

BOIDS : Proies et prédateurs

L'objectif du projet est de concevoir un programme C++ faisant une simulation multi-agent. La simulation multi-agent consiste à donner des règles simples à un grand nombre d'objets et d'observer des phénomènes émergents. Ici, nous allons nous intéresser à deux types de simulation multi-agents, les BOIDS et les prédateurs, dont les règles sont proches de celles des BOIDS.

Boids est le nom d'un programme informatique de vie artificielle développé par Craig W. Reynolds en 1986 qui simule le comportement d'une nuée d'oiseaux (ou d'un banc de poisson selon l'interprétation qu'on lui donne). Il a marqué l'histoire des sciences en montrant qu'un comportement complexe (le vol coordonné d'une nuée d'oiseaux) pouvait *émerger* de règles simples (chaque oiseau adopte son comportement en fonction de celui de ses voisins immédiats). Aujourd'hui des techniques similaires sont utilisées pour animer automatiquement le comportement des foules dans les jeux vidéo ou les films (le roi Lion, le seigneur des anneaux, ...) ou simuler le comportement d'une foule (simulation multi-agent utilisé, par exemple, pour simuler l'évacuation d'un bâtiment).

Bibliothèque graphique

Vous pouvez utiliser la bibliothèque graphique de votre choix. Une bibliothèque simple d'emploi est SFML (<https://www.sfm1-dev.org>). De toute façon, on n'utilisera que les fonctions basiques de la bibliothèque, à savoir afficher des segments de droite dans certaines couleurs et éventuellement afficher des triangles.

Je vous invite à lire les parties qui concerne dans le tutorial de SFML (<https://www.sfm1-dev.org/tutorials/2.5/>).

La simulation

Le projet consiste à simuler une nuée d'oiseaux nommées proies qui ont tendance à se "regrouper" pour leur survie et des oiseaux nommés prédateurs qui chasseront les proies en suivant de simples règles mais qui feront apparaître une stratégie émergente.

Les oiseaux

Une proie comme un prédateur est décrit par :

1. Sa position (un vecteur dans \mathbb{R}^2)
2. Sa vitesse (un vecteur dans \mathbb{R}^2)
3. La force maximale à laquelle il peut être soumis (un réel)
4. Sa vitesse maximale (un réel)

La position d'un oiseau est repéré sur l'écran par le pixel correspondant. On dessinera un segment de longueur de deux pixels orienté dans le sens de la vitesse pour dessiner un oiseau.

Initiation de la simulation

On donnera tous les paramètres voulus dans un fichier de configuration qui sera lu en début de programme.

Le tableau ci-dessous vous donne les paramètres configurables pour la simulation :

Paramètre	Description	Valeur par défaut
n_b	Nombre de proies	256
n_p	Nombre de prédateurs	8
v_{max}	Vitesse maximale	2
f_{max}	Force maximale	0,05
d_a	Distance alignement	25
d_s	Distance séparation	20
Res	Résistance aggrégation	1

Pour une simulation intéressante, il vaut mieux prendre un nombre de proies bien supérieure au nombre de prédateurs.

De même, on prendra par défaut une vitesse maximale de 2 pour les proies comme pour les prédateurs et une force maximale de 0.05 pouvant être appliqué à un oiseau.

A l'initialisation de la simulation, la position et la vitesse initiale de chaque oiseau est tiré aléatoirement, le vecteur vitesse ayant ses composantes tirées au hasard entre -1, 0 et 1.

Lois de la physique régissant le comportement des oiseaux

À chaque instant, tous les oiseaux sont soumis à la mécanique newtonienne :

- L'accélération a subie à un instant donné est égale à la somme des forces (la masse d'un oiseau est considéré égal à 1);
- Après un intervalle de temps $\delta t = 1$, la nouvelle vitesse est $v_{t+1} = v_t + a$
- De même, la nouvelle position est donnée par : $x_{t+1} = x_t + v_{t+1}$

Note : Il est important de penser à limiter la vitesse de l'oiseau lors de sa mise à jour.

