

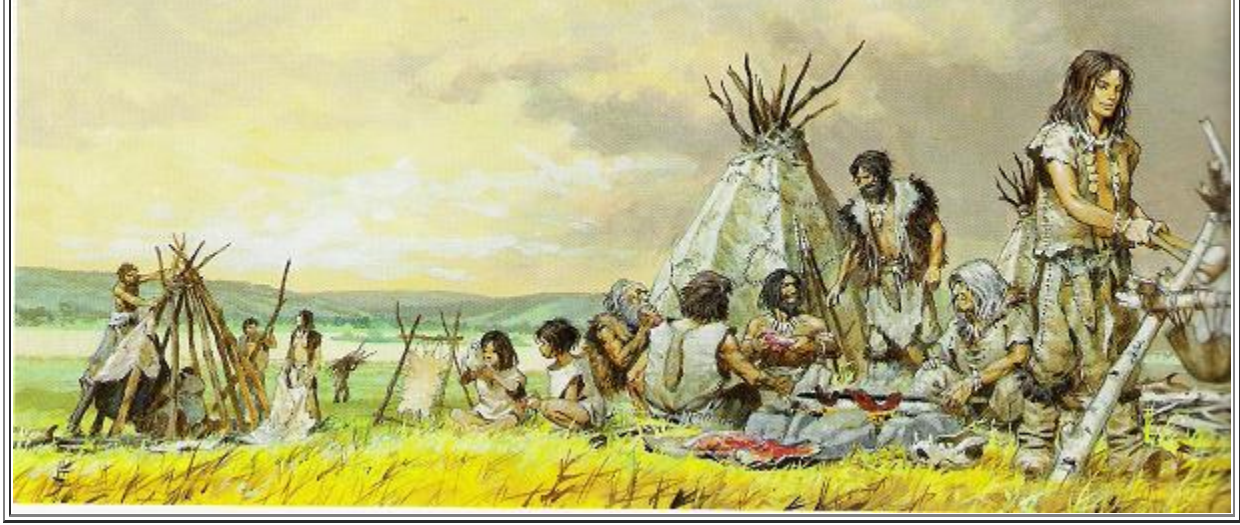
Programmation orientée agents

(HMIN108)

Resp: Jacques Ferber,

TP 4 - Architecture à base d'états finis  
Les chasseurs-cueilleurs

(C) 2016-2018



On cherche à mettre en évidence des architectures d'agent permettant de prendre en compte le processus de décision des agents. Pour cela on va essayer de modéliser le comportement de chasseurs-cueilleurs, en simplifiant à l'extrême leur comportement.

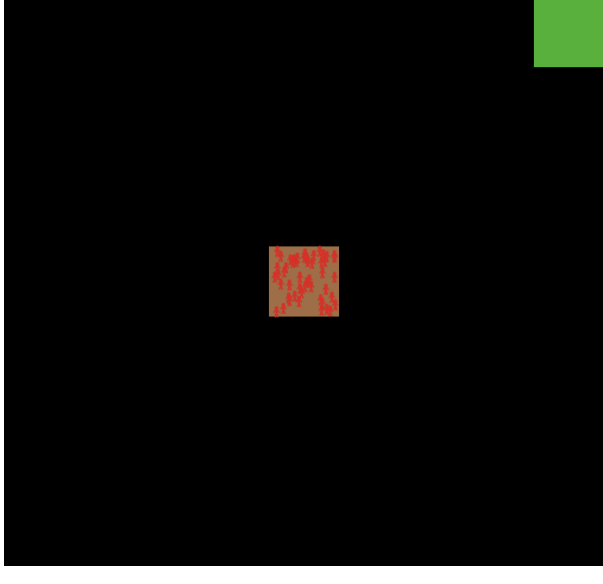
Les chasseurs cueilleurs doivent faire face à un problème important: est il plus intéressant de chasser ou de cueillir? Et si l'on chasse, dans quel coin faut il aller? Et si l'on cueille, quels sont les bons endroits?

On va aborder cette question de la prise de décision au travers d'une analyse progressive du comportement de ces chasseurs cueilleurs.

