
Systeme multi - agents

Les 2 premiers TPs (du 18/10/10 et du 02/11/10) consisteront à traiter ce mini-projet. Certains éléments serviront dans les 2 TPs suivants.

Mini-Projet d'Intelligence Collective

0. Définir un espace 2D (**E2D**) sous forme d'une grille de dimension 50 cases x 50 cases.
1. Simuler la marche aléatoire d'un agent partant d'un point quelconque de cette grille en se dirigeant vers l'un des 8 voisins (laisser une trace de toutes les cases parcourues pour 1000 pas d'itération). (1)
2. Disposer, aléatoirement sur **E2D**, 650 ressources de 3 types différents (notés r1, r2, r3) et équiprobables. (2)
3. Définir la fonction sensorielle "capteur" de cet agent qui renvoie le nombre de ressources de type rx (pour chaque x=1,2,3) présentes parmi les 8 voisines de la position où il se trouve. (3)
4. Procéder à une marche aléatoire de cet agent et tracer les 3 courbes histogrammes des ressources rencontrées. (4)
5. Définir 3 comportements pour cet agent :
 1. marche aléatoire de l'agent vers une case voisine de là où il se trouve (déjà vu précédemment).

2. prise par l'agent, là où il se trouve, d'une ressource en proportion inverse à sa fréquence parmi les 8 voisins : $\text{prob} = 0$ si $\text{freq} = 8$, $\text{prob} = 1$ si $\text{freq} = 0$ (à condition que l'agent n'ait pas déjà pris quelque chose et que la ressource se trouve là).
3. pose par l'agent, là où il se trouve, d'une ressource proportionnellement à sa fréquence parmi les 8 voisins : $\text{prob} = 1$ si $\text{freq} = 8$, $\text{prob} = 0$ si $\text{freq} = 0$ (à condition que l'agent ait une ressource "en main" et que la place soit libre). (5)
6. A chaque temps de cycle un seul de ces 3 comportements est effectuable $\{\text{wa} = \text{walk}, \text{tx} = \text{take}(\text{rx}), \text{px} = \text{put}(\text{rx})\}$. Produire graphiquement la marche aléatoire de l'agent, la liste des comportements enchaînés lors de la marche de cet agent et la liste des types de ressources rencontrées lors de cette marche (sur une centaine de cycles environ). (6)
7. Définir suffisamment d'agents (tous caractérisés par les 3 comportements décrits en (5)) et les lancer dans des marches à partir de positions initiales aléatoires dans l'espace **E2D** rempli de ressources, de sorte à faire apparaître des tas de ressources du même type. (7)
8. Définir un champ d'attraction central de portée limitée dans **E2D** dont les cases alentour sont d'autant plus attractives qu'elles sont proches du centre. Le rôle de ce champ est de modifier la probabilité de se diriger vers 1 des 8 voisins d'une

