

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohammed Khider-BISKRA-
Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie
Département d'Informatique



N°d'ordre :.....

Série :.....

Mémoire

Présenté en vue de l'obtention du diplôme de Magister en Informatique

Option: **Synthèse d'Images et Vie Artificielle (SIVA)**

Sous le thème :

Animation des phénomènes collectifs cellulaires par modèle physique particulaire

Par :

Mr. CHATRA Mohamed

Devant le jury :

DJEDI Nouredine	Pr. Université de Biskra	Président.
CHERIF Foudil	MCA Université de Biskra	Rapporteur.
BABAHENINI Med Chaouki	MCA Université de Biskra	Examineur.
KAZAR Okba	MCA Université de Biskra	Examineur.

Soutenu le :06/10/2010

Dédicaces

« Louange à Dieu, le seul et unique »

A mes très chers parents...

Témoignage d'affection et de grande reconnaissance,

Que Dieu les garde pour moi

A toute ma famille,

Mes très chers frères et sœurs,

Mes chers Grands-parents,

Mes Oncles et Tantes

A tous mes amis,

A tous ceux que j'aime

A tous ceux qui ont participé de loin ou de près à la réalisation de ce travail.

A mon pays,

Avec l'expression de tous mes sentiments de respect,

Je dédie ce modeste travail.

CHATRA Mohamed

Remerciements

Je remercie tout d'abord le bon dieu pour m'avoir donné le courage et la santé pour accomplir ce travail.

*Je tiens à remercier vivement mon directeur de mémoire, le Docteur **CHERIF Foudil** pour m'avoir proposé ce sujet et dirigé mon travail ainsi que pour la marque de confiance qu'ils m'ont manifesté.*

*Je remercie le président de jury monsieur **DJEDI Nouredine**, professeur à l'université de Biskra qui m'a fait l'honneur de présider le jury.*

*Je remercie également monsieur **BABAHENINI Mohamed Chaouki**, maitre de conférences à l'université de Biskra pour m'avoir fait l'honneur d'évaluer mon travail.*

*Je remercie aussi monsieur **KAZAR Okba**, maitre de conférences à l'université de Biskra pour l'intérêt qu'il a bien voulu porter à ce travail en acceptant d'être examinateur.*

Enfin, que tous ceux qui mon ont aidé et encouragé de près ou de loin dans la concrétisation de ce travail, trouvent ici ma gratitude et mes sincères remerciements.

Résumé

Ce travail s'inscrit dans le cadre de l'animation des phénomènes collectifs. Ces phénomènes sont classés dans les effets complexes, et ils sont un sujet de prédilection pour le concept d'émergence. Une grande effervescence règne aujourd'hui sur la modélisation et la simulation de ces modèles, où dans des domaines applicatifs aussi divers que l'architecture, la spécification des nuisances urbaines, le trafic autoroutier ou la biologie. La modélisation des comportements collectifs se divise en deux classes : la première classe concerne les comportements collectifs non délibératifs. Et la classe concerne les comportements collectifs coopératifs qui sont désignés par « action à plusieurs ».

Nous nous intéressons dans notre étude à la première classe qui est celui d'un ensemble d'individus ayant des buts individuels se trouvant en interaction. Apparaissent alors des évolutions structurées de ce collectif. C'est cet effet qui est strictement nommé « foule ».

Les sciences sociales ne disposent pas encore d'une typologie précise de ces formes. La simulation joue, alors, le rôle d'outil de spécification de comportements collectifs. Elle permet, ensuite, l'exploration des conditions de leur émergence. Enfin, elle permet, aussi, d'agir sur elles dans le retour au réel.

Nous proposons un modèle microscopique d'une foule d'humains virtuels. Principalement, nous abordons deux problématiques : d'une part, nous proposons un modèle basé sur un système multi-agents, et d'autre part nous utilisons les réseaux flous pour la description des comportements. Les réseaux flous permettant à l'humain virtuel d'effectuer des engagements dans un environnement virtuel inconnu. Nous avons utilisé l'outil *Massive* qui facilite les tâches de modélisation des comportements et la simulation.

Les résultats obtenus sont très satisfaisants dans le cas d'évitement de collision et l'évitement d'obstacles.

Les mots clés : Animation comportementale, animation de foule, la logique floue, les systèmes multi-agents.

Abstract

This work is part of the emergent collective phenomena. These phenomena are classified into the complex effects, and they are a favorite subject for the concept of emergence. Much excitement prevails today on modeling and simulation of this models, where in applications areas as various as architecture, specification of urban stress, high traffic or biology. Modeling collective behavior is divided in two classes: the first class, non-deliberative emergent collective behavior and the second class for collective cooperative behavior those are designed as "action-to-many".

We are interested in our study by the first class which is set of a group of individuals with individual goals found in interaction. Then appear structured evolutions of this group. This effect is strictly called "crowd".

Social science does not yet have a precise typology of these forms. The simulation is then plays the role of tool of specification collective behavior. It allows then exploration of the conditions of their emergence; it allows also acting on them in the return to reality.

We propose a microscopic model of a crowd of virtual humans. Principally, we tackle two main issues: firstly, we propose a model based on a multi-agent system, and secondly we use fuzzy networks for description of behavior. The fuzzy networks allowing the virtual human to make commitments in an unknown virtual environment. We used Massive tool that facilitates the tasks for modeling behaviors and simulation.

The results obtained are very satisfactory in the case of collision avoidance and obstacle avoidance.

Keywords: Behavioral animation, crowd simulation, fuzzy logic, virtual environment, multi-agent systems.

ملخص

هذا العمل ينتمي إلى دراسة حركة الظواهر الجماعية. هذه الظواهر مصنفة ضمن الظواهر المعقدة وهي الموضوع المفضل لظهور مفهوم البروز. ساد الكثير من الحماس اليوم من أجل نمذجة ومحاكاة هذه الظواهر، والتي اكتسحت مجالات متنوعة مثل الهندسة المعمارية، وصف المناطق الحضرية الكثيفة، طرق المرور السريع و حتى البيولوجيا. نمذجة السلوك الجماعي ينقسم إلى قسمين : السلوكات الجماعية الغير تعاونية و السلوكات الجماعية التعاونية.

نحن مهتمون في دراستنا هذه بالحالة الأولى و التي تمثل مجموعة من الأفراد كل بهدفه متفاعلون فيما بينهم. حيث تظهر تطورات على بنية تشكيل هذه المجموعة. هذا الأثر على الشكل يسمى "الحشد". العلوم الاجتماعية ليس لديها حتى الآن تصنيفا دقيقا لهذه الأشكال. المحاكاة تلعب دور لوصف أداة السلوك الجماعي. حيث تسمح باستكشاف ظروف ظهوره هذه الأشكال. وأخيرا، فإنها تسمح أيضا الرجوع للعمل على واقع ملموس.

اقترحنا في دراستنا هذه نمودجا لحشد افتراضي من البشر. في المقام الأول، تناولنا قضيتين من القضايا المثارة، أولا، اقترحنا نموذج يستند إلى الأنظمة متعددة الوكلاء، و الثانية شبكات غامضة لوصف مختلف السلوكيات. الشبكات الغامضة سمحت للشخص الافتراضي بتقديم تصرفات في محيط افتراضي مجهول. استعملنا برنامج ماسيف Massive الذي سهل لنا عمل نمذجت السلوكيات و المحاكاة. النتائج التي تم الحصول عليها كانت جد مرضية من حيث تفادي الاصطدام بين الأشخاص الافتراضيين و من حيث تجنب الحواجز.

الكلمات المفتاح: الحركة السلوكية، محاكاة الحشود، المنطق الغامض، المحيط الافتراضي، الأنظمة المتعددة الوكلاء.

Table des matières

Introduction générale	18
Animation comportementale	22
1. Contexte et définition	22
2. Domaine d'application	23
2.1 Jeux vidéo.....	23
2.2 Films et effets spéciaux	24
2.3 Validité ergonomique des sites	25
2.4 Mise en situation	25
3 Les premiers travaux	26
4 La boucle de l'animation comportementale	28
5 Conclusion.....	31
Foule d'humains virtuels	33
1. Introduction	33
2. L'humain virtuel.....	33
2.1 Propriétés des humains virtuels.....	33
2.1.1 La perception.....	33
2.1.2. L'émotion.....	34
2.1.3. Le comportement	35
2.1.4. L'action	36
2.1.5. La mémoire	36
2.2 Classification des humains virtuels	36
2.2.1 Les Avatars.....	37
2.2.2 Les acteurs guidés	37
2.2.3 Les acteurs autonomes	37
2.2.4 Les acteurs perceptifs et interactifs.....	38
2.3. Animation d'humain virtuel	38
3. La foule	39
3.1. Comportement de foule.....	39
3.2. Animation de foule.....	40
4. conclusion.....	43

Différentes techniques de modélisation de foules	45
1. Introduction	45
2. Les aspects sociologiques et d'interaction de foule	45
3. Les deux approches de foule	46
3.1 Les modèles microscopiques.....	46
3.1.1 Modèles de force sociale.....	47
3.1.2 Les modèles d'automates cellulaires	47
3.1.3 Les Modèles basés sur les règles.....	51
3.1.3.1 Les « Boids » de Craig Reynolds	51
3.1.3.2 Les systèmes de particules	52
3.1.3.3 Mouvements contrôlés par les lois de la dynamique.....	52
3.1.3.4 Modélisation de foules avec facteur psychologique	55
3.1.3.5 Méthode des « Règles locales »	56
3.2 Les modèles macroscopiques	58
3.2.1 Les modèles de régression	58
3.2.2 Les modèles de choix de route	59
3.2.3 Les modèles de queue (file d'attente)	59
3.2.4 Gaskinetics	59
3.2.5 Modèle hydraulique	60
3.2.6 Modèle de Togawa.....	60
3.2.7 Modèle de Melinek et Booth.....	62
3.2.8 Modèle de Paule.....	63
3.2.9 Le modèle de Predtechenskii et Milinskii	64
3.2.10 Le modèle de Takahashi, Tanaka, et Kose.....	64
4. Les systèmes récents de simulation piétonniers	65
5. classifications des méthodes de foule.....	71
6. Synthèse sur les modèles de foule.....	74
7. Conclusion.....	76
Massive : outil de modélisation et simulation de foule basée sur la logique floue	78
1. Introduction	78
2. Le logiciel Massive	78
2.1 Fonctionnement.....	79
2.2 Les versions de Massive.....	80

2.2.1 Massive Prime	80
2.2.2 Massive Jet	80
2.3 Interface de Massive	81
3. La logique floue	84
3.1 Domaines d'application de logique floue	85
3.2 Principe	85
3.3 Les sous-ensembles flous	85
3.3.1 Concept	85
3.3.2 Définition d'un sous-ensemble flou	86
3.3.3 Caractéristiques d'un sous-ensemble flou	87
3.3.4 Combinaison de plusieurs entrées	87
3.3.5 Opérateurs flous	88
3.3.6 Opérateurs linguistiques	88
3.4 Le raisonnement en logique floue	89
3.4.1 Les implications floues	89
3.4.2 Les règles floues	89
3.4.3 Conjonction de propositions floues	90
3.4.4 Deux cas particuliers : les méthodes de Mamdani et de Larsen	90
3.5 La défuzzification	91
3.5.1 La méthode du maximum	91
3.5.2 La méthode du centre de gravité	92
3.6 Algorithme générale de fuzzification	92
3.7 La logique floue dans Massive	94
4. Conclusion	96
Modèle d'animation	98
1. Introduction	98
2. Observations sur le comportement piétonnier	98
3. Architecture de notre modèle	100
4. Structure du modèle multi-agents adopté	101
4.1 L'environnement	101
4.2 Hypothèses du problème	102
4.3 Description de l'architecture	103
4.3.1 Base de connaissances	104

4.3.2 Module de perception.....	105
4.3.3 Module de réaction.....	105
4.3.4 Module d'action	106
4.3.4.1 Comportement pour sélection du chemin.....	106
4.4 Structure du contrôleur flou	107
4.4.1 Variables linguistiques	107
5 Etude de l'influence de l'environnement individuel	108
5.1 Comportement d'éviter la collision.....	108
5.2 Comportement d'évitement d'obstacles	109
5.2.1 Définition des variables.....	110
5.2.2 Partitionnement des domaines de variables	111
5.2.3 Définir l'ensemble des règles.....	113
Simulation et résultats	116
1. Introduction	116
2. Méthode de simulation	116
2.1 Environnement	119
2.2 Humain virtuel (agent)	121
3 modélisations des comportements.....	122
3.1 Comportement d'évitement de collision	123
3.2 Comportement d'évitement d'obstacles.....	124
4. Simulation d'un phénomène collectif « foule »	125
5. Rapport des résultats	127
6. Conclusion.....	129
Conclusion générale	131
Bibliographie	135

Table des figures

Figure 1.1 : Le jeu vidéo « Les Sims 2 » de Electronic Arts [Pan08].	23
Figure 1.2 : L’environnement virtuel du jeu en ligne « Second Life » [Pan08].	24
Figure 1.3 : Le film d’animation « Mulan » de Disney [Pan08].	25
Figure 1.4 : La simulation d’entraînement Firefighter Training (IRIT) [Pan08].	25
Figure 1.5 : Visite virtuelle de la ville de Grenoble (Artesia) [Pan08].	26
Figure 1.6 : Visite virtuelle de la cité des sciences et de l’industrie (IRISA) [Pan08].	26
Figure 1.7 : la boucle d’animation comportementale [Don04].	29
Figure 1.8 : Hiérarchisation des comportements illustrée par A. Newell dans sa pyramide comportementale [Don04].	31
Figure 2.1 : Structure du Modèle comportemental [Fou06].	33
Figure 2.2 : Acteurs autonomes (Design: Mireille Clavien).	38
Figure 2.3 : interaction entre deux acteurs.	38
Figure 2.4 : Système à base de particules [Hel00].	41
Figure 2.5 : les trois règles de Rynolds [Rey99].	42
Figure 2.6 : Projet de MAGS [Mou03].	43
Figure 3.1 : La simulation de modèle de Helbing avec 10.000 individus [Hel00].	47
Figure 3.2 : Simulation de foule par AC 2D structure étant à la base de grille [Tec01].	49
Figure 3.3 : Carreaux de flux et Champs de Vitesse.	49
Figure 3.4 : interface du logiciel permettant de choisir des carreaux prédéfinis.	50
Figure 3.5 : Deux applications du système Flow Tiles.	50
Figure 3.6 : Capture d’écran d’un brouillard tourbillonnant généré à l’aide de champs de vitesses dynamiques.	50
Figure 3.7 : Les 3 règles du système de Reynolds (le point rouge représente le centre du système de Boids).	51
Figure 3.8 : Installation.	53
Figure 3.9 : Robots unijambistes à rapatrier dans l’enclos.	53
Figure 3.10 : Cyclistes virtuels.	53
Figure 3.11 : Installation.	54
Figure 3.12 : Forces dynamiques d’un cycliste virtuel.	54
Figure 3.13 : Simulation de l’évacuation d’un immeuble.	55
Figure 3.14 : Comportements des agents.	56
Figure 3.15 : Gestion d’une foule dans un musée.	57

Figure 3.16 : Gestion de la foule de la Figure 1 après introduction d'un élément de panique.	57
Figure 3.17 : Différents états émotionnels des individus.	57
Figure 3.18 : Différentes façon de marcher selon les émotions des agents.	58
Figure 3.19 : Simulation dans une gare.	58
Figure 3.20 : Distributions de densité pendant une évacuation dans l'exode [Gal93].	66
Figure 3.21 : Simulation d'une évacuation avec l'exode de bâtiment et de sa visualisation en différé dans 3D avec le vrExodus [Owe98].	66
Figure 3.22 : Densité et simulation de foule dans Simulex.	66
Figure 3.23 : Simulation de légion d'une station de train (légion 2003).	67
Figure 3.24 : simulation 2D par STEPS.	68
Figure 3.25 : L'interface d'Open street [Rey99].	68
Figure 3.26 : Simulation d'évacuation avec Massive WS pour la conception architecturale et l'évaluation.	69
Figure 3.27 : Navigation réactive [Tho00].	69
Figure 3.28 : Les poissons artificiels.	69
Figure 3.29 : Le système ACUMEN est appliqué aux sphères et les écoliers [All02].	70
Figure 3.30 : Les piétons autonomes [Sha05].	71
Figure 3.31 : Graphique de visibilité de syntaxe de l'espace [Tur02].	71
Figure 4.1 : le logiciel Massive [Mas08].	79
Figure 4.2 : Interface de Massive.	81
Figure 4.3 : la fenêtre scène.	82
Figure 4.4 : la fenêtre de corps (squelette).	82
Figure 4.5 : la fenêtre de cerveau.	83
Figure 4.6 : la fenêtre d'action.	83
Figure 4.7 : Caractère non graduel (a) et caractère graduel (b) du degré d'appartenance.	86
Figure 4.8 : Ensemble flou. Le niveau de gris indique le degré d'appartenance.	86
Figure 4.8 : Exemple de fonctions d'appartenance.	86
Figure 4.9 : Caractéristiques d'un sous-ensemble flou A.	87
Figure 4.10 : Fonctions d'appartenance 'distance'.	89
Figure 4.11 : Méthode de maximum.	91
Figure 4.12 : Inconvénient de la méthode du maximum.	92
Figure 4.13 : Méthode du centre de gravité.	92
Figure 4.14 : Les étapes de l'algorithme d'inférence flou.	93
Figure 4.15 : Schéma de principe d'un contrôleur flou.	94

Figure 4.16 : Réseau flou contrôlé la tendance d'orientation pour un agent.	95
Figure 5.1 : L'architecture de notre modèle.	100
Figure 5.2 : Modélisation d'environnement par Massive.	101
Figure 5.3 : modélisation de l'environnement dans Massive.	102
Figure 5.4 : L'environnement de notre simulation.	102
Figure 5.5 : Architecture de notre modèle.	104
Figure 5.6 : La base de connaissance de l'agent.	104
Figure 5.7 : Structure de module perception.	105
Figure 5.8 : Le module stimulus/réponse du module de réaction.	105
Figure 5.9 : Module d'action et ses Interactions.	106
Figure 5.10 : Contrôleur flou multi-E/S.	107
Figure 5.11 : Configuration du contrôleur flou.	107
Figure 5.12 : Répartition floue de la variable Position (P).	108
Figure 5.13 : Répartition floue de la variable Distance (D).	108
Figure 5.14 : Répartition de l'univers de la variable Orientation (O).	108
Figure 5.16 : Eviter les obstacles.	109
Figure 5.15 : Eviter la collision. Les distances capturées dépendent de la position des HV réparties dans le champ de vision.	109
Figure 5.17 : Les champs de vision de l'agent.	111
Figure 5.18 : Représentation de contrôleur flou (D : la distance. P : la position).	111
Figure 5.20 : fonction d'appartenance de distance.	112
Figure 5.20 : Les deux sens de l'inclination.	112
Figure 5.21 : Les étapes de fuzzification.	113
Figure 6.1 : Structure de simulation.	117
Figure 6.2 : Les éléments de notre scène défini en Massive.	118
Figure 6.3 : Les sources lumineuses.	118
Figure 6.4 : Les types de rendu.	118
Figure 6.5 : Les positions des caméras.	119
Figure 6.6 : Localisation des objets.	120
Figure 6.7 : Les obstacles statiques.	120
Figure 6.8 : Modélisation de l'humain virtuel.	121
Figure 6.9 : L'éditeur d'actions.	121
Figure 6.10 : Une vue globale.	122
Figure 6.11 : Les besoins d'un humain virtuel.	122

Figure 6.12 : Réseau flou « éviter la collision ».....	123
Figure 6.13 : Réseau flou « éviter les obstacles ».....	124
Figure 6.14 : Le point de départ.....	125
Figure 6.15 : Comportement d'évitement de collision.....	126
Figure 6.16 : Comportement d'évitement d'obstacles.....	127
Figure 6.17 : Rapport de simulation.....	128
Figure 6.18 : L'orientation de l'humain virtuel.....	129

Liste des tableaux

Tableau 3.1 : Table pour définir les automates cellulaires.....	48
Tableau 3.2 : Les différentes approches dans le domaine [Nur08]......	72
Tableau 3.3 : Comparaison de différents systèmes pour l'animation de grands groupes.....	75
Tableau 4.1 : Les méthodes de Mamdani et de Larsen.	90

Introduction générale

« Il ne faut pas juger de la hauteur d'une montagne
avant d'en avoir atteint le sommet »

Introduction générale

Les ordinateurs ont longtemps été utilisés pour produire des images de synthèses, représentant des mondes virtuels statiques issus de l'imagination des concepteurs. Dans la continuité, l'animation a fait son apparition. Les concepteurs, au travers d'outils dédiés, pouvaient alors animer des personnages et des objets de manière similaire à celle utilisée dans les dessins animés. Allant de pair avec l'évolution de la puissance de calcul, les animations sont devenues de plus en plus complexes à gérer, avec des exigences de qualité accrues. Pour créer des environnements toujours plus réalistes, la notion de simulation physique est utilisée. Son rôle est d'automatiser un certain nombre d'animations sur la base de calculs modélisant des règles physiques, similaires à celles régissant notre propre environnement. Parmi les animations qui utilisent les modèles physiques on a l'animation collective.

Les phénomènes collectifs sont des phénomènes complexes qui sont un sujet de prédilection pour le concept d'émergence. Une grande effervescence règne aujourd'hui sur ce type de sujets, tant dans les sciences humaines et sociales, que dans les sciences de la modélisation et de la simulation, ou dans des domaines applicatifs aussi divers que l'architecture, la spécification des nuisances urbaines, le trafic autoroutier ou la biologie.

On distingue deux grands cas de modélisation des comportements collectifs humains:

- 1- ***les comportements collectifs non délibératifs***: pour ce cas est celui d'un ensemble d'individus ayant des buts indépendants (buts individuels) se trouvant en interaction. Apparaissent alors des évolutions structurées de ce collectif. C'est cet effet qui est strictement nommé "foules". Les effets de panique, bien qu'encore très mal définis appartiennent à cette catégorie.
- 2- ***les comportements collectifs coopératifs (qu'ils soient ou non concurrentiels ou compétitifs)***: pour ce cas est celui où un ensemble d'individus interagissent pour exécuter une tâche ensemble (jouer au football, manifestations, etc...). Il est désigné par "action à plusieurs", contrairement au premier cas appelé "action de plusieurs".

Nous nous intéresserons principalement au premier cas.

Les sciences sociales ne disposent pas encore d'une typologie précise de ces formes. La simulation joue alors le rôle d'outil de spécification de comportements collectifs. Elle permet ensuite d'explorer les conditions de leur émergence, puis d'agir ensuite sur elles dans le retour au réel. Ainsi, nous chercherons à connaître les rôles de l'environnement construit, statique

(bâtiment, places, rues) ou dynamiques (portes automatiques) dans la régulation de ces figures. Ces recherches pourront être exploitées aussi bien pour animer des sites disparus ou pour réguler des ensembles humains dans des sites actuels.

La modélisation est devenue une issue importante dans l'ingénierie et la science. Les approches traditionnelles de modélisation insistent énormément sur la précision et la description exacte des systèmes. L'utilisation des concepts mathématiques comme les équations différentielles, fonctions de transfert, ...etc., est justifiée pour les systèmes bien définis. Mais, quand la complexité augmente, ces concepts deviennent moins efficaces. Le traitement des systèmes complexes nécessite souvent la manipulation d'informations vagues, imprécises, incertaines ou à la fois imprécises et incertaines. L'être humain est compétent dans la manipulation de tels systèmes de façon naturelle. Au lieu de raisonner en termes mathématiques, l'être humain décrit le comportement du système par des propositions linguistiques. Par exemple, un conducteur peut formuler une partie de sa connaissance par : "Si la vitesse est élevée et qu'il y a un virage alors freiner".

Afin de pouvoir représenter ce type d'informations, Zadeh [Zad78] a proposé de modéliser le mécanisme de la pensée humaine par un raisonnement approximatif basé sur des variables linguistiques. Il a introduit la théorie des ensembles flous en 1965, qui constitue une interface entre les mondes linguistiques et numériques. Plus généralement, le terme de la logique floue correspond à tous les développements issus de la théorie des sous-ensembles flous.

