{P}_j à celle de l'agent \vec{P}_i pour obtenir la direction opposée.
- On divise ce vecteur par la distance au carré pour donner un poids infini aux voisins immédiats.
- On somme tous ces vecteurs pour obtenir la résultante.

Principe utilisé : Loi en Carré Inverse (Similaire à Coulomb) : La force diminue avec le carré de la distance, créant une "bulle" de protection quasi-impénétrable autour de l'agent.



FORCE D'ALIGNEMENT (FLUX)



Cette force incite l'agent à synchroniser sa direction et sa vitesse avec la moyenne de ses voisins.

$$\vec{F}_{align} = \left(\frac{1}{|N_i|} \sum_{j \in N_i} \vec{V}_j \right) - \vec{V}_i$$

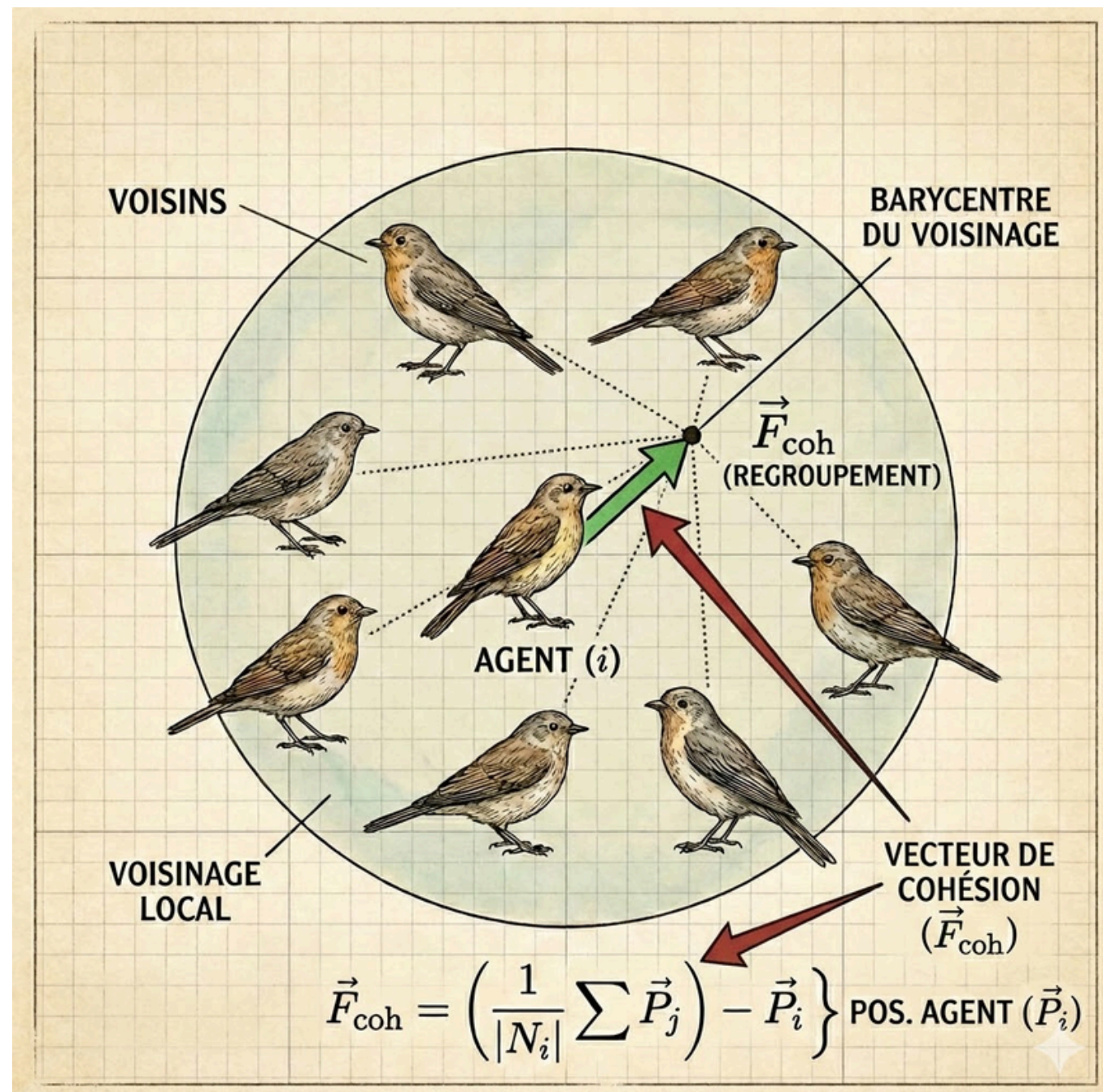
- $\sum \vec{V}_j$: Somme des vitesses des voisins.
- $|N_i|$: Nombre de voisins.

Méthode de Calcul :

- On additionne les vecteurs vitesse de tous les voisins.
- On divise par le nombre de voisins pour obtenir la vitesse moyenne locale.
- On soustrait la vitesse actuelle de l'agent pour obtenir le vecteur de correction (steering force).



FORCE DE COHÉSION (REGROUPEMENT)



Cette force maintient la structure du groupe en attirant l'agent vers le centre géométrique de ses voisins.

$$\vec{F}_{coh} = \left(\frac{1}{|N_i|} \sum_{j \in N_i} \vec{P}_j \right) - \vec{P}_i$$

Variables :

— $\frac{1}{|N_i|} \sum \vec{P}_j$: Barycentre du voisinage.

Méthode de Calcul :

- On fait la moyenne des positions (x, y, z) de tous les voisins (Barycentre).
- On soustrait la position de l'agent à ce barycentre pour créer un vecteur d'attraction.

FORCE DE FUITE (RÉPONSE AU PRÉDATEUR)

Cette force simule la panique : une répulsion exponentielle qui prime sur tout comportement social en présence d'un danger.

