

---

# Chapitre 1

## 1. La simulation comportementale

---

### 1.1. Introduction:

La simulation comportementale est une partie de l'animation qui se rapproche des systèmes réels de par son principe de fonctionnement en assignant aux acteurs ou systèmes animés des comportements indépendants. Ces derniers ne seront alors plus régis par un système global gérant le mouvement de tous les acteurs mais par un mécanisme de décision local placé dans chaque individu. Chaque personnage de la simulation prendra donc les décisions comportementales concernant son mouvement au pas de temps suivant, selon son état et celui de l'environnement l'entourant à cet instant de la simulation. La simulation comportementale est donc un moyen de faire interagir de manière naturelle des acteurs en simulant leurs capacités dans un environnement. Terzopoulos [TT94] a été le premier à définir un modèle comportemental. Un découpage en trois éléments distincts: Capteurs, Module comportemental et Effecteur lui ont permis de réaliser des animations réalistes de bancs de poissons naviguant au milieu de vestiges de l'ancienne Rome.

Depuis, ce modèle a été repris de nombreuses fois [BT98, SDD99] et est quasiment devenu un standard. Chaque individu d'une simulation modifie son comportement en fonction de données fournies par des capteurs et retransmet ses actions à l'environnement à travers les effecteurs. L'utilisation exclusive de ce couple acteurs/effecteurs limite la connaissance et l'effet d'un individu à leurs seules capacités d'acquisition et d'action. La simulation comportementale est utilisée en réalité virtuelle pour permettre l'animation d'entités autonomes

La plateforme de simulation doit intégrer les différents modèles de contrôle du mouvement et des interactions d'objets dans une scène. Un système de simulation est constitué d'un certain nombre d'entités, dont l'état évolue au cours du temps, chaque entité possédant une fréquence propre de calcul. La plateforme doit donc permettre la simulation, sur une même échelle temporelle, de plusieurs entités dont la synchronisation et la communication sont gérées par un moyen temps réel. Les entités communiquent, entre elles, soit par des flots de données, soit par événements.

## 1.2. Modélisation des propriétés des humains virtuels

L'objectif majeur de la modélisation des comportements des acteurs est de construire des agents intelligents autonomes virtuels avec adaptation, perception et mémoire, capable d'agir librement et avec émotion, d'être conscient et imprévisible. (Figure 1.1)

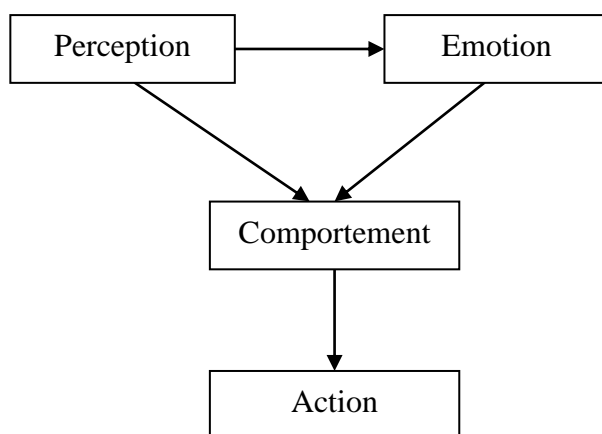


Figure 1.1 : Structure du modèle comportemental

### 1.2.1. La perception

La perception est définie comme la conscience des éléments dans l'environnement à travers des sensations physiques. Il est réalisé en équipant les agents avec des détecteurs visuels, tactiles et auditifs ainsi ils simulent le comportement quotidien humain (aspect visuel, mouvement, réaction, ...). Le sous-système perceptuel le plus important est le système visuel. Une approche basée sur la vision [RMT90] est idéale pour le modelage d'une animation comportementale. Elle offre une approche universelle pour le passage d'information de l'environnement à l'acteur dans le contexte de recherche de chemin.

A un niveau plus haut, nous pouvons décomposer la perception comme suggérée par [BT98]. La perception d'un acteur peut être limitée aux objets et à d'autres acteurs dans le voisinage.

Mais cela limite le nombre de comportements possibles, parce que seules la présence et les caractéristiques d'un objet ou d'un acteur sont impliquées dans la sélection d'un comportement. Les actions des autres acteurs ne sont pas prises en considération.

Le module de perception produit trois types de perception: la perception de la présence d'objets et d'acteurs, la perception des actions d'acteurs et la perception d'acteurs exécutant des actions sur des objets.

### **1.2.2. L'émotion**

L'émotion peut être définie comme l'aspect affectif de la conscience: Un état de sentiment, une réaction psychique et physique (comme la colère ou la crainte), subjectivement expérimenté comme un sentiment fort et physiologiquement qui augmente les changements préparant le corps pour une action vigoureuse immédiate.

Les acteurs doivent être capables de répondre, avec émotion à leur situation et agissant physiquement à cela. Les émotions visibles fournissent des designers avec un moyen direct pour affecter à l'utilisateur un état émotionnel propre à lui. Les acteurs sont donc équipés d'un modèle informatique simple de comportement émotionnel, qui est lié au comportement comme les expressions de visage qui peuvent être employées pour influencer leurs actions.

Une émotion est la réaction d'une personne à une perception. Celle-ci est amenée à répondre par une expression du visage, un geste, ou à choisir un comportement spécifique.

Une émotion arrive entre une perception et la réaction suivante. Deux personnes différentes peuvent avoir des réactions différentes à la même perception, selon la façon dont ils sont affectés par cette perception [OT90].

Les émotions sont causées par la réaction aux objets, les actions d'agents et les événements.

### **1.2.3. Le comportement**

Le comportement est souvent défini comme une réponse d'un individu, d'un groupe ou d'une espèce, à son environnement. Il est habituellement présenté en langage naturel, comme ayant des implications sociales, physiologiques et psychologiques, pas toujours réductibles au mouvement d'un ou plusieurs muscles, ou à une réponse d'effecteurs. On peut également le

décrire comme la manière dont les hommes et les animaux agissent. En d'autres termes, modéliser un comportement pour une créature virtuelle, ne s'arrête pas à définir un ensemble de réactions à son environnement. Il s'agit surtout d'inclure le flux d'informations par lequel l'environnement influence la créature concernée, et la manière dont elle code et utilise ces informations.

L'approche comportementale<sup>1</sup> consiste à faire du comportement, la propriété qui détermine entièrement les actes d'un agent. Dans les modèles qui s'en inspirent<sup>2</sup>, les comportements, qui sont décrits de manière hiérarchique, sont habituellement divisés en modules. Chaque module peut être à son tour décomposé en d'autres modules moins complexes, ou en actions élémentaires. Ces dernières s'exécutant de manière séquentielle ou concurrente. Un schéma classique de simulation basée sur de tels modèles se présente comme suit :

- Au début de l'animation, l'utilisateur rentre pour un acteur, une séquence de comportements (le script) qui sont stockés dans une pile. Le comportement en tête de la pile s'enclenche alors.
- Au terme du comportement courant - lorsque les conditions nécessaires à son déroulement ne sont plus vérifiées- le système dépile le comportement suivant et le déclenche. Ce procédé est répété jusqu'à ce que la pile soit vide.
- Les comportements décomposés en modules sont eux mêmes des piles, et s'exécutent de manière récursive lorsqu'ils sont sélectionnés par le système.

De nombreuses recherches sur l'architecture comportementale et sur les mécanismes de sélection sont effectuées.

Selon G. Hégron et B. Arnaldi [HA92], les systèmes d'animation comportementale se divisent en deux grandes familles qui distinguent les organismes qui agissent sur eux-mêmes de ceux qui agissent sur leur environnement.

### 1.2.3.1. Modèles de transformation internes

Ces modèles s'attachent à décrire de manière réaliste l'évolution interne d'entités vivantes (plantes, être humains...) en fonction de paramètres extérieurs. Il s'agit cependant de rendre compte des modifications de l'enveloppe externe de l'entité à animer (par exemple la croissance d'une plante).