Le comportement des humains virtuels composant une foule est caractérisée par une certaine imprécision du mouvement de ces humains virtuels durant la simulation de leurs modèles. La logique floue est utilisée dans ce travail pour formaliser cette imprécision. Durant la simulation un humain virtuel a une vision sur la distance et la position des obstacles que soient dynamiques ou statiques. Cette vision est imprécise qui engendre un mouvement imprécis de l'humain virtuel.

L'objectif de notre travail est de proposer un modèle de simulation microscopique d'une foule, en utilisant comme application la simulation de foule. Notre travail donc présente une étude du comportement collectif d'humains virtuels dans un environnement virtuel.

La nature de notre problématique nous a suggéré alors, de faire appel à un certain nombre d'approches et techniques :

- l'approche basée sur les systèmes multi-agents comme une philosophie globale de résolution.
- La logique floue comme une technique de contrôle.

- Les sous-ensembles flous comme outil ou méthode de modélisation de comportements d'humains virtuels.

Ce mémoire est organisé en deux parties. La première partie est un développement de l'état de l'art sur tous les aspects qui entrent dans le domaine de l'animation comportementale et la simulation de foule d'humains virtuels. Cette partie est organisée en trois chapitres:

- Le premier chapitre introduit l'animation comportementale avec sa boucle, son domaine d'application, ainsi que les premiers travaux dans ce domaine.
- Le deuxième chapitre présente les propriétés des piétons virtuels, caractérisés par le comportement, l'autonomie, l'adaptation, la perception, la mémorisation, l'émotion, et la conscience. Le comportement d'humain virtuel et la classification des humains virtuels seront traités par la suite.
- Le troisième chapitre est dédié à la présentation des outils développés dans ce domaine. Nous présentons aussi les différentes techniques de modélisation de foules, puis nous présentons les approches de simulation de foule macroscopique et microscopique avec leurs méthodes. Nous présentons à la fin de ce chapitre les différents systèmes de modélisation.

La deuxième partie de ce mémoire traite la réalisation et la présentation des résultats. Elle est constituée de trois chapitres:

- Le quatrième chapitre présente la plateforme de modélisation et simulation « Massive » et la technique utilisée pour obtenir une animation réaliste de foule dense.
- Le cinquième chapitre présente l'architecture de notre modèle qui est modélisée avec Massive. Puis, nous décrivons les différents composants de notre modélisation.
- Le dernier chapitre présente notre modèle de simulation avec les programmes utilisés pour déterminer les différents comportements, nous avons également présenté quelques images résultats pour notre animation. Puis, nous terminons ce chapitre par un rapport de simulation défini par l'outil « FisPro ».

Enfin, nous concluons ce mémoire par des critiques sur ce travail et nous donnons certaines perspectives.

CHAPITRE 1

Animation comportementale

« Plus s'étend et s'approfondit le champ de ma connaissance, plus s'aiguise la conscience de l'étendue de mon ignorance »

Michel Beaud

Animation comportementale

1. Contexte et définition

L'objectif de l'animation est de produire une séquence d'images représentant une discrétisation temporelle de l'état d'un système évoluant au cours du temps. Cette évolution se matérialise par le mouvement des objets ou de la caméra dans la scène, la déformation géométrique des objets et la transformation de leurs attributs non géométriques. Le système est représenté par une suite définissant l'état du système aux instants et correspondant à un échantillonnage régulier du temps. L'animation consiste d'une part à exprimer la (phase de spécification) et d'autre part à les évaluer (phase de calcul). Les modèles de contrôle du mouvement des objets forment le cœur d'un système d'animation et peuvent être classés en trois grandes familles :

- 1) les modèles descriptifs qui s'attachent à reproduire les effets sans en connaître les causes,
- 2) les modèles générateurs qui permettent de décrire les causes qui produisent les effets,
- 3) les modèles comportementaux dont le but sont de simuler le comportement externe ou interne d'entités douées d'une certaine autonomie. Donc les modèles comportementaux peuvent être classés en deux catégories :

3.1 *Les modèles de transformation interne :*

- provoquant des changements externes de l'objet (croissance de plantes ou modèle de muscle par exemple).
- les relations entre l'entité et son environnement sont quasi inexistantes, l'élément important étant de fournir une perception la plus réaliste possible du comportement de la plante (croissance) ou de l'individu (animation faciale, musculaire).

3.2 *Les modèles d'animation externe :*

- définissant le comportement extérieur d'un être, ses actions et ses réactions, de manière individuelle (animal, humain) ou collective (nuée d'oiseaux, banc de poissons, troupeau de mammifères, foules d'humanoïdes).
- sont tous fondés sur un système relationnel entre des acteurs et leur environnement ce qui implique deux types de relations : la perception et l'action.

L'animation comportementale consiste en boucle d'avertissement fermée à niveau élevé, qui offre la capacité de simuler les entités autonomes comme des organisations et des êtres vivants. De tels acteurs peuvent percevoir leur environnement, prendre la décision,

communiquer avec d'autres et exécuter quelques actions, telles que marcher dans la rue ou saisir un objet. Des comportements réalistes des acteurs autonomes évoluant dans les environnements complexes et structurés peuvent être obtenus si et seulement si le rapport entre l'acteur et son environnement environnant peut être simulé. L'information qui doit être extraite ou interprétée de l'environnement dépend du niveau d'abstraction du raisonnement exécuté par les acteurs autonomes.

2. Domaine d'application

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

2.1 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 *creatures* 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 l'animation comportementale a fait naître une nouvelle tendance: la fiction interactive. 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 [Fou06].



Figure 1.1 : Le jeu vidéo « Les Sims 2 » de Electronic Arts [Pan08].



Figure 1.2 : L'environnement virtuel du jeu en ligne « Second Life » [Pan08].

2.2 Films et effets spéciaux

L'animation 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 [Fou06].



Figure 1.3 : Le film d'animation « Mulan » de Disney [Pan08].

2.3 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. 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 [Fou06].



Figure 1.4 : La simulation d'entraînement Firefighter Training (IRIT) [Pan08].

2.4 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. 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. Dans le domaine hospitalier, l'animation comportementale, couplée à la réalité virtuelle, peut être utilisée dans le cadre de soins aux personnes phobiques.

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 [Fou06].



Figure 1.5 : Visite virtuelle de la ville de Grenoble (Artesia) [Pan08].



Figure 1.6 : Visite virtuelle de la cité des sciences et de l'industrie (IRISA) [Pan08].

3 Les premiers travaux

On peut situer historiquement les premiers travaux à la fin des années 80 avec notamment l'article de C. Reynolds [Rey87] sur l'animation de nuées d'oiseaux. Un premier ensemble d'approches a été étudié en parallèle dans la littérature, jusque dans le milieu des années 90, pour la définition du modèle décisionnel:

Stimulus / réponses: cette approche définit le comportement des objets à partir d'un ensemble de capteurs et d'effecteurs reliés entre eux par un réseau de nœuds intermédiaires transformant l'information passée. Ce type d'approche comprend les modèles de type réseaux de neurones. La manière dont un objet se comporte dépend de la perception qu'il a de son environnement et de la manière dont cette perception est transmise à travers le réseau aux effecteurs qui produisent le mouvement des objets. Cette approche a l'avantage de pouvoir générer une quantité très importante de mouvements différents (choix du jeu de paramètres) mais reste par contre à un niveau d'abstraction très faible. De plus il s'agit de modèles de type boîte noire dans lesquels il est impossible de modifier le moindre paramètre sans avoir à reprendre le processus complet de configuration. De plus, le pourcentage de bons contrôleurs

décroissant de manière exponentielle avec la croissance du nombre de paramètres à prendre en compte, il est illusoire de vouloir contrôler un mouvement complexe avec cette approche [Don04].

Règles de comportement: comme l'approche précédente, l'approche par règles de comportement prend en entrée des informations restituant une certaine perception de l'environnement et produit en sortie un certain contrôle sur la motricité des objets. Ici, le comportement des objets est défini par un ensemble de règles. Les comportements possibles d'un objet peuvent être représentés par un arbre de décision, chaque branche représentant un comportement différent. Une action satisfaisant les conditions de l'environnement courant sera choisie par l'application d'un algorithme de parcours d'arbre. Les comportements permis par cette approche sont d'un niveau d'abstraction plus élevé que l'approche précédente. Le cœur du problème dans cette approche réside dans la pondération des différents comportements. La solution la plus simple consiste à faire un choix implicite sur l'ordre des règles (suite de si ... alors ... sinon), mais cette solution ne permet pas la spécification de comportements complexes. Le parcours de l'arbre des possibilités avec pondération des branches permet de ne pas privilégier toujours la même règle, tandis que la prise en compte d'une hiérarchie d'experts permet de confronter plusieurs comportements concurrents, le choix final restant à l'expert de plus haut niveau. On peut ainsi utiliser la panoplie des algorithmes de parcours d'arbre de décision intégrant des fonctions d'évaluation des différentes branches (essais successifs, séparation et évaluation progressive, glouton, parcours ascendant, ...etc) [Don04].

Automates d'états finis: l'automate définit les différents enchaînements possibles entre comportements. Cette approche trouve très rapidement ses limites dès qu'il s'agit de modéliser un comportement un tant soit peu compliqué. La modélisation du comportement d'un conducteur a ainsi montré la nécessité de passer à l'utilisation de plusieurs automates dont il faut définir l'exécution coordonnée.

Le constat que l'on peut faire de ces différents travaux est que ce sont des modèles ad hoc conçus pour être appliqués à chaque fois sur des cas particuliers, dans lesquels les objets et leur environnement sont relativement simples et les champs de perception et d'action limités. Il existe en outre une grande disparité dans les comportements permis par chacune des approches :

- soit le niveau d'abstraction est très faible et seuls des comportements de type réflexes pourront être spécifiés (approche stimulus-réponse),

- soit le niveau d'abstraction est plus élevé mais l'environnement est réduit, voire complètement défini, et donc la perception et l'action de l'entité sont d'un niveau très faible.

Ces modèles demeurent relativement simples, avec des champs de perception et d'action limités, et en outre ils ne prennent pas en compte l'aspect temporel, qui est primordial (mémorisation, prédiction, durée d'une action, séquençage de tâches). Afin de rendre compte de la complexité décisionnelle, il est nécessaire de traiter conjointement les aspects continus et discrets, de coordonner les comportements concurrents et de gérer leur structure organisationnelle, c'est pourquoi les deux premières approches ont été assez vite abandonnées au profit de l'approche à base d'automates d'états finis dans leurs versions parallèles et hiérarchiques : piles d'automates (EPFL, Suisse), ensembles d'automates communicants (PaT-Nets) (Université de Pennsylvanie, Etats-Unis), hiérarchie d'automates parallèles (HCSM) (Université d'Iowa, Etats-Unis) et systèmes de transitions parallèles hiérarchisés (HPTS) (IRISA, France). Le modèle HPTS1 est le seul qui permette de traiter les aspects temporels et non déterministes et qui intègre un langage de spécification couplé à un générateur de code et à un interpréteur. Du côté industriel, un produit (Motivate) a été commercialisé par la société Motion Factory (orientation jeux) pour la spécification du comportement à base de machines à états finis hiérarchiques (HFSM) et il est maintenant intégré au sein du logiciel d'animation Softimage sous le nom de RTK Behavior [Don04].

4 La boucle de l'animation comportementale

L'être humain évolue dans un environnement dynamique. Il est doté de trois capacités fondamentales lui permettant de réagir aux modifications de cet environnement et d'agir sur ce dernier : la perception, la décision, et l'action. Ces trois briques forment la boucle perception-décision-action [Mal97] qui constitue le modèle le plus général représentant un organisme vivant, autonome, évoluant dans un environnement dynamique. La figure 1.7 présente ce modèle, qui est à la base d'un grand nombre de théories relatives au comportement, dans le quel chaque brique possède un rôle bien défini :

- **La perception** alimente le processus décisionnel avec des informations captées sur l'état de l'environnement. Ces informations sont locales ; un organisme, quel qu'il soit, ne peut percevoir que des informations qui se trouvent à proximité de lui. Bien entendu, cette notion de proximité comparativement à la taille du monde, peut être considérée comme ponctuelle.

- **L'action** représente les moyens que l'organisme possède pour modifier l'état du monde qui l'entoure. Une action peut se traduire sous plusieurs formes : parler, manipuler un objet, se déplacer...
- **La décision** joue un rôle central ; elle effectue une corrélation entre la *perception* et l'*action*. La perception alimente le processus décisionnel avec des informations « abstraites » issues des capteurs (ou sens) qui peuvent être mémorisées et/ou directement exploitées. A partir de ces informations, la composante décisionnelle choisit une ou plusieurs actions à réaliser qui sont adaptées au contexte et aux motivations de l'individu.

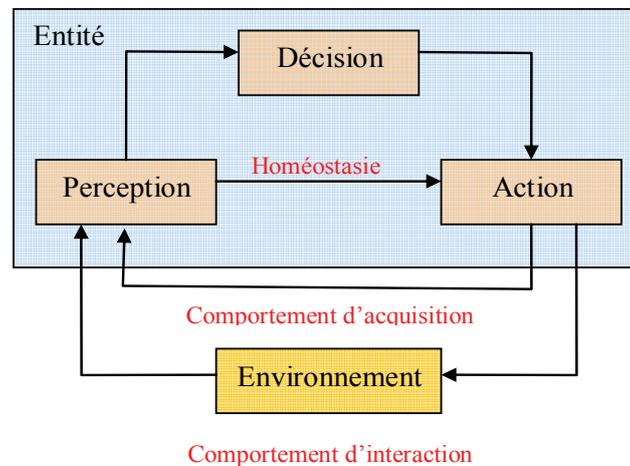


Figure 1.7 : la boucle d'animation comportementale [Don04].

Il existe un certain nombre de retours/interdépendances entre la perception et l'action. Ceux-ci sont de natures différentes :

- **L'homéostasie** est la boucle de régulation interne de l'organisation. Elle traduit la tendance de l'être humain à maintenir leurs paramètres biologiques face aux variations du milieu ambiant. Cette boucle n'agit pas directement au niveau du comportement et ne sera donc pas prise en compte dans la suite de cette étude.
- **Le comportement d'acquisition** est la partie de comportement visant à améliorer la perception. Il peut s'agir, par exemple, de fixer une chose particulière en vue d'obtenir une information nécessaire au processus décisionnel.
- **Les comportements d'interaction** constituent la boucle la plus importante. Supervisées par le processus décisionnel et la perception, les actions modifient l'état de l'environnement, traduisant ainsi l'activité de l'organisme dans son environnement.

Ces informations traduisent le lien ténu qui lie l'état humain et l'environnement dans lequel il évolue. Le comportement semble déterminé par l'environnement mais inversement

l'être humain influe sur ce dernier. Cette relation d'influence traduit l'autonomie des êtres vivants qui peuvent, en fonction de ce qu'ils perçoivent et de leur état interne, décider de leurs propres actions [Lam03].

La boucle d'animation comportementale ou bien les boucles rétroaction viennent d'après la classification de H. Mallot [Mal99]. Effectue aussi une catégorisation des comportements humains en quatre niveaux de complexité :

- 1) Premièrement, les comportements réflexes de type attraction / répulsion peuvent être définis par une interconnexion entre les capteurs et les effecteurs.
- 2) Le deuxième niveau décrit des comportements de type manœuvre qui nécessitent une intégration spatio-temporelle.
- 3) Le troisième niveau s'occupe de l'apprentissage, et est intimement lié aux données sensorielles.
- 4) Enfin, le quatrième niveau, dénommé cognitif, tient compte du but courant de l'individu, et s'inscrit donc dans un processus à plus long terme.

Une approche comparable à cette classification est la théorie du contrôle. Celle-ci est issue de travaux en biologie, avec des expériences sur des êtres vivants en situation d'asservissement. Son principe de base définit que lorsque l'état d'un organisme est différent de celui désiré, l'organisme va modifier son comportement par une série d'actions afin de minimiser cet écart. Ainsi, R. Lord et P. Levy [Rob94] reprennent, avec cette théorie du contrôle, le principe de rétroaction tout en l'étendant à la régulation des comportements sociaux. Pour eux, ces boucles de rétroaction sont organisées hiérarchiquement, avec une notion de complexité croissante, permettant une grande généralité de la prise de décision. Ainsi, chaque niveau traitera d'un ensemble de tâches spécifiques, tout en fonctionnant suivant le principe de minimisation d'erreur. A. Newell [All90] propose une classification de ces niveaux comportementaux hiérarchiques (voir la figure 1.8). Il indique que chaque niveau a un fonctionnement propre, mais peut communiquer avec – et donc influencer – ses voisins immédiats, sans pour autant avoir besoin de connaître leur fonctionnement interne. Ce principe est d'autant plus intéressant en informatique, car directement transposable sous la forme d'un modèle composé de boîtes noires, devant être capables de communiquer des informations. La difficulté réside alors dans cette transmission d'information, devant être suffisamment synthétique tout en conservant un maximum d'expressivité.

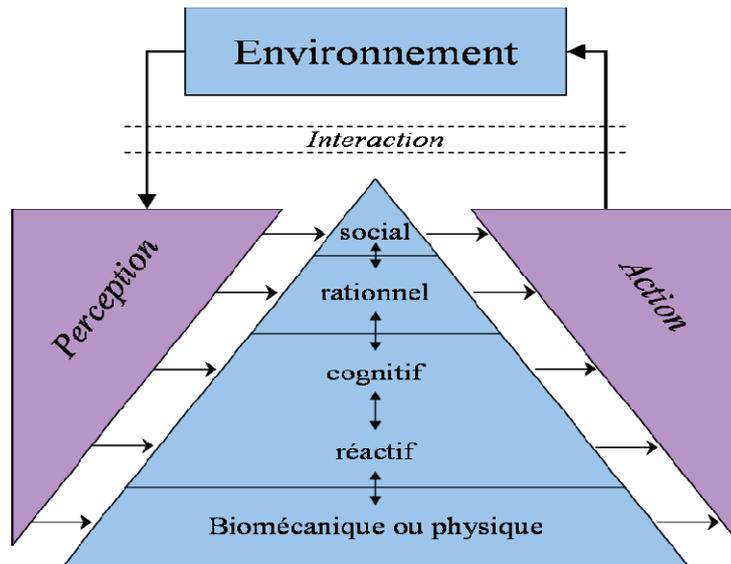


Figure 1.8 : Hiérarchisation des comportements illustrée par A. Newell dans sa pyramide comportementale [Don04].

5 Conclusion

L'animation 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. L'animation comportementale est donc un moyen de faire interagir de manière naturelle des acteurs en simulant leurs capacités dans un environnement.

CHAPITRE 2

Foule d'humains virtuels

« Un ne transmet pas une vérité, chacun doit la trouver seul »

Pierre Daco

Foule d'humains virtuels

1. Introduction

Bien qu'abordée depuis un certain temps déjà, la simulation de foule reste encore aujourd'hui un sujet très ouvert. Ainsi, de nombreuses recherches traitent encore ce vaste sujet, que ce soit dans le seul but de produire une animation dont l'aspect semble réaliste, ou encore d'effectuer des simulations permettant la caractérisation et l'étude de ce phénomène.

Dans ce chapitre et comme un premier temps, nous avons fait une étude sur l'entité qui compose la foule d'humains virtuel. Et comme un deuxième temps, nous définissons la notion de foule.

2. L'humain virtuel

2.1 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 2.1).

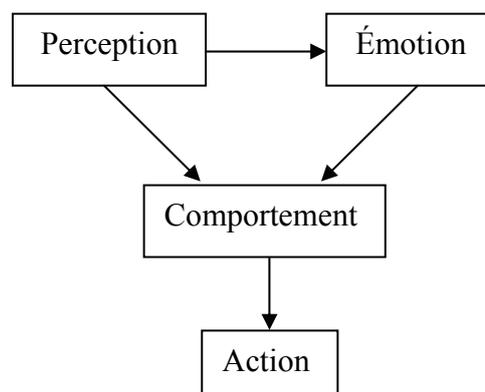


Figure 2.1 : Structure du Modèle comportemental [Fou06].

2.1.1 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 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. 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 [Don04].

2.1.2. 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.

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

Les émotions causées par des événements peuvent être classées selon trois types d'émotions:

- Les émotions causées par des événements potentiels.
- Les événements affectant le destin d'autres.
- Les événements affectant le bien-être de l'acteur.

Chaque classe est caractérisée par des conditions d'apparition pour chacune de ses émotions et variables affectant son intensité. Les émotions auxquelles est soumis un acteur sont dues à sa perception [Don04].

2.1.3. Le comportement

Le comportement est souvent défini comme la voie dans lequel les animaux et les humains agissent. Il est usuellement décrit en termes de langage naturel qui a la signification sociale, psychologique ou physiologique, mais qui n'est pas nécessairement facilement réductible au mouvement d'un ou des muscles.

Le comportement est aussi la réponse d'un individu, groupe, ou espèce à son environnement. Le comportement ne réagit pas seulement à l'environnement, mais inclut aussi le flux d'information par lequel l'environnement agit sur la créature vivante aussi bien sur la façon dont la créature code et emploie cette information.

Le comportement peut être décrit d'une façon hiérarchique. Le modèle comportemental décompose un comportement en comportements plus simple qui peuvent être décomposés plus loin.

Chaque niveau de cette décomposition hiérarchique contient un ou plusieurs comportements exécutés séquentiellement, ou bien concurremment. Un niveau de la hiérarchie contenant plusieurs comportements pour être exécuté séquentiellement, appelé comportement. Chaque comportement d'un ordre de comportement est appelé, cellule comportementale.

La cellule comportementale contient les comportements qui sont exécutés chacun concurremment, ou exclusivement quand les règles d'inhibition sont spécifiées. Les comportements contenus dans une cellule comportementale sont de nature composée ou élémentaire.

Un comportement permet une décomposition récursive dans la hiérarchie. Un comportement élémentaire est placé au fond de la décomposition hiérarchique et encapsule un comportement spécialisé qui contrôle directement une ou plusieurs actions. Un comportement est exécuté récursivement: en fait la hiérarchie du comportement permet d'exécuter les comportements élémentaires des entités à chaque niveau de la structure de la hiérarchie.

Pour contrôler le comportement global d'un acteur, on exploite une pile de comportements. Au début de l'animation, l'utilisateur pousse un ordre de comportements dans la pile de l'acteur. À la fin du comportement actuel le système d'animation passe le comportement suivant de la pile et l'exécute. Ce processus est répété jusqu'à ce que la pile de comportement de l'acteur se vide.

Avec Ce contrôle de comportement employant une pile, un acteur devient plus autonome et crée ses propres sous buts en exécutant le scénario original [Don04].

2.1.4. L'action

Basée 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 amorçant. Une boucle comportementale conduit l'animation, son rôle est de mettre à jour l'état du monde virtuel.

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 [Don04].

2.1.5. 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, 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 [Don04].

2.2 Classification des humains virtuels

Pour bien expliquer les différents types d'humains virtuels et leur évolution, nous allons nous servir de la classification de thalman [Tha96] des humains virtuels :

- les avatars
- les acteurs guidés
- les acteurs autonomes
- les acteurs perceptifs interactifs

2.2.1 Les Avatars

Dans notre terminologie, les avatars correspondent à des représentations fidèles de l'utilisateur. La technique consiste à enregistrer directement les mouvements de l'utilisateur et à les transposer sur un humain virtuel qui devrait avoir aussi son apparence. Pour implanter de tels avatars, il est nécessaire d'avoir des séries de capteurs comme les *Flock of Birds*.

2.2.2 Les acteurs guidés

Les acteurs guidés sont des acteurs qui sont conduits par l'utilisateur mais sans que leurs mouvements ne correspondent directement à ceux de l'utilisateur. Ils sont basés sur le concept de la métaphore directe temps réel [Tha98]. Pour comprendre le concept, prenons un exemple de métaphore traditionnelle: le contrôle d'une marionnette. Une marionnette peut être définie comme une poupée avec des membres déplacés par des fils ou des ficelles. Les doigts humains sont employés pour diriger le mouvement de la marionnette. En Réalité Virtuelle, le meilleur exemple de guidage d'acteurs est la navigation réalisée dans les systèmes d'Environnements Distribués.