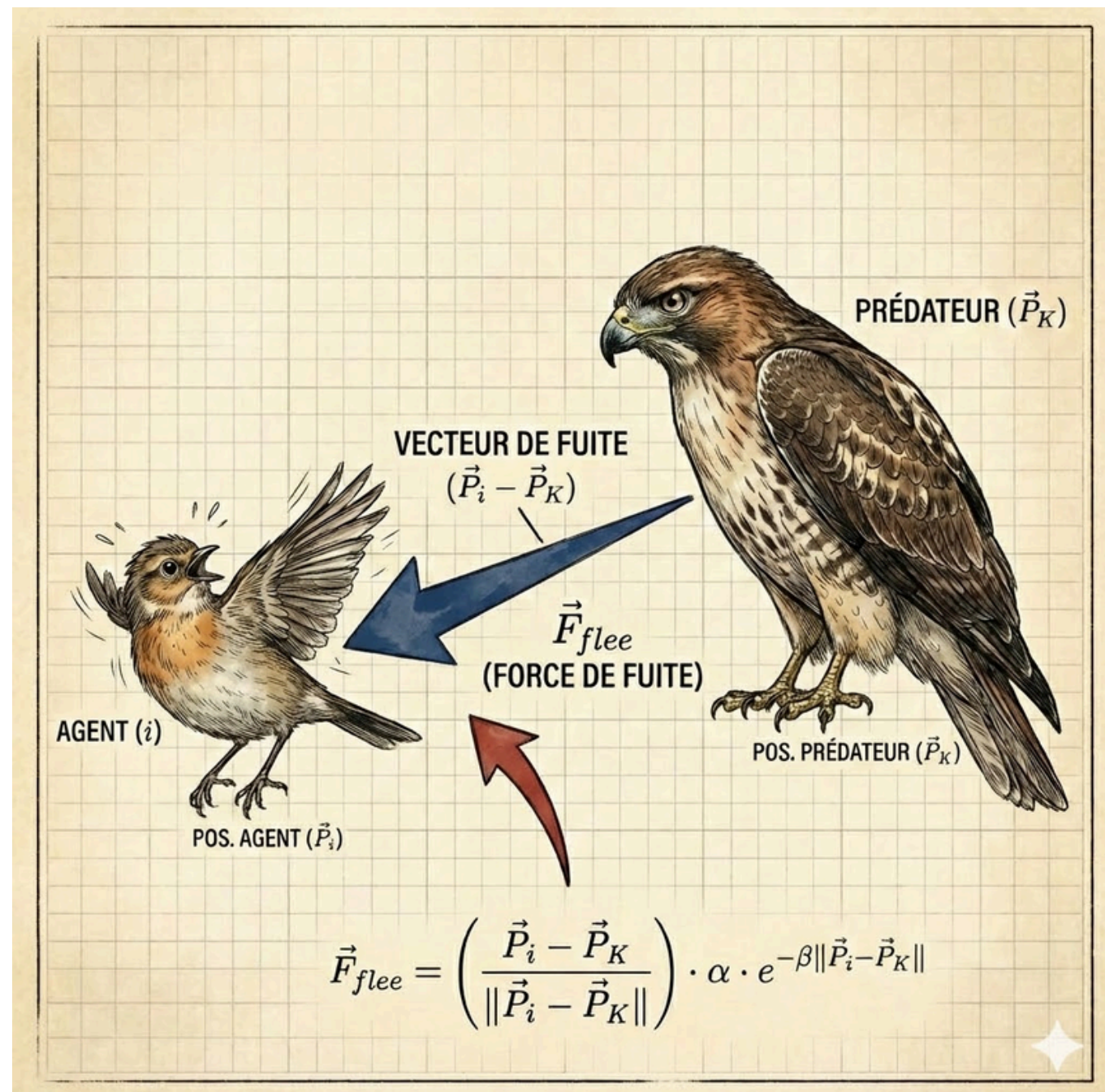
$$\vec{F}_{flee} = \frac{\vec{P}_i - \vec{P}_K}{\|\vec{P}_i - \vec{P}_K\|} \cdot \alpha \cdot e^{-\beta \|\vec{P}_i - \vec{P}_K\|}$$

Variables :

- \vec{P}_K : Position du prédateur.
- \vec{P}_i : Position de l'agent i .
- $\vec{P}_i - \vec{P}_K$: Vecteur directionnel fuyant le prédateur.
- $\|\vec{P}_i - \vec{P}_K\|$: Distance entre l'agent et le prédateur (norme du vecteur).
- $\frac{\vec{P}_i - \vec{P}_K}{\|\vec{P}_i - \vec{P}_K\|}$: Vecteur unitaire directionnel fuyant le prédateur.
- α : Intensité de la peur. (**Magnitude maximale de la force de répulsion**)
- β : Coefficient de l'exposant qui contrôle la rapidité avec laquelle la force diminue avec la distance.
- $e^{-\beta \|\vec{P}_i - \vec{P}_K\|}$: Facteur d'atténuation exponentielle de la distance.

Méthode de Calcul :

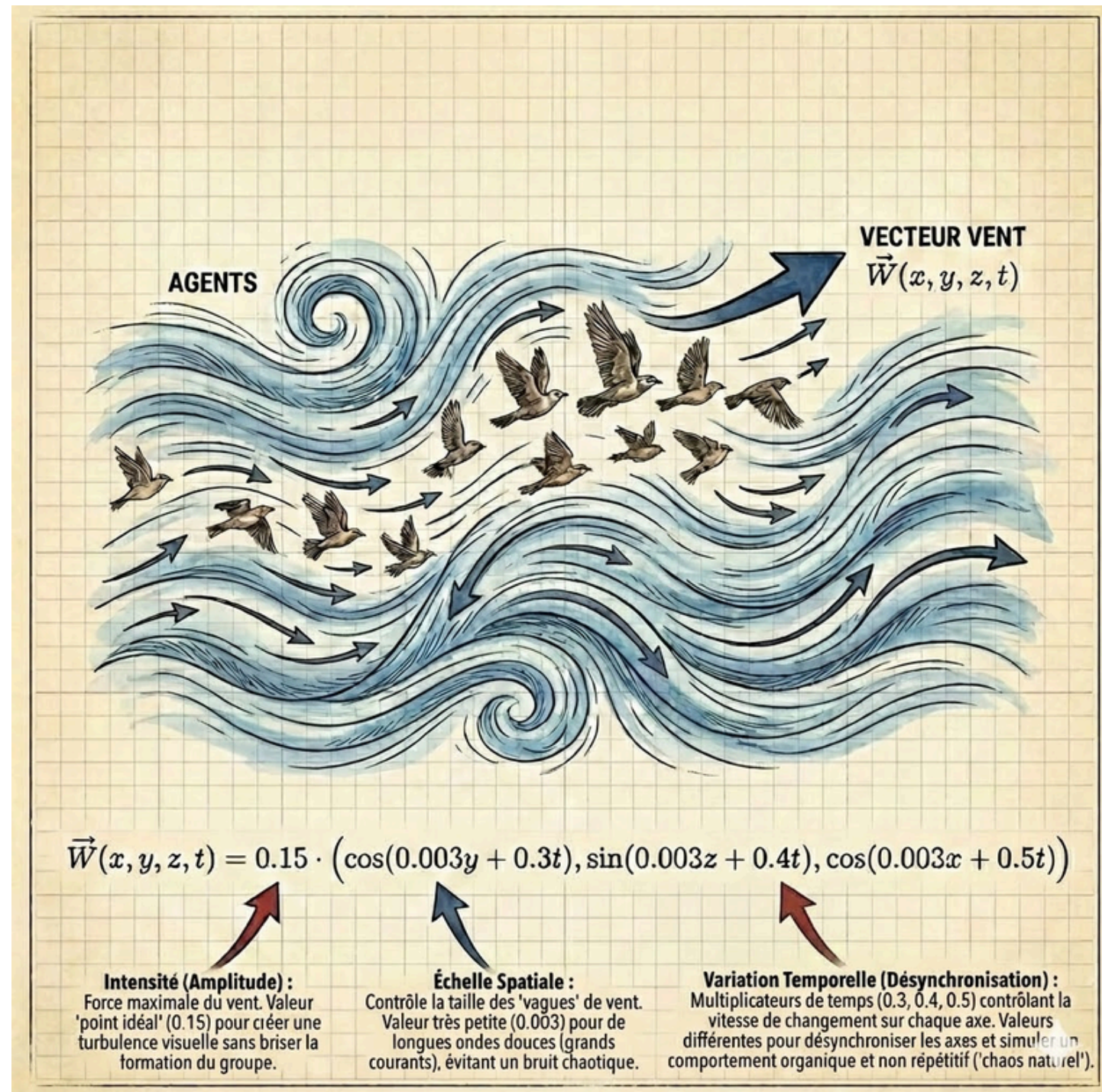
1. Direction : Calcul du vecteur unitaire fuyant le prédateur $\left(\frac{\vec{P}_i - \vec{P}_K}{\|\vec{P}_i - \vec{P}_K\|}\right)$.
2. Urgence : Application d'une exponentielle décroissante de la distance ($e^{-\beta d}$) : la force explose si le danger est proche.
3. Intensité : Multiplication par le coefficient α (peur maximale) pour obtenir la force résultante finale.