---

<sup>1</sup> Souvent appelée behaviorisme - mot dérivé de l'anglais (behavior = comportement)

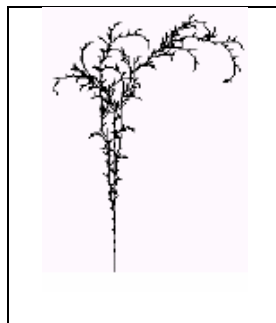
<sup>2</sup> Ils portent l'étiquette de modèles comportementaux

### 1.2.3.2. Modèles de plantes

Les modèles de croissance des plantes sont des systèmes de réécriture fondés sur un axiome et des règles de production appelés L-Systems. A. Lindenmayer [Lin68] définit un L-system comme étant un axiome constitué d'une chaîne de symboles paramétrés et temporisés, et de règles de production qui spécifient comment les symboles se transforment en fonction du temps. Les symboles sont associés à des primitives géométriques, ce qui permet de donner une représentation graphique au L-system.

La réécriture se fait en parallèle sur les différents éléments. Il y a deux manières de parvenir à une représentation graphique: d'une part, les symboles réécrits sont associés à une primitive graphique, et d'autre part, on peut insérer dans la chaîne de production des ordres équivalents à ceux que l'on donne à une tortue Logo. A noter que si la réécriture se fait en parallèle, l'interprétation graphique est complètement séquentielle.

Les règles de production des L-Systems sont des règles discrètes: dans [PHM93], P. Prusinkiewicz définit les dL-Systems en introduisant un temps continu à la place d'une séquence d'étapes de dérivation discrètes. Les paramètres  $w$  du système sont déterminés en continu par une équation différentielle ordinaire tant qu'ils restent dans un domaine  $D$ . Dès que  $w$  atteint une frontière de  $D$ , une règle de production qui dépend de la position sur la frontière, est appliquée. Ensuite, le système est à nouveau régi par l'équation différentielle. La figure 1.2 présente un exemple simple de L-system.



**Figure 1.2 : Exemple de plante générée par un L-system.**

En conclusion, les L-Systems sont utilisés à la fois en animation par ordinateur et en biologie pour simuler la croissance des végétaux. C'est aussi à la biologie (et notamment à la biomécanique humaine) que nous devons les travaux sur les modèles musculaires du corps humain que nous allons aborder maintenant.

### 1.2.3.3. Humanoïdes de synthèse

Du point de vue de l'animation, l'être humain se compose essentiellement d'un squelette, de muscles et d'une enveloppe extérieure, la peau. Pour animer un humanoïde, il faut définir un modèle d'un ou plusieurs de ces composants. La structure utilisée généralement est celle d'une structure osseuse (solide rigide articulé) sur laquelle est plaquée une peau déformable. Deux types de problèmes se posent donc: le mouvement du squelette (par exemple la reproduction de la marche humaine) et la déformation de la peau (par exemple, l'animation faciale).

#### 1.2.3.3.1.1. Reproduction de la locomotion humaine

Le cas de la locomotion humaine est un cas particulier de la locomotion. En effet, contrairement à la nage d'un poisson ou au vol d'un oiseau, la locomotion humaine se distingue par les interactions avec un sol situé à altitude variable. Il n'est donc pas possible d'ignorer complètement les contraintes cinématiques. Différentes techniques sont utilisées:

- Dans le cadre du projet Jack [PB88, LPO+89] de l'université de Pennsylvanie, X. Zhao propose une méthode fondée sur la dynamique et la minimisation d'un critère énergétique. La minimisation de l'effort ou de l'énergie ne correspond pas toujours à une vérité biomécanique, à cause d'une corrélation très forte avec la morphologie du sujet. De plus, l'apprentissage de la marche, ou même la marche au pas des militaires sont des exemples de marche non économique.
- J. Hodgins [Hod96] utilise des techniques classiques de contrôle: il y a un contrôleur différent pour chaque phase de la marche, et ces contrôleurs sont reliés entre eux par des automates finis. Ces modèles sont extrêmement coûteux en temps de calcul.
- Dans le cadre de l'animation interactive, des approches purement cinématiques sont proposées [BMT90]. Plus récemment, R. Boulic [BM96] a intégré des notions de dynamiques en regroupant les articulations dans un arbre de distribution des masses. Ces approches sont intéressantes du point de vue du coût mais permettent difficilement la personification des démarches obtenues. Une amélioration consiste à utiliser des mouvements acquis à l'aide de systèmes vidéo et à les rejouer [FGP+98]. Le coût mémoire est alors plus élevé mais la démarche nettement plus réaliste.
- F. Multon [Mul96] s'intéresse à la construction de modèles biomécaniques de la locomotion humaine. Le cycle de marche est décomposé en différentes phases reliées

par des automates. Pour chacun des cycles des trajectoires articulaires sont interpolées par de simples polynômes de degré 3. L'intérêt de l'approche réside dans le fait que les phases identifiées ne sont pas celles qui sont utilisées habituellement en biomécanique; trois phases sont utilisées pour l'extension du genou et une pour sa flexion, pour la marche comme pour la course. Cette méthode permet en outre l'identification simple de paramètres naturels comme la longueur ou la fréquence des pas.

#### **1.2.3.3.1.2. Animation faciale :**

L'animation faciale consiste à animer le visage d'un être humain. C'est une tâche complexe qui nécessite la modélisation, l'animation et le rendu de la géométrie du visage mais aussi d'éléments distinctifs du visage comme la peau, les cheveux, les lèvres ou la langue. P. Fua [FM97] classe les modèles d'animation faciale en deux catégories:

- les modèles à peu de degrés de liberté, qui se concentrent sur les mouvements de la tête. Ils utilisent des modèles simplistes et sont donc très pauvres graphiquement.
- Les modèles plus sophistiqués; il s'agit en premier lieu d'extraire la géométrie à partir d'images réelles [EBD+97] ou de la générer automatiquement [LTW95]. Ensuite, il faut définir des paramètres de déformation pour contrôler les mouvements du visage.

Trois types de modèles sont utilisés [DB97]: des modèles purement physiques, des modèles purement géométriques ou des modèles anatomiques.

#### **1.2.3.4. Modèles de transformation externes :**

Les modèles dits de transformation externes définissent le comportement extérieur d'un être, c'est à dire ses actions et réactions vis à vis de son environnement, que ce soit de façon individuelle ou collective.

Il existe plusieurs classifications des modèles de comportement externes. D. Zeltzer [Zel90] propose une classification fondée sur le niveau d'abstraction, tandis que S. Donikian [Don94] élabore la sienne selon le type de techniques utilisées. Cette classification paraît plus pertinente ici dans la mesure où les techniques employées sont plus faciles à classer. La plupart des approches de l'animation comportementale que nous allons voir seraient impossibles à classer dans une seule des catégories définies par Zeltzer.