2.2.3 Les acteurs autonomes

Leur rôle est de représenter un service ou un programme comme par exemple pour naviguer. Comme ces acteurs virtuels ne sont pas guidés par les utilisateurs, ils doivent avoir des comportements leur permettant d'agir de manière autonome pour accomplir leurs tâches. Cela requiert de bâtir des comportements pour le mouvement, ainsi que des mécanismes appropriés pour l'interaction. Ce domaine est celui où la recherche de pointe se fait. A titre d'exemple, nous allons prendre le cas de l'évitement d'obstacles, un problème que chacun d'entre nous résout chaque jour un très grand nombre de fois. Les roboticiens ont introduit des techniques mathématiques qui permettent d'évaluer, à partir d'une configuration donnée le ou les chemins possibles, pour aller d'un point à un autre en évitant les obstacles. Cette démarche très valable pour un bras de robot se déplaçant mécaniquement dans un environnement connu et statique, est peu applicable au comportement d'un humain dans un environnement souvent mal connu et parfois dynamique. Lorsqu'on se déplace sur un trottoir, il ne faut pas seulement éviter de se cogner aux parcomètres, mais surtout éviter le piéton qui vient en face; sans oublier que ce piéton va aussi tenter de vous éviter. Ce type de comportement, de nature dynamique, ne peut être simulé en animation qu'avec une démarche de type sensorielle. C'est effectivement le sens de nos recherches. Nos humains virtuels ont maintenant une vision, une audition et un toucher de synthèse. Ils peuvent se diriger vers un bruit déterminé ou éviter la

table qu'ils voient ou enfin réagir même dans l'obscurité en reconnaissant qu'ils ont cogné un mur. Ils peuvent même jouer au tennis en basant leur jeu sur celui de l'adversaire.



Figure 2.2 : Acteurs autonomes (Design: Mireille Clavien).

Le principe repose sur une boucle dite de perception-action. A chaque instant, pour chaque être virtuel, on crée une perception de l'espace virtuel, c'est-à-dire que l'on simule dans l'ordinateur ce que chaque être peut voir, entendre et toucher à l'aide de sens virtuels. Ainsi l'humain virtuel aura un vrai champ de vision, il pourra entendre les bruits produits par d'autres êtres ou se cogner à une table. Sur la base de ce que ces êtres ressentent, ils sont capables de prendre des décisions simples. Par exemple, un être virtuel, attiré par un son en dehors d'une pièce, pourra chercher la porte et lorsqu'il l'aura aperçue, pourra trouver le chemin jusqu'à cette porte et sortir et ceci sans aucune intervention d'un vrai humain.

2.2.4 Les acteurs perceptifs et interactifs

Nous définissons un acteur perceptif et interactif comme un acteur conscient d'autres acteurs et des gens réels. Un tel acteur est aussi supposé être autonome bien sûr. De plus, il est capable de communiquer interactivement avec les autres acteurs quel que soit leur type et avec les gens réels (voir la figure 2.3).



Figure 2.3 : interaction entre deux acteurs.

2.3. Animation d'humain virtuel

Dans le cadre de l'animation comportementale, il est un organisme vivant qui retient l'attention : l'être humain. Il s'agit d'un être vivant qui exhibe un comportement très complexe. Il est à la fois doté de capacités de réaction à l'environnement et de raisonnement. L'immense potentiel d'interaction entre l'être humain et son environnement, ainsi que sa capacité, souvent exploitée, de réaliser plusieurs tâches simultanément, traduit cette complexité. Elle soulève de nombreux problèmes dans le cadre de l'animation

comportementale et met en évidence le besoin d'outils et de modèles évolués de description du comportement.

L'animation d'un humain virtuel de synthèse se décompose généralement en plusieurs niveaux : le niveau géométrique, le niveau cinématique, le niveau contrôle du mouvement, le niveau réactif et enfin le niveau cognitif.

Le niveau géométrique fournit la représentation de l'humain virtuel dans l'environnement virtuel, qui se caractérise par un grand nombre de degrés de liberté. Les valeurs associées à ces degrés de libertés sont calculées dans la couche cinématique, en charge de la gestion de l'animation. La couche de contrôle du mouvement, offre des interfaces de haut niveau permettant de spécifier les gestes qui doivent être effectués par l'humain virtuel. Le réalisme de l'animation dépend de ces trois premiers niveaux et plus précisément de la précision de représentation du modèle géométrique, mais aussi de la qualité visuelle des mouvements qui peuvent être effectués.

Le niveau réactif définit des enchaînements d'actions atomiques permettent de traduire le comportement de l'humain virtuel sous la forme de ses interaction avec l'environnement et effectue donc le lien avec l'animation. Enfin, le niveau cognitif symbolise les capacités de raisonnement dont l'humain virtuel peut être doté. Il est dépendant d'une représentation abstraite de l'environnement, qui lui permet de raisonner sur ses actions et leurs conséquences. Ces deux niveaux permettent d'obtenir un certain réalisme au travers de la cohérence globale du comportement exhibé par l'humain virtuel. Cette cohérence dépend principalement de la pertinence de l'enchaînement des actions par rapport à l'état perceptible de l'environnement et aux motivations de l'humain virtuel. Enfin, la brique d'évolution traduit le phénomène biologique du même nom, que l'animation comportementale n'adresse pas réellement et qui est plutôt du ressort de la vie artificielle [Lam03].

3. La foule

3.1. Comportement de foule

Il est connu que dans les mouvements de foules, les flux de piétons marchant dans des directions opposées ont tendance à se séparer, conduisant ainsi à la formation dynamique de files de piétons allant dans la même direction. Lorsque la densité des piétons dans un espace devient grande, il est montré qu'il est possible d'effectuer une approximation de la tendance générale en utilisant les lois sur l'évolution des fluides. Dans les situations de panique, les piétons souhaitent se déplacer beaucoup plus vite que d'habitude et de ce fait vont jusqu'au contact physique avec les autres. Ils ont aussi tendance à développer des comportements

mimétiques de ceux les précédant dans le flux. Turner [Tur64] utilise une théorie des normes pour suggérer que l'homogénéité se développe dans une foule non pas parce que des personnes ayant les mêmes prédispositions se sont assemblées spatialement, mais parce que les membres de la foule ont tendance à se conformer à une norme émergente.

Selon Yamori [Yam98], cette notion de norme est accompagnée inévitablement de la notion de régulation, comportement qui n'est pas inné mais acquis comme élément normatif au sein d'une certaine macrostructure ou collectivité. Cependant, de petits groupes peuvent servir de levier pour faire bouger de plus grandes unités, jouant ainsi le rôle du cœur.

Boles [Bol81] a observé des structures de bandes au sein des foules de piétons se déplaçant sur des trottoirs et il a montré que les piétons avaient tendance à utiliser ce type de structure afin de maintenir le maximum d'efficacité au sein du trafic piétonnier. K. Yamori a cherché à étudier ces structures de bande d'un point de vue macroscopique [Yam98]. Il postule que pour que ce type de structure existe, il est nécessaire qu'une densité critique de personnes soit atteinte, et il cherche à déterminer les facteurs qui déterminent l'émergence de ces structures. Milgram et Toch [Mil69] ont noté qu'une structure en anneau émergeait lorsque les membres d'une foule disposaient d'un point d'intérêt unique commun. Cette formation circulaire peut être observée au sein des foules regroupées par exemple autour d'artistes de rue. Un des problèmes cruciaux mis en avant par Yamori [Yam98] concerne les relations entre les comportements micro et macroscopiques au sein d'une foule. L'enjeu est d'expliquer comment un individu peut affecter une collectivité et comment en retour la collectivité va agir sur le comportement individuel au cours du temps. Yamori se focalise sur l'étude de la formation des structures de bandes macroscopiques sur des passages-piétons dans un cas de trafic piétonnier très dense comme c'est le cas au Japon. De l'étude, il postule qu'une masse critique doit être atteinte afin qu'émerge une structure en forme de bandes.

3.2. Animation de foule

La simulation macroscopique a été historiquement la première approche utilisée pour simuler les déplacements de personnes, du fait de son faible coût de calcul. En effet, dans cette approche, le piéton n'est pas traité individuellement mais en tant qu'élément constitutif d'une matière plus macroscopique. Henderson [Hen71] propose, dans le cas de faibles densités, l'utilisation d'un modèle gazeux assimilant les déplacements de personnes à ceux de molécules de gaz en faisant un parallèle sur leurs densités respectives. D'autres modèles ont été proposés comme le modèle hydraulique pour les foules à forte densité, assimilant le mouvement de personnes dans un espace à celui d'un liquide dans un tuyau.

D'autres modèles statistiques fondés sur un grand nombre d'observations ont été proposés. Ils donnent des équations sur le temps d'évacuation d'un immeuble en traitant séparément les zones de circulation horizontale et verticale. Le lecteur intéressé trouvera une description détaillée de ces modèles dans l'étude bibliographique de début de thèse de Sébastien Paris [Par07] sur la caractérisation des niveaux de services et la modélisation des circulations de personnes dans les lieux d'échange.

Les systèmes à base de particules sont des systèmes physiques qui permettent de décrire des forces d'attraction et de répulsion pouvant être associées à des obstacles ou à des entités simulées. Les forces appliquées à une entité sont additionnées de manière à déterminer sa nouvelle direction. Ce modèle fait l'analogie entre le déplacement d'individus en forte densité, n'ayant quasiment pas de degrés de liberté dans leurs déplacements et l'écoulement de particules dans des compartiments. I. Peschl [Pes71] indique que cette modélisation est pertinente pour le cas où une forte densité de personnes est confinée et en état de panique. Quand le flot se dirige vers une issue unique, un phénomène d'agglutination se produit et le débit de l'issue devient quasi nul. Braun et al [Bra03] proposent une extension du modèle d'Helbing en ajoutant la dimension sociale de la connaissance d'autrui au sein du groupe : lorsque les entités se connaissent, elles peuvent s'entraider pour sortir d'une situation critique. Ils modélisent ainsi des forces d'attraction entre agents, pondérées par des facteurs d'altruisme et de dépendance. Les modèles à base de particules exhibent un comportement macroscopique réaliste dans le cas d'une densité de population importante.

Cependant, ces modèles ne prennent pas en compte ni la perception visuelle, ni les règles sociales.

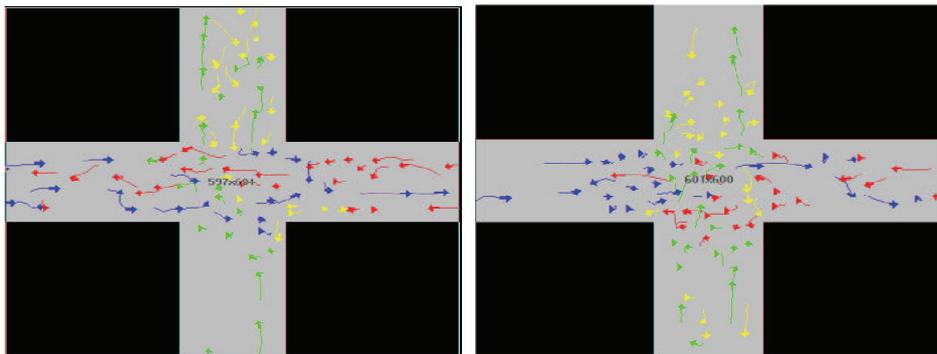


Figure 2.4 : Système à base de particules [Hel00].

S. Goldenstein et al [Gol01]. Proposent une approche dynamique en trois couches pour modéliser le comportement de foules de personnages évoluant au sein d'environnements virtuels. La première couche, la plus locale, est un système dynamique non linéaire décrivant le mouvement de l'agent modélisé comme une particule par deux paramètres : son angle

d'orientation et le module de sa vitesse. Les obstacles et les cibles définissent ainsi un ensemble de répulseurs et d'attracteurs. Des paramètres comportementaux comme l'agressivité est prise en compte en utilisant une distance-seuil indiquant à partir de quelle distance un obstacle est pris en compte. La seconde couche dite environnementale consiste en une structure de données cinématiques maintenant dynamiquement les relations avec les objets en mouvement et les obstacles à l'aide d'une triangulation de Delaunay. La troisième couche permet d'avoir une vision globale nécessaire pour planifier des tâches et déterminer ainsi l'orientation désirée pour s'approcher d'une cible.

Le modèle de nuée introduit par Reynolds [Rey87] est un modèle à base de règles de comportement permettant de définir le comportement d'un oiseau en fonction du comportement des entités les plus proches. Il propose ainsi d'utiliser seulement trois règles pour définir le comportement d'un oiseau au sein de la nuée, tout en définissant un rôle particulier au meneur chargé de suivre une trajectoire :

- la séparation afin d'éviter d'éventuelles collisions avec les voisins ;
- l'alignement afin de réguler sa vitesse sur celle de ces voisins ;
- la cohésion afin de rester proche de ses voisins.

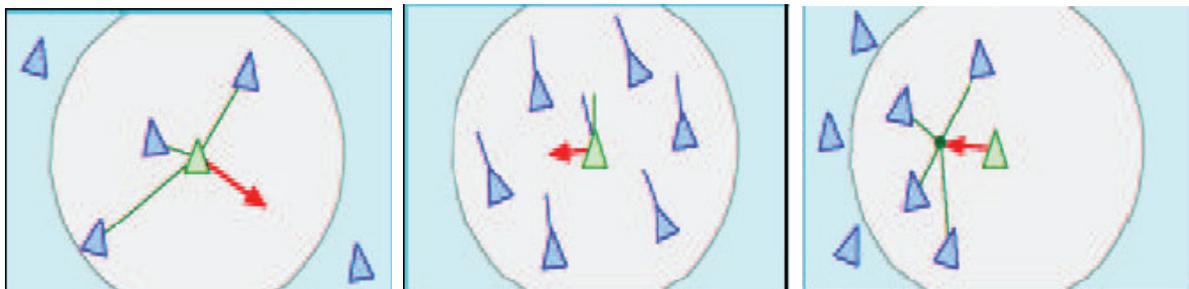


Figure 2.5 : les trois règles de Reynolds [Rey99].

Ulicny et al [Uli02] utilisent une approche en plusieurs niveaux pour modéliser le comportement d'un individu au sein d'une foule en combinant les approches à base de règles et d'automates d'états finis. Toutes ces approches sont confrontées au problème de la détermination des plus proches voisins, ce qui constitue un des goulets d'étranglement en ce qui concerne la détermination du nombre de piétons simulables en temps réel. Plusieurs approches ont été proposées pour optimiser ce calcul en utilisant des structures de données telles que des treillis (bin-lattice) [Rey99] ou des K-d arbres [Har00]. D'autres méthodes fondées sur l'exploitation de connaissances sur l'environnement informé ont été proposées, permettant ainsi de modéliser des comportements qui utilisent la nature de l'espace de circulation et sa localité. Des études se sont focalisées sur les niveaux d'autonomie et de contrôle pour la simulation de foules, de manière à proposer des comportements réalistes de

foules au sein d'environnements virtuels à un moindre coût calculatoire. Ainsi Soraia Raupp Musse [Mus00c] a introduit dans sa thèse différents niveaux de contrôle des foules : le guidage, la programmation et l'autonomie.

Dans le cadre du projet MAGS, les auteurs s'intéressent à la simulation de plusieurs milliers de piétons dans une représentation discrétisée de la ville de Québec (voir la figure 2.6). Ils traitent le problème du goulet d'étranglement du calcul de la perception en limitant la vision à un cône de 90 degrés de secteur angulaire et de 20 cellules de rayon (pour une carte 300x300). Des travaux complémentaires ont été effectués sur l'optimisation de la visualisation en temps réel de foules en utilisant des imposteurs hiérarchiques et des modèles d'ombrage temps réel d'imposteurs. De manière à augmenter le réalisme des animations, Ashida et al [Ash01] ont effectué des analyses statistiques sur les déplacements des piétons le long d'un trottoir. Ils exhibent des actions subconscientes qu'ils intègrent dans leur moteur d'animation à l'aide d'un processus stochastique pour contrôler leur activation. Tsutsuguchi et al. [Tsu00] Posent un modèle de gestion de l'animation de foule dépendant de la distance à l'observateur : au delà d'une certaine distance à l'observateur, la position des piétons n'est pas mise à jour à chaque image mais à une fréquence moins élevée.



Figure 2.6 : Projet de MAGS [Mou03].

4. conclusion

La foule n'aurait pas de valeur que si l'élément constitutif de la foule est l'individu. Pour cela, nous avons présenté dans ce chapitre l'élément principal de la foule virtuel qui est l'humain virtuel. Nous avons présenté ses propriétés pour donner plus de réalisme à l'animation de foule basée sur l'autonomie de chaque humain virtuel.

CHAPITRE 3

Différentes techniques de modélisation de foules

« Nous sommes constamment à la recherche d'analogies avec nos expériences antérieures ; sinon, toute action serait impossible. Notre langage et toute notre culture sont bâtis sur ces analogies. »

Robert M.

Différentes techniques de modélisation de foules

1. Introduction

Ce chapitre vise à présenter une vue d'ensemble de la littérature sur le domaine de la simulation de foule virtuelle, des études connexes sur des facteurs sociologiques, différentes modélisations de foule, et des aspects d'interaction. En outre, on propose une classification des méthodes de simulation de foule afin de présenter les travaux relatifs sur le domaine de la simulation de foule. Enfin, on signale que de nombreuses applications visant à recréer des comportements de foules ont été développées. Il serait impossible d'en faire une liste complète. En plus, nous présentons quelques travaux utilisant différents modèles de comportement.

2. Les aspects sociologiques et d'interaction de foule

Les sociologues et les psychologues ont étudié beaucoup d'aspects des conventions sociales, la proxémique d'individu et de groupe (étude de l'espace personnel), l'importance des relations entre groupes et les effets d'interaction surgissant dans un groupe de personnes.

L'espace personnel est le secteur immédiat entourant un individu. Cet espace peut être employé pour fournir à l'interaction d'autres personnes ou être isolé dans les autres (concept d'intimité), selon l'intention et la perception individuelles de l'espace [Jef98b]. Proxémiques est l'étude de l'espace et des foyers personnels sur l'utilisation sociale de l'espace pour atteindre une communication confortable et réaliser les niveaux préférés de la participation interpersonnelle [Aie87] [Hal98].

Dans le monde physique, les normes sociales règlent le comportement. Le comportement des gens est influencé par des expériences antérieures, observant autres et identifiant les repères qui spécifient si son comportement serait acceptable socialement [Jef98a]. Les directives qui définissent son comportement peuvent s'appeler les conventions sociales. L'espace personnel, l'intimité, l'espace de groupe, et les facteurs environnementaux sont des paramètres qui peuvent influencer des conventions sociales. Les normes sociales règlent le comportement dans la société. Elles sont parfois non-dites et peuvent être acquises par l'expérience. Elles peuvent jouer un rôle important dans le comportement d'utilisateur et ses espérances au sujet du comportement des autres. Les normes de groupe peuvent « se développer à travers des instructions données par les superviseurs, des événements critiques dans l'histoire du groupe, ou comportement à partir des situations passées » [Fel84].

Une définition acceptable de foule est un groupe d'individus placés dans le même environnement physique ou ils ont un objectif commun ou les deux. La foule peut être inactive ou en activité. Dans le dernier cas, les groupes ou les individus suivent quelques tendances et partagent quelques buts et intentions. Puis, les individus dans une foule peuvent agir d'une manière différente que quand ils sont seuls ou dans un petit groupe, voulant dire que les individus peuvent perdre leurs comportements et intentions individuels et assument le comportement de foule ou de groupe [Ben95]. Quelques comportements sociaux peuvent surgir de rapports sociaux entre les individus qui forment temporairement une masse ou une foule.

Quelques chercheurs ont étudié les conventions requises pour établir la communication entre les groupes et les individus en mondes virtuels [Jef98b]. Par exemple, Becker et Mark (1998) ont défini quelques conventions d'interaction entre groupes afin d'établir l'adhésion de groupe et indiquer l'intimité. Cependant, les conventions sociales peuvent développer le thème de produire l'entité, dont le genre est un composant [Kau98]. Pour ce qui concerne les environnements virtuels et distribués, Rodden (1996), Greenhalgh et Benford, (1997) ont étudié l'impact des environnements virtuels de collaboration à grande échelle peuplés par les utilisateurs de l'internet [Bow99].

3. Les deux approches de foule

Là existent beaucoup de méthodologies appropriées de simulation de foule. Ici nous présentons plusieurs systèmes qui ont été développés pour la dynamique d'animation ou d'évacuation. Elle toute partage la condition pour les modèles piétonniers de mouvement, mais diffère dans les techniques pour produire des chemins de mouvement. D'abord nous examinerons les méthodes microscopiques, puis les méthodes macroscopiques.

3.1 Les modèles microscopiques

Les modèles microscopiques étudient le comportement et les décisions de différents piétons et de leur interaction avec d'autres piétons dans la foule. Les modèles microscopiques décrivent le comportement d'espace-temps de différents piétons. Il y a deux sous-catégories : modèles de force sociale et modèles des automates cellulaires (AC). La différence entre eux est dans la discrétisation de l'espace et du temps. Les modèles de force sociale [Hel00] décrivent le comportement microscopiquement piétonnier par les domaines sociaux (les forces sociales virtuels) induits par le comportement social des individus. Dans l'approche d'AC, l'espace à l'étude est représenté par une grille uniforme des cellules avec les états locaux selon un ensemble de règles qui décrivent le comportement des piétons [Che04]. Ces

règles calculent l'état d'une cellule particulière comme en fonction de son état précédent et des états des cellules adjacentes. Les modèles microscopiques sont plus intéressants du point de vue d'animation de foules virtuelles des agents avec des comportements autonomes réalistes ; donc, nous expliquerons ces méthodes plus en détail.

3.1.1 Modèles de force sociale

Le modèle de force sociale est une approche microscopique pour simuler le mouvement de piétonnier. Les forces sociales « virtuelles » sont similaires à de vraies forces telles que l'interaction répulsive, forces de frottement, dissipation, et fluctuations. Il résout les équations du mouvement de Newton pour chaque individuel. Ce modèle peut être appliqué pour simuler les scénarios de mouvement dans un monde réel.

A l'inverse d'autres modèles, les modèles de force sociale décrivent le comportement piétonnier plus exactement. Cependant, ils sont conçus pour être aussi simples que possible. Chaque agent est représenté dans le plan de locomotion par un cercle avec son propre diamètre et le modèle décrit des coordonnées, des vitesses, et des interactions continues avec d'autres objets. Chaque paramètre de force ayant une interprétation normale, est propre pour chaque piéton, et est souvent choisi aléatoirement ou trouvé à l'aide d'une expérience. Les forces sociales modélisent le comportement de foule humaine avec une combinaison des facteurs socio-psychologie et physique. Le modèle de force sociale dérivé le plus important de forces sociales est le modèle de Helbing comme illustré dans la figure 3.1 [Nur08].

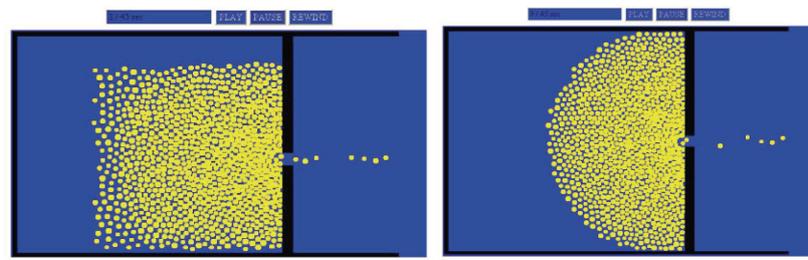


Figure 3.1 : La simulation de modèle de Helbing avec 10.000 individus [Hel100].

3.1.2 Les modèles d'automates cellulaires

AC est une approche d'intelligence artificielle pure, la simulation des modèles comme des spécifications mathématiques des systèmes physiques dans lesquels l'espace et le temps sont discrets et quantifiés physiquement et prennent un ensemble fini de valeurs discrètes. Un automate cellulaire se compose d'une grille uniforme régulière (2D rangée) avec un ou plusieurs variables discrètes à chaque emplacement (cellules) (Figure 3.2). Les murs et d'autres obstacles fixes sont noirs, alors que les cellules blanches sont des zones qui peuvent être occupés par des piétons [Dij00] [Kir03].