Principe utilisé : Décroissance Exponentielle (Réponse aux Stimuli) : En biologie, la réaction de fuite n'est pas linéaire ; elle s'active brutalement lorsqu'un seuil de proximité critique est franchi.



FORCE DU VENT (CHAMP VECTORIEL)



Cette force applique une turbulence environnementale continue et cohérente spatialement.

$$\vec{W}(x, y, z, t) = 0.15 \cdot \begin{pmatrix} \cos(0.003y + 0.3t) \\ \sin(0.003z + 0.4t) \\ \cos(0.003x + 0.5t) \end{pmatrix}$$

Variables :

- t : Temps écoulé.
- 0.003 : Fréquence spatiale (échelle des tourbillons).

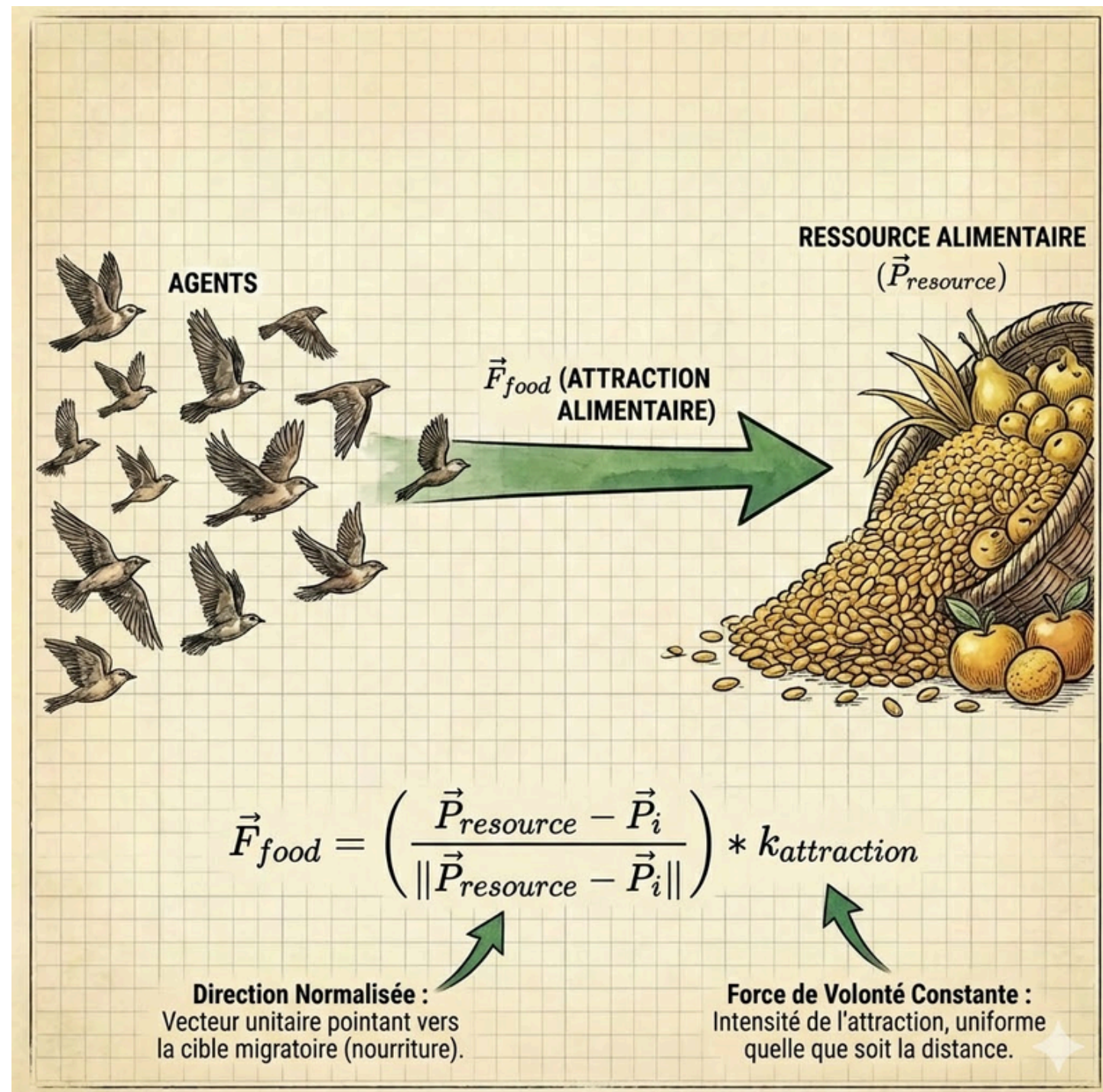
Méthode de Calcul :

- On utilise des coordonnées croisées (Y pour calculer X) dans des fonctions trigonométriques.
- On ajoute une composante temporelle t pour faire évoluer le champ.
- On assemble le vecteur résultant pour obtenir une direction de vent fluide.

Principe utilisé : Champs Vectoriels / Bruit Pseudo-Aléatoire : Utilisation de fonctions harmoniques couplées pour simuler des turbulences naturelles (non chaotiques) similaires au Bruit de Perlin.



ATTRACTION ALIMENTAIRE



Cette force représente l'objectif global du groupe (migration vers une ressource).

$$\vec{F}_{food} = \frac{\vec{P}_{resource} - \vec{P}_i}{\|\vec{P}_{resource} - \vec{P}_i\|} \cdot k_{attraction}$$

Variables :

