

Une sélection d'action prenant en compte opportuniste et personnalité*

Tony Dujardin, Philippe Mathieu, Jean-Christophe Routier

Équipe SMAC
Laboratoire d'Informatique Fondamentale de Lille
Université des Sciences et Technologies de Lille
59655 Villeneuve d'Ascq cédex
{Dujardin|Mathieu|Routier}@lifl.fr
<http://www.lifl.fr/SMAC>.

Résumé : Nous proposons un mécanisme de sélection d'action pour systèmes multi-agent situés et cognitifs autorisant une grande variété de comportements. Notre sélection d'action permet la gestion de plusieurs buts avec des priorités dynamiques, la possibilité pour l'agent de se détourner temporairement d'un but principal afin de réaliser un but secondaire valorisé par l'environnement et oriente le choix de l'action à exécuter en fonction des préférences de l'agent. Pour cela notre mécanisme de sélection d'action prend en compte trois facteurs que sont les buts à priorité dynamique, l'opportunisme et la personnalité de l'agent.

Mots-clés : Sélection d'action, Agent cognitif, Comportement

1 Introduction

La simulation de comportements cognitivement plausibles (Devigne *et al.*, 2005a) posent plusieurs problématiques, parmi lesquelles se trouvent la représentation de connaissances, les mécanismes de planification proposant des résolutions à l'achèvement des buts de l'agent (sous forme de plans construits à partir des connaissances de l'agent) et également la conception de mécanisme de sélection d'action (qui choisissent l'action que l'agent va exécuter parmi les actions exécutables présentes dans les plans de l'agent). C'est dans ce dernier point que se situe notre proposition. Un mécanisme de sélection d'action permet par le choix de l'action à exécuter d'obtenir un certain nombre de comportements différents plus ou moins réalistes. Nous proposons un mécanisme de sélection d'action s'appliquant aux systèmes multi-agent, composés d'agents cognitifs et géographiquement situés dans l'environnement, dans lesquels la résolution des buts des agents s'effectue sous la forme d'arbres de planification (un arbre-et/ou par but).

*Ce travail est cofinancé par le CPER TAC de la région Nord-Pas de Calais, les fonds européens du FEDER et une allocation de recherche MESR.

Nous ne nous intéressons pas ici, à la construction des arbres de planification, nous supposons que la racine est un but à résoudre et que les feuilles sont des actions exécutables (dont les conditions d'exécutions sont satisfaites). Notre but n'est pas que nos agents résolvent leurs buts de manière optimale (en nombre d'actions par exemple), mais que cette résolution soit réaliste et cohérente par rapport aux comportements que l'on désire leur attribuer. Pour cela, notre mécanisme de sélection d'action permet à partir des arbres de planification de l'agent, de sélectionner l'action à exécuter en respectant le comportement de l'agent. Ce comportement est défini à l'aide de trois facteurs, qui sont les priorités des buts, l'influence de l'environnement (dont l'opportunisme fait partie) et la personnalité de l'agent. Les **priorités des buts** permettent de choisir l'action à exécuter suivant le but qu'elle résolve. L'**influence de l'environnement** permet qu'un agent se détourne temporairement de l'exécution d'actions résolvant un but prioritaire, afin d'effectuer une action mise en valeur par l'environnement. La **personnalité de l'agent** est un facteur indépendant de toute simulation prenant en compte les préférences de l'agent sur les actions qu'il peut effectuer. A l'aide de celle-ci, notre agent est capable de choisir entre deux actions indistinguables par leur but ou l'environnement.

Nous définissons en section 2, les concepts basiques employés dans cet article. Dans la section 3 nous présentons, notre mécanisme de sélection d'action prenant en compte chacun des points mentionnés ci-dessus. En section 4 nous nous positionnons par rapports à des travaux connexes, avant de conclure.

2 Définitions préalables

Définissons quelques éléments de l'arbre de planification. Une **action** possède un ensemble de conditions nécessaires à son exécution et un ensemble d'effets issus de son exécution. Par exemple,

Action : ouvrir la porte

Conditions : l'agent a la clé, la porte est verrouillée.

Effets : la porte est ouverte.

Un **arbre-et/ou** est une structure arborescente alternant **nœud-et** et **nœud-ou** de père en fils. Les nœuds-ou représentent les actions permettant de satisfaire une condition (un but ou un nœud-et), dans les nœuds-et se trouvent les conditions nécessaires à l'exécution du nœud père. Nos arbres-et/ou présentent en racine le but à réaliser et en feuilles les actions pouvant être effectuées (les conditions d'exécution sont satisfaites).