L'écran est considéré ici comme un tore. Un oiseau qui sort de l'écran par un côté arrivera par le coté opposé.

Comportement des proies durant la simulation

Les proies suivent essentiellement les règles introduites par Reynolds pour les Boids, à savoir :

Force d'alignement

Cette force traduit le fait que les oiseaux d'une nuée ont tendance à voler dans la même direction. Cette force est directement liée à la moyenne des vitesses des proies volant dans le voisinage d'une proie :

$$F_a(b) = \frac{1}{n} \sum_{b' \in \mathcal{V}(b)} v(b')$$

où $\mathcal{V}(b)$ est l'ensemble des proies situées à une distance inférieure à une distance d_a qui par défaut vaut 25. Pour garantir l'homogénéité des forces, il est en plus nécessaire de normaliser $f_{al}(b)$ en :

- normalisant le vecteur;
- en le multipliant par la vitesse maximale;
- en lui soustrayant la vitesse de l'oiseau;
- en limitant sa norme à la force maximale que peut subir l'oiseau.

Force de séparation

La force de séparation traduit le fait que deux proies ne peuvent pas être trop proche.

Dans le premier cas, dès que deux proies sont suffisamment proches, elles changent de direction pour éviter la collision. Cette force est déterminée par :

$$F_s(b) = \sum_{b' \in \mathcal{V}'(b)} \frac{1}{d(b, b')} (pos(b) - pos(b'))$$

où b est la proie pour laquelle on calcule la force, $\mathcal{V}'(b)$ est l'ensemble des proies situées à une distance inférieures à d_s valant par défaut 20. n est le nombre de voisins (le cardinal de $\mathcal{V}'(b)$), $pos(b)$ la position de b et $pos(b')$ la position de b' et enfin $d(b, b')$ est la distance entre la proie b et la proie b' .

La force doit être ensuite normalisée de la même façon que la force d'alignement.

Force de cohésion

La force de cohésion pousse une proie à se déplacer vers la position moyenne de ses voisins.

Formellement, cette force est définie par

$$F_c(b) = \left(\left(\frac{1}{n} \sum_{b' \in \mathcal{V}(b)} pos(b') \right) - pos(b) \right) / Res$$

où $\mathcal{V}(b)$ est le même voisinage que pour la force d'alignement et Res est une constante (pouvant être égale à 1) qui traduit une certaine résistance à se mettre en groupe.

Là encore, il faudra normaliser la force de la même manière que les précédentes

Force de dispersion

La force de dispersion est une force traduisant la volonté de la proie d'échapper à son prédateur.

Formellement, cette force est définie par : $F_d(b) = \sum_{p \in \mathcal{V}_p(b)} \frac{1}{d^2(b,p)} (pos(b) - pos(p))$

où $\mathcal{V}_p(b)$ est l'ensemble des prédateurs situés à moins de d_a de distance de la proie.

Il faudra encore là normaliser le vecteur comme pour les forces précédentes.

Force totale

La force totale appliquée à la proie est la somme pondérée des quatres forces calculés :

$$F = F_a + 3 \frac{F_s + F_d}{4} + F_c$$

Comportement des prédateurs

Les prédateurs sont soumis qu'à deux forces : une force d'attraction vers les proies et une force de répulsion envers les autres prédateurs (afin de ne pas s'aggréger et pouvoir avoir un chance d'entourer des proies)

La force d'attraction sera

$$F_a(p) = \sum_{b \in \mathcal{V}(p)} \frac{1}{d^2(b,p)} (pos(b) - pos(p))$$

où $\mathcal{V}(p)$ est l'ensemble des proies se trouvant à une distance inférieure à d_a du prédateur.

La force de répulsion sera :

$$F_r(p) = \sum_{p' \in \mathcal{V}_p(p)} \frac{1}{d^2(p',p)} (pos(p) - pos(p'))$$

Comme pour les proies, il faudra "normaliser" de la même façon les deux proies avant de les sommer.