L'état d'un automate cellulaire est complètement spécifié par les valeurs des variables comme chaque cellule. Un automate cellulaire évolue dans des étapes discrètes, avec la valeur de la variable d'une seule cellule et on affecte par les valeurs des variables aux cellules voisines. Les variables de chaque cellule sont mises à jour simultanément ont se basés sur les valeurs des variables dans leur voisinage durons l'étape de temps précédente et selon un ensemble de règles locales [Wol01]. Ces règles décrivent le comportement de prise de décision (intelligent) des automates, de ce fait créant et émulation de comportement réel. Chaque automate évalue ses opportunités selon une base cas par cas. Le comportement global de groupe émergent est un résultat des interactions des règles locales comme chaque piéton examine les cellules disponibles dans son voisinage.

Les AC en général fournissent à un cadre pour les modèles discrets des interactions localement homogènes. Elles sont caractérisées par les propriétés fondamentales (L, S, N, f) montré dans le tableau ci-dessous. L'acceptation d'un treillis régulier et d'un voisinage uniforme est compatible avec les géométries comme ceux dans le tableau puisque l'ensemble d'états, S, contient également des informations sur, comme une cellule soit accessibtable ou non (par exemple, des portes ou des murs entre les cellules) [Nur08].

Tableau 3.1 : Table pour définir les automates cellulaires	
L	Se compose d'une grille discrète régulière des cellules
$t \rightarrow t+1$	L'évolution prene place dans des étapes de temps discret
S	Ensemble d'états finis.
$F: S^2 \rightarrow S$	Chaque cellule évolue selon la même règle (fonction de transition), qui dépend seulement de l'état de la cellule et d'un nombre fini de cellules voisinages.
$N: \forall c \in N, \forall r \in L: r+c \in L$	La relation de voisinage est locale et uniforme



Figure 3.2 : Simulation de foule par AC 2D structure étant à la base de grille [Tec01].

Parmi les travaux de AC on a le projet de Stephen Chenney, Stephen Chenney est un chercheur à l'université du Wisconsin à Madison aux Etats-Unis. Il a publié en 2004 un article au Siggraph sur un de ses projets intitulé « Flow Tiles » [Che04] (qu'on pourrait traduire par « carreaux de flux »). Cet article présente une nouvelle technique pour concevoir des champs de vitesses. Un champ de vitesse est en fait une grille où est attribuée à chaque case une vectrice vitesse. Ainsi, lorsqu'un élément de la foule passe au dessus d'une case, le vecteur spécifique à celle-ci lui est affectée et détermine ainsi la direction qu'il va prendre et sa vitesse. Le comportement des agents de la foule est donc ici lié à sa position sur un terrain. Une autre technique utilisant les flux, appelée « Superposition » consiste à combiner plusieurs flux de base, pour générer, par exemple, du vent qui va ainsi diriger plusieurs objets dans la même direction. Cette méthode ne permet cependant pas de gérer correctement les comportements des agents lorsque ceux-ci s'approchent des bords du terrain ou lorsqu'ils rencontrent un obstacle. En effet, ce sont des forces globales et linéaires qui s'appliquent au groupe entier d'individus. Le système de Stephen Chenney présente l'avantage de contrôler la vitesse d'un seul individu donné et non l'ensemble. Il permet également d'appliquer des forces non linéaires. En effet, chaque carreau est composé d'un champ de vitesse (voir la figure 3.3), lui-même composé de plusieurs vectrices vitesses applicables aux agents de la foule. Cela permet entre autre d'avoir des mouvements fluides à l'intérieur d'un carreau mais aussi de pouvoir détourner les objets lorsqu'un obstacle se présente devant eux.

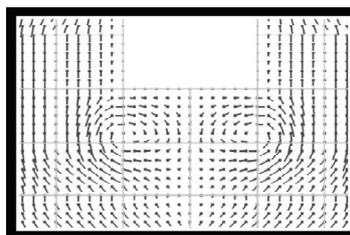


Figure 3.3 : Carreaux de flux et Champs de Vitesse.

Stephen Cheney a donc mis au point une application qui permet à l'utilisateur de créer ses propres terrains à partir de carreaux de flux et d'ainsi pouvoir facilement de larges espaces d'influence.

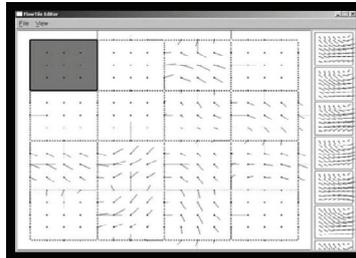


Figure 3.4 : interface du logiciel permettant de choisir des carreaux prédéfinis.

Lorsque l'utilisateur a choisi un carreau de flux, le programme affiche les carreaux susceptibles d'être placés à ses côtés. En effet, il faut que les vitesses sur les bords des carreaux concordent pour éviter les sautes d'animation lorsqu'un agent passe d'un carreau à l'autre.

Ce programme peut aussi bien servir à modéliser le comportement d'objets flottants sur une rivière ou la gestion d'une foule dans les rues d'une ville (voir la figure 3.5). Il est très utile lorsqu'il s'agit de générer des foules massives d'individus car les calculs sont simples.

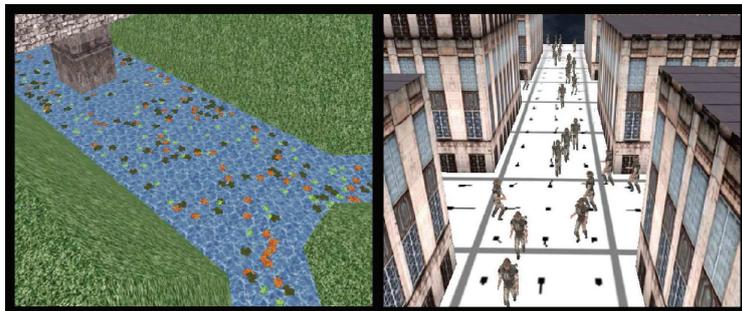


Figure 3.5 : Deux applications du système Flow Tiles.

De plus, il est possible d'utiliser des champs de vitesses dynamiques à savoir que les vectrices vitesses changent au cours du temps (voir figure 3.6)

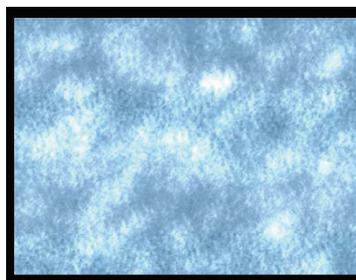


Figure 3.6 : Capture d'écran d'un brouillard tourbillonnant généré à l'aide de champs de vitesses dynamiques.

Le comportement présenté ici sert donc à générer des foules d'individus « sans but précis ». Ils se contentent juste de suivre un chemin, ils ne sont pas acteurs du système. Autrement dit le comportement d'un individu ne modifie pas le comportement global du groupe.

Les modèles AC [Tec01][Kir03][Che04][Tor07], bien que rapide et simple à implémenter, ne tenez pas compte du contact entre les agents. La surface au sol est discrète, et les individus peuvent seulement se déplacer à une cellule libre adjacente. Cette approche de damier offre des résultats réalistes pour des foules de plus faible densité, mais des résultats peu réalistes quand les agents dans des situations à haute densité sont obligatoires dans les cellules discrètes. De plus longs chemins (non-local) plus réalistes dans la grille peuvent être obtenus par précomputation des chemins vers des buts et en les stockant dans la grille [Los03].

3.1.3 Les Modèles basés sur les règles

3.1.3.1 Les « Boids » de Craig Reynolds

En 1986, Craig Reynolds mis au point un algorithme permettant de créer des agents autonomes appelés « Boids » [Rey87]. Ces derniers avaient ainsi l'habilité d'improviser leurs actions et leurs déplacements. Il a ainsi voulu recréer des déplacements d'animaux, tels une nuée d'oiseaux ou un banc de poissons, et ce de façon totalement automatique. La simplicité de ce système a rapidement conduit Craig Reynolds au rang de référence en matière d'animation comportementale.

Le système des Boids permet de modéliser un comportement dit émergent. Il est effectivement impossible de prédire quelle direction prendra une nuée d'oiseaux à une instante t . La complexité comportementale résulte ici de l'interaction des Boids entre eux. Ces derniers doivent respecter un nombre limité de règles simples comme illustré dans la figure 3.7, telles que :

- **La cohésion** : pour former un groupe, chaque Boid doit se rapprocher des autres.
- **La séparation** : chaque Boid doit garder une distance minimale avec ses congénères. Cela permet d'éviter les collisions entre deux Boids.
- **L'alignement** : Les Boids doivent suivre le même chemin pour rester groupés.

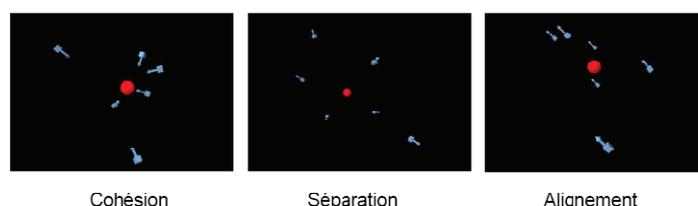


Figure 3.7 : Les 3 règles du système de Reynolds (le point rouge représente le centre du système de Boids).

La combinaison de ces trois règles permet de calculer une force propre à chaque Boid qui lui permet d'avancer dans une direction donnée. L'utilisation de différents paramètres à la fois globaux (qui agissent sur tout le système) mais aussi locaux (propre à chaque Boid, telle la vitesse maximale) permet de décupler les possibilités d'animation. Ce système permet donc de calculer les trajectoires de nombreux éléments et ce sans avoir recours à la méthode du « keyframing ». En effet, il fallait auparavant établir à un instant donné toutes les positions de chaque élément de la nuée. Ce travail, à la fois long et fastidieux, ne permettait pas d'obtenir le même genre de résultat qu'avec les Boids au niveau de la fluidité d'animation. Le système de Craig Reynolds donne à chaque Boid deux fonctions à savoir qu'il est acteur du système mais qu'il subit aussi les lois qui régissent ce système. Bien qu'âgée d'une vingtaine d'années, cette technique reste la plus utilisée pour modéliser des foules d'individus que ce soit en temps réel ou en pré-calculé.

3.1.3.2 Les systèmes de particules

Bouvier (1997) a employé des systèmes de particules adaptés pour étudier des mouvements de foule où des êtres humains sont modélisés comme ensemble interactif de particules. Le mouvement des personnes est basé dessus Forces newtoniennes aussi bien que sur des buts et des décisions humains. Ils ont présenté le concept de la « décision charge » et des « champs de décision » modélisés en employant des notions des charges électriques dans le sens qu'une particule avec une charge électrique sera influencée par un champ électrique comme une personne avec une charge de décision [Mus00b].

3.1.3.3 Mouvements contrôlés par les lois de la dynamique

David Brogan, Ronald Metoyer et Jessica Hodgins sont trois chercheurs américains à l'Institut de Technologie de Géorgie aux Etats-Unis. Ils ont mis au point en 1998 deux installations dans le cadre de leurs recherches sur les interactions de l'humain dans des environnements virtuels [Dav98]. Ils présentent une approche pour peupler des environnements virtuels en utilisant une simulation dynamique pour générer les différents mouvements des personnages qui répondent en temps réel aux actions de l'utilisateur. L'animation est donc entièrement gérée par la machine mais reste cependant liée au comportement de l'utilisateur.

La première installation conçue par ces trois chercheurs est un jeu dans lequel un troupeau d'animaux unijambistes robotiques se déplace dans un environnement virtuel. L'utilisateur navigue avec ce troupeau en conduisant une sorte de pédalo statique en portant des lunettes

dans lesquelles est diffusée l'application en temps réel. Le but du jeu est de rassembler le troupeau de robots dans un enclos, un peu à la manière d'un chien de berger.



Figure 3.8 : Installation.



Figure 3.9 : Robots unijambistes à rapatrier dans l'enclos.

Chaque robot est une entité virtuelle capable de percevoir à la fois ses congénères mais aussi la position de l'utilisateur dans l'espace virtuel. Il utilise cette connaissance pour éviter les collisions avec tous ces individus.

La seconde application est une reconstitution virtuelle du circuit de Cyclisme des jeux Olympique d'Atlanta (1996). Le but était de créer un environnement virtuel où un athlète pourrait s'immerger à la fois visuellement et physiquement avec la présence d'autres cyclistes virtuels (voir la figure 3.10). Un tel système peut s'avérer utile pour les professionnels limités par le temps ou de mauvaises conditions météorologiques. En effet, ils pourraient s'entraîner avec ce genre de programme.



Figure 3.10 : Cyclistes virtuels.

L'utilisateur monte donc sur un vélo de course (voir la figure 3.11) qui tanguent entre -12 et 12 degrés pour permettre la simulation des côtes et de pentes du circuit. Le vélo est muni de capteurs qui mesurent la vitesse de la roue arrière ainsi que la direction de la roue avant (via la rotation du guidon, entre -20 et 20 degrés).

La roue arrière est couplée à un générateur qui permet de la faire tourner au moment où le cycliste se trouve dans une pente. Cela permet d'augmenter l'effet immersif de l'installation, celui-ci étant déjà procuré par le casque que porte l'utilisateur qui diffuse les images de l'environnement virtuel.



Figure 3.11 : Installation.

Le principal intérêt de cet environnement virtuel est la présence de cyclistes virtuels dont le comportement est dynamique. Rien n'est prédéfini. Leurs déplacements se font en fonction des différentes forces appliquées sur leurs vélos, à savoir l'interaction « cycliste- vélo » ainsi que l'interaction « sol-vélo » comme illustré dans la figure 3.12.



La force appliquée sur le guidon dirige le vélo

Les roues tournent sur le sol sans glisser

Les forces appliquées par le cycliste sur les pédales sont transmises à la roue arrière et entraînent le vélo

Figure 3.12 : Forces dynamiques d'un cycliste virtuel.

En ce qui concerne l'animation comportementale, les cyclistes virtuels sont capables d'adopter des stratégies de courses assez sophistiquées comme des blocages, des échappées solitaires, etc...

Cette modélisation de foule est donc très complexe car chaque individu a son propre comportement. Il peut avoir un comportement de groupe ou bien solitaire.

3.1.3.4 Modélisation de foules avec facteur psychologique

Nuria Pelechano, Kevin Brien, Barry Silverman et Norman Badler sont quatre chercheurs américains à l'Université de Pennsylvanie à Philadelphie. Ils ont mis au point une application permettant de simuler l'évacuation d'un immeuble dans une situation de panique comme illustré dans la figure 3.13. Ils ont ainsi inclus à leur foule virtuelle un modèle psychologique [Nur05].

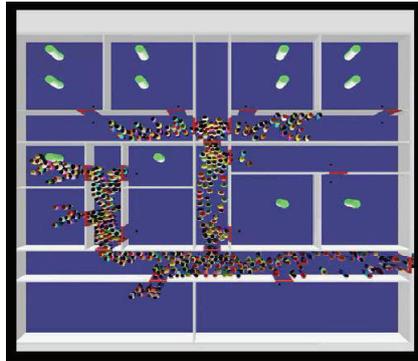


Figure 3.13 : Simulation de l'évacuation d'un immeuble.

Cette simulation est en fait la réunion de deux applications :

- **MACEMACES : Multi Agent Communication for Evacuation Simulation.** Ce programme exécute des algorithmes de détermination de chemins, et permet aux agents de la foule d'explorer un environnement inconnu. Il permet aussi à ces derniers de communiquer entre eux et donc de partager les informations sur l'environnement.
- **PMFserv: Performance Moderator Functions server.** Cette librairie a permis de développer l'aspect psychologique du projet à savoir gérer les différentes émotions des individus et donc leur comportement. Cela permet de contrôler à différents degrés la capacité, le stress, les émotions, les décisions et la motivation des agents et d'ainsi ajouter plus de réalisme au projet.

Les comportements ne sont en aucun cas prédéfinis ou programmés via des procédures. La seule intervention de l'utilisateur réside dans l'attribution de paramètres aux agents. A partir de ceux-ci, les individus réagiront individuellement et mèneront au caractère émergent du système. Le programme place les agents dans différentes pièces. Seuls les agents d'une même pièce peuvent communiquer et partager leurs informations. Différents rôles sont attribués aux agents, ces derniers dépendent de deux attributs : « chef » et « expérimenté » :

- Un **chef expérimenté** connaît la structure interne du bâtiment et va aider les autres personnes pendant l'évacuation.
- Un **chef non expérimenté** correspond aux gens qui arrivent à bien gérer leur stress et qui aident les autres tout en explorant l'immeuble à la recherche de nouveaux chemins.

- Un agent qui n'est *ni chef, ni expérimenté*, est un agent qui dépend des autres et qui panique dans de telles situations au point d'être incapable de prendre des décisions.

Le système en résultant est donc un système multi agents, ce qui correspond à un ensemble d'entités qui s'auto organisent et dont la force réside dans la simplicité de l'individu, le nombre d'agents et la communication. On se rend compte ainsi que tous les individus finissent par se diriger vers la bonne sortie en évitant les obstacles sur leur passage (poteaux ou même des agents tombés à terre) (voir la figure 3.14). On a à faire ici aussi à un système émergent, les comportements individuels finissent par déboucher sur un comportement de groupe non programmé.

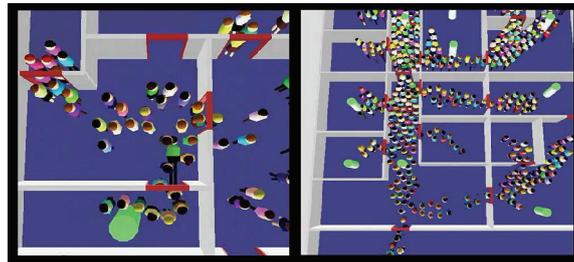


Figure 3.14 : Comportements des agents.

3.1.3.5 Méthode des « Règles locales »

Un modèle comportemental utilisant des règles locales est un système qui applique ses règles sur chaque individu qui compose la foule. L'individu est doté d'une perception des autres et son comportement est calculé en fonction de celle-ci.

Daniel Thalmann est Professeur et Directeur du Laboratoire de Réalité Virtuelle (VRlab) à l'EPFL (Ecole Polytechnique Fédérale de Lausanne). C'est un pionnier dans la recherche sur les Humains Virtuels. Ses intérêts de recherche actuels incluent les Humains Virtuels en temps réel, les Environnements Virtuels Immersifs, la simulation de foules et l'interaction multimodale. Daniel Thalmann a travaillé avec Soraia Raupp Musse sur un modèle comportemental d'êtres humains en temps réel appelé « ViCrowd » [Mus00a] (Virtual Crowd, foule virtuelle). Ce projet fait partie de la thèse de Soraia Raupp Musse. Ce travail est basé sur plusieurs concepts de sociologie et permet par exemple de simuler l'évacuation d'un lieu due à une situation de panique comme illustré dans les figures 3.15 et 3.16 et ainsi de pouvoir gérer par la suite de tels évènements dans la vie réelle à l'aide de l'analyse de la simulation virtuelle.

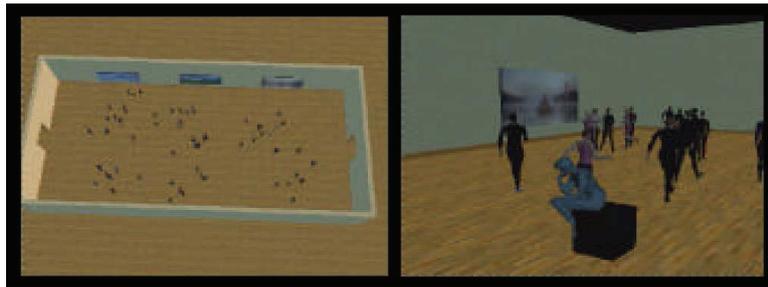


Figure 3.15 : Gestion d'une foule dans un musée.



Figure 3.16 : Gestion de la foule de la Figure 1 après introduction d'un élément de panique.

La hiérarchie de « ViCrowd » se compose de foules virtuelles, de groupes et d'individus. Comparé à d'autres travaux, celui-ci privilégie l'animation de groupe et non la singularité des individus (qui dans d'autres projets similaires sont autonomes et « intelligents »). Ceci permet de réduire considérablement les temps de calculs. ViCrowd est basé sur le principe des « flocking system » (gestion de foules) mais utilise aussi des règles de comportement simples utilisant différentes conditions pour gérer les événements et les réactions des êtres virtuels comme illustré dans les figures 3.17 et 3.18. Ce projet devant représenter les mouvements et réactions d'humains, un modèle sociologique a été utilisé pour faire émerger les affinités et les répulsions entre les individus.

Ces derniers disposent aussi d'un panel d'émotions (« triste », « calme », « état normal », « heureux », « très heureux », etc.) attribuées par l'utilisateur. L'état émotionnel de chaque individu influe sur ses mouvements (posture, vitesse de marche, etc...) (voir la figure 3.17 et la figure 3.18).

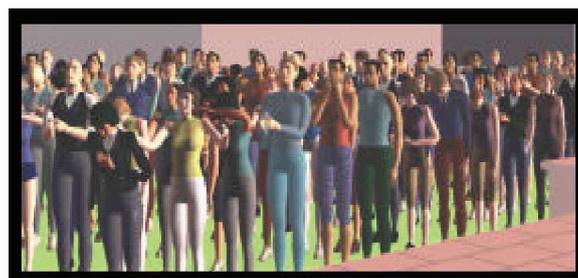


Figure 3.17 : Différents états émotionnels des individus.

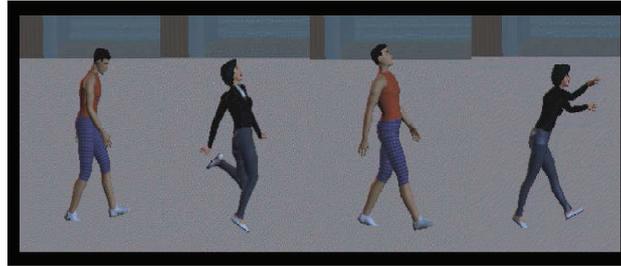


Figure 3.18 : Différentes façon de marcher selon les émotions des agents.

Les émotions et donc le comportement de la foule sont liés au type d'endroit où les agents se trouvent (voir la figure 3.19). Par exemple, dans une gare, ils seront stressés et marcheront plus vite que dans un parc. De plus, certaines actions sont prédéfinies pour les individus. Par exemple, toujours dans une gare, ils pourront acheter un ticket de train, passer un coup de fil, attendre quelqu'un, etc.



Figure 3.19 : Simulation dans une gare.

3.2 Les modèles macroscopiques

Les modèles de simulation macroscopique sont historiquement les premiers modèles informatiques ayant permis des simulations de foules. En effet, ceux-ci simulent la totalité des individus par un ensemble de fonctions mathématiques (une vue globale des comportements de foule comme des flux), ne nécessitant qu'une puissance de calcul assez faible. Ainsi, ces modèles abandonnent tout comportement individuel évolué, au profit d'une simulation très rapide d'un grand nombre d'entités. Nous verrons que le but de ces modèles réside dans la caractérisation d'un phénomène particulier généralement reproductible dans des conditions environnementales figées.

3.2.1 Les modèles de régression

Les modèles de régression utilisent des relations statistiquement établies entre les variables de flux pour prévoir des opérations de flux de piétonnières d'écoulement dans des circonstances spécifiques. Les caractéristiques de ce flux dépendent de l'infrastructure (escaliers, couloirs, etc.) [Mil98][Nur08].

3.2.2 Les modèles de choix de route

Les modèles de choix de route décrivent la manière de trouver le piéton en ce basant sur le concept de l'utilité. Les piétons choisissent leurs destinations pour maximiser l'utilité de leur voyage (confort, temps de déplacement, etc.) [Hoo03] [Nur08].

3.2.3 Les modèles de queue (file d'attente)

Les modèles de queue utilisent les modèles de chaîne de Markov [Lov94] pour décrire comment les piétons se déplacent un nœud dans un réseau à l'autre. Les nœuds sont habituellement des chambres, et donc les liens sont habituellement des portails ou des portes. Des modèles de chaîne de Markov sont définis par un ensemble d'états avec des probabilités de transition. À chaque étape d'extrapolation, un état successeur est sélectionné par un échantillonnage à partir de distribution de transition ou identifiant de successeur le plus probable. Les probabilités de transition d'état sont estimées à partir de la fréquence relative des transitions entre les prototypes de comportement observés dans les données d'apprentissage, en prenant le prototype de comportement le plus proche chaque instant. En considérés seulement les transitions causant le changement d'état [Nur08].