Un **but** est un ensemble de conditions qui doivent être satisfaites. A chaque but b de l'agent a correspond l'arbre de planification associé (noté $Plan_a(b)$).

$$\forall a \in Agent, \forall b \in But_a,$$

$$Plan_a : \begin{array}{l} But_a \rightarrow Arbre \\ b \mapsto Plan_a(b) \end{array}$$

Une **action exécutable** est représentée par une feuille d'un arbre de planification.

$$\forall a \in Agent, \forall b \in But_a,$$

$$\begin{array}{lcl} ActionExecutable_a : & But_a & \rightarrow Feuille \\ & b & \mapsto x \end{array}$$

$$ActionExecutable_a(b) \in \{x \mid x = Feuille(Plan_a(b))\}$$

Une **alternative** est une séquence d'actions permettant de résoudre un but à partir d'une action exécutable. Dans un arbre-et/ou, une alternative est composée de toutes les actions issues d'un nœud-et et d'un choix parmi les actions issues d'un nœud-ou.

$$\begin{array}{lcl} Alternative_a : & (Arbre \times ActionExecutable_a) & \rightarrow \{Action\}^* \\ & (Plan_a(b) \times feuille) & \mapsto \{x\}^* \end{array}$$

Chaque arbre de planification contient les différentes alternatives (les différentes suites d'actions possibles) permettant de résoudre le but. Une fois les arbres de planification construits, le mécanisme de sélection d'action va devoir choisir une action à effectuer parmi toutes les actions exécutables.

$$\begin{array}{lcl} \forall a \in Agent, \forall b \in But_a, Selection : & Agent & \rightarrow Action \\ & a & \mapsto x \in ActionExecutable_a(b) \end{array}$$

Le mécanisme de sélection d'action attribue une note ϕ_a à chaque action exécutable et retient celle ayant reçu la note maximale.

$$\begin{array}{lcl} \forall a \in Agent, \forall b \in But_a, \phi_a : & ActionExecutable_a(b) & \rightarrow \mathbb{R} \\ & x & \mapsto NoteAction \end{array}$$

$$\begin{array}{l} \text{Prochaine action : } x \in ActionExecutable_a(b) \mid \\ \phi_a(x) = \max_{y \in ActionExecutable_a(b)} \{\phi_a(y)\} \end{array}$$

Classiquement, cette note correspond à l'évaluation de l'action suivant les facteurs pris en compte par le mécanisme de sélection d'action. C'est donc dans la notation des actions exécutables, que l'on peut intervenir pour différencier les comportements. Le choix des facteurs pris en compte dans le calcul de ϕ_a est donc primordial pour obtenir un mécanisme de sélection d'action permettant des comportements variables et réalistes.

3 Notre mécanisme de sélection d'action

Dans cette partie nous détaillons incrémentalement notre mécanisme de sélection d'action prenant en compte dans un premier temps, uniquement les buts à priorités dynamiques (notée π), puis nous ajouterons l'influence de l'environnement (avec l'accomplissement noté acc et l'opportunisme noté opp) et enfin nous prendrons en compte la personnalité de l'agent (notée α). Nous présentons également la manière de combiner ces facteurs afin d'obtenir des comportements variables et réalistes.

3.1 L'influence de buts

L'importance d'un but est intrinsèquement variable, un but permettant la survie de l'agent est certainement plus prioritaire que d'autres. De plus, l'importance d'un but est parfois fonction d'un paramètre, par exemple la faim : j'ai moins faim si je viens de

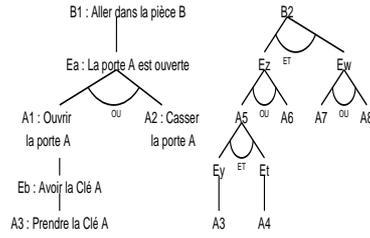


FIG. 1 – Deux arbres de planification, le premier concerne le but $B1$ (les noms des actions et conditions sont données dans celui-ci afin de montrer un exemple concret d'arbre-et/ou), le second le but $B2$. Les buts $B1$ et $B2$ ont respectivement les priorités $\pi_a(B1)$ et $\pi_a(B2)$. Si $\pi_a(B1) < \pi_a(B2)$ alors les actions exécutable $\{A2, A3\}$ sont moins prioritaires par rapport aux actions exécutable $\{A3, A4, A6, A7, A8\}$. $A3$ étant une action exécutable des deux buts, elle aura la priorité donnée par le but le plus prioritaire $B2$. Ainsi la note des actions exécutable correspond à la priorité du but : $\phi_a(A2) = \phi_a(A3) = \pi_a(B1)$ et $\phi_a(A3) = \phi_a(A4) = \phi_a(A6) = \phi_a(A7) = \phi_a(A8) = \pi_a(B2)$.