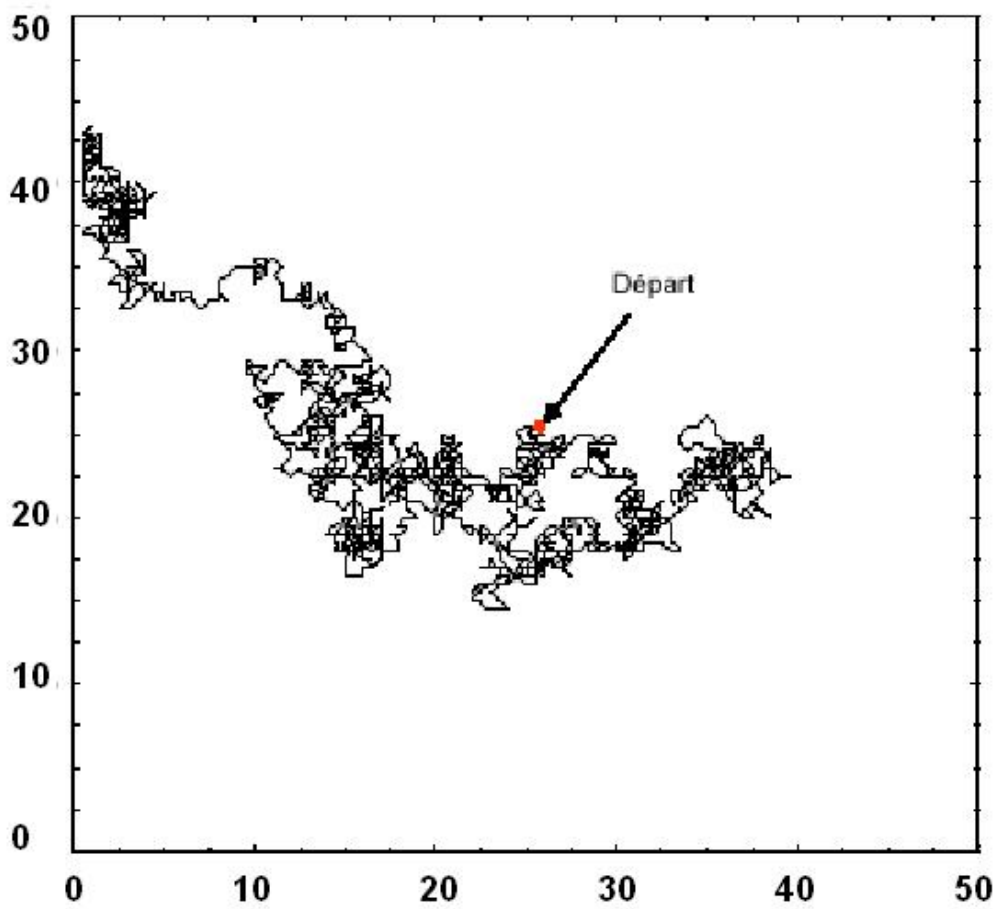
case donnée (jusqu'ici walk s'effectuait vers 1 des 8 voisins tous équiprobables). (8)

9. Procéder à une marche aléatoire pour un agent dans ce nouvel **E2D** (sans ressource mais avec champ central attractif) à partir d'une case initiale pas trop éloignée de ce champ central. Réitérer de telles marches en partant de cette même case initiale. Laisser une trace de toutes les cases parcourues au terme de toutes ces marches (effectuer une ou deux dizaines de marches). (9)
10. Rendre le champ (non plus central et statique) mais dynamique en définissant un comportement supplémentaire de l'agent :
 1. marquer (d'une unité de marquage) chaque case empruntée par l'agent dans une marche aléatoire. (10)
11. Procéder à un lâcher d'un certain nombre d'agents, aléatoirement positionnés initialement dans l'espace **E2D** rempli de ressources. Produire l'espace final, résultant d'un nombre suffisant d'itérations, pour voir apparaître des tas et des chemins privilégiés. (11)
12. On tentera de rendre les unités de marquages présentes en chaque case "évaporables". On procèdera alors à un lâcher comme en (11). Produire l'espace final. (12)
13. On fera disparaître progressivement les ressources au cours du temps (moins vite quand elles sont dans un tas qu'en dehors) et apparaître aléatoirement d'autres