3.2.4 Gaskinetics

Dès 1971, L.F. Henderson propose d'assimiler les déplacements de personnes dans des conditions de faible densité à ceux de molécules de gaz. En se basant sur la théorie cinétique des gaz de Maxwell-Boltzmann, il fait le parallèle entre la densité des piétons et celle des particules d'un gaz, en assimilant leur différence d'états (arrêt, marche, course) aux différences d'énergie des gaz. Ce modèle s'appuie sur deux hypothèses concernant les piétons, en laissant de côté toutes règles sociales (comme les affinités liant certaines personnes) :

- Leurs positions et vitesses sont indépendantes les unes des autres ;
- Leur vitesse est indépendante de leur position.

Ces premiers constats, même s'ils peuvent paraître évidents aujourd'hui, sont à la base de toute simulation d'humanoïdes virtuels.

Donc Gaskinetics utilisé une analogie avec la dynamique de fluide ou de gaz pour décrire comment la densité et la vitesse de foule changent durs avec le temps utilisant des équations différentielles partielles [Hen71].

3.2.5 Modèle hydraulique

Un modèle hydraulique a été proposé par J. Archea [Arc79] pour simuler des foules de forte densité. Il assimile ici le mouvement des personnes au travers de couloirs, escaliers, et portes, à celui de l'eau au travers de tuyaux, vannes, ou autres. La simulation de l'évacuation du bâtiment se fait en deux étapes :

- Initialisation du mouvement de l'ensemble des individus recevant simultanément un message d'alerte (ouverture des vannes) ;
- Simulation de l'évacuation (écoulement de l'eau dû à la force de gravitation).

Ce modèle se base sur les hypothèses suivantes :

- Prise de conscience et décision de l'évacuation immédiate et simultanée de tous les individus.
- Répartition uniforme des occupants.
- Débits uniformes.
- Flux unidirectionnels.
- Trajets prédéfinis.

Même si ce modèle a trouvé de nombreuses applications, notamment du fait de sa facilité de mise en œuvre, il souffre de la simplicité des hypothèses sur lesquelles il repose : en réalité les personnes ne partent pas en même temps, et leurs directions ne sont pas aussi facilement prévisibles.

3.2.6 Modèle de Togawa

K. Togawa [Par07] propose un modèle basé sur des observations réalisées sur une période de dix ans dans des établissements recevant du public. Il donne ainsi une formule permettant de calculer la vitesse de marche d'une foule par rapport à sa densité :

$$V = V_0 \cdot \rho^{-0.8}$$

- Avec $V_0 = 1.3 \text{ m.s}^{-1}$, $\rho =$ densité en personnes/m²

Le débit est alors déduit par :

$$N = \rho \cdot V = V_0 \cdot \rho^{0.2}$$

- Avec N=débit en personnes/s/m, $1.3 < N < 1.7$

K. Togawa suggère quelques modifications suivant le degré d'inclinaison du plan de circulation :

- Si le sens est ascendant, V est réduite de 2% par degré d'angle d'inclinaison ;

- Si le sens est descendant, la vitesse augmente pour des pentes faibles, mais diminue pour des pentes fortes.

Ce modèle a été appliqué dans le cadre de la simulation de l'évacuation d'un immeuble en cas d'incendie. Le produit de sortie de cette simulation est un temps total d'évacuation, défini comme la somme des temps :

- De détection du feu ;
- D'alarme ;
- De préparation ;
- Et d'évacuation net, qui est lui-même la somme des temps d'évacuation des escaliers et des circulations horizontales.

Le temps d'évacuation des circulations horizontales est calculé ainsi :

$$T = \frac{P}{C_c \cdot \sum L_e} + \frac{L_h}{v}, \text{ avec}$$

- T = temps d'évacuation en secondes
- P = nombre de personnes dans le bâtiment
- L_e = largeur des zones de circulation et des issues en mètres
- C_c = coefficient de circulation (évalué à 1.3 pers. / m / s)
- L_h = longueur totale maximale des chemins d'évacuation
- v = vitesse de circulation (généralement 0.6 m / s)

Le temps de parcours d'un escalier est quant à lui calculé par, quel que soit le sens de parcours :

$$t = 4H, \text{ avec}$$

- t = temps de parcours en secondes, et H = hauteur d'un étage en mètres

K. Togawa [Par07] remarque que, pour l'évacuation des établissements à plusieurs étages, la largeur seule des escaliers n'est pas suffisante pour prédire leur débit. Il pense que les paramètres les plus significatifs sont la hauteur des marches et l'allure des personnes. Il constate que le temps de parcours d'un escalier est inversement proportionnel à son angle et sa hauteur, et que la vitesse moyenne en descente est de 0.5 m/s. Pour conclure, le modèle de K. Togawa propose un ensemble de règles permettant une évaluation approximative du débit maximal admissible en sortie d'un bâtiment. Il semble difficile d'intégrer telles quelles ces règles dans un modèle plus complexe, mais elles peuvent tout de même servir comme valeur de comparaison lors d'une première étape de validation.

3.2.7 Modèle de Melinek et Booth

S. J. Melinek et S. Booth [Par07] ont amélioré le modèle de K. Togawa en prenant en compte les éventuels phénomènes de saturation. Ils découpent un bâtiment en cellules (une par étage) et calculent le temps d'évacuation pour chacune d'elles. Les personnes sont supposées être à proximité des escaliers, et leur flot continu et ne rencontrant pas d'obstacle. Il est aussi admis que les personnes quittant le rez-de-chaussée ne réduisent pas le débit des niveaux supérieurs. Il est défini :

- Q_r = occupation du niveau r en personnes
- b_r = largeur de l'escalier entre les niveaux r et $(r-1)$ en mètres
- $N0$ = débit unitaire par mètre de largeur des escaliers descendants (généralement 1.1 pers / m / s)
- t_s = temps mis par une personne pour descendre librement un étage (généralement 16 s)
- La population P_r des niveaux r et supérieur se calcule ainsi :

$$P_r = \sum_{i=r}^n Q_i$$

- Le temps minimum pour descendre l'escalier situé entre les niveaux r et $(r-1)$ est :

$$\frac{P_r}{N' \cdot b_r}$$

- Le temps pour que la queue de la foule atteigne le rez-de-chaussée est : $r \cdot t_s$
- Le temps minimum d'évacuation T_r des niveaux r et supérieurs est donc :

$$T_r = \frac{P_r}{N' \cdot b_r} + r \cdot t_s, \quad r \in \{1, \dots, n\}$$

Soit :

$$T_r = \sum_{i=r}^n \frac{Q_i}{N' \cdot b_r} + r \cdot t_s, \quad r \in \{1, \dots, n\}$$

Dans les cas où plusieurs escaliers sont disponibles à un étage r , b_r est considéré comme étant la somme de leurs largeurs. En supposant que le nombre d'occupants et la largeur d'escaliers soient identiques à chaque niveau, on pose $Q = Q_r$ et $b = b_r$. On obtient donc le temps minimum T_r d'évacuation des étages r et supérieurs par :

$$T_r = \frac{(n-r+1) \cdot Q}{N' \cdot b} + r \cdot t_s$$

Deux cas sont alors possibles pour le temps net d'évacuation T_e :

$$\frac{Q}{N' \cdot b} > t_s \quad T_r \text{ est maximum pour } r = 1$$

$$\frac{Q}{N' \cdot b} < t_s \quad \begin{cases} T_e = \frac{n \cdot Q}{N' \cdot b} + t_s & (\text{saturation à chaque niveau}) \\ T_r \text{ est maximum pour } r = n \\ T_e = \frac{Q}{N' \cdot b} + n \cdot t_s & (\text{aucun saturation}) \end{cases}$$

Malheureusement, les temps prédits par ce modèle sont généralement bien inférieurs à ceux observés en situation réelle (avec un écart d'environ deux minutes). S. J. Melinek et S. Booth proposent donc de rajouter systématiquement ces deux minutes aux temps calculés car ils estiment qu'elles correspondent au délai de réponse à l'alarme ainsi qu'au temps d'accès aux escaliers. Comme pour le modèle de K. Togawa, le modèle présenté ici fournit des approximations quant aux débits maximum de personnes dans des situations particulières. Les corrections approximatives apportées par S. J. Melinek et S. Booth [Par07] peuvent amener à douter quant à la généralisation de leur modèle. En effet, il est difficilement concevable de pouvoir approximer la complexité du comportement humain par une approximation constante du temps de réponse à un stimulus externe. Ainsi, d'autres situations nécessiteront certainement des corrections différentes.

3.2.8 Modèle de Paule

J. L. Pauls [Par07] oriente son étude sur l'observation d'exercices d'évacuation d'immeubles de bureaux de grande hauteur au Canada. Celui-ci justifie l'utilisation de son modèle par le fait que de tels exercices, menés en situations réelles, peuvent être considérés comme potentiellement dangereux. Son étude porte surtout sur l'évacuation au niveau des escaliers, en établissant la loi suivante :

$$f = 0.206 \cdot \rho^{0.27}, \text{ Avec}$$

- f = débit unitaire en pers /s/m de largeur d'escalier
- ρ = nombre de personnes par mètre de largeur d'escalier

J. Pauls [Par07] affine son modèle quantitatif par des précisions qualitatives provenant de diverses observations (vidéos, photographies). Il développe ainsi le concept de largeur effective d'un escalier en constatant qu'un espace d'une quinzaine de centimètres est systématiquement laissé entre les évacuants et les murs latéraux. La largeur effective correspond donc à la largeur réellement utilisée d'un escalier, soit la largeur réelle diminuée de trente centimètres. Ainsi, il indique que la formule permettant de calculer le débit est :

$$Q = 0.206 \cdot (L - 0.3) \cdot \left(\frac{P}{L - 0.3} \right)^{0.27}, \text{ avec}$$

- Q = débit en pers/s
- L = largeur effective en mètres
- P = nombre de personnes évacuant ($P = p \cdot L$)

Le temps d'évacuation est alors lié à la population initiale du bâtiment et à la largeur effective des escaliers par :

$$\frac{L}{P} = 8040 \cdot T^{-1.37}, \text{ avec}$$

- L = largeur effective en mètres
- P = nombre de personnes évacuant
- T = temps d'évacuation en secondes (au maximum 600)

3.2.9 Le modèle de Predtechenskii et Milinskii

V. M. Predtechenskii et A. I. Milinskii [Par07] proposent un modèle fluide déterministe basé sur l'analyse statistique d'un très grand nombre de mouvements de foules. Ces travaux ont eu un impact fort en Europe, et continuent encore aujourd'hui à être utilisés aux États-Unis, notamment par la Society of Fire Protection Engineers. Ils introduisent les notions d'intensité et de débit unitaire (produit de la densité et de la vitesse du mouvement) afin de déterminer la capacité d'une issue.

La grandeur fondamentale de ce modèle est la densité locale de la foule, exprimée comme le rapport entre la somme des projections horizontales des individus et la surface totale occupée par la foule. L'encombrement d'une personne est ici calculé en fonction de son physique, de son âge ainsi que de sa tenue vestimentaire, et correspond environ à une ellipse :

$$s = \frac{\pi}{4} \cdot l \cdot e, \text{ avec}$$

- s = surface au sol occupée par l'individu en m²
- l = largeur de l'individu en mètres
- e = épaisseur de l'individu en mètres

3.2.10 Le modèle de Takahashi, Tanaka, et Kose

K. Takahashi, T. Tanaka et S. Kose [Par07] proposent un modèle en 1988 au Japon. Ils décrivent un bâtiment sous la forme de huit types d'éléments, dont deux représentants des espaces virtuels :

- Une pièce est un espace pourvu d'issues, contenant initialement des personnes.
- Un couloir est un espace pourvu d'une entrée et d'une sortie, initialement vide. Il peut être divisé en unités pour les besoins du calcul.
- Un escalier est un espace reliant entre eux deux étages, fonctionnant comme un chemin.
- Un vestibule est un espace avec une entrée et une sortie vers un escalier, fonctionnant comme un chemin et permettant, par comptage, de juger de l'état d'évacuation d'un étage.
- Un hall est un espace à entrées et sorties multiples, connectées par des chemins multiples.
- Un refuge est une destination finale d'évacués.
- Un lien est un espace imaginaire sans capacité qui régule un flot d'évacués à l'intersection de deux éléments.
- Une zone d'entassement est un espace imaginaire qui reçoit les évacués ne pouvant passer immédiatement à l'élément suivant.

Le modèle suppose des mouvements homogènes d'évacuants (vitesse constante, modèle fluide) et une répartition initiale uniforme dans les pièces. Au départ de l'évacuation, les occupants de chaque pièce vont vers la sortie, en privilégiant la plus proche si elles sont multiples. Ainsi, une pièce avec plusieurs sorties sera découpée en unités. Pour les cas où l'issue ne se trouve pas dans un coin, on découpe celle-ci en deux unités.

4. Les systèmes récents de simulation piétonniers

Les modèles commerciaux de la circulation de piétonniers sont dans les classes microscopiques et macroscopiques. La classe microscopique étudie les caractéristiques de différents piétons telles que la vitesse et l'interaction avec autre piétons, alors que la classe macroscopique est concernée par des groupes de piétons plutôt que différentes caractéristiques, et leur analyse se concentre sur une densité, systèmes de grand échelle. Puisque nous sommes principalement intéressés par les simulations où chaque agent est conduit par ses propres buts, sa propre personnalité et un processus décisionnel. Nous nous concentrerons sur les systèmes microscopiques.

Cette section examine certains systèmes qui sont les plus appropriés pour la simulation de foule qui ont été développés dans l'industrie et dans le milieu universitaire.

Exodus : il a été développé par le groupe « Fire Safety Engineering Group » à l'université de Greenwich [Gal93] [Owe98]. (Les figures 3.20 et 3.21 illustrent quelques projectiles d'écran du commutateur d'Exodus) [Nur08].



Figure 3.20 : Distributions de densité pendant une évacuation dans l'exode [Gal93].

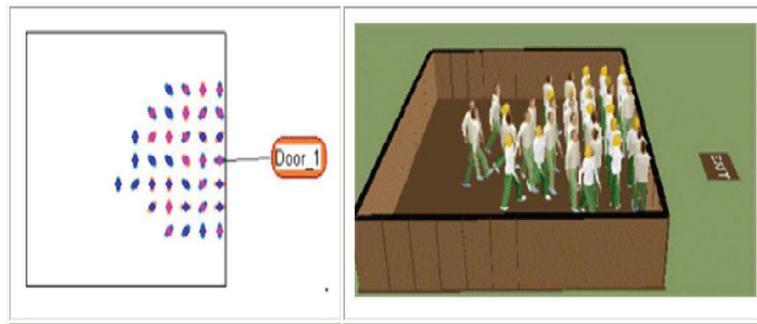


Figure 3.21 : Simulation d'une évacuation avec l'exode de bâtiment et de sa visualisation en différé dans 3D avec le vrExodus [Owe98].

Pedroute : il a été à l'origine développé par « London Underground Limited » et il a été utilisé intensivement pour modéliser des paramètres de foule dans les réseaux souterrains à travers le monde [Buc94] [Nur08].

CROSSES (Crowd Simulation System for Emergency Situations) est un système de simulation de foule pour des situations d'urgence, vise à fournir un outil de réalité virtuelle pour former des personnes pour répondre effectivement aux situations d'urgence urbaines en employant un modèle pour produire et simuler une foule virtuelle [Uli,Th01]. Ce système multi-agents fournit des agents autonomes et des comportements préétablis avec l'environnement virtuel et avec les vrais participants humains [Nur08].

Simulex: il a été développé comme modèle d'évacuation avec les possibilités de la simulation d'un grand nombre de personnes dans les bâtiments géométriquement complexes [Tho95]. La figure 3.22 montre la représentation de la distance et la simulation d'une foule [Nur08].

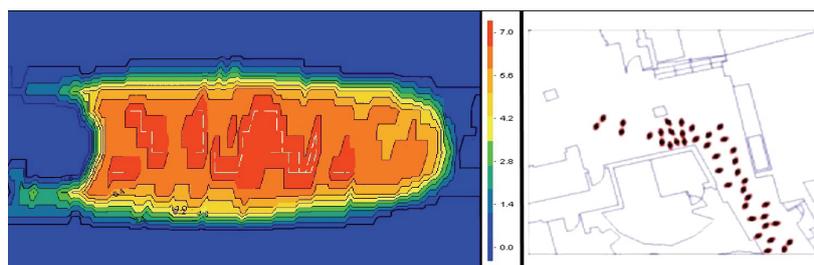


Figure 3.22 : Densité et simulation de foule dans Simulex.

Rampage : c'est un système de simulation de particules pour animer des explosions et d'autres primitifs élémentaires ; il divise le comportement humain en réactions réflexes et la prise de décision basée sur la connaissance obtenue à partir de la scène [Bou95]. Ses principes pour des simulations de comportement humain sont basés sur l'équation de gaz de Boltzmann [Nur08].

Egress : la technologie d'AEA [Ahn06] a lancé le développement de l'Egress en 1991 (technologie d'AEA 2002). C'est un logiciel commercial pour la simulation de foule. Le modèle utilise des techniques d'intelligence artificielle pour déterminer comment une personne réagirait dans une série de circonstances de feu et de fumée. Le rendement de l'Egress inclut l'analyse de temps d'évacuation [Nur08].

Legion : ce système n'a pas été conçu comme système d'analyse comportementale de foule mais en tant que d'investigation pour l'étude des systèmes interactifs de grande échelle. Le modèle informatique simplifie la représentation comportementale des individus. D'abord, le modèle utilise seulement quatre paramètres (point de but, vitesse, distance à d'autres, et temps de réaction) et une règle de décision (fondée sur l'hypothèse du principe de moindre-effort) pour représenter la nature complexe de l'individu comportements. La figure 3.23 illustre la simulation de légion d'une station de train avec le système Legion [Nur08].

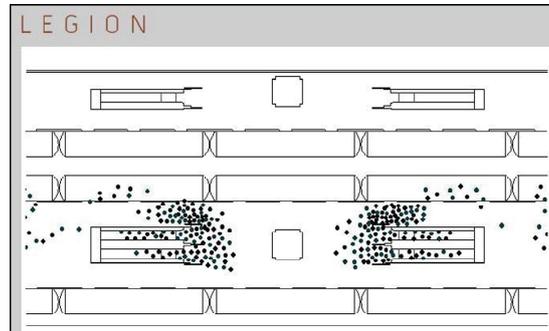


Figure 3.23 : Simulation de légion d'une station de train (légion 2003).

STEPS : c'est un modèle à base d'agent avec la géométrie des automates cellulaires (voir la figure 3.24) [Mot03]. Chaque individu occupe une cellule à un moment donné et se déplace vers une direction désirée si la prochaine cellule est vide. Chaque cellule occupée, a ses propres caractéristiques, facteur de patience, et comportement habituel [Nur08].

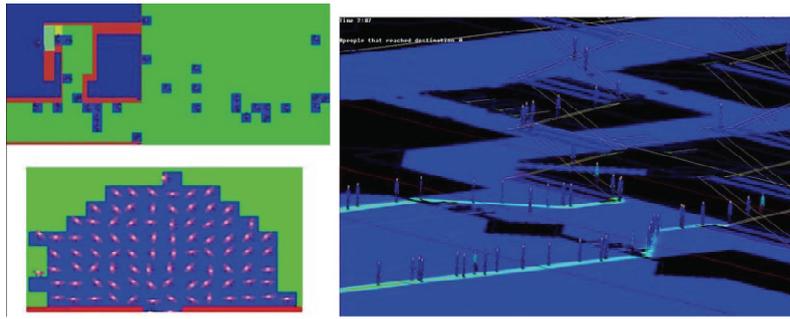


Figure 3.24 : simulation 2D par STEPS.

ViCrowd : représente un modèle pour produire automatiquement des foules humaines basées sur les propriétés de groupe au lieu des individus [Mus, al 98]. Les individus dans un groupe suivraient les caractéristiques des groupes au lieu de leurs propres spécifications pour satisfaire les conditions en temps réel. Il est basé sur le système de flocage de Reynolds mais inclut une définition simple des règles comportementales utilisant des événements et des réactions conditionnels. Un modèle sociologique est employé pour manipuler les effets d'affinité et de répulsion qui peuvent émerger dans la simulation de foule et créer des comportements plus complexes.

OpenSteer : fournit une trousse à outils des comportements de direction définis en termes d'agent mobile abstrait (voir la figure 3.25) [Rey99] [Nur08].

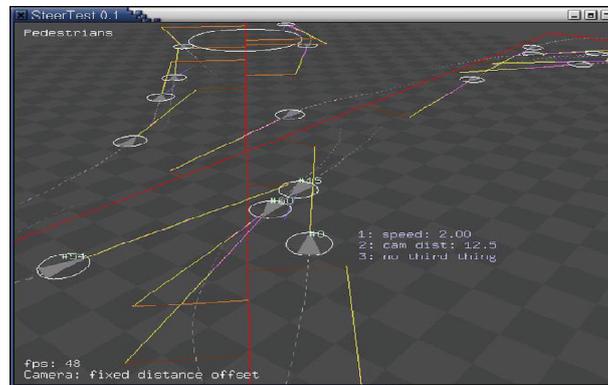


Figure 3.25 : L'interface d'Open street [Rey99].

Massive SW. Ce système est basé sur la technologie de la vie artificielle, utilisant une combinaison des règles simples avec la logique floue, développée par Massive Software, Inc. (Massive Software, Inc. 2005). Pour plus des détails voir le chapitre 4 qui est consacré à la présentation de Massive. La figure 3.26 montre la simulation d'évacuation d'une foule [Nur08].

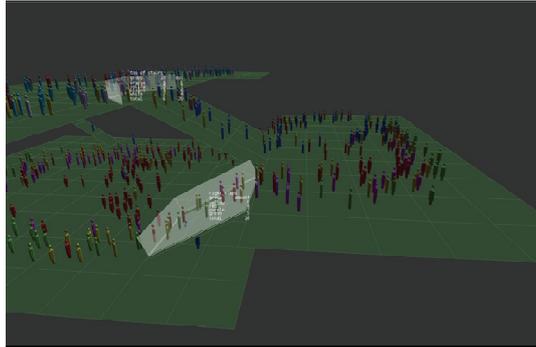


Figure 3.26 : Simulation d'évacuation avec Massive WS pour la conception architecturale et l'évaluation.

Reactive Navigation : les agents de Lamarche et de Donikian (2000, 2004) se déplacent dans les environnements virtuels complexes représentés avec une structure topologique hiérarchique extraite à partir de la géométrie de l'environnement virtuel (voir la figure 3.27). Cette structure permet la recherche rapide de chemin aussi bien qu'un algorithme réactif efficace de navigation. Pour éviter la collision tout en atteignant leurs cibles, ils emploient un processus itératif d'optimisation. La prévision de collision est basée sur des calculs de voisinage ; elle crée des relations de voisinage dans les foules clairsemées et des relations de courte distance dans les foules denses [Nur08].



Figure 3.27 : Navigation réactive [Tho00].

Artificial fishes : le TU et le Terzopoulos (1994) ont développé une approche de la vie artificielle pour simuler l'aspect, le mouvement, et le comportement de différents poissons dans le monde marin virtuel et également les comportements complexes de groupe observés dans de vrais écosystèmes aquatiques (voir la figure 3.28). Chaque poisson se comporte comme agent autonome montrant le comportement de poissons [Nur08].



Figure 3.28 : Les poissons artificiels.

ACUMEN : c'est un système pour synthétiser et identifier les mouvements globaux dans un environnement virtuel avec une interface de langage naturel [All02]. Ses composants principaux incluent une interface interactive pour la commande globale basée sur une collection de paramètres d'élargissement d'un modèle existant.

Le système ACUMEN a employé la représentation paramétrisée d'action [Bin00] pour capturer la sémantique du mouvement global pour la génération et l'identification. Le mouvement est basé en grande partie sur un système de particule comme le modèle de la simulation de groupe, utilisant les forces dynamiques agissant sur les corps rigides de produire le mouvement désiré (voir la figure 3.29) [Nur08] [Chi00] [Mal87].