manger et de plus en plus faim tant que je n'ai pas mangé. Afin de réaliser cette importance des buts, nous affectons à chaque but une priorité π dynamique (une fonction variante) ou statique (une fonction constante).

$$\forall a \in Agent, \pi_a : \begin{matrix} But_a & \rightarrow & \mathbb{R} \\ b & \mapsto & ValeurBut \end{matrix}$$

$$\pi_a(b) : \forall a \in Agent, \forall b \in But_a, 0 \leq \pi_a(b) \leq MAX,$$

$$\forall x \in ActionExecutable_a(b), \phi_a(x) = \pi_a(b).$$

Cette priorité influence le mécanisme de sélection d'action qui privilégiera les actions qui résolvent des buts de priorités importantes (FIG.1). A chaque pas de temps le choix des buts à résoudre est donc remis en cause. Cela permet la prise en compte de nouveaux buts et de la dynamique de buts existants, ce qui peut totalement modifier l'ordre de réalisation des buts. Cependant cette influence ne prend pas en compte l'environnement, notre agent adopte donc un comportement proche des personnages que l'on trouve dans le jeu The Sims, où les buts des personnages sont donnés par l'utilisateur et organisés dans une file (FIFO). Pour mesurer le manque de prise en compte de l'environnement, nous traiterons un exemple bien connu dans ce jeu, qui est celui du journal et du courrier. Le Sims pour prendre son courrier et son journal aura ce comportement : il sortira de la maison, prendra le courrier, le déposera sur la table la plus proche à l'intérieur de la maison, ressortira, prendra le journal et le déposera à l'intérieur. Ce comportement est très coûteux en déplacement et peu rationnel. Un comportement plus rationnel aurait été de sortir une seule fois, de prendre le courrier et le journal et de les déposer à l'intérieur. Pour réaliser ce genre de comportement, il faut permettre au mécanisme de sélection d'action de prendre en compte l'environnement (en plus des priorités sur les buts) et de réaliser des actions de priorités plus faibles, à moindre coût. Pour cela nous avons mis en place l'influence de l'environnement sur le comportement de l'agent.

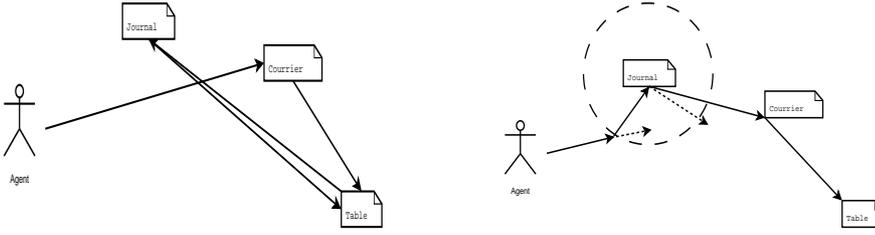


FIG. 2 – Influence de l’opportuniste sur la sélection d’action. Dans cet exemple, l’agent a a deux buts, B_j le but de déposer le journal sur la table, B_c le but de déposer le courrier sur la table, avec $\pi_a(B_c) > \pi_a(B_j)$. Pour réaliser ces buts l’agent doit prendre ces objets (actions P_j et P_c) et les déposer ensuite sur la table (actions D_j et D_c). **A gauche**, un mécanisme de sélection ne prenant pas en compte l’opportuniste ($\phi_a(D_c) = \phi_a(P_c) = \pi_a(B_c)$ et $\phi_a(D_j) = \phi_a(P_j) = \pi_a(B_j)$) l’agent exécute donc dans l’ordre les actions : P_c, D_c, P_j et D_j , ce qui est très coûteux en déplacement. **A droite**, notre mécanisme de sélection prenant en compte l’environnement : au départ la séquence d’actions est identique (P_c, D_c, P_j et D_j) mais en passant près du journal (d’une distance inférieure à $seuilOpp_a$ symbolisé par le cercle en tirets) l’agent se détourne du courrier pour aller prendre le journal par opportuniste, une fois le journal pris l’agent ne va pas le déposer, car l’action P_c redevient la plus prioritaire, il y a donc une seconde remise en cause de l’ordre de résolution des buts. L’agent finit donc par prendre le journal, puis le courrier et les déposer sur la table exécutant la séquence (P_j, P_c, D_c et D_j).