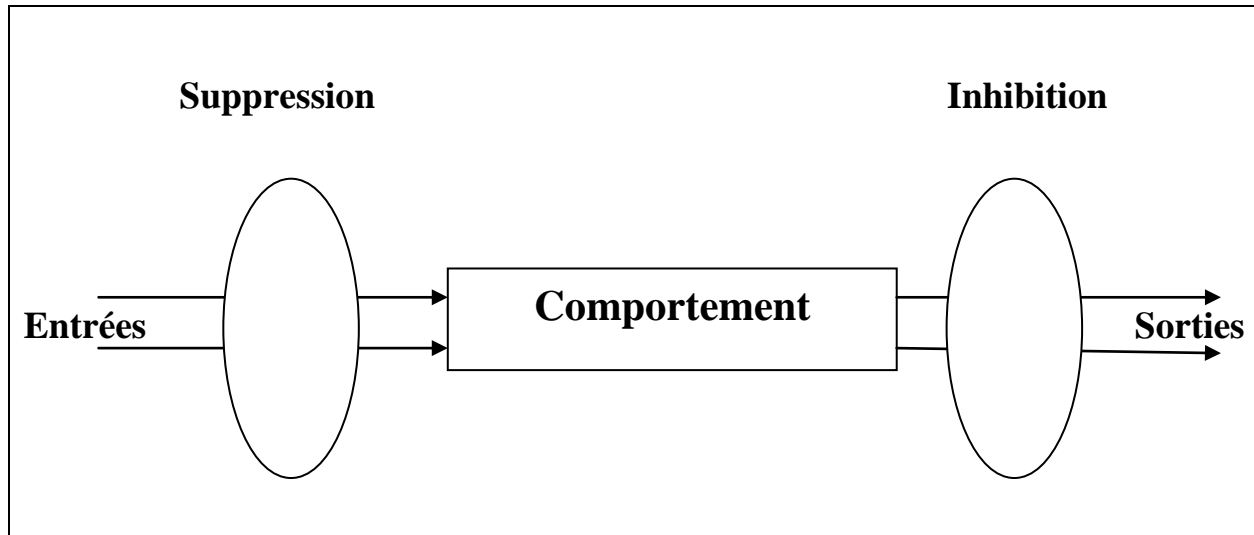
##### **1.2.3.4.1. Approche de R. Brooks**

Brooks fait partie des pionniers ayant étudié les architectures dédiées à la génération de comportements. En 1986 il définit un modèle se fondant sur une décomposition en

comportements de base [Bro86]. Ces comportements sont représentés à l'aide d'automates à états finis temporisés fonctionnant de manière asynchrone. Les capteurs stockent les données dans des registres qui sont accessibles à tous les comportements. Ces comportements ont également entre eux un mode de communication basé sur des messages. Ils peuvent agir sur tous les effecteurs.

Le regroupement de ces comportements forme des niveaux de compétence. Ces systèmes sont hiérarchisés selon le principe de subsumption: les niveaux supérieurs manipulent les entrées et sorties des systèmes inférieurs, les contrôlant de manière indirecte. Pour cela, Brooks a introduit deux opérateurs: un opérateur d'inhibition et un opérateur de suppression.

L'opérateur de suppression agit sur les entrées des niveaux inférieurs en bloquant l'accès à certains capteurs pour les remplacer par ses propres données. Grâce à cette méthode, il contrôle l'information sur laquelle travaille le niveau inférieur. Le deuxième opérateur, l'inhibition, travaille sur les sorties des niveaux inférieurs en empêchant la diffusion de messages sur la sortie du comportement. Ces deux mécanismes sont qualifiés par le terme anglais "subsumption".



**Figure 1.3 : Opérateurs de « subsumption »**

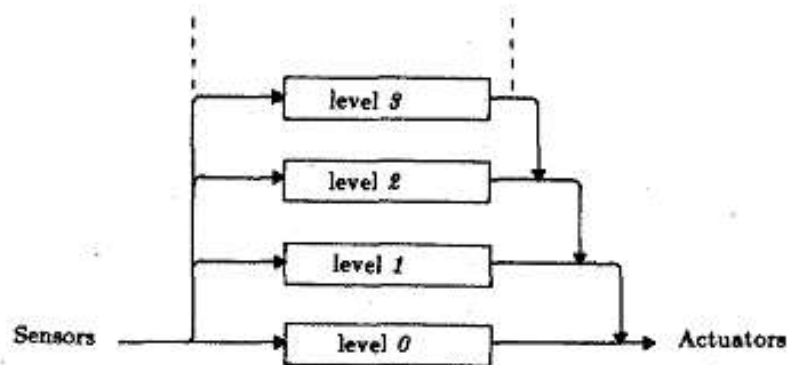
Brooks décompose le problème du contrôle d'un robot mobile en niveaux de compétence dont chacun définit un comportement pour le robot mobile, par exemple : avoid objets, wander, explore... (Figure 1.4). Un niveau supérieur implique un comportement plus spécifique. Le principe de l'architecture de subsumption est qu'on peut ajouter des niveaux



supérieurs dans les niveaux actuels sans devoir modifier ces derniers. Les niveaux supérieurs peuvent manipuler les entrées et sorties des niveaux inférieurs (Figure 1.5).



**Figure 1.4 : Décomposition du contrôle d'un robot mobile en comportements**



**Figure 1.5 : Architecture de subsumption. Le contrôle est décomposé en niveaux de compétence. Un niveau supérieur subsume le rôle des niveaux inférieurs s'il veut prendre le contrôle**

Un niveau de compétence est construit par un ensemble de comportements de base. Ils se communiquent par des messages. Quand un niveau supérieur veut prendre le contrôle et subsume un niveau inférieur, il utilise deux opérateurs de subsumption : suppression et inhibition. L'opérateur de suppression supprime les messages entrants des niveaux inférieurs et les remplace par ses propres données. L'opérateur d'inhibition bloque les messages sortants des niveaux inférieurs. Les deux opérateurs sont effectifs pendant un certain temps (Figure 1.6). La figure 1.7 montre un exemple d'architecture de subsumption pour le contrôle du robot mobile avec trois niveaux de compétence (avoid objects, wander et explore).