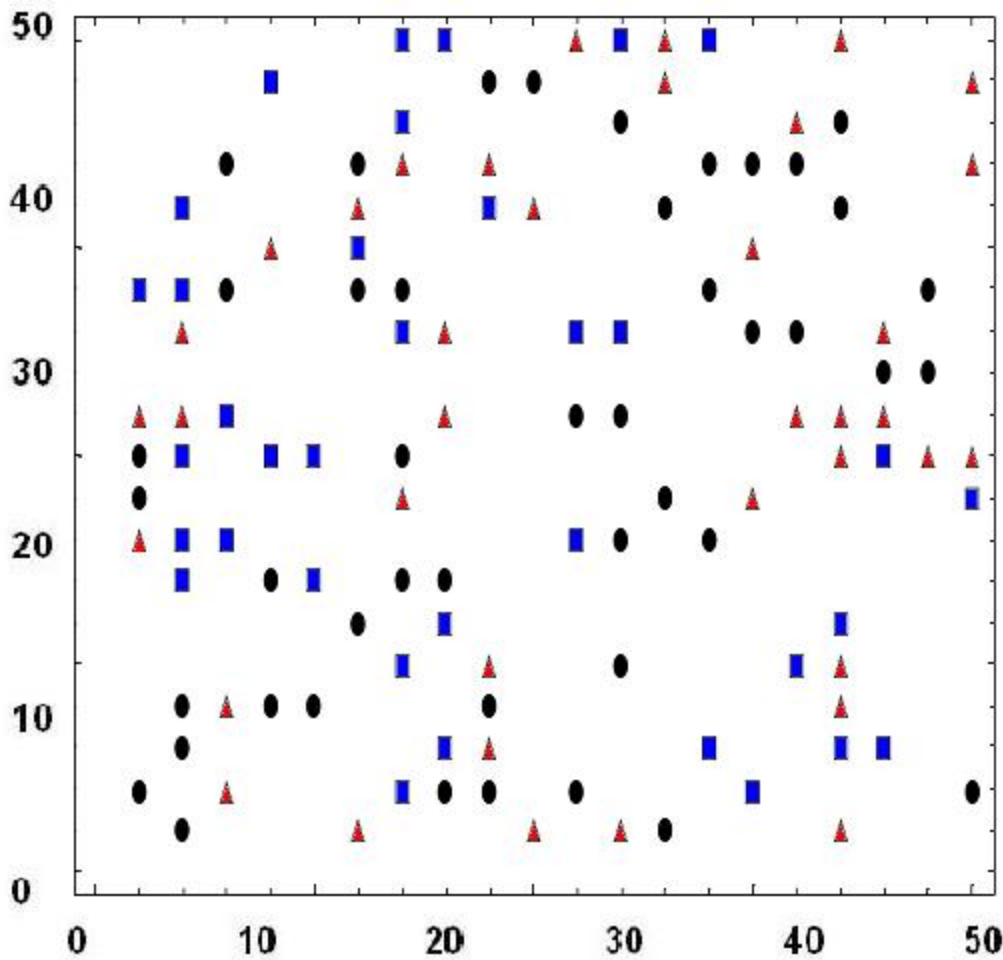
ressources (le même nombre ou pas, mais en dehors des tas).

14. On procèdera à l'affichage des agents (et pas seulement des traces de leurs déplacements) sur la grille représentant **E2D** et à la gestion en temps réel (à chaque instant du cycle) de leurs positions (ie: les voir se déplacer, cela nécessitera peut-être l'utilisation de boucles de temporisation pour la commodité visuelle de l'utilisateur).
15. On construira une interface d'utilisation conviviale de définition, de paramétrage, d'exécution, d'interruption et d'affichage des résultats de cette application.
16. On procédera au passage de la dimension $\text{dim}(\mathbf{E2D}) = 50 \times 50$ à $\text{dim}(\mathbf{E2D}) = 1000 \times 1000$ (par exemple). On décrira les changements nécessaires opérés sur le programme afin de résoudre les problèmes de calcul temps réel et d'affichage (réduction / zoom de grille).
17. On ajoutera toute autre fonctionnalité à cette plateforme susceptible d'intéresser l'utilisateur, par exemple (non contractuel) :
 1. possibilité de « suivre » un agent choisi par l'utilisateur, le voir se déplacer, prendre une ressource, (éventuellement le voir la transporter), la poser, repartir, ...
 2. possibilité de définir un niveau de fatigue des agents, qui nécessiterait de se rapprocher d'un tas pour « manger » avant de pouvoir repartir, ...

(1) par exemple produire ce genre de graphique



(2) par exemple produire ce genre de graphique



(3) la fonction `capteur(a,i,j,x)` renverra le nombre de ressources de type x présentes parmi les 8 voisins de la case $\langle i,j \rangle$ où se trouve l'agent a .