3.2 L’influence de l’environnement : opp

Pour se détourner temporairement d’un but (déposer le courrier) afin de réaliser un but proche (prendre le journal), un agent doit prendre en compte son environnement. Nos agents situés peuvent évaluer la distance entre eux et un objet de l’environnement, ainsi le mécanisme de sélection d’un agent peut valoriser la note ϕ_a d’une action car la cible sur laquelle se porte l’action est proche et détourne temporairement la résolution du but le plus prioritaire. L’influence de l’environnement sur le comportement de l’agent a est appelé opportuniste est notée opp_a .

$$\forall a \in Agent, opp_a : Action \rightarrow \mathbb{R}$$

$$x \mapsto ValeurOpp$$

$$cible : Action \rightarrow Agent$$

$$x \mapsto t$$

$$opp_a(x) = \max(1, \frac{seuilOpp_a}{d_a(cible(x))}),$$

$seuilOpp_a$ est le seuil du caractère d’opportuniste de l’agent a ,

$d_a(cible(x))$ est la distance entre l’agent a et la cible (noté t) de l’action x .

Ainsi un agent peut se détourner temporairement d’un but principal afin de réaliser un but proche.

$$\forall a \in Agent, \forall b \in But_a, \forall x \in ActionExecutable_a(b), \phi_a(x) = \pi_a(b) * opp_a(x)$$

Remarques. Suivant la valeur de $seuilOpp_a$, nous pouvons obtenir des agents plus au moins opportunistes. Notre choix d'utiliser la multiplication pour combiner π_a et opp_a s'explique par la formule de opp_a :

- Si $d_a(cible(x)) \geq seuilOpp_a$, la valeur de opp_a est de 1, avec la multiplication, l'opportunisme n'a pas d'influence (FIG.2 à gauche).
- Si $d_a(cible(x)) < seuilOpp_a$, la valeur de opp_a est supérieure à 1 et augmente en fonction de la réduction de la distance agent-cible, augmentant ainsi l'influence de l'opportunisme (FIG.2 à droite).

3.3 L'influence de l'environnement : acc

Notre agent choisit les actions à effectuer suivant la priorité du but qui se trouve en racine de l'arbre de planification et de la distance entre l'agent et la cible de l'action. Nous avons introduit une deuxième influence de l'environnement, que nous avons appelé le caractère d'accomplissement des buts de l'agent, noté acc_a pour l'agent a . Ce caractère valorise les alternatives des buts qui peuvent être accomplies rapidement. Ainsi comme pour l'opportunisme, le caractère d'accomplissement des buts poussera l'agent à se détourner de la résolution d'un but pour accomplir les actions d'un but qui peut être résolu en peu d'actions. Plus un agent à une valeur importante pour ce caractère, plus il estimera que le fait de finir un but est plus important que de faire une action d'un autre but de même priorité mais dont l'alternative est constituée de plus d'actions (FIG.1, pour le but B1, l'agent privilégiera l'action A2 qui termine le but, à l'action A3 qui demande l'exécution de deux actions pour terminer le but et cela avec $seuilAcc_a \leq 2$).

$$\forall a \in Agent, \forall b \in But_a,$$

$$acc_a : Action \rightarrow \mathbb{R}$$

$$x \mapsto ValeurAcc$$

$$\forall x \in ActionExecutable_a(b), nbA_a(x) = \|alternative(Plan_a(b), x)\|$$

$seuilAcc_a$ est le seuil du caractère d'accomplissement de l'agent a ,

$nbA_a(x)$ est le nombre d'actions de l'alternative d'action exécutable x .

$$acc_a(x) = \max(1, \frac{seuilAcc_a}{nbA_a(x)})$$

$$\forall x \in ActionExecutable_a(b), \phi_a(x) = \pi_a(b) * opp_a(x) * acc_a(x)$$

Le mécanisme de sélection d'action prend donc en compte les préférences sur les buts à accomplir et l'influence de l'environnement, tant au niveau de la distance entre l'agent et une cible qu'au niveau de l'accomplissement d'un but.

Remarques. La valeur de $seuilAcc_a$ permet d'affiner le caractère d'accomplissement de l'agent. Ce caractère est différent de l'opportunisme car il est basé sur le nombre d'actions à effectuer entre l'action exécutable et le but, mais il ne tient pas compte de la distance à parcourir pour la réaliser. Dans un arbre-et/ou, nous pouvons calculer le nombre d'actions d'une alternative d'action exécutable x (noté $nbA_a(x)$) par la formule suivante :

Fils donne les fils (successeurs directs) d'un nœud

$$\begin{aligned} Fils : \text{nœud} &\rightarrow \{\text{nœuds}\}^* \\ x &\mapsto \{x\}^* \end{aligned}$$

Pere donne le père (prédécesseur direct) d'un nœud

$$\begin{aligned} Pere : \text{nœud} &\rightarrow \text{nœud} \\ x &\mapsto x \end{aligned}$$