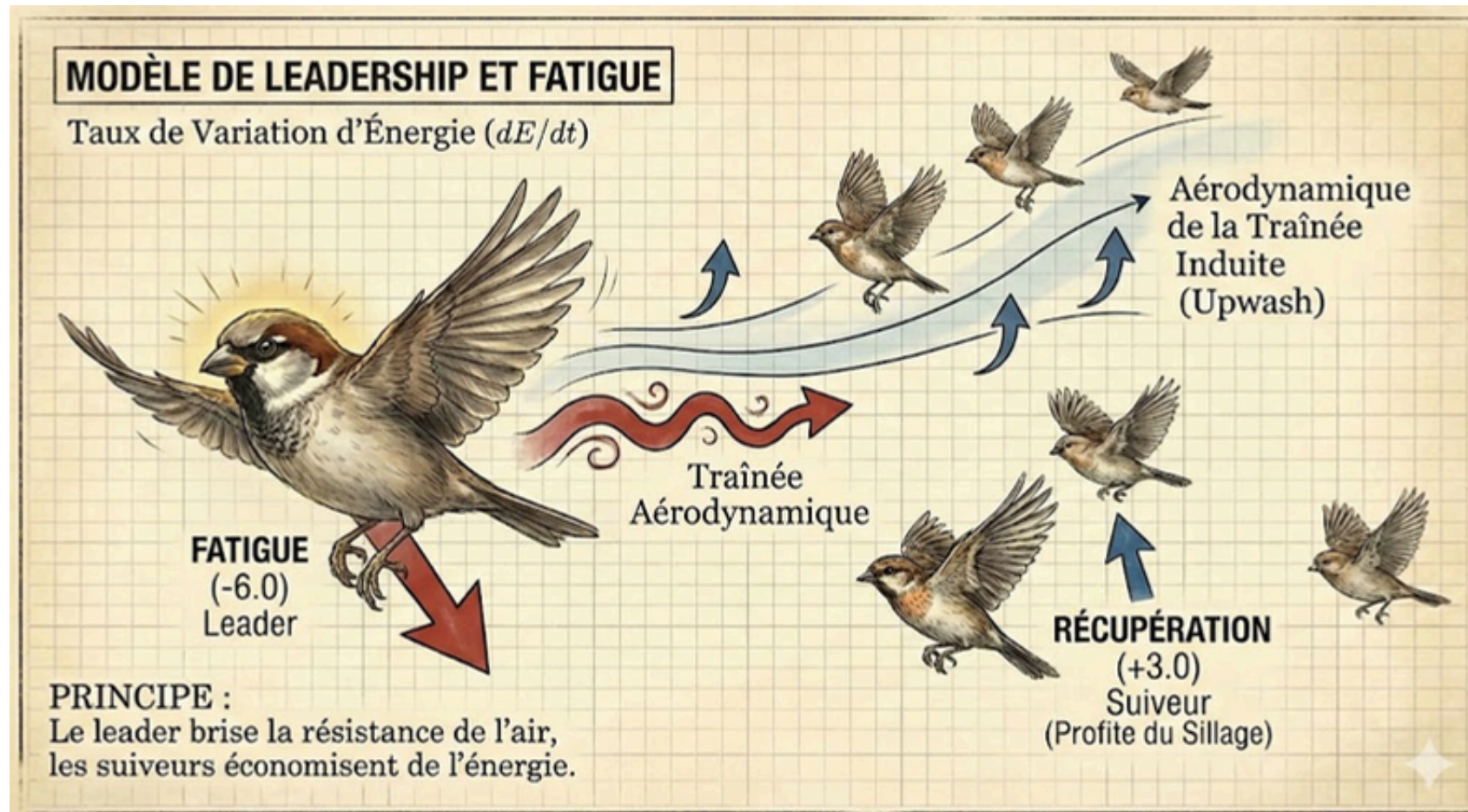
- $\vec{P}_{resource}$: Cible migratoire.
- $k_{attraction}$: Force de volonté constante.

Méthode de Calcul :

- On calcule le vecteur normalisé pointant vers la nourriture.
- On le multiplie par une constante k pour que l'attraction soit uniforme quelle que soit la distance.



TAUX DE VARIATION D'ÉNERGIE



Cette équation différentielle régit la consommation ou la récupération d'énergie selon la position aérodynamique.

$$\frac{dE}{dt} = \begin{cases} -6.0 & \text{si Leader (Trainée Aérodynamique)} \\ +3.0 & \text{si Suiveur (Profite du Sillage)} \end{cases}$$

Variables :

— dE/dt : Variation d'énergie par seconde.

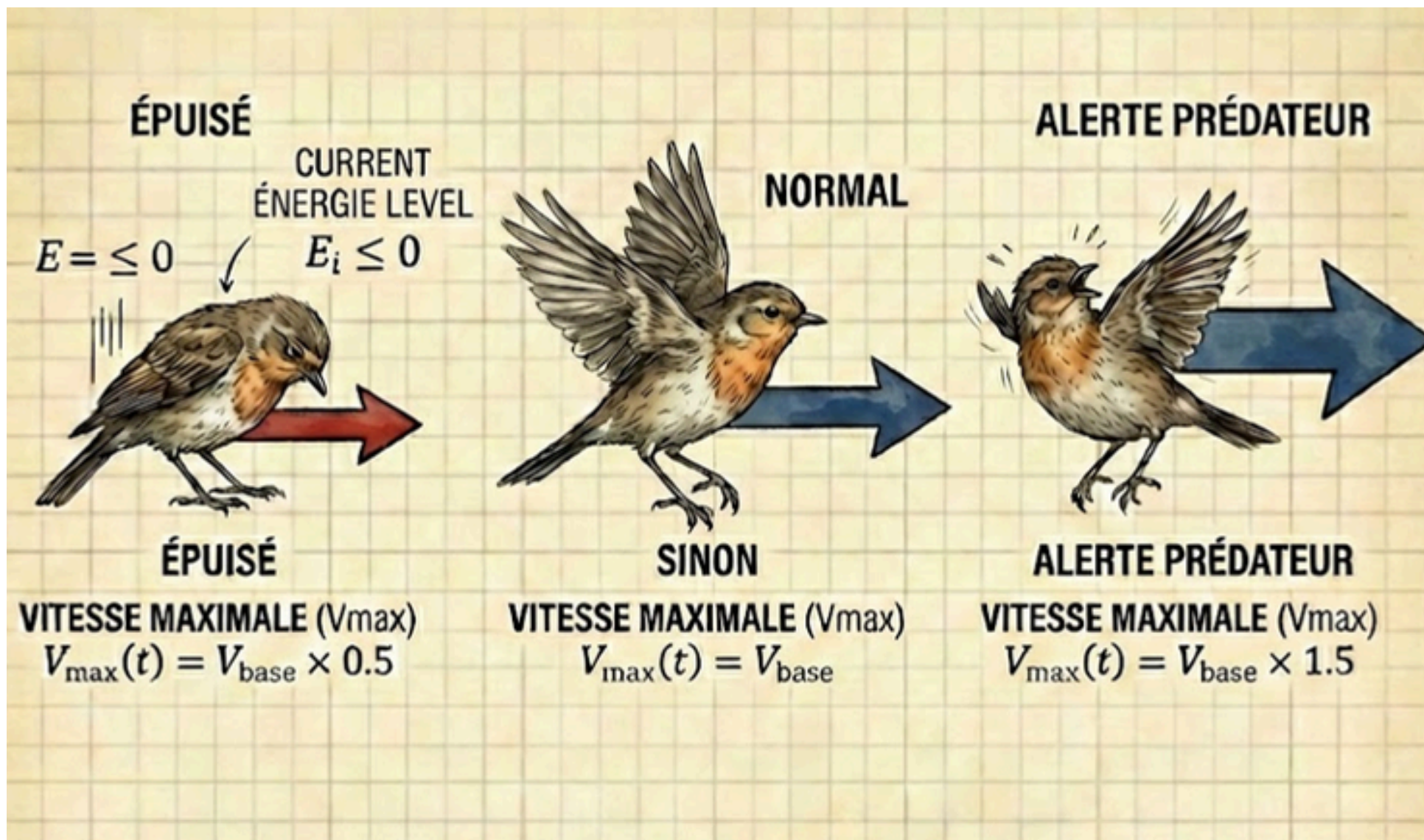
Méthode de Calcul :

- On vérifie à chaque pas de temps si l'oiseau a un voisin devant lui (produit scalaire positif).
- Si non (Leader), on soustrait une grande quantité d'énergie.
- Si oui (Suiveur), on ajoute une quantité moyenne d'énergie.