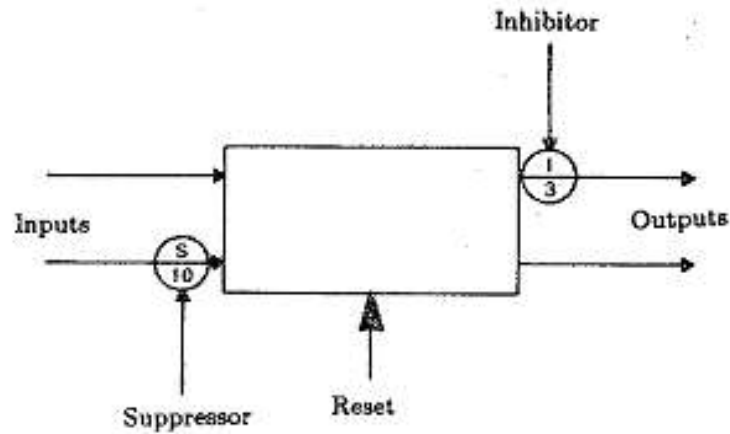


Figure 1.6: Opérateurs de subsumption. Le temps est noté dans le cercle

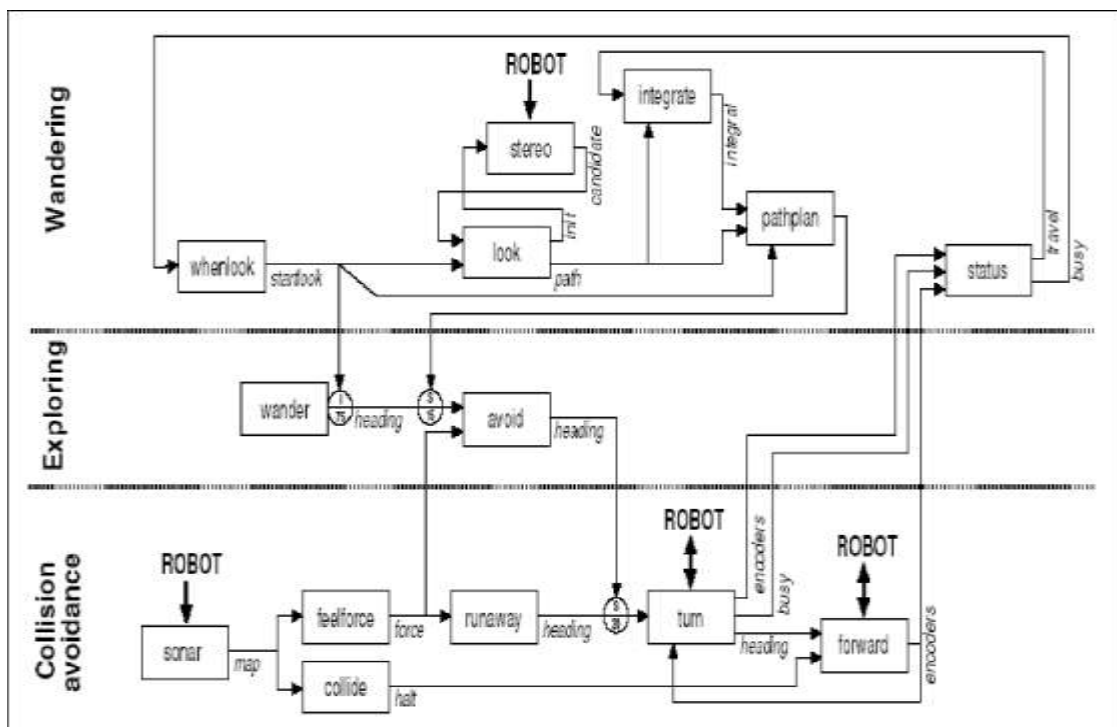


Figure 1.7: Exemple d'architecture de subsumption avec trois niveaux de compétence

Brooks a réussi à utiliser l'architecture de subsumption pour contrôler les comportements des robots mobiles virtuels et physiques tels qu'errer en évitant des obstacles ou atteindre une cible.

Mais, l'utilisation d'une telle architecture est alors une tâche très complexe car il faut définir manuellement les valeurs de temps effectif pour chaque opérateur de subsumption, l'interaction entre les comportements de base dans un niveau de compétence et l'interconnexion entre les niveaux de compétence. De ce fait, il est alors difficile de reconfigurer cette architecture pour faire des tâches différentes. [Tra07]

### 1.2.3.4.2. Les Boids de C. Reynolds

Dès 1986, C. Reynolds s'est attaché à construire un modèle comportemental permettant d'animer un ensemble d'individus à l'aide de la description du comportement individuel de chaque animal appartenant au groupe [Rey87]. Le terme «boids» représente le nom générique de ces entités. Ce modèle est constitué de trois comportements de base décrivant le déplacement d'un boid en fonction des positions et vitesses des individus à proximité de lui. La cohésion permet de garder une formation. La séparation permet d'éviter les agrégats, et l'alignement est utilisé pour obtenir un déplacement uniforme des individus. En sommant les vecteurs résultants de ces opérations, Reynolds obtient le vecteur de déplacement désiré.

Trois règles de comportement sont définies: éviter les collisions, harmoniser sa vitesse à celle de ses voisins et rester proche de ses voisins. La figure 1.8 présente un exemple de nuée d'oiseaux dont le comportement respecte les règles précédentes.

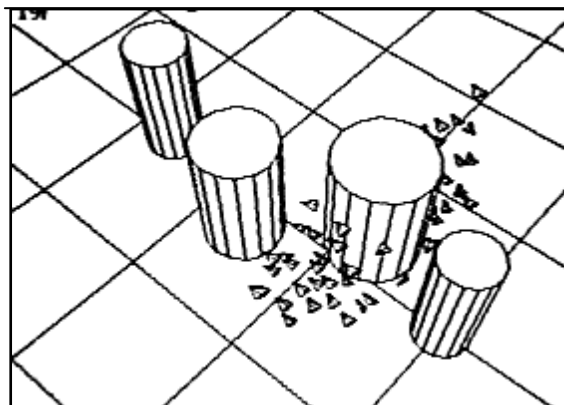


Figure 1.8 : Règles de comportement: une nuée d'oiseaux évitant un obstacle

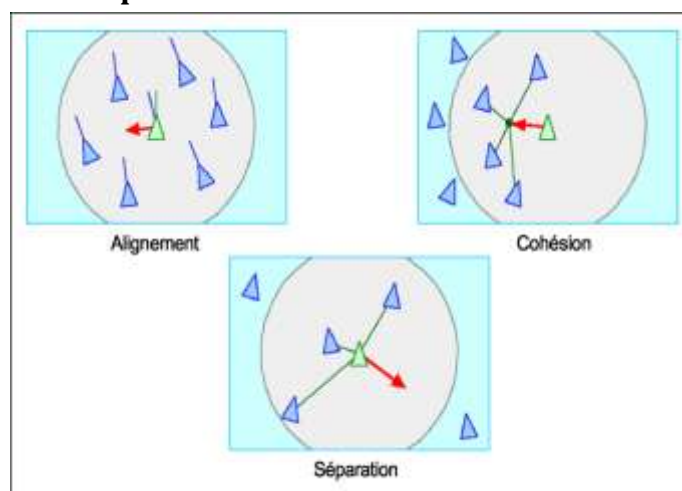


Figure 1.9 : Comportements primaires des boids.

Cette technique paraît relativement simple mais permet, par émergence, d'obtenir des comportements complexes. Cette méthode sera, par la suite, utilisée avec succès dans l'industrie cinématographique pour animer des bancs de poisson, ou des hordes de chauvesouris. Les applications de ce modèle restent limitées à des animations de groupes d'individus.

En 1999, C. Reynolds étend son modèle pour contrôler le déplacement d'entités autonomes [Rey99]. Il propose alors une série de techniques afin d'obtenir des comportements de proie/prédateur, de suiveur dans un groupe, ou plus simplement d'évitement d'obstacles.

Ces comportements s'insèrent dans une structure hiérarchique globale se scindant en trois parties: un module de sélection de l'action qui, en fonction des buts et stratégies, va définir les comportements primaires à appliquer. Ceux-ci seront ensuite transformés en ordres d'animation.

La principale difficulté réside dans la construction de ces comportements primaires, souvent dépendants du support sur lequel ils sont appliqués.

#### **1.2.3.4.3. Approche basée comportement: Mataric**

J. Mataric [Mat99] s'appuie sur une collection de comportements pour contrôler l'animation de ses entités. Un comportement est un processus ou des lois de contrôle qui résolvent ou satisfont un objectif. Par exemple l'évitement d'obstacle satisfait l'objectif de ne pas avoir de collisions.

Chaque comportement reçoit une information des capteurs de l'entité et éventuellement une information provenant d'autres comportements. A l'aide, éventuellement de données propres, il envoie alors une réponse aux effecteurs de l'entité ou à d'autres comportements. L'architecture forme alors un réseau structuré.

Cette architecture utilise une représentation uniforme de l'échelle de temps. L'exécution de ces comportements est parallèle et distribuée. De plus, chaque comportement doit avoir un temps de réponse très court dans le but de satisfaire à des exigences de temps réel.

L'ajout de comportements se fait de manière incrémental: à partir de comportements de réactifs comme l'évitement d'obstacles, il est possible d'introduire des comportements plus complexes de suivi de trajectoire, d'exploration... des comportements sont donc ajoutés jusqu'à ce que leurs interactions conduisent au résultat escompté.

Ces modèles ont une représentation du monde distribué sur différents comportements augmentant les performances.

Ces modèles sont, en général très performants dans des environnements dynamiques lorsqu'ils impliquent plusieurs robots. L'aspect réactif entraîne une limitation au niveau de la complexité des actions à réaliser puisqu'il faut rester à des échelles de temps quasi-réactives.

#### 1.2.3.4.4. Terzopoulos et la simulation comportementale

Terzopoulos a initié le modèle général utilisé en simulation comportementale [TT94]. Un environnement virtuel est simulé pour créer des comportements de recherche de nourriture, de prédation et d'accouplement. Ce système est composé de trois sous systèmes responsables de la perception, de la sélection du comportement et du contrôle moteur des entités. Des senseurs virtuels fournissent des informations sur la dynamique de l'environnement. Le système comportemental établit un lien entre la perception et le système moteur en déterminant les paramètres de contrôle des actions en fonction des perceptions couplées avec un générateur d'intention.