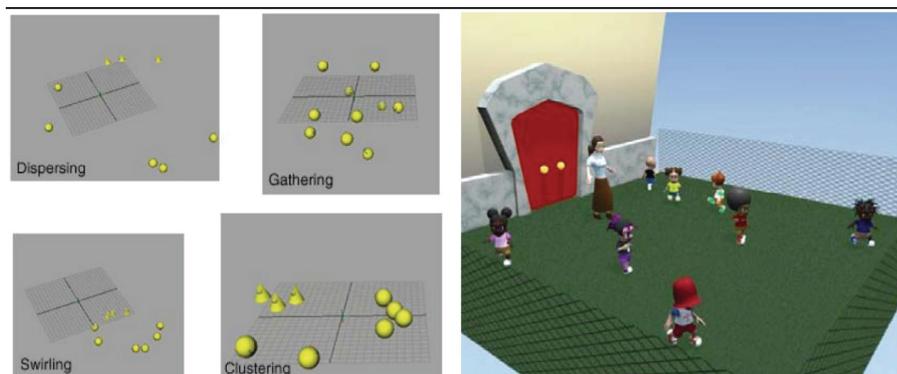


Figure 3.29 : Le système ACUMEN est appliqué aux sphères et les écoliers [All02].

Autonomous Pedestrians : il a été préconisé par Terzopoulos, il contient un moteur d'intégration de l'approche de la vie artificielle. Il intègre des individus perceptuels, des comportements, et des composants cognitifs dans un modèle des piétons [Sha05]. L'environnement est représenté par les structures de données hiérarchiques qui soutiennent efficacement des requêtes perceptuelles sur des piétons autonomes qui conduisent leurs réponses comportementales et maintiennent leur capacité de prévoir leurs actions à un niveau local et global [Nur08].

Les agents exécutent six routines réactives de base de comportement : évitez l'obstacle statique, évitez l'obstacle statique à un tour complexe, maintenir la séparation dans une foule mobile, éviter les piétons approchants, éviter les piétons dangereusement étroits, et vérifier les obstacles relatifs de nouvelles directions. Les agents exécutent l'action d'évitement de collision; donc, si une intersection avec un autre piéton est sur le point de se produire, qu'ils détectent en employant « un secteur sûr avant, » l'agent s'arrêtera, essaiera de tourner pour faire face loin, et pour attendre jusqu'à ce que l'espace soit disponible autour de sa position actuelle (voir la figure 3.30) [Sha07] [Yu07].

Des arbres de décision sont construits pour différents comportements, et leurs probabilités sont accordées pour obtenir les comportements désirés.



Figure 3.30 : Les piétons autonomes [Sha05].

Space syntax : les planificateurs urbains ont proposé l'analyse de graphique de visibilité pour le mouvement piétonnier [Tur07] (voir la figure 3.31). Cette méthode examine comment les points visuellement accessibles soient dans un secteur pour déterminer la direction probable de la navigation d'un piéton. Cette approche employer le champ visuel pour guider le mouvement normal, sans considérer d'autres facteurs socio-économiques (atteignant pour une destination, rencontrant des personnes, prenant quelque chose, etc.) ou physique granulaire (particules mobiles vers un but, poussée, formant des ruelles, etc.). Pour simuler une foule relativement grande, ils calculent « une architecture visuelle exo-somatique » où les raccords entre les endroits mutuellement évidents dans une configuration sont préenregistrés dans une table de consultation [Nur08].

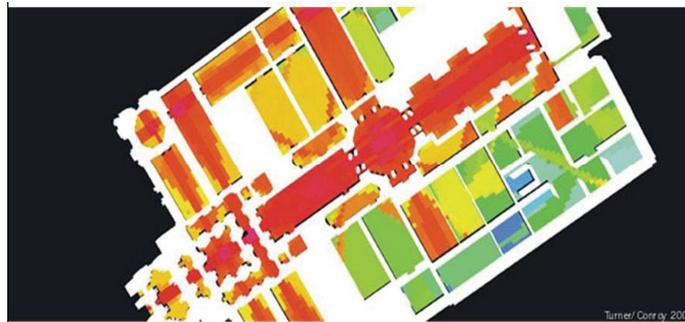


Figure 3.31 : Graphique de visibilité de syntaxe de l'espace [Tur02].

5. classifications des méthodes de foule

Après avoir passé en revue les études dans le domaine des foules, nous proposons dans cette section deux classifications pour les travaux effectués autour de cette partie. La première est liée aux méthodes employées pour représenter les mouvements des foules et des comportements. La seconde propose, quant à elle, une classification des foules selon leurs applications.

La classification suivante décrit les méthodes, qui peuvent être employées pour représenter les foules:

- Méthodes de foules à base d'image: définissent les méthodes de transformation de l'image et de vision à saisissent l'information de vraies foules.
- Modèles de foule à base physique: utilisés pour décrire le mouvement de foule sur la base de fondements de la physique.
- Modèles procéduraux: basés sur la description d'équations paramétriques.
- Modèles comportementaux: fournissent un outil pour simuler des foules sur des fondements basés comportement.

Le tableau 3.2 représente une classification de plusieurs méthodes et systèmes de modeler de groupes ou de foules. Différent paramètres caractérisent chaque méthode comme la communication entre les individus, l'apprentissage et l'utilisation de temps réel.

	Communication ou signale	Individus ou des rôles	Temps réel	Apprentissage	Structure spatiale pour le mouvement
Modèles de forces sociaux	Non	Certains	Oui	Non	Cont
Basée sur les règles	Non	Non	Oui	Non	Cont
AC	Non	Non	Oui	Non	Grille 2D
Simulex	Non	Certains	Non	Non	Cont
Egress	Non	Certains	Non	Non	Grille hexagonal
ViCrowd	Non	Oui	Oui	Non	Cont
OpenStreet	Non	Certains	Oui	Non	Cont
Legion	Certains	Certains	Non	Non	Cont
Exodus	Non	Certains	Non	Non	Grille 2D
Steps	Non	Certains	Non	Non	Grille 2D
Massive SW	Non	Oui	Non	Non	Cont
Reactive navigation	Non	Oui	Oui	Non	Cont
Artificial fishes	Non	Oui	Oui	Oui	Cont
ACUMEN	Oui	Oui	Oui	Non	Cont
Crosses	Non	Oui	Oui	Non	Cont
Autonomous Pedestrians	Non	Oui	Oui	Non	Cont
Space Syntax	Non	Non	Oui	Non	Cont
MACES + HiDAC	Oui	Oui	Oui	Oui	Cont

Tableau 3.2 : Les différentes approches dans le domaine [Nur08].

Les méthodes présentées dans le tableau 3.2 peuvent être choisies en fonction de la nature des foules requises dans les applications. En effet, il y a différents types de contraintes selon le but de l'application. Par exemple, si l'application consiste à modéliser un mouvement de foule pendant une évacuation, aussi bien la commande interactive que la simulation de

comportement ne peut être importante. Nous avons classifié ces applications selon quatre types et ce sur la base des différents buts:

- **Les applications** de divertissement incluent des productions et des jeux de film. Dans ce cas, des groupes aussi bien que des individus doivent être facilement commandés et dirigés. Quelques exemples sont les films récents "AntZ" [Ant98] et "désinsectise la vie" [Bug]. Les défis incluent la simulation de plusieurs personnes aussi bien que la génération automatique des individualités. En outre, la commande facile des groupes représente une condition de ce genre d'application.

- **La simulation de mouvement de foule** vise à évaluer le mouvement de plusieurs personnes dans un environnement contraint. Elle implique normalement une visualisation simple (incorporation de foule) et un compromis fort avec les résultats numériques afin d'évaluer l'environnement. Quelques logiciels commerciaux, par exemple SIMULEX [TM95] et LEGION [Sti96] sont des exemples de ce genre d'application de foule. Les défis principaux dans ce cas incluent une méthode réaliste d'action pour l'évitement de collisions, un raccordement fort avec l'environnement et a compromis avec les résultats numériques et statistiques.

- afin de **peupler les environnements virtuels de collaboration** par exemple dans un système de CVE (collaborative virtual environment), la foule a besoin d'une simulation en temps réel, aussi bien que beaucoup d'aspects de l'interactivité pour guider et diriger les foules pendant la simulation. La programmation du comportement de foule représente un défi important dans les environnements virtuels de collaboration, où les participants peuvent créer et agir l'un sur l'autre avec des foules. Quelques expériences ont été faites dans le contexte du projet de COVEN [TS97].

- le dernier type d'application de foule est la **modélisation comportementale des foules**. Dans ce cas, le but est de fournir les comportements autonomes ou semi autonomes qui peut être appliqués pour une self-animation des agents formant la foule. Afin de traiter les contraintes dues au grand nombre d'acteurs (centaines ou milliers), ce genre d'application présente les défis suivants: description des aspects sociologiques et sociaux qui surgissent dans les foules, en traitant les comportements et le raccordement dirigés et émergents avec environnement virtuel. En outre, des optimisations et des simplifications sont exigées afin de modéliser et simuler l'intelligence de foule et les capacités de décision.

6. Synthèse sur les modèles de foule

Il y a eu de recherche et développement considérable dans la simulation de foule ou de groupe, spécialement étudié la dynamique d'évacuation. Néanmoins, il reste des points pour l'amélioration. Le centre principal des applications commerciales est de validation leurs systèmes en termes de sortie (débits, densités, secteurs de congestion, temps d'évacuation, etc.). Ils emploient des approches macroscopiques ou microscopiques. Les modèles microscopiques les plus utilisés généralement dans les applications d'industrie sont les simulations de particules et modèles d'automate cellulaire (AC). Ces méthodes se sont avérées manquer du réalisme quand elles sont appliquées aux humains virtuels 3D pour des systèmes d'animation parce qu'elles l'un ou l'autre regard plus près des particules que au vrai mouvement d'humain (les modèles de forces sociales et simulation de particules en général) ou sont limités aux configurations de damier (AC).

En revanche, la recherche pour développer les agents autonomes s'est concentrée sur leur capacité de diriger de grands environnements virtuels complexes tout en évitant des obstacles statiques et d'autres agents. Cependant, la plupart des cas, ignorent les problèmes qui surgissent en ayant affaire avec les foules à haute densité. Ces systèmes appliquent habituellement quelques ensembles de règles pour éviter la collision basée sur modifier la vitesse ou la trajectoire des agents. En conséquence, ces modèles sont suffisants pour les foules moyen-basse densité. Cependant, quand la foule est très dense, ils rapportent le comportement émergent artificiel tel que des individus cessant et attendant l'espace pour éclaircir. Il n'y a aucun concept de contact de corps-à-corps menant à pousser des agents dans une foule ou des individus traîné par la foule. D'autres effets comme tomber, dommages, incapacitation, et d'autres qui marchent au-dessus de l'agent tombé sont également ignorés. Le système HiDAC il a en tant que son objectif principal à traiter toutes ces configurations qui émergent dans de vraies foules à haute densité [Pe107].

En termes de navigation globale, les systèmes édités pour la simulation de foule supposent que des agents aient les informations complètes sur l'environnement. Un agent peut accéder à la structure interne entière des algorithmes d'environnement et d'utilisation tels que A* (plusieurs techniques ont été exploitées à réalisez le temps réel en effectuant la planification globale de chemin pour de grands groupes d'agents), ou bien l'environnement est discrétisé comme grille qui stocke des potentiels ou des cartes de distance que les agents suivront atteindre localement le but. Il est essentiel de procurer à un agent d'explorer les environnements en partie connus et apprenez les nouveaux dispositifs. Un autre aspect crucial

des foules qui est ignoré ailleurs est ces personnes ont la capacité de communiquer avec d'autres pour échanger l'information de navigation saillante. Notre travail se concentre également sur améliorer le réalisme du comportement des agents en les permettant pour avoir des informations partielles sur l'environnement et pouvoir prolonger leur mémoire (ou cartes mental) comme ils explorent l'environnement et communiquent avec d'autres individus dans la foule. Les agents peuvent également montrer différents comportements basés sur différents rôles [Pel06]. En conclusion, quelques facteurs psychologiques doivent être incorporés en vue de la modification l'exécution globale d'un individu basé sur son état mental

Le tableau 3.3 montre une comparaison de certains des modèles les plus significatifs dans l'animation de foule. Ceci souligne les caractéristiques principales dans des simulations multiagent et des évacuations piétonnières pour comparer les contributions de notre modèle (MACES + HiDAC) à d'autres.

	Réponse de collision	Secousse de particules corrigée	Méthode de comportement
Modèles de forces sociaux	oui	Non	Les forces
Basée sur les règles	Non	Pas besoin	Les règles
AC	Non	Pas besoin	AC
Simulex	Oui	Non	Cartes de distance
Egress	Non	Pas besoin	AC
ViCrowd	Non	Pas besoin	Les règles
OpenStreet	Non	Pas besoin	Les règles
Legion	Non	Pas besoin	Moindre-effort
Exodus	Non	Pas besoin	AC
Steps	Non	Pas besoin	AC
Massive SW	Non	Pas besoin	Les règles + la logique floue
Reactive navigation	Non	Pas besoin	Les règles
Artificial fishes	Non	Pas besoin	Les règles
ACUMEN	Oui	Non	Simulation de particules
Crosses	Non	Pas besoin	Les règles
Autonomous Pedestrians	Non	Pas besoin	Approche vie artificielle
Space Syntax	Non	Pas besoin	Graphiques de visibilité
MACES + HiDAC	Oui	Oui	Forces sociales prolongées

Tableau 3.3 : Comparaison de différents systèmes pour l'animation de grands groupes.

7. Conclusion

Nous avons présenté dans ce chapitre les différents modèles et techniques existant pour la simulation de foule virtuelle, ici nous avons remarqué que la simulation comportementale et la création d'un environnement virtuel habité et peuplé par un grand nombre de piétons d'une manière réaliste est introduite dans plusieurs domaines d'applications tels que : le graphisme, la robotique, les jeux vidéo et les films. Dans ce contexte deux approches ont été traitées qui sont les méthodes microscopiques et macroscopiques. La méthode macroscopique étudie la foule dans sa totalité et ne considère pas l'interaction entre les individus, et elle est caractérisée par son faible coût de calcul, mais ils ne prennent pas en compte ni la perception visuelle, ni les règles sociales.

CHAPITRE 4

Massive

« C'est en postulant l'impossible qu'on obtient
parfois le possible » Wolfgang Goethe

Massive : outil de modélisation et simulation de foule basée sur la logique floue

1. Introduction

L'animation de personnages d'une manière réaliste est un défi majeur des applications de réalité virtuelle. Il devient en effet nécessaire de proposer aux utilisateurs des mondes qui ne soient plus de simples vitrines statistiques mais des environnements crédibles peuplés. De plus de nouvelles applications de simulation s'appuient désormais sur une modélisation de l'humain aussi bien au niveau de sa représentation physique, qu'au niveau de ses capacités comportementales. Le fonctionnement de comportement d'un être humain est représenté par la boucle perception-décision-action qui est une notion intuitive de l'architecture décisionnelle de piéton. L'humain voit ce qui se passe autour de lui, décide de son action en fonction de sa perception et de ses connaissances, et agit sur lui-même (par exemple en apprenant une nouvelle notion) ou pour modifier son environnement (par exemple, il peut déplacer un objet). L'organisme humain est en interaction constante avec son environnement par le biais de capteurs (perception) et d'effecteurs agissant directement sur l'environnement.

2. Le logiciel Massive

Nous avons développé notre propre animation de foule au sein du logiciel Massive, pour pouvoir ensuite réaliser un film d'animation. Pour cela, nous utilisons le système incorporé dans Massive, la logique floue (Fuzzy logic) Il s'agit d'une technique de contrôle.

Nous présentons dans ce chapitre le logiciel Massive (définition, interface, la fonctionnalité, ...ect), ainsi que la la méthode de contrôle (la logique floue).

Le mot Massive est une abréviation de la phrase : (Multiple Agent Simulation System in Virtual Environment) en français, c'est un système de Simulation Multi-Agents dans un Environnement Virtuel c'est-à-dire est un logiciel utilise pour la simulation de foules virtuelles.

Développé par *Stephen Regelous*, la principale fonction de Massive est de permettre de créer rapidement et facilement des foules de plusieurs milliers (ou millions) d'individus, appelés *agents*, qui puissent agir chacun différemment en fonction de leur environnement, grâce à un système de logique floue. En fonction de ce qui se trouve à proximité de l'agent, ses mouvements et ses déplacements (préalablement capturés à l'aide de la motion capture ou animé dans un logiciel tierce) se modifieront.

En plus de ces capacités en intelligence artificielle I.A., Massive offre aussi de nombreuses autres fonctions comme la simulation de vêtements ou un rendu de base. Plusieurs comportements sont disponibles préconfigurés, dont des supporteurs dans un stade, des émeutiers ou encore de simples passants qui marchent et discutent entre eux.

C'est à la demande de **Peter Jackson** que Massive a été développé dans les studios de **Weta Digital**, à Wellington en Nouvelle-Zélande, par Stephen Regelous, en 1996. L'objectif était de réussir à réaliser des scènes de batailles réalistes sans avoir à diriger des milliers de figurants (et à les payer), pour la trilogie du Seigneur des anneaux. Depuis, le programme a évolué en un produit complet, a été utilisé dans de nombreux autres films et Stephen Regelous dirige maintenant la compagnie Massive Software, dont la principale ressource est ce programme.



Figure 4.1 : le logiciel Massive [Mas08].

2.1 Fonctionnement

Pour avoir une simulation réaliste de foule, ce programme est un système multi-agents qui donne une intelligence artificielle préprogrammée à chaque agent. Chacun reçoit une multitude d'actions possibles. Lors des premiers tests, chaque agent avait ordre de combattre un ennemi qu'il voyait et s'il n'en voyait pas, il devait courir jusqu'à ce qu'il en trouve ; au visionnage des tests, on remarquait qu'il y avait des agents qui couraient vers l'extérieur comme s'ils fuyaient le champ de bataille alors qu'ils cherchaient juste un ennemi. Suivirent des tests comme celui dit de « l'homme cerveau », nom donné au seul agent capable de marcher sur un terrain fort accidenté et qui pouvait analyser le terrain autour de lui, repérer les pentes trop abruptes et ne pas y aller.

2.2 Les versions de Massive

2.2.1 Massive Prime

Massive Prime: est un programme pour la création de la simulation comportementale. Une scène Massive se compose d'un ou plusieurs agents autonomes qui peuvent réagir entre eux et de leur environnement. Les agents se composent d'un corps et d'un cerveau. Leur environnement comprend les autres agents et un modèle facultatif de terrain. Le corps est une hiérarchie des segments reliés, chacune représentée par un primitif géométrique plein. Le cerveau est un réseau de logique floue qui peut prélever l'information du corps et activer de divers paramètres dans le corps.

Massive exécute les fichiers de cdl et les fichiers de MAS.

- Cdl (Creature Description Language) représente le langage de description de créature. Un fichier de cdl contient une description complète du corps et du cerveau d'un agent.
- Mas (Massive) Une classe de fichier MAS est un fichier Massive d'installation.

Il y a trois manières de créer le mouvement dans Massive :

- par contrôle direct des angles de jointures et des transformations en utilisant la fenêtre de modélisation du cerveau.
- par contrôle de déclenchement et de mélange des actions.
- par les dynamiques de corps rigide

Ces trois approches peuvent être combinées. Ainsi une simulation de dynamique d'un corps rigide (RBD) pourrait être attachée à un agent qui est contrôlé par des actions, et des forces pourraient être appliquées à partir des nœuds de sorties de la fenêtre du cerveau. Également la fenêtre cerveau peut être employée pour modifier des actions pour permettre le comportement tel que l'adaptation de terrain et la variation paramétrique du mouvement.

La fenêtre de vue montre une image perspective de la simulation. La position et l'orientation d'appareil photo « caméra » peuvent être commandées interactivement, l'appareil-photo peut être contraint aux agents dans diverses manières, et des appareils-photo peuvent être lus de et écrits aux dossiers.

Les aspects physiques d'un agent sont édités dans la page de corps de la fenêtre d'icônes. Ceci inclut les propriétés de dynamique, le tissu, la géométrie, l'éclairage ...etc.

2.2.2 Massive Jet

Massive jet est une version de Massive, un programme pour la création de la simulation comportementale. La différence entre Massive jet et la version de Massive prime est que le Massive jet ne permet pas de créer de nouveaux agents autonomes, ni l'édition des cerveaux

ou du mouvement d'agent. Les utilisations de Massive jet pré-ont construit les agents prêts à fonctionner ou d'autres agents construits dans la version de Massive.

2.3 Interface de Massive

L'interface de Massive peut être divisée en deux éléments principaux : la fenêtre des nœuds et la fenêtre de vue (simulation). Dans la fenêtre des nœuds il y a les menus, la zone de travail de nœud, des nœuds disponibles de page, et le dialogue de propriété.

Les menus sont divisés en menu principal et menu de choix de page.

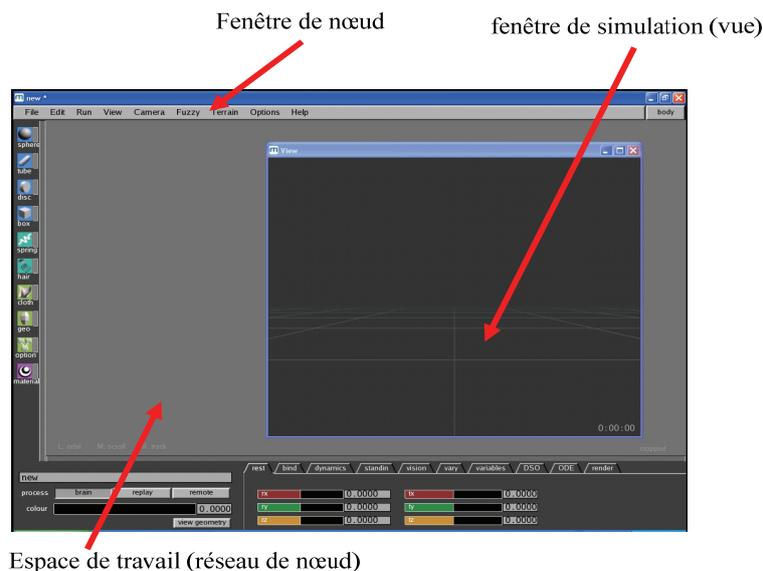


Figure 4.2 : Interface de Massive.

L'aspect de ces éléments changera quand nous passons d'une fenêtre à une autre dans l'interface, mais il y a beaucoup de similitudes de la manière que l'information est accédée et rapportée par l'intermédiaire de ces éléments, ainsi vous le trouverez utile de se familiariser avec l'approche globale à l'interface. En plus de ces fenêtres primaires, Massive fournit un certain nombre de d'autres fenêtres d'édition qui nous aideront en créant nos simulations. L'élément principal dans l'interface de fenêtre des nœuds est la zone de travail des nœuds, et il peut représenter une de quatre pages (scène, corps, cerveau ; action).

- La fenêtre de scène est employée pour éditer les lumières, les caméras, les groupes, et les options de rendu de terrain dans la scène.

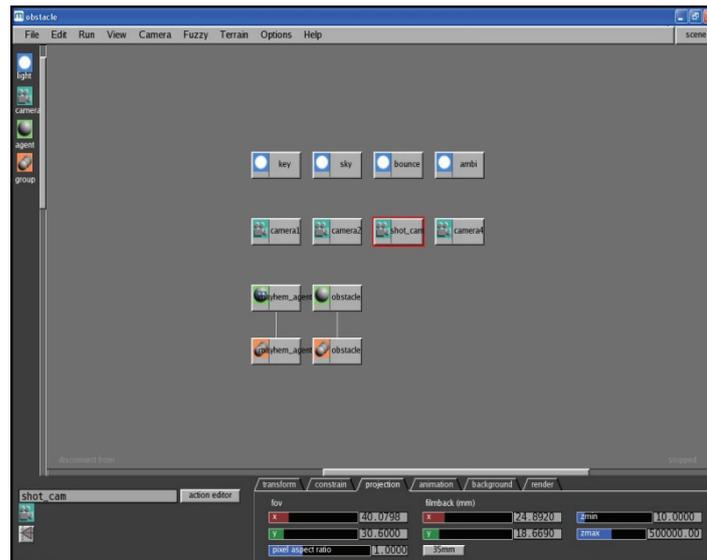


Figure 4.3 : la fenêtre scène.

- La fenêtre de corps est employée pour éditer les caractéristiques physiques d'un agent telles que ses segments, ressorts, géométrie, tissu, matériaux, options et variation.