Principe utilisé : Aérodynamique de la Trainée Induite (Upwash) : Le leader brise la résistance de l'air, créant des vortex porteurs dont profitent les suiveurs pour planer (économie d'énergie).



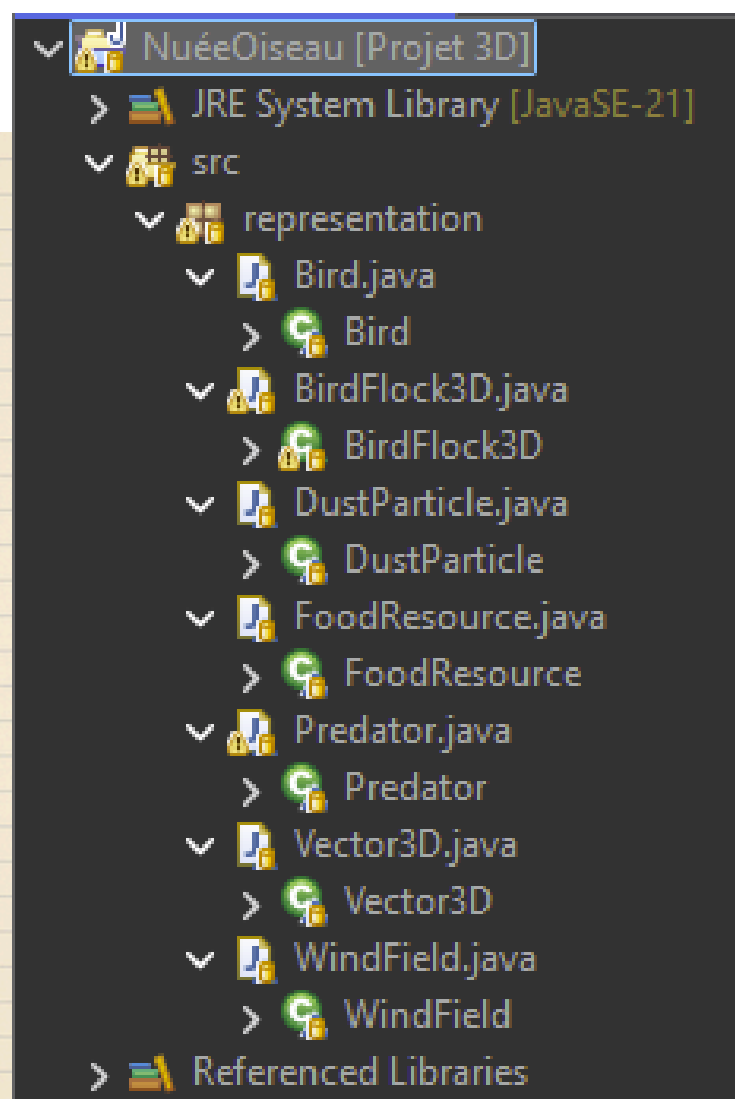
CONSÉQUENCE MÉCANIQUE : FATIGUE ET VITESSE



Cette règle modifie les capacités physiques de l'agent en fonction de son état physiologique interne.

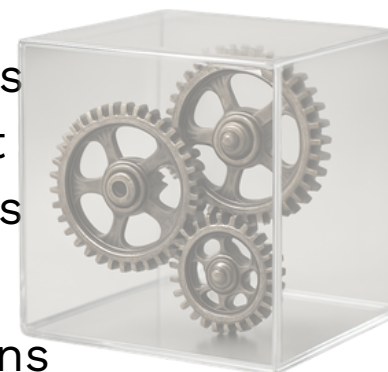
$$V_{max}(t) = \begin{cases} V_{base} \times 0.5 & \text{si } E_i \leq 0 \text{ (Épuié)} \\ V_{base} \times 1.5 & \text{si Alerte Prédateur (Adrénaline)} \\ V_{base} & \text{sinon} \end{cases}$$

ARCHITECTURE DU PROJET



BirdFlock3D.java correspond au main de notre projet, elle définit plusieurs variables pour lancer la simulation et quelques options utilisateurs

Vector3D.java nous permet d'instancier plusieurs fonctions qui nous seront utiles pour les calculs de vecteurs



FoodResource.java gère la génération des différents points d'intérêt (nourriture) qui attire les oiseaux

Predator.java gère toute la logique concernant le prédateur avec notamment ses périodes de chasse/repos etc...



Bird.java gère toute la logique des oiseaux, mouvement, direction, état, regroupement etc...



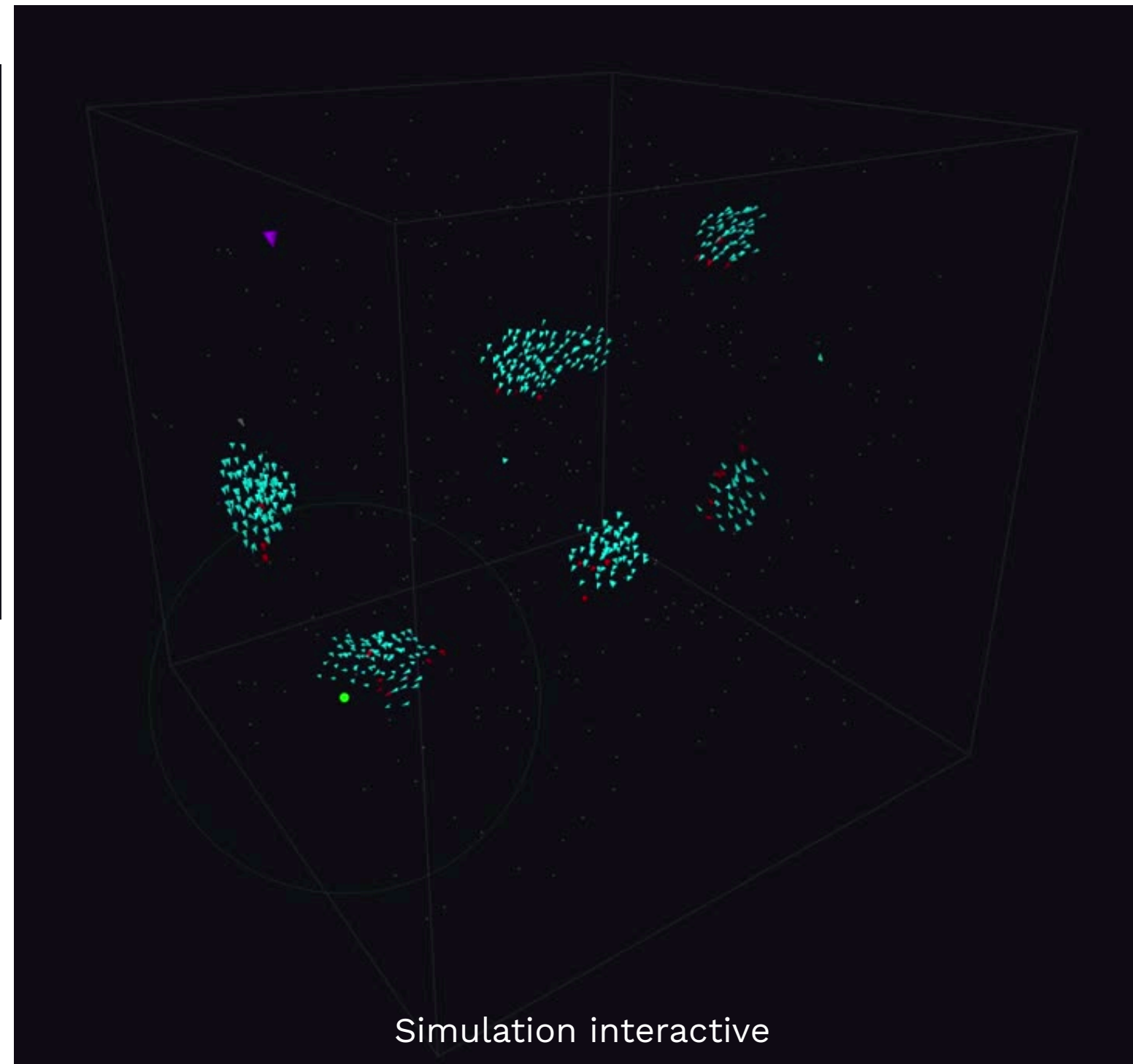
WindField.java nous permet de générer les forces et directions du vent