Le système moteur comprend un modèle dynamique de l'entité accompagné d'un ensemble de contrôleurs dédiés à la production d'un mouvement spécifique. Il décompose un ordre de haut niveau (tourner à gauche par exemple) en activations des contrôles moteurs nécessaires pour réaliser cette action (muscles).

Le résultat présente les évolutions de poissons, dans des fonds sous-marin, établies à partir de contractions musculaires, conséquence de la perception de l'environnement.

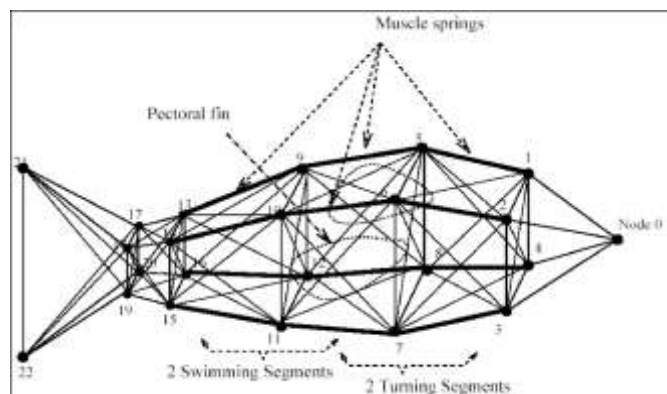


Figure 1.10 : Modèle dynamique masse/ressort d'un poisson de Terzopoulos

### 1.2.3.4.5. Approche basée théorie des schèmes de R. Arkin

Les théories des schèmes et des réseaux neuronaux correspondent aux deux principaux modèles informatiques utilisés pour représenter le fonctionnement du cerveau. Le schème est défini comme étant l'unité de base du comportement: il encapsule à la fois des connaissances lui permettant de réagir et un processus correspondant à la façon dont il doit réagir. Le comportement global émerge du contrôle des activités concurrentes des différents schèmes.

En 1998, R.Arkin propose une approche basée sur les schèmes moteurs, appliquée à la navigation de robots autonomes [Ark98]. La particularité de cette approche est la représentation des schèmes moteurs selon un format unique: le vecteur d'action. Ce vecteur est généré selon des méthodes à base de champs de potentiels considérant les cibles comme des attracteurs et les obstacles comme étant répulsifs.

La coordination entre les schèmes moteurs se fait alors simplement en additionnant l'ensemble des vecteurs calculés par les schèmes actifs, après multiplication de chaque vecteur par un poids dynamique associé à chaque schème. Il en résulte un unique vecteur d'actions correspondant à la direction que le robot doit suivre. Ainsi, chaque unité comportementale participe à l'émergence du comportement global.

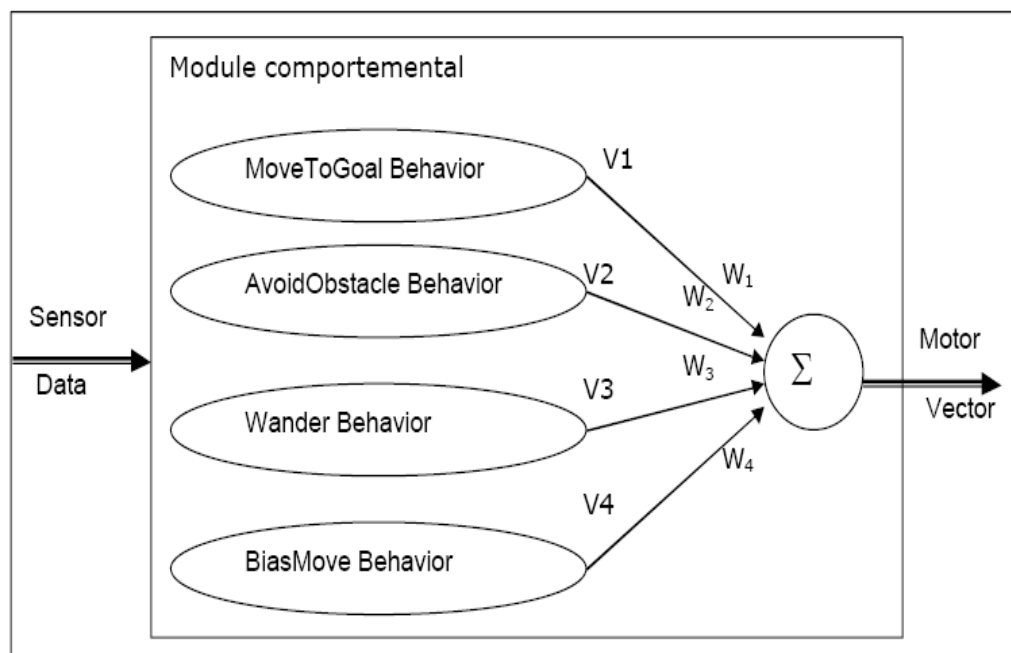


Figure 1.11 : Contrôle comportemental à base de schèmes moteurs

Un schème moteur génère un vecteur d'action à partir des informations fournies par les schèmes de perception. Selon les informations perçues dans l'environnement, chaque schème peut créer des stimuli utiles au schème moteur auquel il est attaché.

Le principal inconvénient de cette méthode est lié aux minimums locaux même si de nombreuses techniques permettent de limiter les risques de se retrouver dans une solution non optimale. Cette problématique est fortement liée à des problèmes de déplacement de robots autonomes et peut difficilement s'appliquer à d'autres domaines.

#### 1.2.3.4.6. Approche ascendante de P. MAES

L'architecture ascendante de P. Maes [Mae89] s'appuie sur une sélection de l'action qualifiée d'ascendante. A partir de comportements de base d'un agent (s'approcher d'une source, boire...), on établit un réseau non hiérarchique de nœuds d'actions. Un seul nœud peut être exécuté simultanément sachant que, lors de l'activation, plusieurs nœuds peuvent être sélectionnés. Le choix se porte alors sur le nœud de niveau le plus élevé, ce niveau décroissant automatiquement à chaque pas de réaction de l'agent.

Un nœud perçoit l'environnement de l'agent sous forme de pré-conditions correspondant à la présence ou à l'absence de certaines caractéristiques de l'environnement au moment de leur évaluation. Un nœud est éligible seulement si toutes ses pré-conditions sont satisfaites, alors que tous les nœuds participent au choix du nœud à exécuter. Chaque nœud calcule son niveau d'activation en fonction des motivations de l'agent et de sa connexion avec d'autres nœuds. Ces connexions permettent de propager une excitation ou une inhibition, c'est-à-dire d'augmenter ou de réduire le niveau d'activation d'autres nœuds.

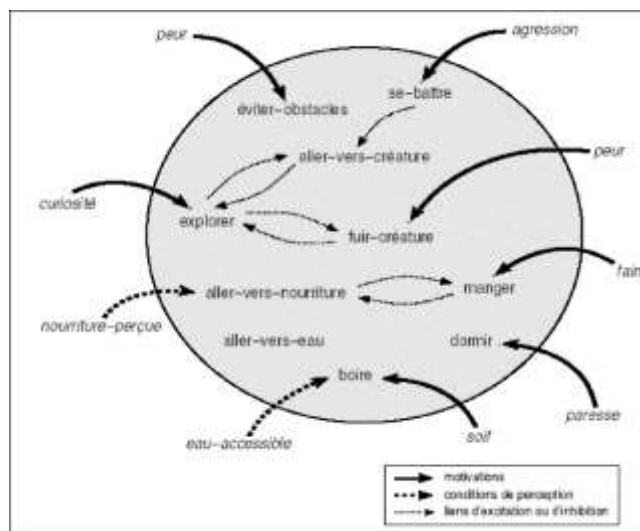


Figure 1.12 : Exemple d'architecture ascendante de P. MAES

Ce mécanisme est intéressant lorsqu'il faut interrompre un comportement en situation d'urgence ou pour profiter d'opportunités. Néanmoins, il ne gère pas les comportements conflictuels en particulier, car aucun mécanisme ne permet de combiner les actions des comportements.