(4) on choisira de représenter, pour chaque type de ressource, le nombre de déplacements effectués entre deux découvertes successives d'un même type de ressource

(5) bien commenter cette partie de code

la probabilité de prise par l'agent a en $\langle i,j \rangle$ de la ressource rx est $t_prob(a,i,j,x) = 1 - n/8$ où $n = \text{capteur}(a,i,j,x)$. on suggère de tirer aléatoirement un nombre N entre 0 et 1, si $N < t_prob(a,i,j,x)$ l'agent ne procède pas à la prise de la ressource sinon il la prend.

la probabilité de pose par l'agent a en $\langle i,j \rangle$ de la ressource rx est $p_prob(a,i,j,x) = n/8$ où $n = \text{capteur}(a,i,j,x)$. on suggère de tirer aléatoirement un nombre N entre 0 et 1, si $N < p_prob(a,i,j,x)$ l'agent ne procède pas à la pose de la ressource sinon il la pose.

(6) la partie graphique de la marche aléatoire sera du même genre qu'en (1), la liste

des comportements enchaînés lors de cette marche sera du genre

"wa:wa:wa:wa:t1:wa:wa:wa:wa:wa:wa:p1:wa:wa:p2:" et la liste des ressources rencontrées sera du genre "--:r3:--:--:r1:r1:--:r2:--:--:r1:--:" de sorte qu'on pourra les superposer afin de constater les décisions qui auront été prises par l'agent :

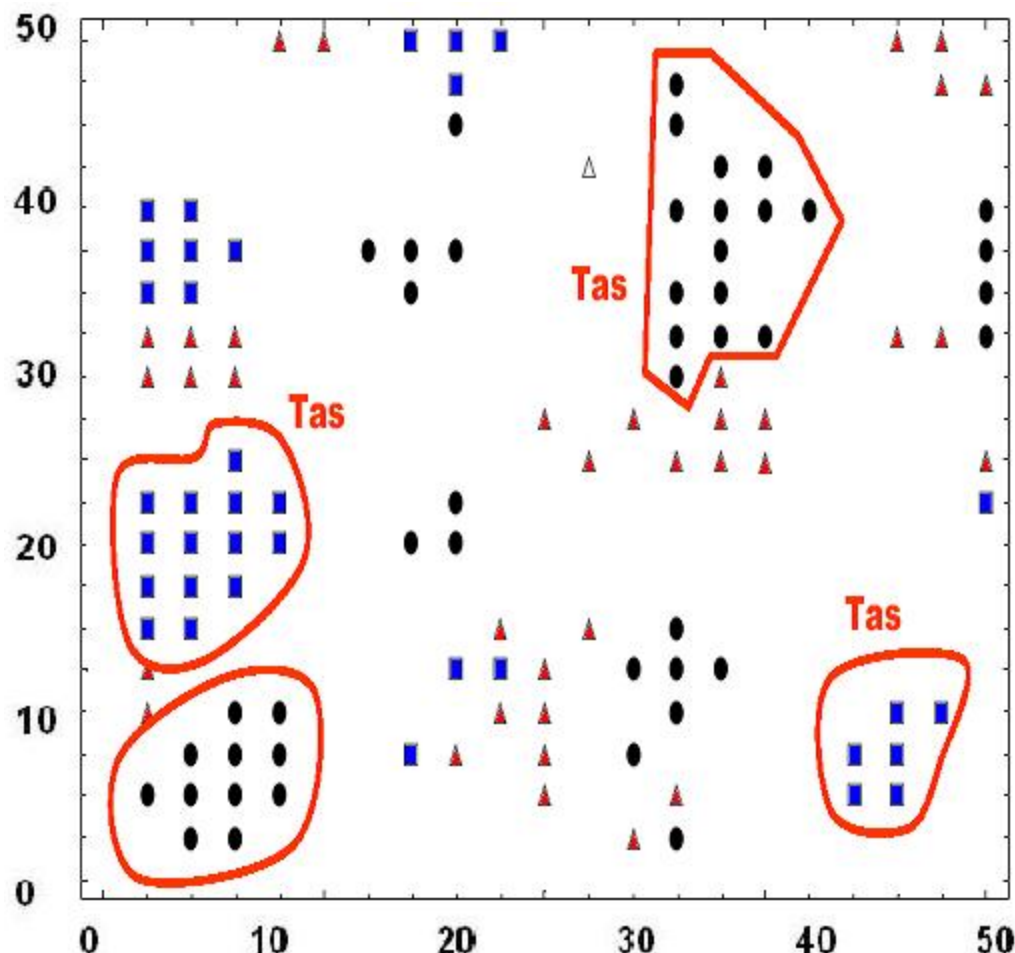
"--:r3:--:--:r1:--:r1:--:r2:--:--:r1:--:" l'agent se trouve successivement sur une case où il y a : rien puis r3 puis 2 fois rien puis r1 puis rien puis etc. (ATTENTION)