Calcul de $nbA_a(x)$: $nbA_a(x) = (N(x))$

où $N(x)$ récupère le nombre de nœuds *Actions* de l'alternative sélectionnée.

$$N(x) = 1 + (P(Pere(x))).$$

où $P(x)$ calcule le nombre de nœuds *Actions* parmi les ancêtres du nœud courant.

$$P(x) = N(Pere(x)) + Et(x) \forall s \in Fils(Pere(x)) \setminus x$$

où $Et(x)$ sélectionne la plus petite alternative en nombre d'actions.

$$Et(x) = A(f) \mid f \in Fils(x) \text{ et } A(f) = \min_{k \in Fils(x)} (A(k))$$

où $A(x)$ compte les nœuds *Action* parmi les fils issus d'un choix.

$$A(x) = 1 + Et(f), \forall f \in Fils(x)$$

Remarque. Notre choix d'utiliser la multiplication pour combiner π_a , opp_a et acc_a s'explique de la même manière que pour l'opportunisme, c'est-à-dire avoir une influence seulement lorsque le nombre d'actions est inférieur au seuil et dont l'importance augmente en fonction de la réduction du nombre d'actions présentes dans l'alternative.

Dans cette formule, l'influence de l'environnement permet au mécanisme de sélection d'action de prendre en compte l'environnement, mais ne permet pas de distinguer deux actions de même but et de même environnement. Prenons par exemple, un agent qui doit aller dans une pièce fermée par une porte et qui a deux possibilités d'action : ouvrir ou casser la porte. Il est donc nécessaire de mettre en place un troisième facteur permettant de définir des préférences sur les actions que l'agent peut accomplir (la personnalité de l'agent, noté α).

3.4 Personnalité de l'Agent

Chaque agent possède sur les actions, ses propres préférences qui sont définies indépendamment de toute simulation. Ces préférences sont des valeurs attribuées aux actions qui permettent d'exprimer qu'un agent peut posséder une réticence envers une action, cette **réticence** doit dévaloriser l'alternative où se trouve l'action. Un agent peut également aimer faire une action, cet **attirance** doit valoriser l'alternative où se trouve l'action. Enfin un agent peut vouloir ne jamais faire une action, cette **inhibition** doit être prise en compte comme une inhibition de l'alternative (une réticence absolue). La personnalité de l'agent (noté α) représente l'impact des préférences sur le choix d'une alternative (et donc la sélection de son action exécutable). Pour noter une alternative, nous nous intéressons à l'ensemble des actions appartenant à celle-ci. C'est-à-dire uniquement à la présence de l'action dans l'alternative.

Pour composer la personnalité de l'agent (α) avec l'influence des buts et de l'environnement, nous devons sélectionner l'intervalle pour la notation des préférences des actions afin d'obtenir la note α et choisir la manière de composer cette note avec les autres

influences. Après avoir comparé plusieurs compositions sur différents intervalles, nous avons choisi de retenir le produit dans l'intervalle $[0, N]$. Cela nous permet de répondre à nos attentes sur l'absence d'influence ($\alpha = 1$), l'attrance ($\alpha \in]1, N]$), la réticence ($\alpha \in]0, 1[$) sur les actions à accomplir et l'inhibition (en prenant $\alpha = 0$, dans ce cas $\phi_a = 0$). Ce choix nous permet donc d'écrire que :

$$\forall a \in Agent, \forall b \in But_a,$$

$$\begin{aligned} \phi_a : ActionExecutable_a &\rightarrow \mathbb{R} \\ x &\mapsto NoteAction \end{aligned}$$

$$\forall x \in ActionExecutable_a(b), \phi_a(x) = \pi_a(b) * opp_a(x) * acc_a(x) * \alpha(x)$$

La valeur de α est la combinaison des préférences des actions d'une alternative. Nous avons étudié et retenu quatre combinaisons possibles (permettant la réalisation de notre définition de la personnalité d'un agent) afin d'obtenir α . Ces quatre combinaisons sont le **minimum**, la **moyenne des N minima**, la **moyenne géométrique** et la **moyenne harmonique**. Pour calculer la valeur des α dans un arbre-et/ou nous nous basons sur le fait qu'au final l'action choisie sera l'action exécutable ayant obtenu la note maximale. Nous partons donc de l'action exécutable (en feuille), pour sélectionner les nœuds correspondant son alternative. Dans un premier temps nous prenons l'ensemble des actions de la branche allant de la feuille jusqu'à la racine. En effet ces actions ne nécessitent pas de choix de la part de l'agent, car elles font partie de l'alternative. Ensuite nous nous occupons des conditions (nœuds-ou) de l'alternative. En effet les fils de ces nœuds peuvent représenter un choix que l'agent aura à effectuer.