#### 1.2.3.4.7. Humains virtuels de N. Badler

Le système Jack [BPW93] s'appuie sur la planification de tâche et la simulation biomécanique pour animer des humains virtuels.

Ce modèle se décompose en deux parties: la première partie utilise un réseau d'automates qui contrôlent des comportements de haut niveau. La deuxième partie se concentre sur les capacités motrices qui manipulent la géométrie d'un agent. Cette couche de bas niveau est gérée par une boucle réactive appelée SCA (Sense Control Action).

B Reich [Rei97] a étendu ce modèle en se concentrant sur la locomotion et en décomposant la boucle réactive en un moteur d'animation et un ensemble de comportements de bas niveaux appelés comportements instantanés. A chaque comportement est associée une fonction d'évaluation qui est utilisée pour déterminer l'action qui sera choisie. Ainsi à chaque pas de la simulation, le système évalue pour chaque état possible de la simulation au pas de temps suivant, une valeur correspondant à l'efficacité de l'action choisie. Pour cela, il suffit de sommer les valeurs d'évaluation de chacun des comportements instantanés.

On retrouve deux types de comportements instantanés: Les comportements de niveau 0 correspondant aux comportements réflexes (évitement d'obstacle) et des fonctions permettant de satisfaire les contraintes physiques (inertie, gravité). Les comportements de niveau 1 regroupent les comportements permettant de résoudre les buts que l'entité cherche à atteindre (essentiellement des comportements d'attraction).

Les comportements de niveau 0 sont toujours actifs alors que les comportements de niveau 1 sont activés par les comportements de plus haut niveau appelé PaTNets (Parallel Transition Networks).

Ces comportements de haut niveau se basent sur des automates exécutés en parallèles. Chaque automate représente une action que l'humain virtuel peut exécuter en complément de sa locomotion. Il possède aussi un espace permettant de mémoriser des valeurs dans le but de le guider dans ses actions. Ce modèle a été mis en œuvre dans une simulation d'humain virtuel appelé Jack. (Figure 1.13).





**Figure 1.13 : Jack dans une Ford Fiesta**

#### **1.2.3.4.8. Architecture de sélection de l'action de B. Blumberg**

Dans le but de faciliter la compréhension des modèles comportementaux, B. Blumberg a choisi de séparer comportements et compétences motrices [BDI+02]. Dans ce cas, un comportement correspond à un système régi par les objectifs qu'il cherche à atteindre. Il s'appuie pour cela sur son système sensoriel. Chaque comportement peut générer une séquence d'actions. Cette séquence sera transcrite pour modifier les caractéristiques géométriques de l'entité par un module appelé contrôleur. On associe donc à une compétence motrice un contrôleur.

Une des particularités de ce modèle est de ne gérer qu'un comportement à la fois. Cependant, les résultats des différents modules comportementaux influent sur le résultat final. La sélection du comportement s'effectue à l'aide d'une architecture connexionniste. Chaque nœud représente un comportement de base qui est relié au système perceptif et éventuellement à d'autres nœuds afin de lier les comportements entre eux. Un système d'évaluation en fonction des flux d'entrée permet d'évaluer chacun de ces nœuds. La dernière étape consiste alors à sélectionner celui ayant la valeur la plus élevée.

Le niveau perceptif s'appuie sur la notion de filtres qui permettent d'extraire les événements et les objets pertinents des informations brutes en provenance de l'environnement.

La compétence motrice est définie par le comportement actif. Les autres comportements ne proposent alors que des actions secondaires à appliquer si la première tâche s'est terminée avant la fin du temps imparti. Ils peuvent aussi proposer des méta-actions précisant la méthode pour exécuter l'action principale.

A tous ces éléments viennent se greffer deux mécanismes: la fatigue et l'inhibition permettant de trouver un équilibre entre des comportements persistants et des comportements trop opportunistes. Ainsi, la fatigue est un paramètre réduisant l'influence des comportements activés dans le but de pouvoir activer les comportements les moins prioritaires. D'un autre côté, l'inhibition va influencer les liens entre les comportements de manière à en favoriser certains au détriment d'autres.

Enfin, Blumberg a regroupé les comportements en groupes. Chaque groupe comprend tous les comportements pouvant mener à la résolution d'une problématique. Ainsi le groupe «se déplacer d'un endroit à un autre» comprend les comportements «déplacement dans différentes directions» et «Évitement d'obstacles». Certains comportements peuvent appartenir à plusieurs groupes. Dans ce cas son évaluation entraîne, soit une influence pour la sélection de l'action à l'intérieur du groupe, soit un événement sur un contrôleur.

Cette architecture complexe a permis de réaliser des animats au comportement efficace tel que le chien. Elle permet de lier efficacement simulation comportementale et animation mais reste néanmoins difficile à mettre en œuvre.

#### **1.2.3.4.9. Approche orientée émotion de P. Becheiraz**

P. Becheiraz a fait le choix de créer une structure comportementale pour animer des humains virtuels utilisant les émotions afin de leur rajouter de la crédibilité [BT98]. En complément des modules perception/comportement/action, il ajoute un module d'émotion. Le module comportemental utilise alors ce nouveau module d'émotion ainsi que la perception pour sélectionner un comportement. (Voir Figure 1.1)

Un potentiel d'émotion est calculé pour un groupe d'éléments engendrant une émotion. Si le potentiel est au dessus d'un certain seuil, sa valeur est utilisée pour calculer l'intensité de l'émotion. L'ajout d'une mémoire d'émotion permet d'obtenir des acteurs aux comportements variés car possédants des expériences différentes.

Enfin, les comportements se décomposent en une hiérarchie de comportements élémentaires pouvant être traités de manière séquentielle, ou concurrentielle. Des liens d'inhibition donnent la possibilité d'éviter les conflits lorsque deux comportements mutuellement exclusifs doivent être exécutés en parallèle.

Le module d'action autorise l'exécution parallèle et séquentielle des actions permettant une grande variété d'expression mélangeant gestes et animations faciales.

#### 1.2.3.4.10. Modèle HPTS de S. Donikian

Dans le but de fournir un modèle comportemental pour des environnements virtuels, S. Donikian a développé une architecture basée sur une hiérarchie de modules constitués d'automates parallèles: HTPS (Hierarchical Parallel Transition System) [Don01]. Elle fut mise en œuvre dans plusieurs exemples de simulation de conducteurs autonomes.

Donikian se base sur la composition de plusieurs sous-automates qui vont travailler sur la sélection de l'action. Un automate est composé d'entrées, de sorties, de paramètres de contrôle, d'une boîte aux lettres, et éventuellement d'autres automates.

Les paramètres de contrôle permettent d'influencer le comportement de l'automate. Ils sont gérés par l'automate lui-même ou par un automate extérieur.

Une fonction d'intégration permet de gérer, trier ou synthétiser l'ensemble des réponses fournies par les sous parties. On retrouve parmi les types de fonctions d'intégration, 5 types de fonctions.

Le premier type est une fonction classique manipulant des opérateurs de calcul ou de comparaison sur des types de base (entiers, réels et booléens). Le deuxième type de fonction se base sur une certaine saturation ou sur un seuillage pour réaliser une fonction de filtrage. L'utilisation de priorités permet de définir un troisième type de fonction dans le cas de réponses concurrentes. Un opérateur temporaire de retard permet d'ordonner les données. Enfin une fonction d'intégration peut être composée de plusieurs des fonctions précédemment décrites.