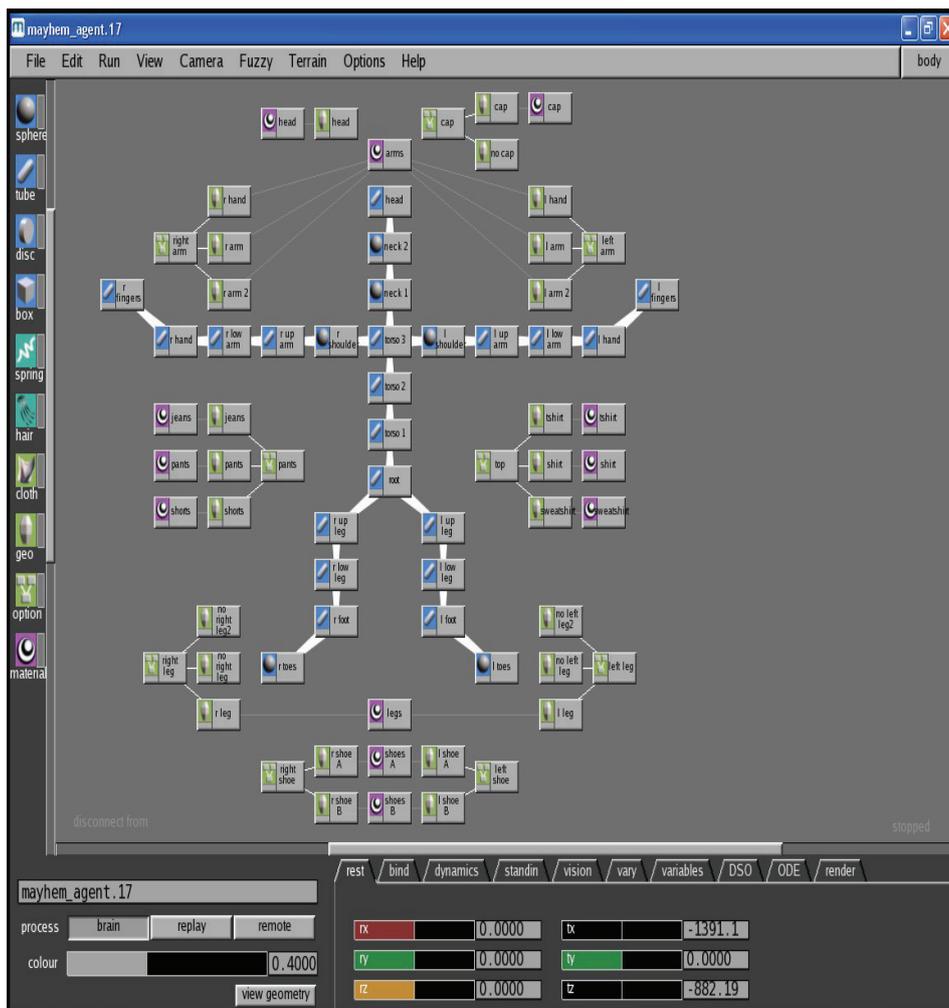


Figure 4.4 : la fenêtre de corps (squelette).

- La fenêtre de cerveau est employée pour éditer les caractéristiques de fonctionnements de cerveau d'un agent.

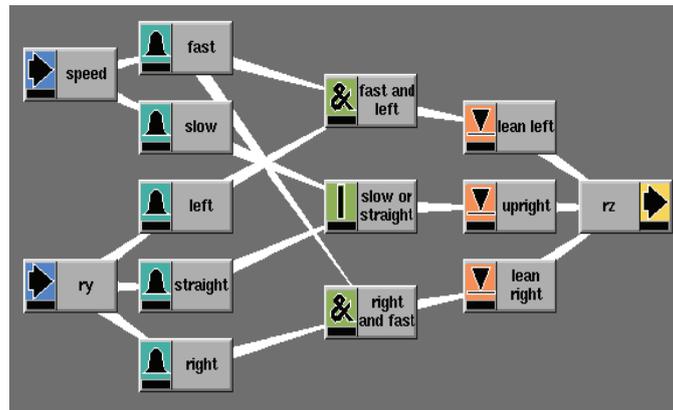


Figure 4.5 : la fenêtre de cerveau.

- La fenêtre des actions est employée pour concevoir les mouvements d'un agent sous forme d'un ou plusieurs arbres de mouvement. Un arbre est une partie de l'agent et une fois déclenché, cause l'agent à faire une transition d'une action à une autre d'une façon logique et facilement contrôlable. La page des actions produit également une session de capture de mouvement qui prend la liste des arbres de mouvement.

Pour la majorité d'agents, seulement un arbre de mouvement est nécessaire. Cependant, quelques agents peuvent courir des actions sur des voies séparées, telles qu'un agent de cheval-et-cavalier. Massive permet à l'utilisateur de créer plus d'un arbre par agent et d'assigner chaque arbre à sa propre voie.

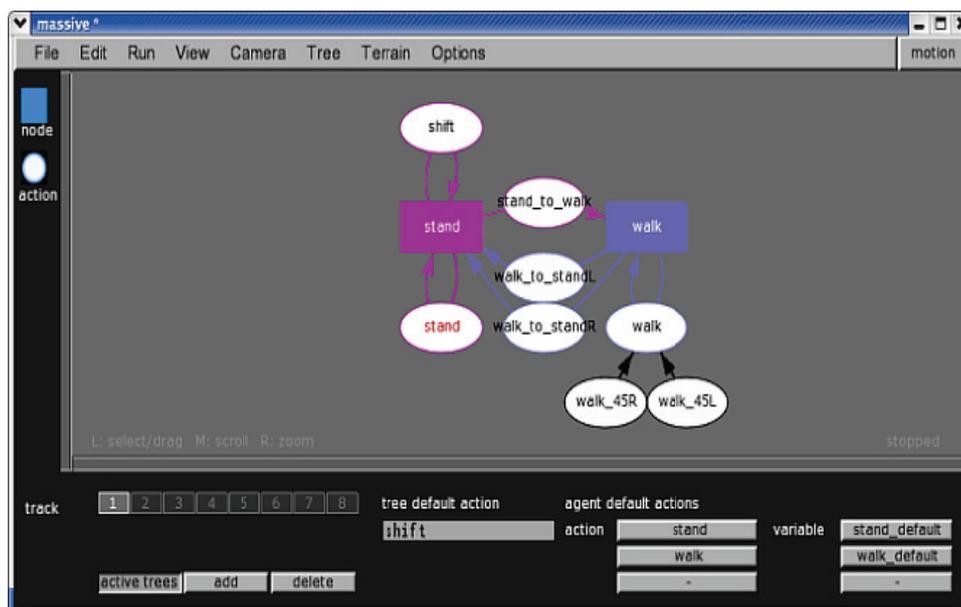


Figure 4.6 : la fenêtre d'action.

3. La logique floue

Le logiciel Massive permet d'utiliser la logique floue pour caractériser le cerveau d'un agent. La logique floue (*fuzzy logic*, en anglais) a été introduite en 1965, par Lotfi A. Zadeh, pour répondre aux problèmes auxquels sont confrontés de nombreux systèmes complexes qui doivent traiter des informations qui sont de nature imparfaite.

On distingue principalement deux types d'imperfection :

- Les *incertitudes* qui ont trait à la véracité de l'information, c'est-à-dire à sa conformité à une réalité.
- Les *imprécisions* qui concernent davantage le contenu de l'information.

La théorie des probabilités a été introduite pour gérer le premier type d'imperfection et plus précisément les *incertitudes* liées à la notion d'*occurrence* d'évènements *précis*. Si cette approche est bien adaptée à ce type particulier d'incertitude, son cadre est cependant trop limitatif pour rendre compte de l'ensemble des imperfections.

En effet, d'une part, le jugement incertain peut s'appuyer sur des informations plus subjectives, où l'on ne dispose d'aucune connaissance de nature fréquentielle et donc probabiliste ; par exemple ces informations peuvent provenir de l'appréciation subjective d'un expert humain. D'autre part, la théorie des probabilités n'est pas adaptée à la gestion du second type d'imperfection qui concerne la nature imprécise des informations.

Aujourd'hui, de nombreux systèmes sont amenés à analyser et à traiter des *informations imprécises* qui composent souvent le monde réel. On peut mentionner notamment les systèmes qui visent à automatiser les actions humaines comme, par exemple, les systèmes d'aide au diagnostic ou encore, les systèmes de reconnaissance des formes [Bez92] [Ped90].

Face à ces imperfections liées très souvent à la perception du monde réel, L. A. Zadeh a introduit les deux théories suivantes :

- La *théorie des sous-ensembles flous* [Zad65] pour modéliser l'*imprécision* en s'approchant notamment du raisonnement humain, robuste face à une appréciation imprécise de nature plutôt *qualitative*.
- La *théorie des possibilités* [Zad78], formalisée et étendue notamment par Dubois et Prade [Dub80], pour gérer les *incertitudes* d'évènements de nature non probabiliste qui peuvent être *précis* ou *imprécis*.

3.1 Domaines d'application de logique floue

La logique floue est une technique utilisée en intelligence artificielle et utilisée dans des domaines aussi variés que l'automatisme (freins ABS), la robotique (reconnaissance de formes), la gestion de la circulation routière (feux rouges), le contrôle aérien, l'environnement (météorologie, climatologie, sismologie, analyse du cycle de vie), la médecine (aide au diagnostic), l'assurance (sélection et prévention des risques) et bien d'autres.

Elle s'appuie sur la *théorie mathématique des ensembles flous*. Cette théorie, introduite par Zadeh, est une extension de la théorie des ensembles classiques pour la prise en compte d'ensembles définis de façon imprécise. C'est une théorie formelle et mathématique dans le sens où Zadeh, en partant du concept de fonction d'appartenance pour modéliser la définition d'un sous-ensemble d'un univers donné, a élaboré un modèle complet de propriétés et de définitions formelles. Il a aussi montré que cette théorie des sous-ensembles flous se réduit effectivement à la théorie des sous-ensembles classiques dans le cas où les fonctions d'appartenance considérées prennent des valeurs binaires ($\{0,1\}$).

3.2 Principe

À l'inverse de la logique booléenne, la logique floue permet à une condition d'être en un autre état que *vrai* ou *faux*. Il y a des degrés dans la vérification d'une condition.

3.3 Les sous-ensembles flous

La théorie des sous-ensembles flous est une théorie mathématique du domaine de l'algèbre abstraite. Pour représenter mathématiquement l'imprécision relative à certaines classes d'objets et sert de fondement à la logique floue.

Les sous-ensembles flous (ou parties floues) ont été introduits afin de modéliser la représentation humaine des connaissances, et ainsi améliorer les performances des systèmes de décision qui utilisent cette modélisation.

Les sous-ensembles flous sont utilisés soit pour modéliser l'incertitude ou l'imprécision, soit pour représenter des informations précises sous forme lexicale assimilable par un système expert.

3.1.1 Concept

La théorie des sous-ensembles flous généralise le concept de sous-ensemble "classique", souvent qualifié de *net* ou *booléen*, en introduisant un caractère graduel à la notion d'appartenance d'un élément à un sous-ensemble (voir la figure 4.7).

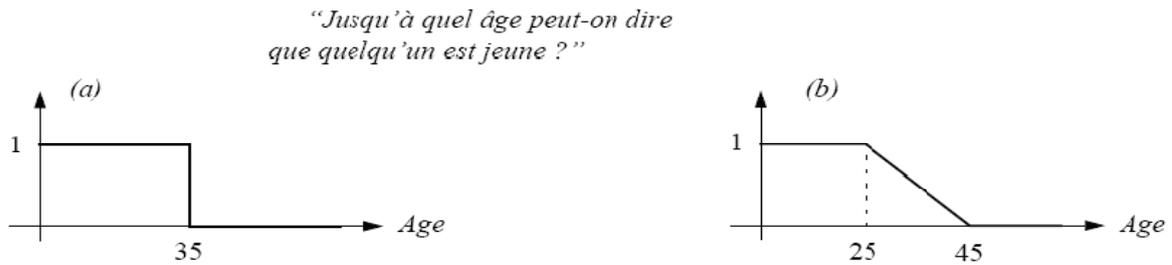


Figure 4.7 : Caractère non graduel (a) et caractère graduel (b) du degré d'appartenance.

Le caractère graduel permet de représenter des connaissances qui sont imprécises, soit de par leur aspect qualitatif (langage naturel), soit de par le biais de leur obtention (instrument de mesure). Il découle, alors, de la notion de sous-ensemble flou un ensemble de concepts permettant : une meilleur approche de la réalité, l'introduction de transitions graduelles entre différentes catégories ou encore l'apport d'une notion d'appartenance partagée d'un élément à plusieurs classes.

3.3.2 Définition d'un sous-ensemble flou

Soit E un ensemble dénombrable ou non. Un sous-ensemble flou A de E est défini par une fonction d'appartenance μ_A à valeurs dans $[0,1]$ qui associe à chaque élément x de E un degré d'appartenance $\mu_A(x)$ indiquant le niveau d'appartenance de x à A .

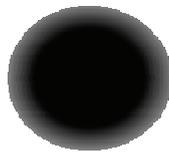


Figure 4.8 : Ensemble flou. Le niveau de gris indique le degré d'appartenance.

L'ensemble E est appelé *ensemble* ou *univers de référence*¹.

On peut noter, pour le cas particulier où la fonction d'appartenance μ_A prend ses valeurs dans $\{0,1\}$, que le sous-ensemble flou A est un sous-ensemble classique.

Il existe différents types de fonctions d'appartenance (voir figure 4.8) que l'on choisit généralement régulières et convexes pour leurs bonnes propriétés [Zad65].

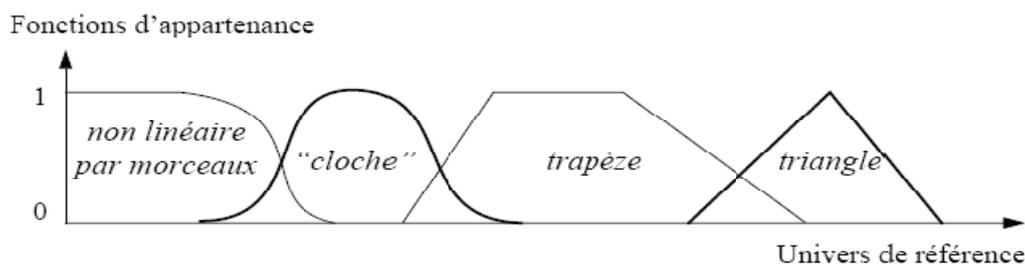


Figure 4.8 : Exemple de fonctions d'appartenance.

3.3.3 Caractéristiques d'un sous-ensemble flou

Nous allons définir, dans ce paragraphe, les caractéristiques les plus utiles à la représentation d'un sous-ensemble flou noté A et qui ont trait principalement à la forme de sa fonction d'appartenance μ_A . La figure 4.9 illustre l'ensemble de ces caractéristiques sur un exemple simple.

- **Le support**

On appelle support de A , noté $\text{Supp}(A)$, l'ensemble des éléments de E pour lesquels le degré d'appartenance à A n'est pas nul : $\text{Supp}(A) = \{x \in E / \mu_A(x) > 0\}$.

- **Le noyau**

On appelle noyau de A , noté $\text{Noy}(A)$, l'ensemble des éléments de E pour lesquels le degré d'appartenance à A est égal à 1 : $\text{Noy}(A) = \{x \in E / \mu_A(x) = 1\}$.

- **La hauteur**

On appelle hauteur de A , noté $h(A)$, la plus grande valeur prise par la fonction d'appartenance associée à A : $h(A) = \sup_{x \in E} \mu_A(x)$.

A est dit **normalisé** si $h(A) = 1$.

- **Les α -coupes**

On appelle α -coupe de A , noté A_α , l'ensemble des éléments de E pour lesquels le degré d'appartenance à A est au moins égal à α : $A_\alpha = \{x \in E / \mu_A(x) \geq \alpha\}$.

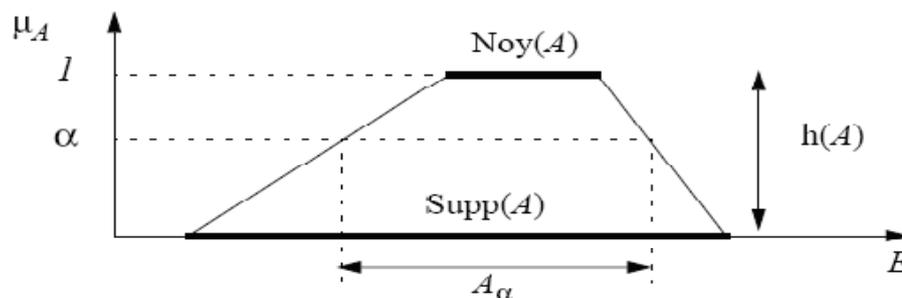


Figure 4.9 : Caractéristiques d'un sous-ensemble flou A .

3.3.4 Combinaison de plusieurs entrées

Dans le cas d'une combinaison de plusieurs entrées (« Si le ciel est bleu et si j'ai le temps »), deux cas se présentent :

- Les entrées sont liées par une fonction logique « ET » : dans ce cas, on peut considérer comme première approche seulement l'entrée ayant le degré de vérification le plus faible. En fait, il suffit de choisir un opérateur $\&$ tel que $\&(x, y) \leq \min(x, y)$ où $\&$ est appelée une **t-norme**. \min est la plus *optimiste* des t-normes.

- Les entrées sont liées par une fonction logique « OU » : dans ce cas, on peut considérer comme première approche seulement l'entrée ayant le degré de vérification le plus élevé. En fait, il suffit de choisir un opérateur t tel que t est appelée une **t-conorme**. \max est la plus *pessimiste* des t-conormes.

Il est techniquement possible de représenter toutes les opérations binaires de base en se basant sur la logique floue. En effet, à partir des opérateurs *ET*, *OU* et *NON* (*AND*, *OR*, *NOT*), on peut représenter les 8 opérations de base :

- OU (OR) : $A \text{ OR } B = \max(A, B)$;
- ET (AND) : $A \text{ AND } B = \min(A, B)$
- NON (NOT) : $\text{NOT } A = 1 - A$;
- OU EXCLUSIF (XOR) : $A \text{ XOR } B = (A \text{ OR } B) \text{ AND NOT } (A \text{ AND } B) = A + B - 2 \times \min(A, B)$;
- NON-OU (NOR) : $A \text{ NOR } B = 1 - \max(A, B)$;
- NON-ET (NAND) : $A \text{ NAND } B = 1 - \min(A, B)$;
- NON-XOR (NXR) : $A \text{ NXR } B = 1 + 2 \times \min(A, B) - (A + B)$;
- SUIVEUR (NOP) : $\text{NOP } A = A$.

Par ailleurs, la dimension décimale des variables de la logique floue permet d'effectuer des combinaisons non binaires :

- Le produit : $A \cdot B$ ou $A \times B$ (équivalent en binaire à l'opération AND)
- L'addition : $A + B$ (équivalent en binaire à l'opération OR)

3.3.5 Opérateurs flous

Les opérateurs flous (ou *fuzzy*) peuvent être implémentés de diverses manières et une même application peut d'ailleurs faire appel à des implémentations différentes judicieusement choisies selon le contexte.

3.3.6 Opérateurs linguistiques

Dans le langage humain, on utilise très souvent les opérateurs linguistiques comme : très, près de, loin, plus ou moins...etc. Ces opérateurs s'appellent les modificateurs des ensembles flous, la définition de ces opérateurs est subjective, ce qui donne comme résultat une infinité d'opérateurs proposés par différents auteurs.

Prenons la propriété floue : 'distance' dans l'ensemble de références de distance d'obstacle. Cette propriété est jugée différemment par les diverses personnes et très souvent ce jugement

dépend de l'instant. Pour cette raison, on a décidé de ne pas fixer de limites pour les fonctions d'appartenance.

L'application des opérateurs linguistiques permet de créer les ensembles modifiés comme par exemples : 'proche', 'moyenne' ou 'loin'.

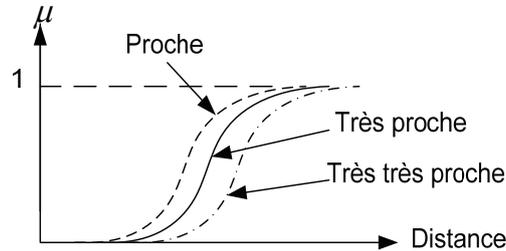


Figure 4.10 : Fonctions d'appartenance 'distance'.

L'opérateur le plus utilisé pour l'expression du modificateur 'très' est la concentration ou la fonction d'appartenance est donnée par : $\mu_{con(A)}(x) = \mu_A^2(x)$

3.4 Le raisonnement en logique floue

3.4.1 Les implications floues

Dans Massive une implication floue est une relation \mathfrak{R} entre les deux ensembles X et Y quantifiant le degré de vérité de la proposition :

$$SI (x \in A) \text{ ALORS } (y \in B),$$

Où A et B sont des sous-ensembles flous de X et Y respectivement.

3.4.2 Les règles floues

Comme nous montrons dans le titre 3.3.4 les systèmes flous utilisent des règles d'inférence pour modéliser les relations entrée/sotie et donc pour inférer connaissances à partir de quelque fait connu. Ces règles ont la forme :

$$SI \langle \text{antécédent} \rangle \text{ ALORS } \langle \text{conséquence} \rangle$$

Où $\langle \text{antécédent} \rangle$ et $\langle \text{conséquence} \rangle$ sont des phrases (suites de proposition et variables en général) liées par l'opérateur ET dans le cas de $\langle \text{conséquence} \rangle$ et par les opérations ET, OU, NON dans le cas de $\langle \text{antécédent} \rangle$. De cette façon sont réalisés les systèmes basés sur la connaissance.

Les phrases des règles floues sont de type linguistique, et ont la forme : (V est L)

Où : V est une variable linguistique

L est l'étiquette d'un ensemble flou

Comme en logique classique, on peut effectuer des raisonnements sur les sous-ensembles flous, par exemple un raisonnement du type:

$$SI (x \in A) ET (y \in B) ALORS (z \in C),$$

Où A, B, C sont des sous-ensembles flous.

En logique classique, ce peut être :

$$SI (x > 5) ET (y > 7) ALORS (xy > 35)$$

En logique floue, on aura une règle du genre :

$$SI (\text{vitesse est grande}) ET (\text{obstacle est proche}) ALORS (\text{freinage est fort})$$

Tout le problème réside dans la détermination du sous-ensemble flou des z satisfaisant l'implication, connaissant x , y et les fonctions d'appartenance de A, B, et C. Le sous-ensemble des solutions est flou car un z donné réalise l'implication avec un degré plus ou moins grand.

3.4.3 Conjonction de propositions floues

Le contrôle d'un processus par commande floue nécessite de considérer les différentes valeurs possibles pour les variables d'entrée, et donc différentes règles pour chaque situation envisagée. Ainsi, on considère un ensemble de propositions floues, du genre :

$$\left\{ \begin{array}{ll} SI (x \in A_1) ET (y \in B_1) ALORS (z \in C_1) & P_1 \\ SI (x \in A_2) ET (y \in B_2) ALORS (z \in C_2) & P_2 \\ & \dots \\ SI (x \in A_n) ET (y \in B_n) ALORS (z \in C_n) & P_n \end{array} \right.$$

Pour déterminer le sous-ensemble flou solution de la conjonction des n propositions, il faut utiliser un opérateur d'agrégation permettant de faire la synthèse des solutions de chaque P_i .

Comme on le verra dans l'exemple de la figure 4.15.

3.4.4 Deux cas particuliers : les méthodes de Mamdani et de Larsen

Examinons deux possibilités de choix pour \mathfrak{R} , ET, et agrégation.

Méthode de :	Mamdani	Larsen
Implication floue \mathfrak{R}	\mathfrak{R}_M	\mathfrak{R}_L
Opération de conjonction ET	Minimum	Produit
Opération d'agrégation	Maximum	Maximum

Tableau 4.1 : Les méthodes de Mamdani et de Larsen.

Remarque : la méthode de Mamdani est aussi appelée méthode min-max.

