

Projet d'intelligence artificielle : Simulation de Société d'Agents Coopératifs

Jason CROMBEZ
Marine LAVAUX
Rémi PALANDRI

7 janvier 2013

1 Sujet

Ce projet consiste à définir une société d'agents où chaque agent aura un rôle. Pour les agents humains ce rôle pourra être :

- cultivateur
- éleveur
- chasseur
- porteurs d'eau
- constructeur
- cuisinier
- chef de clan (un seul agent)

Il y aura également des agents animaux qui pourront être soit sauvages soit domestiquables.

Ce clan et ces animaux vivront dans un univers clos, représentable par une île (grille 2D rectangle), constitué d'une plaine dont une partie sera occupée par le clan (village) et d'une jungle.

Cet univers sera soumis à des intempéries (dont on n'est pas obligé de représenter la nature exacte : foudre, éboulements, inondations, ...) et à des antagonismes (chasses d'animaux qui veulent fuir les humains et survivre à leurs attaques, attaques par les animaux des humains qui veulent fuir et survivre eux aussi, ...) dont l'effet ultime sera toujours la diminution (la mort), dans une certaine proportion (simulée par des tirages aléatoires), du nombre d'agents (humains ou animaux). Les agents auront néanmoins la faculté de se reproduire si certaines conditions sont satisfaites (conditions de ressources alimentaires, ...).

Les agents animaux chercheront essentiellement à se nourrir et à se reproduire. Leur capacité à se nourrir pourra être entravée par le manque d'étendue dont ils peuvent profiter (surface de l'île moins celle du village et des cultures). Leur capacité à se reproduire pourra dépendre de leur capacité à se nourrir et de leur nombre (plus ils sont nombreux plus ils peuvent se rencontrer).

Les objets abris peuvent servir à faire diminuer les probabilités de mort des agents humains, les huttes de stockage servent de réserves et permettent de survivre même si le temps ne permet ni les récoltes, ni la chasse. Ces objets peuvent être détériorés et devront être reconstruits ou réparés.

Toutes les activités prennent du temps et de l'énergie vitale à tous les agents (humains ou animaux). Ils doivent tous se nourrir et se reposer. Ces caractéristiques peuvent aussi être simulées par différentes jauges.

L'agent chef de clan est un peu particulier dans la mesure où il pourrait décider de certaines affectations comme décider combien d'agents de tel type il faut consacrer à la tâche qui leur incombe, décider quels tâches accomplir, combien de temps y consacrer, etc

(voir quels sont les paramètres utiles et sur lesquels le chef de clan pourra jouer).

L'unité de surface cultivable rapporte une certaine quantité de nourriture végétale pour autant qu'elle soit cultivée pendant un certain temps par un certain nombre d'agents cultivateurs. Il en est de même concernant l'élevage des animaux domestiques.

2 Analyse du sujet

2.1 L'univers

L'univers est clos, il s'agit d'une île. Celle-ci est générée automatiquement à l'aide de Perlin Noise, c'est également le cas en ce qui concerne les différents types de terrains (forêts, lacs, ...) les coefficients peuvent être ajustés afin de créer des cartes ayant leur spécificités propres.

La carte est divisée en cellules unitaires, celles-ci contiennent les informations relatives au terrain, aux agents et bâtiments présents sur la cellule. Ces informations sont réparties en deux catégories, celles qui sont mutuellement exclusives (une cellule ne peut être que d'un seul type de terrain) et celles s'ajoutent les unes aux autres (une cellule peut à la fois contenir un bâtiment, plusieurs agents animaux et plusieurs agents humains).

2.2 Les agents

Qu'ils s'agissent des animaux ou des humains, la modélisation de ceux-ci à une racine commune. Il s'agit de la classe *Etre* dans le fichier *etre.py*. Cette représentation dont hérite tous les types d'agents regroupe ce qui peut être factorisé. Il s'agit pour une part de tout ce qui concerne la gestion des entités en lien avec l'interface et d'autre part d'attributs non spécifiques à un type d'agent particulier (le genre par exemple, qui sera utilisé dans le cadre de la reproduction ou bien l'état de l'agent, vivant ou mort).

Tous les types d'agents disposeront d'une fonction *run* qui sera jouée à chaque tour de la simulation, certains comportement tels que la mort si la jauge de nourriture tombe à zéro seront également universels.

2.2.1 Les animaux

Il existe deux types d'agents animaux, les domesticables et les sauvages. Sur l'interface graphique ceux-ci sont représentés respectivement par des moutons et des loups.

Les éléments factorisables à ces deux types d'agents sont regroupés au sein de la classe *Animal* se trouvant dans le fichier *Gaia.py* et héritant de la classe *Etre*. Les classes *Sheep* et *Wolf*, également présentes dans *Gaia.py* héritent de cette classe *Animal*. Ce type d'héritage multiple étant possible en *Python*. Ce que l'on factorise au niveau de la classe *Animal* consiste essentiellement en des comportements (méthodes). Il s'agit des fonctions concernant la reproduction.

Comme précisé dans le sujet, la reproduction dépend de la possibilité de deux animaux de sexe opposé à se rencontrer ainsi que du fait que ceux-ci sont suffisamment bien alimentés. Les fonctions nécessaires sont donc de rendre possible la vérification de ces conditions (en terme de genre et d'état de la jauge de nourriture), ainsi que la recherche d'un partenaire pour se reproduire si ces conditions sont vérifiées, le fait de le rejoindre ainsi que l'enfantement.

Dans le cadre de la modélisation de ces comportements nous avons fait certaines hypothèses, celles-ci pouvaient être simplificatrices ou bien avoir un caractère arbitraire. L'hypothèse simplificatrice que nous avons faite est de considérer que le temps de gestation est nul, de plus l'individu qui voit le jour est instantanément considéré comme étant un adulte. Les choix arbitraires que nous avons faits se situent au niveau de tous les paramètres qu'il nous a fallu fixer, il s'agit par exemple du champ de vision des animaux, le rayon de ce champ de vision est une constante arbitraire et il en va de même pour le niveau de la jauge de nourriture qui définit si l'animal a faim ou bien est disposé à se reproduire.

- Les moutons

Les moutons se nourrissent d'herbe, ils peuvent donc décider de se nourrir quelle que soit la nature du terrain sur lequel ils se trouvent. Ils n'ont donc pas besoin de chercher leur nourriture.

La fonction *run* du mouton dépend essentiellement de l'état de sa jauge de nourriture si il est assez alimenté il cherchera à se reproduire si ce n'est pas le cas soit (cette condition est déterminée par le hasard mais pondérée par la jauge de nourriture) il se mettra à errer ou se nourrira (la probabilité qu'il se nourrisse sera d'autant plus importante que la jauge de nourriture sera basse).

- Les loups