Dans le cas d'un automate ne comportant aucune sous-partie, le résultat ne sera que fonction des entrées, de son état interne et des paramètres de contrôle. Finalement, la boîte aux lettres permet de communiquer avec d'autres automates (qu'ils soient supérieurs d'un point de vue hiérarchique ou de simples sous parties.). Les messages correspondent, soit à des requêtes pour obtenir le statut d'un automate (actif, inactif, suspendu), soit à des ordres pour contrôler l'activité d'un automate (lancement, suspension, reprise, terminaison).



**Figure 1.14 : Ville de Rennes virtuelle peuplée par des agents basés sur l'architecture HPTS**

Le principal avantage de cette méthode réside dans sa modularité: en effet, un automate permet notamment de décrire aussi bien un module comportemental, que des capteurs ou des effecteurs. De plus, l'aspect distribué permet de répartir les automates en fonction de la puissance de calcul disponible. Cependant, leur utilisation est fortement liée à la plateforme de simulation GASP.

#### **1.2.3.4.11. Approche basé classifieurs: C. Sanza**

Dans le but de fournir un système comportemental adaptatif, Sanza [SDD99] s'est appuyé sur un système de classifieurs (aCS) pour modéliser le comportement de ses entités. Il se base donc sur une base de règles pour définir le comportement de ses entités La sélection de la règle, ou du classifieur, à appliquer s'effectue en deux phases:

La première étape présélectionne les classifieurs dont la partie condition correspond à l'état de l'environnement. La seconde permet de choisir parmi tous les candidats potentiels celui qui devrait fournir la meilleure réponse à l'état courant. Ainsi, en pondérant la force de chacune des règles avec sa spécificité (c'est-à-dire la précision avec laquelle la partie condition du classifieur décrit son environnement) et en ajoutant un bruit on obtient une valeur représentative de la règle. Une sélection stochastique permet alors de déterminer la règle choisie. Il ne reste plus qu'à activer l'effecteur déterminé par la partie action du classifieur.

En plus de la mécanique de sélection de l'action, ce modèle rajoute un élément de coordination entre entités: La communication, et par conséquent l'échange de règles, permet de profiter de l'expérience de plusieurs entités simultanément afin d'obtenir un système plus robuste.

Il ajoute également la possibilité de créer un dialogue entité/utilisateur au moyen d'un générateur syntaxique.

L'aspect adaptatif et collectif du modèle a permis la réalisation d'une simulation (une séquence d'un match de football) dans laquelle chaque entité est contrôlée par ce système. Les résultats sont pertinents et accompagné d'une forte émergence de comportements collectifs. Cependant, ce système présente l'inconvénient de n'activer qu'un effecteur à la fois, limitant ainsi les applications du modèle. De plus, il n'offre pas la possibilité d'intégrer d'autres méthodes de contrôle des entités dans le but de pouvoir comparer les résultats face à des situations prédéterminées.

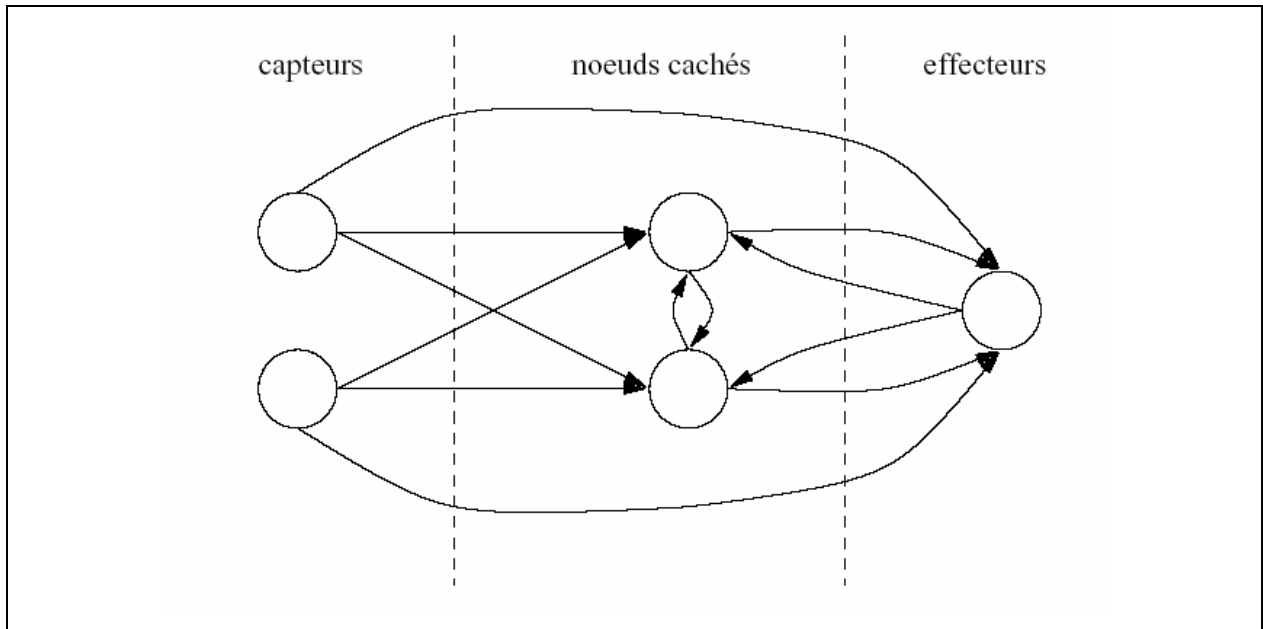
#### **1.2.3.4.12. Architectures basées sur les réseaux de neurones**

Un réseau de neurones artificiel est constitué par l'interconnexion de cellules élémentaires: les neurones formels. Le neurone formel, à l'instar de son homonyme biologique, a pour unique tâche de transmettre un influx nerveux sous certaines conditions définies par une fonction de transfert. Le principe de fonctionnement du neurone formel de McCulloch et Pitts est le suivant [MP43]: Les potentiels présents sur chaque entrée du neurone sont additionnés puis une fonction d'activation décide de propager le signal ou non sur son unique sortie. C'est l'organisation du réseau de neurones qui permet à l'ensemble d'exprimer tout son potentiel. En simulation comportementale, les architectures sont essentiellement basées sur le modèle de la perception multicouche de Marvin Minsky et Seymour Papert [Rey99]. Dans ce dernier, les neurones sont organisés en couches: une couche d'entrée, une couche de sortie et une couche cachée lui permettant de résoudre les problèmes linéairement séparables. La modification des poids (représentant leur force d'activation) de chaque nœud peut se faire dynamiquement à l'aide d'un système d'apprentissage (une méthode d'apprentissage par renforcement ou, plus rarement, un algorithme génétique).

Bien qu'intéressante par ses capacités réactives importantes, l'approche stimulus-réponse des réseaux de neurones est d'un niveau d'abstraction faible et, ne permet donc, généralement que la modélisation de comportements instinctifs (généralement liés aux mouvements d'une entité virtuelle ou à son déplacement dans l'environnement). De plus, cette technique manque de souplesse d'adaptation, la modification de l'environnement ou d'une réaction à un stimulus nécessitant un nouveau calcul complet des paramètres du réseau. Enfin, les réseaux obtenus par adaptation dynamique à l'environnement sont souvent complexes et, par conséquent,

relativement opaques si on souhaite analyser en profondeur les mécanismes du comportement généré.