"wa:wa:wa:wa:t1:wa:wa:wa:wa:wa:wa:wa:p1:" en chacune de ces positions il décide de faire : wa puis wa (il néglige r3) puis 2 fois wa puis t1 (prise de r1) puis wa puis etc.

(ATTENTION) il faut noter qu'après 4 actions wa l'agent se retrouve sur une case contenant r1, il décide de prendre r1 (5° action = t1) il se retrouve donc (sans avoir bougé) sur la même case où il n'y a plus rien, il décide alors de faire wa (6° action = wa). autrement dit la suite "--:r3:--:--:r1:--:r1:--:r2:--:--:r1:--:" ne représente pas forcément des cases distinctes. un entrelacement de ces 2 suites indiquerait l'état des cases entre chaque transition (matérialisée par l'action en question :

"--:wa:r3:wa:--:wa:--:wa:r1:t1:--:wa:r1:wa:--:wa:r2:wa:--:wa:--:wa:r1:wa:--:wa:p1:". on pourra écrire cette fonction d'entrelacement.

(7) par exemple produire ce genre de graphique (la trace rouge matérialisant les Tas a été faite à la main, donc ici de façon subjective)



(8) on pourra définir l'action sur la position $\langle i,j \rangle$ de ce champ central d'origine $\langle i_0,j_0 \rangle$ à partir de la distance euclidienne $d(i_0,j_0,i,j) = [(i_0 - i)^2 + (j_0 - j)^2]^{1/2}$ entre $\langle i_0,j_0 \rangle$ et $\langle i,j \rangle$, de sorte que, hors champ central, la probabilité naturelle $1/8$ d'être choisie pour une case $\langle i,j \rangle$ sera augmentée d'un facteur $\mu(i,j)$ si cette case est dans ce champ.

ce facteur sera calculé selon une fonction affine décroissante de $d(i_0,j_0,i,j) \in [0,20]$ et nulle pour $d > 20$, à valeurs dans $[0,m]$.

la portée du champ est donc limitée à une distance 20 (par exemple).

la valeur du paramètre m sera à choisir au mieux.

il vient donc par exemple que $\mu(i,j) = \max(0, m \cdot (1 - d(i_0,j_0,i,j)/20))$ et ainsi chaque case $\langle i,j \rangle$ voisine de la case courante $\langle i_c,j_c \rangle$ sera pondérée par $1/8 + \mu(i,j)$.

il faudra normaliser ces pondérations de sorte que la somme des poids des 8 cases voisines de $\langle i_c,j_c \rangle$ soit égale à 1.