Sachant que le choix entre chaque alternative est fait en fonction de la composition ayant obtenu la meilleure note et que les fonctions de compositions proposées, nous permettent toutes d'appliquer la relation d'additivité suivante : $\max(Composition(B \cup C \cup D), Composition(B \cup F \cup G)) = note(B) + \max(Composition(C \cup D), Composition(F \cup G))$, nous pouvons élaguer notre parcours en pré-sélectionnant la meilleure alternative parmi celles présentes dans les sous-arbres issus de nœuds-ou. Pour cela, il suffit d'appliquer la règle de maximisation dans le sous-arbre de chaque nœud-ou.

α est la note attribuée à l'alternative d'une action exécutable.

$$\begin{aligned} \alpha : ActionExecutable_a &\rightarrow \mathbb{R} \\ x &\mapsto NoteAlternative \end{aligned}$$

Comb est une fonction utilisée pour combiner les préférences des différents nœuds.

$$\begin{aligned} Comb : \{Actions\}^* &\rightarrow \mathbb{R} \\ \{x\}^* &\mapsto NoteAlternative \end{aligned}$$

Calcul de α : α est la note pour la personnalité de l'agent combinant l'ensemble des nœuds de l'alternative de l'action x , $\alpha(x) = Comb(N(x))$

$N(x)$ récupère les *Actions* de l'alternative sélectionnée.

$$N(x) = \begin{cases} \emptyset & \text{si } x = \text{racine} \\ \{x\} \cup (P(Pere(x))) & \text{sinon} \end{cases}$$

$P(x)$ collecte les *Actions* parmi les ancêtres du nœud courant.

$$P(x) = N(Pere(x)) \cup \bigcup_{s \in Fils(Pere(x)) \setminus x} Et(s)$$

$Et(x)$ sélectionne la meilleure alternative des nœuds et. Cette alternative est celle qui pour l'ensemble des nœuds la composant obtient la meilleure note en prenant en compte l'ensemble des actions de l'alternative.

$$Et(x) = A(f) \mid f \in Fils(x) \text{ et } Comb(A(f)) = \max_{k \in Fils(x)} (Comb(A(k)))$$

$A(x)$ récolte les nœuds *Action* parmi les fils issus d'un choix.

$$A(x) = \{x\} \cup \left(\bigcup_{f \in Fils(x)} Et(f) \right)$$

Dans notre approche, nous pouvons remarquer que les facteurs pris en compte par le mécanisme de sélection d'actions sont représentés par la totalité des arbres de planification. En effet, la personnalité de l'agent se retrouve dans chaque nœud *Action* des arbres, l'influence des buts est présente aux racines, l'accomplissement est construite suivant la taille des alternatives des arbres et l'opportunisme est calculé sur les feuilles (actions exécutables).

3.5 Autres Facteurs

Notre mécanisme de sélection d'action utilise les influences des buts, de l'environnement ainsi que la personnalité de l'agent pour choisir l'action à effectuer, ce qui détermine le comportement de l'agent. Ces différents facteurs sont paramétrables afin d'être utilisés dans le plus grand nombre de simulations possibles. Néanmoins pour obtenir un comportement plus réaliste nous ajoutons deux autres facteurs que sont la revalorisation multi-but et l'inertie de l'action exécutée.

Parmi les arbres de planification d'un agent, il est possible qu'une action apparaisse plusieurs fois en tant qu'action exécutable (exemple de A3 en figure 1). Cette action fait donc "avancer" plus vite un ou plusieurs buts, c'est pourquoi elle doit être valorisée par la **revalorisation multi-but**. Pour valoriser cette action, nous lui affectons la valeur de la somme probabiliste de ces m notes ϕ après les avoir normalisées (m étant le nombre d'alternatives ayant la même action exécutable x).

MAX : la borne supérieure pour π_a

$seuilOpp$: la borne supérieure opp_a

$seuilAcc$: la borne supérieure acc_a

N : la borne supérieur pour α

$$Bornes = MAX * seuilOpp_a * seuilAcc_a * N$$

$$\forall a \in Agent, \forall b \in But_a, \forall x \in ActionExecutable_a(b),$$

$$E = \{x' \mid \exists b' \in But_a, x' \in ActionExecutable_a(b'), x = x'\}$$

$$\phi_a(x) = Bornes * \sum proba_E \left(\frac{\pi_a(b) * opp_a(x) * acc_a(x) * \alpha(x)}{Bornes} \right).$$

Afin d'éviter les phénomènes d'oscillations entre différentes alternatives et ainsi assurer une certaine persistance, la valeur $\phi_a(x)$ de l'action x sélectionnée à l'instant t sera valorisée par un pourcentage noté *Inert* correspondant à l'**inertie** de l'action. Ainsi pour qu'une action y prenne la place de l'action x à $t+1$, $\phi_a(y)$ devra être supérieure à $\phi_a(x)_{valorisee}$. Si c'est le cas la valeur $\phi_a(y)$ sera également valorisée par le même pourcentage *Inert*. Une fois l'action exécutée, nous répercutons l'inertie sur le père de l'action afin de persister dans l'alternative.