### Les Sensor Actuator Network (SAN)



**Figure 1.15 : Sensor Actuator Network**

Van de Panne utilise des réseaux de neurones, baptisés Sensor Actuator Network pour animer des créatures virtuelles sur des terrains accidentés. L'utilisateur spécifie la configuration de la créature (effecteurs et capteurs). Les capteurs sont connectés à la fois à la couche cachée et aux effecteurs. Le poids d'activation des différents nœuds est calculé à l'aide d'un algorithme stochastique. Les résultats obtenus sont des créatures capables de se déplacer en fonction des mouvements générés par les activations successives des effecteurs en fonction de l'environnement (certains rampent, d'autres sautent, marchent, ...).

#### 1.2.4. L'action

Basé sur l'information perceptuelle, le mécanisme comportemental d'un acteur détermine les actions à exécuter. Les actions peuvent avoir plusieurs degrés de complexité. Un acteur peut se développer dans son environnement ou bien agir réciproquement avec l'environnement ou encore communiquer avec d'autres acteurs.

Les actions sont exécutées en employant une architecture de mouvement commune. Le module d'action gère l'exécution des actions employées par un comportement en animant un modèle d'homme générique basé sur une hiérarchie de nœud. Il permet l'exécution simultanée

ou séquentielle d'actions en gérant des transitions lisses entre des actions finales et des actions amorçantes [BMT95]. Une boucle comportementale conduit l'animation, son rôle est de mettre à jour l'état du monde virtuel. Le but est de déclencher est d'arriver à la solution.

A chaque itération, le temps est incrémenté, le monde virtuel est mis à jour avec en particulier une mise à jour de l'état de chaque objet et acteur. Dans le cas d'un acteur, la perception est d'abord exécutée, après ses émotions sont produites avant que son comportement et ses actions ne soient exécutés.

### **1.2.5. La Mémoire**

La mémoire est d'habitude définie comme le pouvoir ou le processus de reproduction ou de rappel de ce qui a été appris et conservé, particulièrement par les mécanismes associatifs.

La mémoire est aussi le dépôt pour des informations apprises et à conserver, générées à partir de l'activité d'un organisme ou d'une expérience.

La mise en œuvre de la mémoire pour un acteur n'est pas très complexe, comme la mémoire est déjà un concept clef en informatique. Par exemple, dans [NT97], les auteurs proposent une troisième mémoire visuelle globale qui permet à un acteur de retenir l'environnement, de le percevoir et de s'adapter à ses changements.

## **1.3. Domaines d'application de la simulation comportementale**

Les domaines d'application de la simulation comportementale sont variés et s'étendent des domaines ludiques tels que les jeux vidéo aux domaines de la médecine et des soins des phobies.

### **1.3.1. Jeux vidéo**

Avec l'évolution de la puissance de calcul des ordinateurs, allant de paire avec la puissance des cartes graphiques, une bonne qualité de rendu graphique peut désormais être obtenue en utilisant relativement peu de temps processeur. Cette évolution permet aux concepteurs de jeux vidéo de se concentrer sur le comportement des personnages avec lesquels les joueurs peuvent interagir. Deux exemples notables sont les jeux de créatures de la société Cyberlife, et Black and White de la société Lionhead, qui proposent des mondes peuplés de créatures autonomes. Ces créatures peuvent prendre des décisions seules, l'intervention du joueur, qui peut cependant, dans certaines circonstances influencer sur leur comportement est de tenter de leur apprendre des choses. Le développement de la simulation comportementale a fait

naître une nouvelle tendance: la fiction interactive [CCM02]. L'idée est, ici, de fournir les grandes lignes d'un scénario, tout en laissant le joueur libre d'interagir à son gré avec des entités semi-autonomes. Ces entités suivent la ligne directrice du scénario mais peuvent prendre des décisions pour interagir «intelligemment» avec le joueur. Cela donne une grande sensation de liberté, tout en pouvant générer des situations imprévues faces aux actions de l'utilisateur.

Dans le domaine de jeu, outre la qualité des animations et des graphiques, la crédibilité et la cohérence du comportement sont primordiales pour le réalisme et l'implication du joueur.

### 1.3.2. Films et effets spéciaux

La simulation comportementale a fait son apparition dans le monde de la cinématographie et des effets spéciaux depuis déjà un certain temps. La tendance actuelle est d'accroître son utilisation, à travers le développement d'outils de modélisation du comportement pour les logiciels de production d'animation en 3D. Cette tendance s'explique par un besoin de productivité accru, à travers des scènes de plus en plus complexes mais allant de paire avec des exigences de qualité croissantes. L'un des meilleurs exemples actuels est le deuxième volet de l'adaptation cinématographique du *seigneur des anneaux*. Pour la grande bataille du *gouffre de Helm*, un outil spécifique: MASSIVE, a été utilisé pour reproduire les comportements de foule et modéliser la composante décisionnelle des entités. Dans le même domaine, divers outils sont désormais disponibles, parmi lesquels AI-Implant de la société BIOGRAPHICS, qui peut être utilisé sous la forme de module pour 3D studio Max et Maya, ou bien encore les modules RTK-Crowd et RTK-Behavior de SOFTIMAGE. En termes de conception, ces outils offrent de grands avantages:

- Le travail du concepteur est simplifié par la gestion automatisée des comportements de groupe;
- Les modèles décisionnels permettent de générer automatiquement des animations adaptées au contexte;
- Les scènes paraissent plus réelles car elles sont plus variées.

Grâce à l'automatisation, ces techniques permettent de décrire, de manière implicite, des scènes d'une complexité inégalable grâce à l'aide d'outils dédiés.



### 1.3.3. Validité ergonomique des sites

La création de lieux publics pose des problèmes d'ordre ergonomique, autrement dit, relatifs à la qualité de leur utilisation. Des problèmes peuvent se poser quant à la bonne navigation à l'intérieur des lieux, la lisibilité des divers panneaux de direction, ou lors de situations de panique [HFV00]. L'animation comportementale, à travers des modèles se basant sur l'analyse du comportement humain, peut être utilisée pour effectuer des validations sur des maquettes virtuelles. Les éventuels problèmes peuvent alors être détectés et corrigés avant la construction des divers aménagements. Dans ce cadre, son utilisation offre des atouts qui sont d'ordres sécuritaire et ergonomique mais aussi d'ordre économique.

### 1.3.4. Mise en situation

L'interaction avec des agents autonomes, à travers la réalité virtuelle, permet de mettre un être humain en situation dans le cadre d'un scénario. Ces capacités peuvent être utilisées dans un cadre pédagogique où l'interaction avec des humanoïdes intelligents aura pour conséquences d'augmenter la rapidité d'apprentissage par l'intermédiaire de moyens d'interaction plus proches de la réalité [LTG+00].

Un autre domaine d'application concerne l'armée où la définition du comportement d'humanoïdes permet de tester des scénarios catastrophes ou de mettre les soldats en situation [TRG+03]. Dans le domaine hospitalier, l'animation comportementale, couplée à la réalité virtuelle, peut être utilisée dans le cadre de soins aux personnes phobiques [KV03]. L'idée consiste à plonger la personne, par l'utilisation de la réalité virtuelle, dans un milieu qui déclenche sa phobie. Dans le cadre du soin de l'agoraphobie, par exemple, les simulations de foules s'avèrent utiles, pour peupler les environnements virtuels et fournir au patient un milieu réaliste.

## 1.4. Conclusion

Ces domaines d'applications variés, justifient les recherches effectuées en simulation comportementale, et particulièrement le besoin d'établir un lien avec les études effectuées sur le comportement humain. Les modèles proposés, pour permettre d'obtenir un réalisme suffisant, doivent généralement prendre en compte les caractéristiques propres au comportement humain.

Nous avons présenté au niveau de ce chapitre la modélisation des propriétés des humains virtuels ainsi nous avons cité la plupart des modèles comportementaux utilisés. Dans le chapitre suivant, nous allons présenter un comportement très utilisé au niveau de la simulation comportementale : c'est le comportement *de recherche de chemin*.