DustParticle.java gère l'affichage des particules de poussière permettant de représenter avec leur mouvement : la force et la direction du vent

RENDU DE LA SIMULATION

Tableau des statistiques

--- PRÉDATEUR ---	
Mangés (Total)	: 192
Quota Actuel	: 0/2
État:	CHASSE
--- OISEAUX ---	
Population	: 450
Panique	: 2
Fatigués	: 0
■ Leader (Fatigue)	■ Suiveur
■ Alerte	■ Épuisé
. Poussière (Vent)	





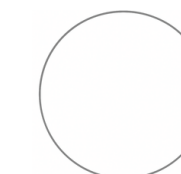
Les oiseaux sont représentés par les cônes de petite taille

-  Éclaireur
-  Suiveur
-  En alerte
-  Épuisé



Le prédateur est représenté par le cône de taille moyenne

-  En chasse
-  Repu



Rayon d'action du cri du prédateur



Les points d'intérêts tel que la nourriture sont représentés par une sphère verte

 Point d'intérêt

 Rayon d'attractivité

Simulation interactive

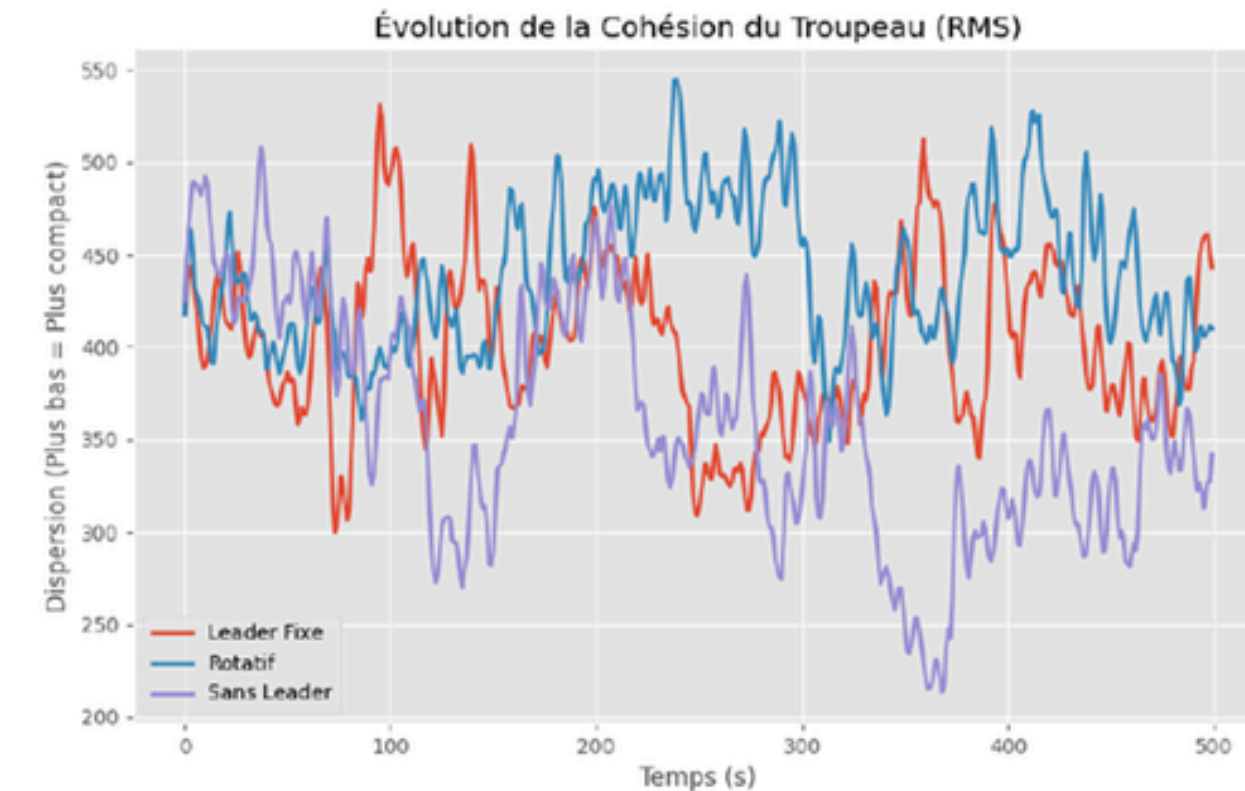


Cohésion & vitesse du groupe

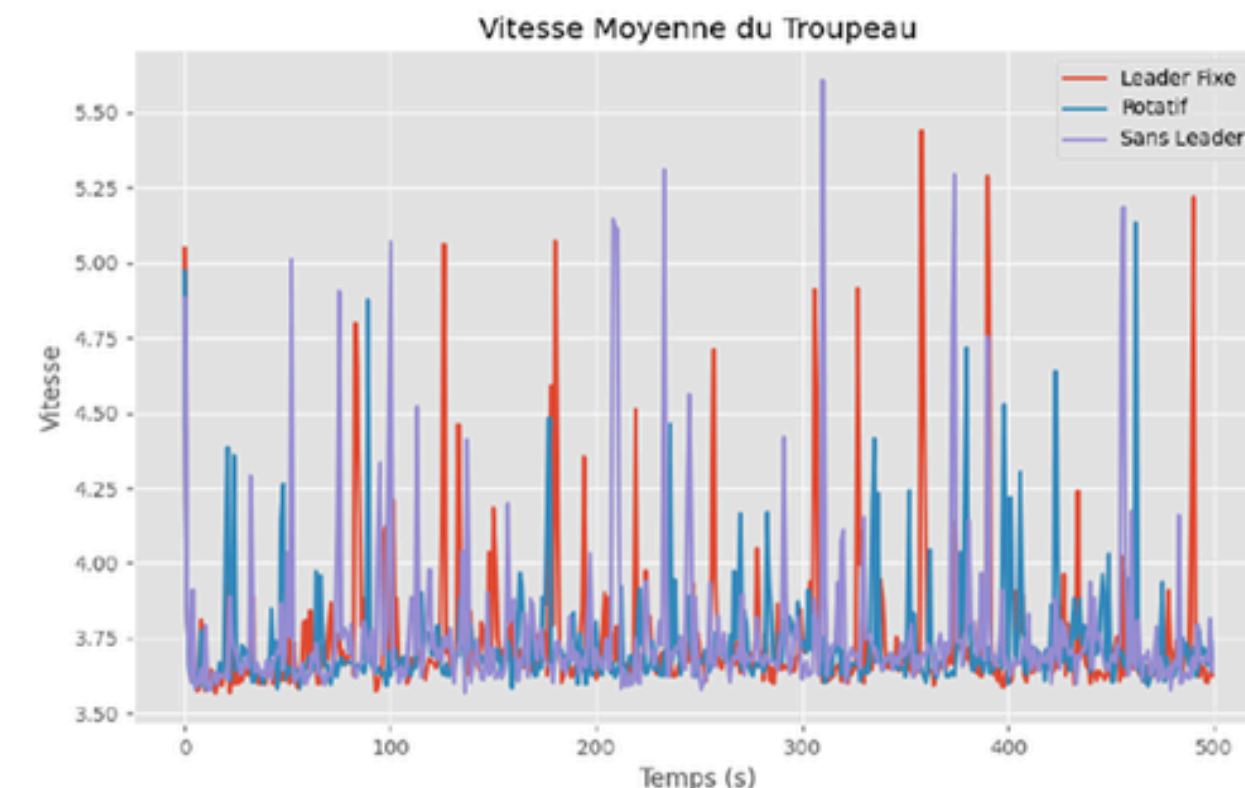
Points clés