4 Travaux connexes

La notation d'actions est un problème important, de nombreux travaux de recherche (Bryson, 2003; Pirjanian, 1997) ont été faits dans ce domaine. Parmi eux on peut citer Pattie Maes (Maes, 1990) et Alejandro Guerra-Hernández (Guerra-Hernández, 1998) qui construisent un réseau reliant les actions suivant les préconditions et effets des actions. Ces réseaux ont un fonctionnement semblable aux réseaux de neurones, qui permettent à chaque neurone de transmettre de l'énergie à son successeur à partir d'une valeur seuil d'énergie reçue ainsi que la réception d'énergie négative appelée inhibition. D'autres approches à base de règles comme SOAR (Laird *et al.*, 1987) obtiennent des comportements permettant des résolutions optimales des buts des agents. Les comportements de ces agents sont déterminés principalement par ces règles qui peuvent évoluer (par exemple par des phases d'apprentissage). Dans SOAR, la sélection d'action est réalisée par des opérateurs. Un opérateur est un type abstrait et sa concrétisation reste à réaliser et à adapter selon le comportement désiré. A la différence nous proposons un mécanisme de sélection d'action concret qui permet l'obtention de comportements différents. Ainsi deux agents ayant les mêmes arbres de planification peuvent résoudre leurs buts de manières différentes. Pour cela, nos agents possèdent des préférences pour chaque action, ces préférences constituent la personnalité de l'agent. A partir de ces préférences (indépendantes de la simulation) et de leurs combinaisons avec des influences (liées à la simulation), notre mécanisme de sélection d'action permet cette différenciation des comportements tout en gardant la cohérence dans le choix des actions à effectuer. Dans ce sens, nous nous rapprochons des travaux de Bernd Schmidt (Schmidt, 2005) sur le modèle PECS (Physical conditions Emotional state Cognitive capabilities Social status) qui fonctionne en quatre étapes :

Approche PECS	Notre proposition
Calculer les valeurs des variables d'état interne	Calculer les priorités des buts
Calculer l'intensité des chaque motivation	Calculer l'intensité des influences
Sélectionner la motivation la plus importante	Sélectionner l'action ayant eu la meilleure note ϕ
Effectuer l'action sélectionnée	Effectuer l'action sélectionnée

Néanmoins, nous apportons en plus de la notion d'influence, la notion de personnalité qui permet d'avoir des préférences, réticences ou inhibitions sur certaines actions. De plus un mécanisme de sélection d'action doit permettre, à la fois l'opportunisme tout gardant une certaine persistance (éviter les oscillations entre les actions), comme la proposition de Blumberg (Blumberg, 1994). Toby Tyrrell (Tyrrell, 1993) décrit les problèmes qui doivent être pris en compte pour évaluer un bon mécanisme de sélection d'action. Nous satisfaisons un grand nombre de ces critères. Le tableau suivant reprend les principaux critères énoncés par Tyrrell et précise comment notre proposition les respecte.

Critères de Tyrrell	Notre proposition
Persistance : continuer jusqu'à l'achèvement d'une action, pour éviter le coût d'un changement d'action.	L'inertie de l'action en cours jusqu'à la fin de celle-ci, y compris pendant un déplacement nécessaire à son exécution.
Activations proportionnelles à l'état courant : dans les systèmes homéostatiques, le besoin d'avoir des activations proportionnelles à l'état courant.	l'influence des buts dynamiques peuvent être utiliser comme des variables homéostatiques.
Concurrence équilibrée : les nœuds aidant à la réalisation d'un seul but ne doivent pas être dévalorisés par rapport aux nœuds achevant plusieurs buts.	Pas de discrimination des nœuds achevant un seul but, mais valorisation des nœuds multi-but.
Continuité des séquences d'actions : besoin d'une autre persistance car changer de séquence d'actions a un coût élevé.	L'inertie est transmise au père de l'action exécutée, permettant la persistance sur l'alternative.
Interruptibilité si nécessaire : interrompre une séquence d'actions de priorité relativement faible pour en effectuer une de priorité plus importante.	Notre inertie permet l'interruptibilité. Les influences de l'environnement peuvent valoriser des actions, permettant d'interrompre l'alternative en cours.
Opportunisme : permettre d'interrompre une séquence d'actions afin de profiter de l'avantage qu'offrent des opportunités.	Même définition pour notre opportunisme.
Combinaison de préférences : prendre en compte des priorités de nœuds de hauts niveaux pour choisir parmi les nœuds de bas niveaux.	Prise en compte de toutes les actions de l'alternative et de l'influence des buts dans la sélection parmi les actions exécutables.
Combinaison Flexible des stimuli : utiliser d'une fonction arbitraire pour combiner les valeurs des stimuli.	Préférences des actions, plusieurs fonction de combinaison entre les influences et la personnalité.