1. Comportement simple

a) Village et une zone de cueillette

On crée un village au centre du monde. Le village, représenté sous la forme d'un carré marron, a une taille d'environ 20 patches de côté.. On crée aussi une zone de cueillette en haut à droite, comme le montre la figure suivante:



b) Les animaux

On suppose qu'il existe des animaux sauvages qui évoluent aléatoirement dans le monde. On suppose de plus que ces animaux, sont capables de se reproduire avec un taux de reproduction allant de 0.005 à 0.05 à chaque tour (c'est à dire que chaque animal a une probabilité  $p$  avec  $0.5\% < p < 5\%$  d'engendrer un nouvel animal à chaque tour).

On suppose de plus que le nombre max d'animaux est fixé à 400. (comme tous les paramètres, cela peut faire l'objet de sliders).

Vitesse des animaux : 0.2

**Note:** vous pouvez ajouter d'autres informations, le fait que les animaux broutent de l'herbe par exemple (en reprenant des modèles des TP précédents), mais ce n'est pas fondamental ici..

2. Les chasseurs-cueilleurs

On suppose qu'il existe un ensemble de chasseurs (le nombre des chasseurs est un paramètre global de la simulation, à mettre sous la forme d'un slider donc).

Ces chasseurs peuvent soit essayer de chasser des animaux sauvages, soit aller cueillir des plantes sauvages. Dans un premier temps on suppose que ce choix est déterminé par une variable `cueilleur?` qui indique si le chasseur a plutôt un comportement de chasseur ou plutôt un comportement de cueilleur..

Un chasseur cherche un animal, le rattrape à la chasse, le tue, et revient au village. Quand il revient au village, une certaine énergie devient disponible correspondant à la valeur du paramètre (`energie-chasse`) entre 1 et 100. Cette énergie est globale à la simulation: elle correspond à l'énergie que récupère le village lorsqu'un chasseur rapporte un trophée à la chasse.

Un cueilleur va dans le coin des plantes et rapporte une certaine quantité d'énergie (paramètre `energie-cueillette`) entre 1 et 100 aussi, en revenant au village.

Le taux de cueilleur peut être défini par un paramètre global qui détermine combien il y a de cueilleurs parmi les chasseurs-cueilleurs.

**Important:** on représentera le comportement d'un chasseur sous la forme d'une machine à état fini (comme on l'a vu dans le cours). De ce fait le comportement de base de la procédure `to go` est le suivant:

```
to go
  ask animaux [ sheep-live ]
  ask chasseurs-cueilleurs [ run ctask]
  ;i ...

end
```