- 3 modes comparés : leader fixe • leader rotatif • sans leader
- Meilleure cohésion en mode sans leader (RMS minimal ≈ 358)
- Leader rotatif : cohésion plus « lâche » (≈ 439) → réorganisations fréquentes
- Vitesse moyenne proche entre modes ($\approx 3.7-3.8$) : performance comparable
- Leader fixe : fatigue du leader partiellement compensée par la poussée collective

Lecture : une légère « respiration » de cohésion peut améliorer la réactivité.



(a) Cohésion (Distance RMS)

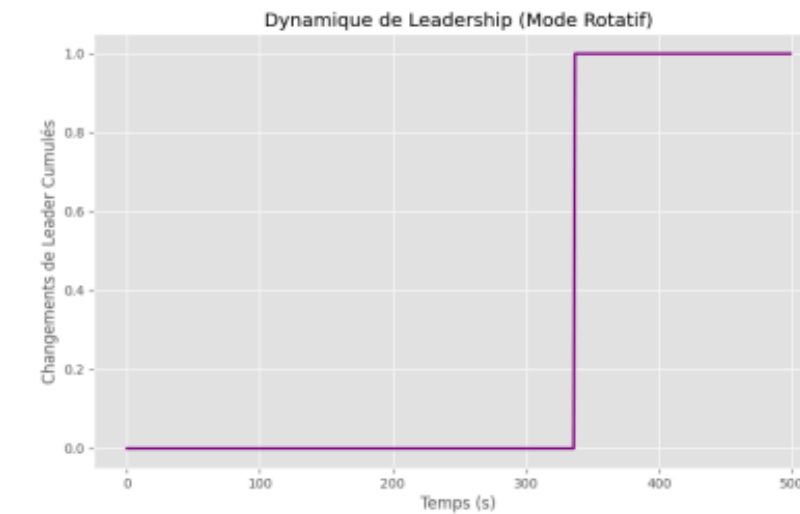


(b) Vitesse Moyenne

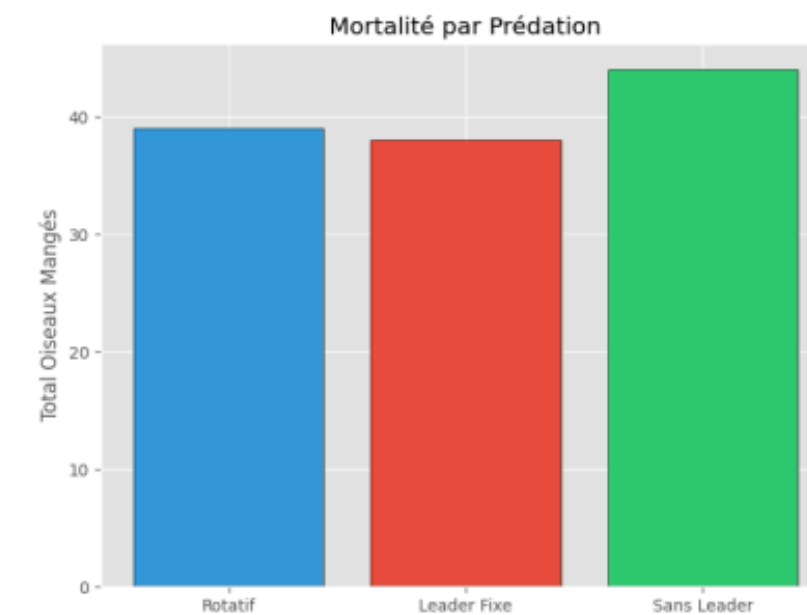
Mortalité & leadership

Points clés

- Mortalité sous prédation : leader fixe (≈ 38) et rotatif (≈ 39) < sans leader (≈ 44)
- Leadership rotatif : changements de rôle déclenchés par un seuil de fatigue (≈ 350 s)
- Sans leader : structure fragile \rightarrow dispersion plus forte \rightarrow prédation accrue
- Conclusion : une hiérarchie améliore la survie ; le rotatif répartit l'effort tout en restant flexible