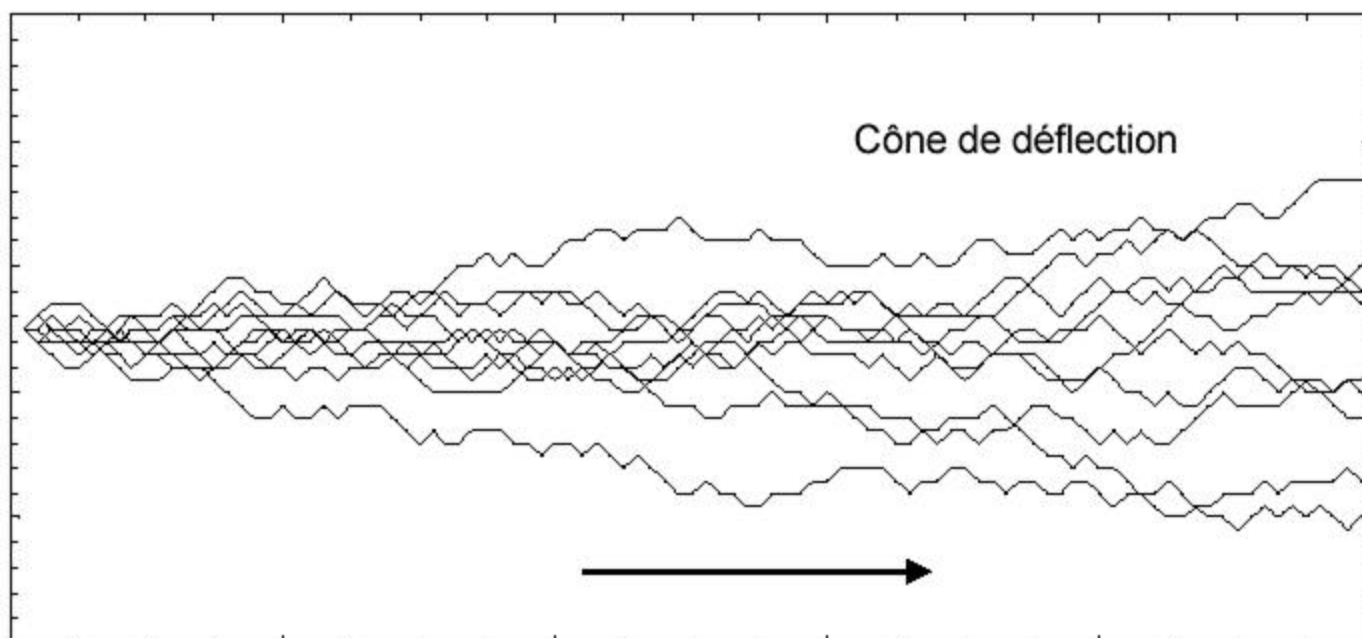
autrement dit la probabilité pour chaque case $\langle i,j \rangle$ vaudra $\text{prob}(i,j) = [1/8 + \mu(i,j)] / [1 + \sum \mu(i,j)]$

pour déterminer à partir de la case $\langle i_c,j_c \rangle$ vers quelle case $\langle i,j \rangle$ on se déplace, on suggère de partitionner l'intervalle $[0,1]$ en 8 sous-intervalles de longueurs $\text{prob}(i,j)$ de sorte qu'un tirage aléatoire d'un nombre dans $[0,1]$ tombera dans un de ces sous-intervalles avec une fréquence proportionnelle à $\text{prob}(i,j)$ et désignera ainsi la case $\langle i,j \rangle$ choisie.

(9) on demande de produire un graphique des marches de l'agent du genre de celui-ci, mais pour l'espace **E2D**



Dans un champ (tropical) attractif prédisposant l'agent à aller de gauche à droite (des zones claires vers les zones foncées) tout en le laissant libre d'aller vers le haut ou vers le bas, on constate que plusieurs marches du même agent produisent une sorte de zone conique incluant tous ses déplacements (cône de déflexion).



(10) bien commenter cette partie de code relative au nouveau comportement de l'agent. penser à modéliser l'espace de sorte que chaque case $\langle i,j \rangle$ de **ET2D** puisse être affectée du nombre d'unités de marquage présentes en $\langle i,j \rangle$ à l'instant t . on écrira une fonction $unites(i,j,t)$ dédiée à ce calcul.

(11) cette fois, à chaque instant du cycle, l'agent dépose une unité de marquage sur la case où il se trouve ET exécute l'une des 3 actions définies en (5). toutes ces unités de marquage se cumulent en chaque case (quel que soit l'agent qui l'y a déposée). on

cherche à produire une image du genre de celle demandée en (7) avec en plus des couleurs (en niveau de gris) sur chaque case matérialisant la quantité d'unités de marquage qui y sont déposées (d'autant plus foncé que le nombre d'unités est élevé).

(12) on pourra altérer la quantité $unites(i,j,t)$ en la multipliant par un facteur d'atténuation $k(t)$