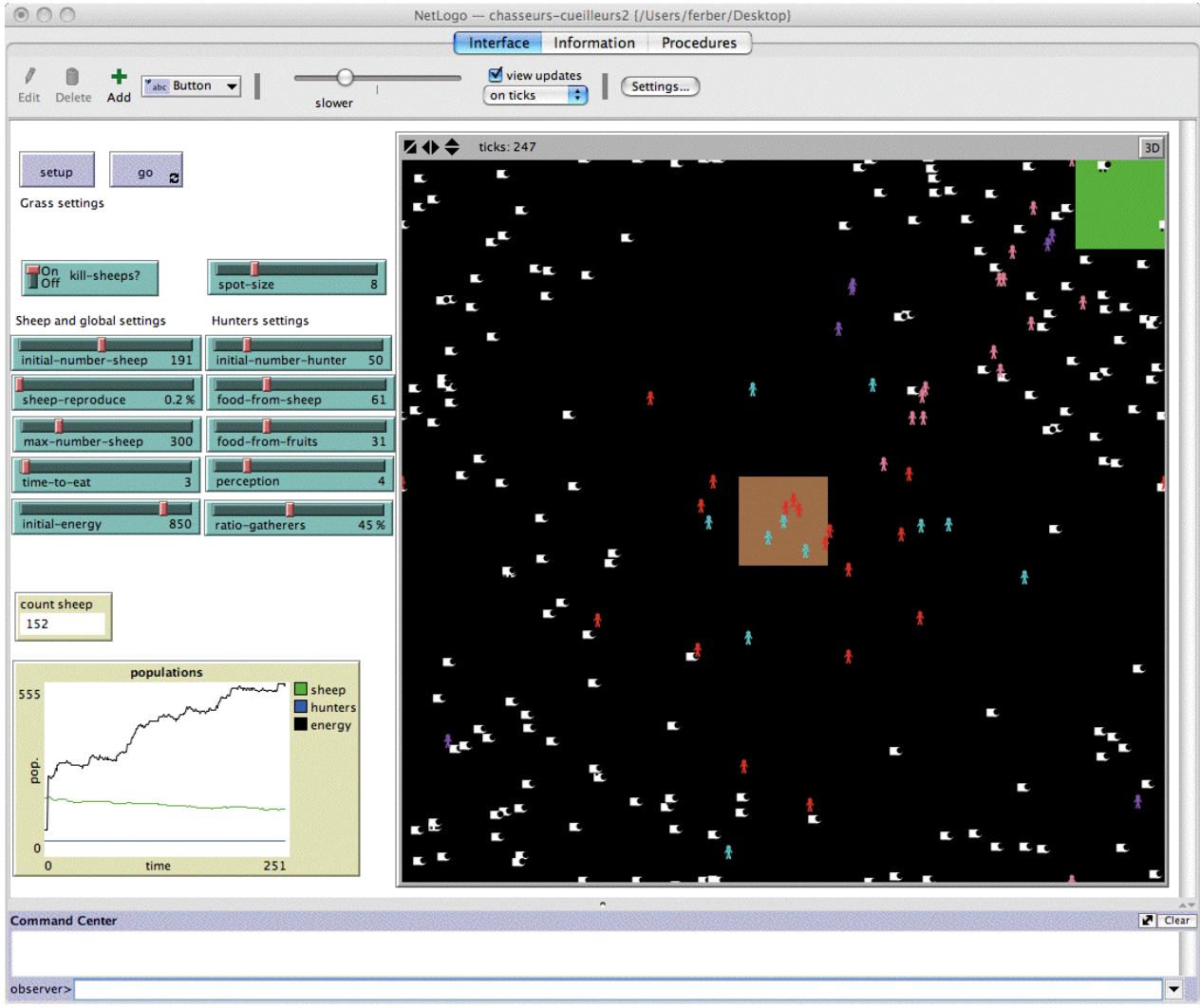
Le comportement d'un chasseur-cueilleur comprend cinq activités principale: décider de ce qu'il va faire, aller chasser, ramener le produit de la chasse au village, aller cueillir des plantes, ramener le produit de la cueillette au village.

**NOTE: Il est important que vous programmez en implémentant une architecture à base de machine à état fini.**

Les paramètres globaux de la simulation:

population-initiale, nombre-init-animaux, taux-reproduction-animaux (entre 0.5% et 5%), nombre maximaux d'animaux (autour de 400), energie-initiale (du village), energie-de-chasse (environ 60), energie-de-plante (environ 20), taux-cueilleurs (entre 0 et 100), consommation de l'energie globale de chaque chasseur-cueilleur À chaque tour (entre 0 et 5, par pas de 0.1 au max).

La figure suivante montre un "snapshot" d'une telle simulation (attention, les paramètres sont en anglais, car c'est une habitude pour moi de toujours écrire les programmes en anglais).



3. La prise de décision

Pour l'instant la prise de décision de devenir chasseur ou cueilleur est déterminé à la création par le rapport cueilleur/population totale. On suppose d'autre part que les humains consomment de l'énergie à chaque tour, chaque humain consommant une certaine quantité d'énergie globale du village.

Maintenant on va essayer que chaque agent prenne lui-même sa décision de savoir s'il va chasser ou aller cueillir. Ce comportement suit souvent des règles complexes qui prennent en compte non seulement des choix individuels, mais aussi les connaissances du milieu et surtout la culture de la société. Mais dans notre cas, on va juste se baser sur des choix élémentaires.

**Le choix est celui-ci:** si le taux de succès à la chasse est trop faible, alors la probabilité augmente pour chacun de choisir le comportement de cueillette. Ce taux de succès peut être déterminé comme la quantité d'énergie ramenée de la chasse par rapport au nombre de chasseurs.

Si ce taux est faible, alors l'individu aura tendance à choisir le comportement de cueilleur. Au contraire un taux fort correspond à un grand succès dans la chasse et l'individu aura tendance à aller tenter sa chance en tant que chasseur.