(a) Fréquence des changements de leader



(b) Total cumulé des décès

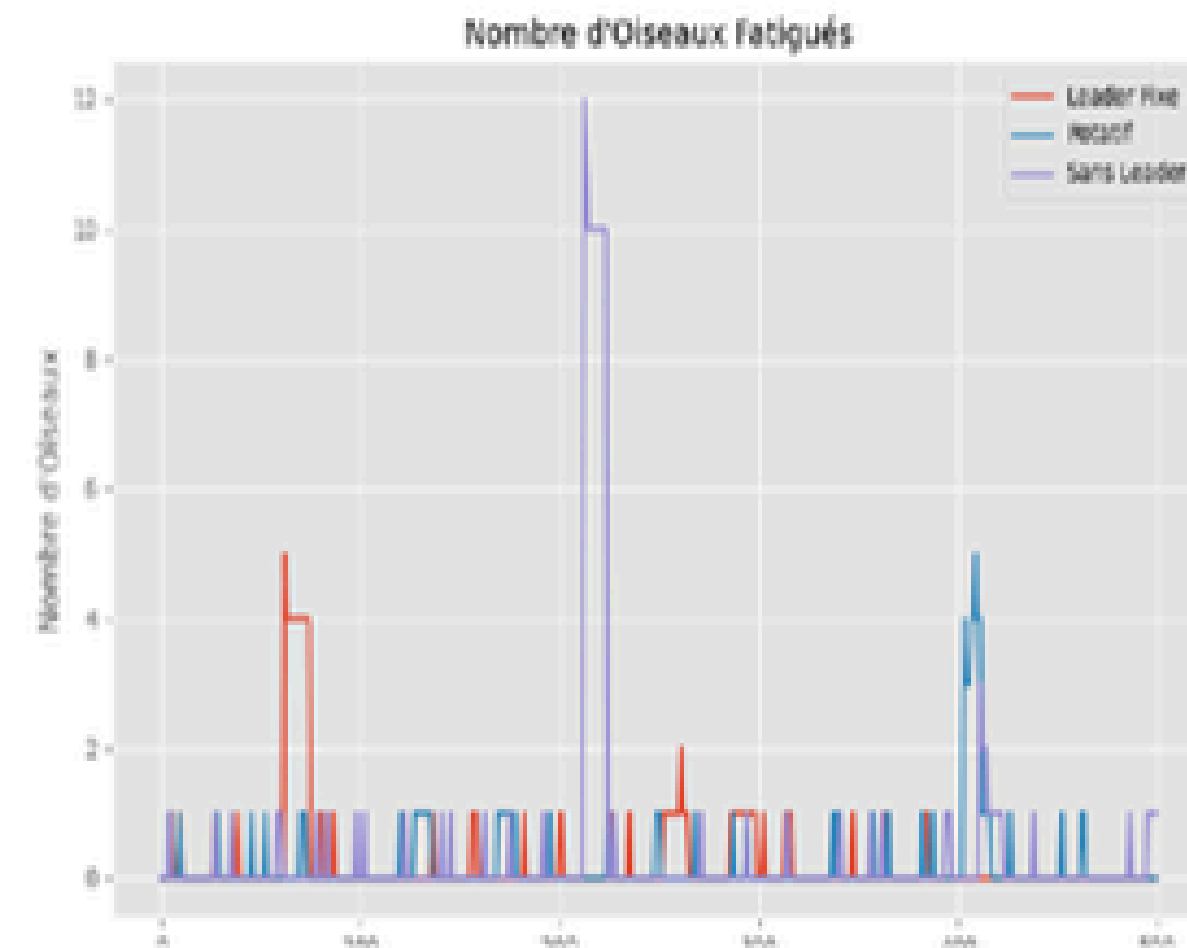
Synthèse — le coût caché du leadership fixe

Même mortalité ≠ même endurance : la stratégie "fixe" s'épuise.

Points clés

- Leader fixe : fatigue qui s'accumule → stratégie non durable
- Leadership rotatif : fatigue répartie → système stable (= 0 épuisement)
- Sans leader : pics d'épuisement plus forts → organisation fragile
- Message : la rotation protège l'endurance du groupe (logique "vol en V")

Nombre d'oiseaux fatigués (au cours du temps)



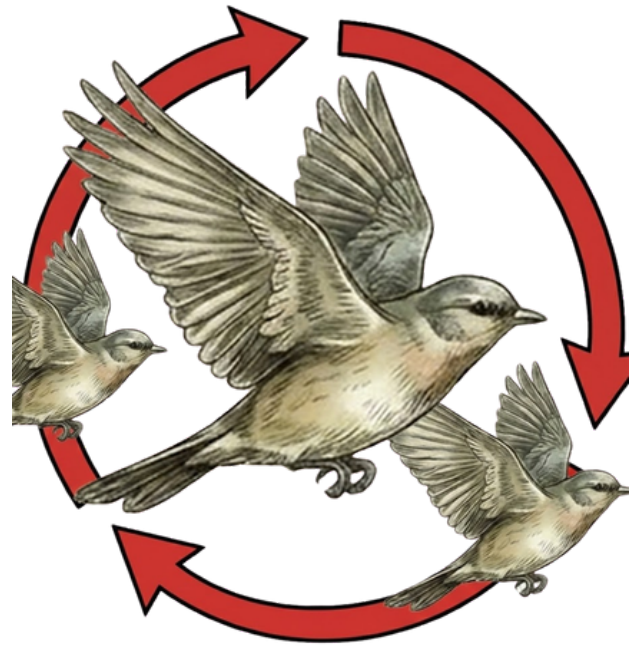
Pour améliorer notre modèle et notamment constater une vraie différence entre les modes de leader fixe et les leader en rotation



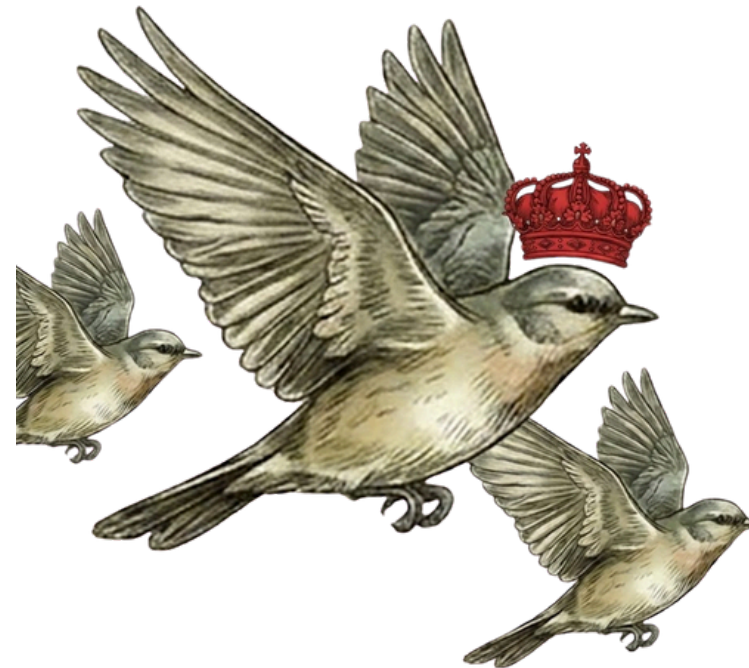
Modification du nombre d'oiseau
500 individus → 5000 individus



Augmentation de la durée de simulation
3 minutes → 24 heures



Rotation du leader



Leader fixe



**Vol désordonné
NON VIABLE**

Contrairement à ce que nous pensions **sur 5 minutes** ces deux modes ne présentent pas de différences significatives.

Cependant sur de **plus longues et réalistes périodes** le leader fixe n'est pas tenable car ce dernier s'épuisera à un moment donné ce que nous vérifierons en lançant la simulations sur une journée.



**Merci pour
votre
attention**