5 Conclusion et perspectives

Une simulation réaliste nécessite de prendre en compte un ensemble de contraintes (Tyrrell, 1993) permettant une grande variabilité de comportements. Nous proposons un mécanisme de sélection d'action concret permettant de prendre en compte ces contraintes dans une approche cognitive. Pour cela dans un système multi-agent cognitifs et géographiquement situés, nous utilisons un mécanisme de sélection d'action basé sur trois grands facteurs : l'influence des buts, l'influence de l'environnement et la personnalité de l'agent. Ces influences permettent la gestion de plusieurs buts de priorités dynamiques, le détournement temporaire de l'exécution d'actions résolvant un but afin d'exécuter une autre action valorisée par l'environnement ou par la personnalité de l'agent, le tout dans une remise en cause à chaque pas de temps de l'ordre de réalisation des buts à accomplir.

Actuellement, nous expérimentons notre proposition sur la plateforme COCOA (Devigne *et al.*, 2005b) afin de valider les choix qui ont été faits. Les modules de base de connaissances, de planification et de mécanisme de sélection d'actions sont actuellement pris en compte pour la notation d'actions. Le module de perception est le seul à ne pas influencer la notation d'action, cette prise en compte permettrait d'obtenir une influence du type : "quand je vois un objectif j'ai plus envie de le réaliser que lorsque je ne le vois pas". Nous pensons également prendre en compte les actions destructrices, c'est-à-dire l'influence de l'impact d'une action sur l'ensemble des arbres de planification et ainsi pouvoir inhiber une action capable de rendre insatisfiable un but ou avoir une réticence pour une action ralentissant un but. Ceci nous permettrait de réaliser un autre critère de Tyrrell qui est ne pas choisir la meilleure action parmi les actions exécutables, mais la meilleure action pour la résolution de tous les buts. Nous pourrions également prendre en compte l'ordre des actions dans la notation : une action très lointaine aurait une influence moindre par rapport à une action proche. Enfin il serait intéressant d'utiliser notre mécanisme de sélection d'action mono-agent, pour permettre la coopération entre agents. Un agent laisserait ce qu'il doit faire pour aller aider un autre agent, car son action est moins importante que l'action de l'autre agent. Pour cela, il faut pouvoir juger de l'importance d'une action, c'est-à-dire sélectionner l'action la plus importante parmi les actions que l'agent peut faire et l'aide qu'il peut donner aux autres agents.

Références

- BLUMBERG B. (1994). Action selection in hamsterdam : Lessons from ethology. In *Proceedings of the 3rd int. conf. on the simulation of Adaptive behaviour*.
- BRYSON J. J. (2003). Action selection and individuation in agent based modelling. In *Agent 2003 : Challenges of Social Simulation*.
- DEVIGNE D., MATHIEU P. & ROUTIER J.-C. (2005a). Interaction-based approach for game agents. In *19th European Conference on Modelling and Simulation (ECMS'05)*.
- DEVIGNE D., MATHIEU P. & ROUTIER J.-C. (2005b). Team of cognitive agents with leader : how to let them acquire autonomy. In *Proceedings of IEEE Symposium on Computational Intelligence and Games (CIG'05)*.
- GUERRA-HERNÁNDEZ A. (1998). Modeling behavior : an action selection approach. In *Workshop on Distributed Simulation, Artificial Intelligence, and Virtual Environments*.
- LAIRD J. E., NEWELL A. & ROSENBLOOM P. S. (1987). SOAR : An architecture for general intelligence. *Artificial Intelligence*, p. 1–64.
- MAES P. (1990). How to do the right thing. *Connection Science Journal, Special Issue on Hybrid Systems*.
- PIRJANIAN P. (1997). An overview of system architecture for action selection in mobile robotics.
- SCHMIDT B. (2005). Human factors in complex systems : The modelling of human behaviour. In *19th European Conference on Modelling and Simulation (ECMS'05)*.
- TYRRELL T. (1993). *Computational Mechanisms for Action Selection*. PhD thesis, University of Edinburgh.