Avec ces définitions la fonction d'appartenance du sous-ensemble flou des solutions de la proposition P_i :

$$SI (x \in A_i) \text{ ET } (y \in B_i) \text{ ALORS } (z \in C_i)$$

Est :

$$\text{Mamdani : } f_{S_i}(z) = \min(\min(f_{A_i}(x), f_{B_i}(y)), f_{C_i}(z))$$

$$\text{Larsen : } f_{S_i}(z) = (f_{A_i}(x)f_{B_i}(y))f_{C_i}(z)$$

Et dans les deux cas les solutions sont agrégées par max, le sous-ensemble flou S solution du système des n propositions a pour fonction d'appartenance :

$$f_S(z) = \max(f_{S_1}(z), f_{S_2}(z), \dots, f_{S_n}(z)),$$

Où si $1 \leq i \leq n$, S_i est le sous-ensemble flou solution de la proposition P_i .

Nous nous intéressons par la méthode de Mamdani car le logiciel Massive il utilise la.

3.5 La défuzzification

Le processus de commande floue doit fournir une solution numérique pour être exploitable. Par exemple pour un processus de contrôle, l'organe de commande nécessite un signal d'entrée précis. Il reste donc une étape : l'interprétation du sous ensemble flou des solutions, la *défuzzification*.

Plusieurs possibilités existent, mais il semble qu'on ne puisse pas déterminer de manière systématique la meilleure méthode de défuzzification. Par ailleurs, la rapidité de l'algorithme de défuzzification est cruciale pour un contrôle en temps réel. Il existe deux méthodes.

3.5.1 La méthode du maximum

Elle consiste à choisir comme solution de défuzzifier l'abscisse du maximum de la fonction d'appartenance des solutions.

Si plusieurs points conviennent, on peut par exemple utiliser une variante, la méthode de la moyenne des maxima, qui consiste à prendre comme solution la moyenne des abscisses des maxima. Dans notre cas :

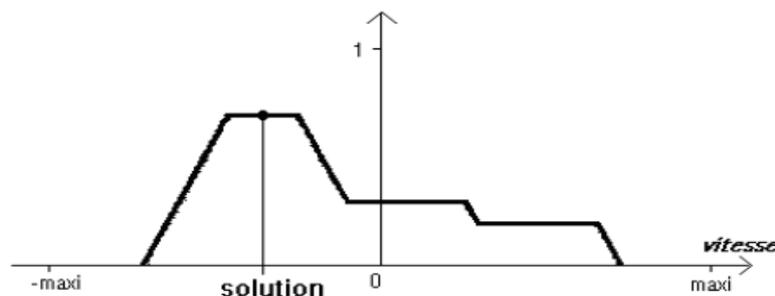


Figure 4.11 : Méthode de maximum.

L'avantage de cette méthode est qu'elle ne nécessite pas une grande puissance de calcul. Cependant, elle présente un gros inconvénient : de faibles variations du sous-ensemble flou solution peuvent entraîner des sauts importants du signal de sortie, comme l'illustre la figure 4.11. Ceci est particulièrement gênant si le signal défuzzifié alimente un organe de contrôle nécessitant un signal régulier.

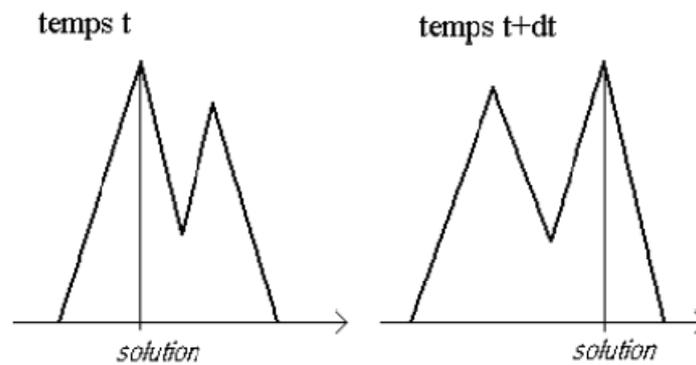


Figure 4.12 : Inconvénient de la méthode du maximum.

3.5.2 La méthode du centre de gravité

Cette méthode est la plus souvent utilisée et donne généralement les meilleurs résultats. Elle consiste à prendre comme solution l'abscisse du centre de gravité des solutions.

Pour le pendule inversé :

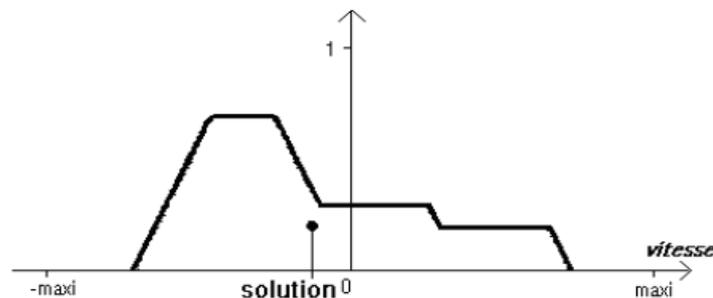


Figure 4.13 : Méthode du centre de gravité.

Par comparaison avec la méthode du maximum, les résultats fournis sont bien plus stables vis-à-vis de variations minimales du sous-ensemble flou solution. Mais en contrepartie, elle exige une plus grande puissance de calcul.

3.6 Algorithme générale de fuzzification

Les applications pratiques de la logique floue sont nombreuses et diverses. Néanmoins il est impossible de passer sous silence le "*contrôle flou*". Les réalisations industrielles n'ont pas tardé à apparaître (automatisme, robotique, intelligence artificielle...). Un contrôleur flou se comporte comme un approximateur capable de calculer un résultat non-flou depuis

plusieurs variables non-floues elles aussi, mais par le biais d'un processus de raisonnement flou.

Algorithme Inférence_flou ;

Début

Fuzzification les données d'entrées ;

Définir la base de connaissance ;

La décision ;

Défuzzification ;

Fin.

- Le premier traitement des données soumises consiste en la fuzzification, soit la transformation de données non-floues (crisp) en données floues.
- Le noyau du contrôleur se décompose en deux parties, l'une statique, l'autre dynamique. La partie statique consiste en une base de connaissances floues. Elle traduit le jugement de l'expert. Ce procédé est particulièrement adapté pour traduire des appréciations « à l'estime ». Les processus mentaux de décision ne sont pas des équations.
- La partie dynamique du noyau est en fait le moteur d'inférences floues. C'est lui qui appliquera les règles et agrègera les résultats.
- L'obtention du résultat crisp s'obtiendra par l'opération de défuzzification. Le schéma qui suit tente d'en résumer le principe :

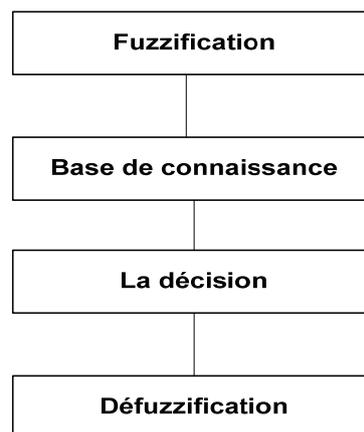


Figure 4.14 : Les étapes de l'algorithme d'inférence flou.

La figure suivante illustre les étapes de l'algorithme :

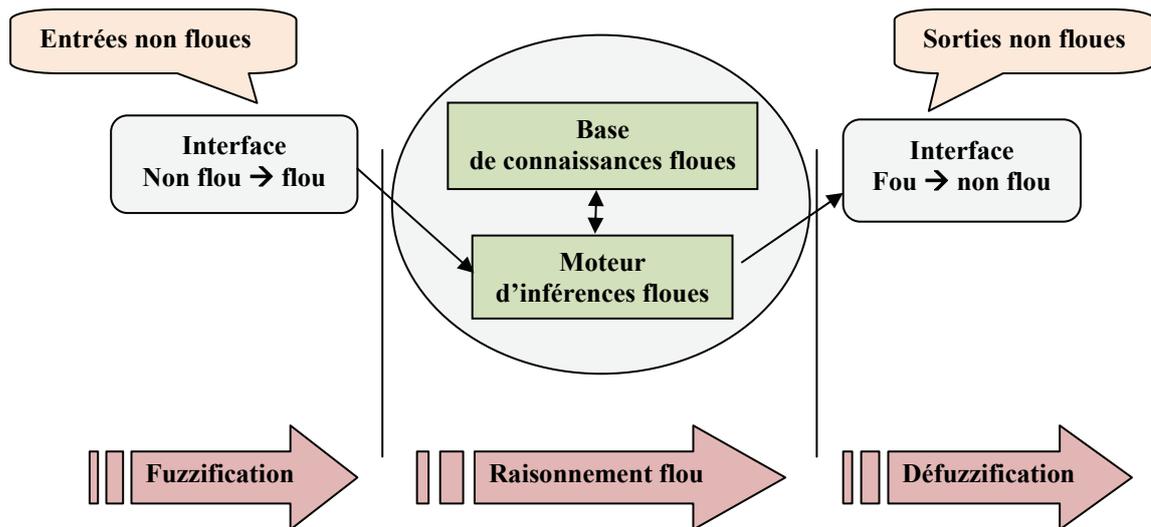


Figure 4.15 : Schéma de principe d'un contrôleur flou

L'algorithme d'inférence flou est l'opération complète pour obtenir les décisions, c'est-à-dire définir les comportements des humains virtuels, à l'aide de la logique floue.

Pour comprendre bien l'algorithme nous détaillons chaque étape dans le titre d'évitement des obstacles

3.7 La logique floue dans Massive

Les nœuds de la logique floue dans Massive représentent un réseau de logique floue, qui comprend un cerveau d'agent. Pour créer un réseau de logique floue l'utilisateur édite un graphe connecté des icônes qui représentent différents nœuds dans le réseau (voir la fenêtre de nœuds dans l'exemple de figure 4.15). Les données d'entrée sont accédées via les nœuds d'entrer. Les valeurs d'entrée floues sont définies dans les nœuds flous (fuzz nodes).

Les règles sont représentées par les connexions faites avec les nœuds *and* et les nœuds *or*. Les valeurs des sorties floues sont mises dans des nœuds de défuzzification (defuzz nodes). Les sorties sont résolues et assignées en nœuds de sortie. Le nœud temporisateur et le nœud bruit sont des nœuds d'entrée spéciaux.

Chaque icône a une barre d'activité qui indique la valeur de sortie pour le nœud. Les valeurs positives sont affichées en rouge et les valeurs négatives en bleu.

Le schéma suivant est un réseau flou pour contrôler la tendance d'orientation pour un agent.

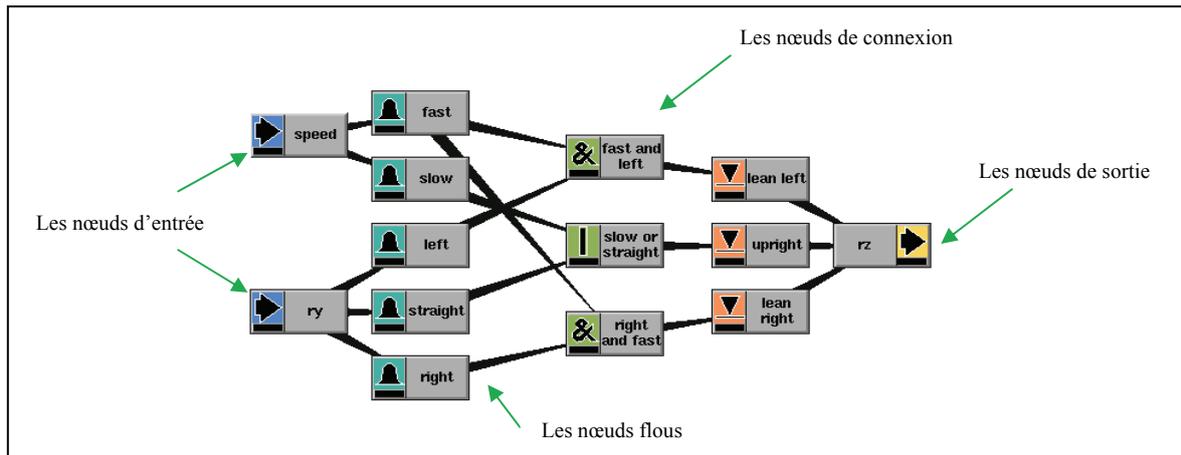


Figure 4.16 : Réseau flou contrôlé la tendance d'orientation pour un agent.

A gauche de schéma on remarque les nœuds d'entrée qui échantillonnent des informations à partir des sentiments d'agent et des états internes. Ces nœuds sont connectés aux nœuds flous (fuzz nodes) où les valeurs floues rapide (*fast*) et lente (*slow*) et les valeurs floues d'orientation gauche, tout droite et droite (*lean right*, *up right*, *lean left*) sont définies. Ceux-ci sont alors combinés en règles qui infèrent le suivant :

Entrées : vitesse, tendance ;

Sorties : orientation ;

Les valeurs floue basée sur la vitesse : *fast*, *slow* ;

Les valeurs floue basée sur la tendance : *left*, *straight*, *right* ;

Les valeurs floue de sorties pour l'orientation : *lean left*, *up right*, *lean right* ;

Règles:

If **fast** and **left** then **lean left**

If **slow** or **straight** then **upright**

If **right** and **fast** then **lean right**

Les valeurs floues de sortie sont définies dans les trois nœuds « defuzz » à droite « *lean left*, *up right*, *lean right* » et elles sont défuzzifiées et appliquées à l'agent dans le nœud de sortie appelé 'rz'. Une bonne méthode pour construire une logique floue est d'écrire des règles qui représentent des instructions sensibles comme celles écrites ci-dessus. Alors il est facile de suivre qu'est ce qu'un cerveau est en train de faire.

4. Conclusion

Nous avons présenté dans ce chapitre la plateforme utilisée pour notre simulation avec la technique de contrôle employée, le logiciel Massive parmi les outils de modélisation et d'animation d'une foule dense d'humains virtuels. Nous avons remarqué dans le chapitre précédent que chaque outil s'appuie sur son technique de modélisation, pour Massive nous remarquons que la logique floue est la technique utilisée pour la modélisation des comportements, cela est dû que la logique floue ressource des problèmes liés à certains domaines comme l'intelligence artificielle, l'animation comportementale et la robotique, et donne des résultats satisfaisants. Ainsi, la logique floue est un sur-ensemble de la logique conventionnelle (booléenne) étendue pour introduire la notion de degré de vérité, prenant des valeurs entre « complètement vrai » à « complètement faux ».

CHAPITRE 5

Modèle d'animation

« L'homme est pour partie corps, esprit et imagination, son corps est imparfait, son esprit est faible, mais son imagination le rend singulier entre tous »

Karl Mansfield

Modèle d'animation

1. Introduction

Dans les années récentes, Les travaux de recherche engagés par les infographistes cherchent comment modéliser les comportements des humains réels pour l'appliquer sur des humains virtuels synthétisés. Ces recherches engendrent un axe appelé les modèles comportementaux offrant la possibilité de simuler des entités autonomes comme des organes vivants ou des êtres humains. Des études psychologiques ont montré que le comportement humain peut être décomposé en une boucle perception-décision-action comme il est montré dans les chapitres précédents. Dans ce chapitre nous allons présenter l'architecture de notre modèle que nous avons défini et nous démontrons son adéquation à la modélisation de différents composants. Notre modèle est appliqué pour construire une foule virtuelle décrite selon le principe de la boucle perception-décision-action. Les humains virtuels autonomes ainsi construits sont utilisés dans des simulations de foule en environnement virtuel inconnu, ce qui entre dans le cadre d'une problématique d'étude des systèmes d'animations collectifs.

2. Observations sur le comportement piétonnier

Nombreuses sont les études qui ont été conduites sur le problème du comportement piétonnier. Dans le domaine de l'étude des flots, des statistiques sont effectuées en vue d'analyser le comportement piétonnier pour assurer, à plus long terme une certaine qualité de navigation. Au cours de ces études, un certain nombre de caractéristiques de ce comportement ont pu être mises en évidence [Hel01]:

- Les piétons montrent une certaine aversion à prendre des détours ou à marcher dans une direction opposée à leur direction désirée (celle les menant vers leur but), même dans des environnements peuplés. Cependant, ils ont tendance à choisir la route la plus rapide qui n'est pas forcément la plus courte. En général, les notions de détour et de confort de navigation sont prises en compte, en vue de minimiser l'effort à fournir.
- Chaque piéton possède sa vitesse individuelle correspondant à la vitesse la plus confortable, autrement dite, celle qui permet de minimiser la dépense d'énergie. Du point de vue de la personne, cette vitesse est corrélée à plusieurs facteurs comme le sexe et l'âge par exemple. Le respect de cette vitesse dépend aussi de facteurs externes tels que la raison du déplacement (la personne est-elle pressée ou non?) ou encore de la densité de la foule.

- Les piétons tentent de conserver une certaine distance avec les autres piétons et les obstacles de l'environnement. Cette distance varie en fonction du contexte, elle se réduit si le piéton est pressé ou si la foule devient dense. Dans le cas de piétons stationnaires (attente d'un train, attente d'un bus...), la tendance est de se répartir uniformément dans l'espace. Cependant, autour des endroits attractifs (statue, monument...), la densité des piétons augmente. Les individus se connaissant peuvent former des groupes; ces groupes ont alors tendance à exhiber un comportement similaire à celui d'un piéton seul.
- Lorsque la densité des piétons devient grande, la tendance générale ressemble à l'évolution d'un fluide. Au milieu de foules statiques, on peut remarquer une tendance à créer des flots; en environnement dense, on peut remarquer la formation de zones de navigation regroupant des piétons allant dans la même direction.
- Dans les situations de paniques, les individus ont tendance à développer des comportements « aveugles » et essaient de se déplacer beaucoup plus vite que de normale. Les interactions entre les piétons deviennent alors physiques, allant jusqu'à la bousculade. Les personnes blessées ou tombées constituent alors des « obstacles » gênant la progression. Finalement, un comportement mimétique est en œuvre; les piétons ont tendance à faire comme les autres, à aller où vont les autres, au détriment de l'exploitation d'opportunités de sorties différentes.

Ces résultats exposent les tendances générales associées à la navigation. D'autres études se sont plus particulièrement intéressées à la manière d'éviter une collision et à la direction adoptée. Selon Goffman, en accord avec le premier point suscit , les pi tons sont r ticents aux changements de direction, et lorsqu'ils y sont oblig s, ils privil gient les plus petits changements minimaux. Un pi ton aura donc tendance   effectuer un  vitement de collision en minimisant l'angle de d viation. Cependant, comme le fait remarquer Lee, l'organisation globale des flux pi tonniers est similaire   celle observ e pour les v hicules respectant le code de la route. Autrement dit, en Angleterre nous trouverons une tendance   l' vitement   gauche, alors qu'en Alg rie, la tendance sera   l' vitement   droite. Dans le cas de situations relativement denses, l'application de cette r gle permet d'obtenir un consensus global. Les flots s'organisent, ayant pour cons quence une circulation plus fluide. Des d tails suppl mentaires sur le comportement de navigation en environnements urbains (travers e de rue, attention visuelle...) peuvent  tre consult s dans la th se de G. Thomas [Tho99].

3. Architecture de notre modèle

L'architecture de notre modèle est illustrée dans la figure ci-dessous, représente les éléments essentiels qui montrent la scène de notre modèle : les agents et l'environnement virtuel. L'agent (humain virtuel) dans le monde virtuel est capable d'extraire des informations concernant les positions des obstacles et les autres humains virtuels dans son environnement par sa vision (perception), ces informations permettent à l'humain virtuel de prendre une bonne décision, c'est-à-dire réagir de façon autonome.

D'autre par on doit déterminer l'environnement par certains objets comme : les obstacles, les piétons et les groupes d'agents.

L'agent est caractérisé par sa position, son orientation et un ensemble des comportements individuels. Ces comportements déterminent l'aspect psychologique de l'agent.

Généralement pour donner un aspect plus réaliste de l'animation de foule, la notion de groupe doit être introduite. Pour cela nous introduisons la notion de groupe dans notre simulation comme illustré dans la figure 5.1. Dans ce cas, nous remarquons le comportement de groupe apparu comme une seule entité, donc il se comporte d'une manière différente que s'il est seul, en gardons l'objectif de notre étude qui est les comportements collectifs non délibératifs.

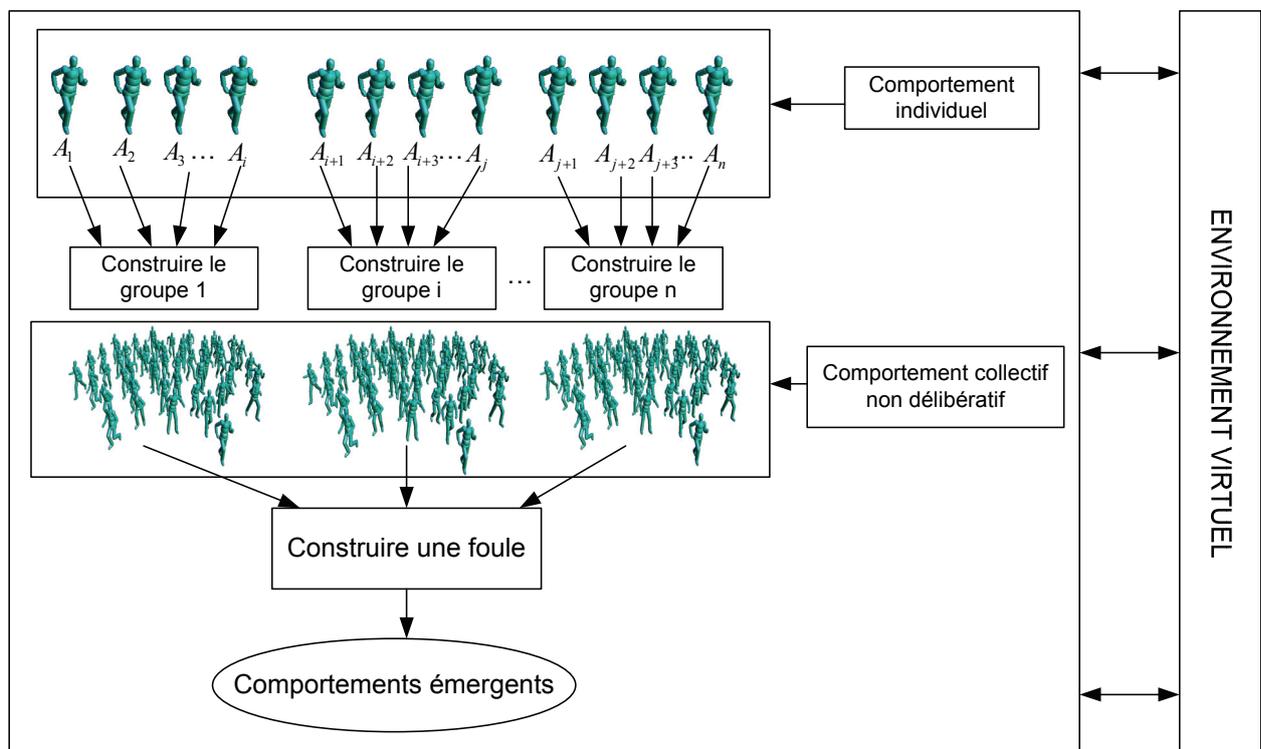


Figure 5.1 : L'architecture de notre modèle.

4. Structure du modèle multi-agents adopté

L'analyse présentée ici nous amène à proposer pour des raisons de souplesse et de modularité, une approche basée sur les systèmes multi-agents privilégiant à la fois la distribution de décision, l'autonomie et la réactivité des agents, ainsi que la coordination des actions en cas de conflits. Pour ce faire, nous avons adapté des agents autonomes de type multi-groupes pour la résolution du problème.

Chaque humain virtuel (agent) a une mission spécifique au sein du système multi-agents et doit posséder un niveau substantiel d'autonomie à travers des capacités d'ordonnancement : autonomie d'opération, perception de son environnement, évitement de collision, etc.

L'approche multi-agents adoptée propose d'une part, de distribuer le contrôle et les connaissances nécessaires à la résolution d'un problème (situation complexe) entre un ensemble d'entités appelées humains virtuels.

Comme Massive est un logiciel basé sur les systèmes multi-agents nous allons présenter la notion de système multi-agents.

4.1 L'environnement

Pour la modélisation de l'environnement, Massive offre deux méthodes pour construire un environnement, sont résumé dans les deux point suivants :

- **Environnement modélisé directement par Massive** : dans cette méthode Massive facilite la modélisation des terrains à l'aide des objets géométriques préconfigurés comme (sphère, cube, cylindre, tube,) ces objets sont placés et repartis dans une surface plate. Les environnements créés par Massive sont des terrains simples avec manque de réalisme.

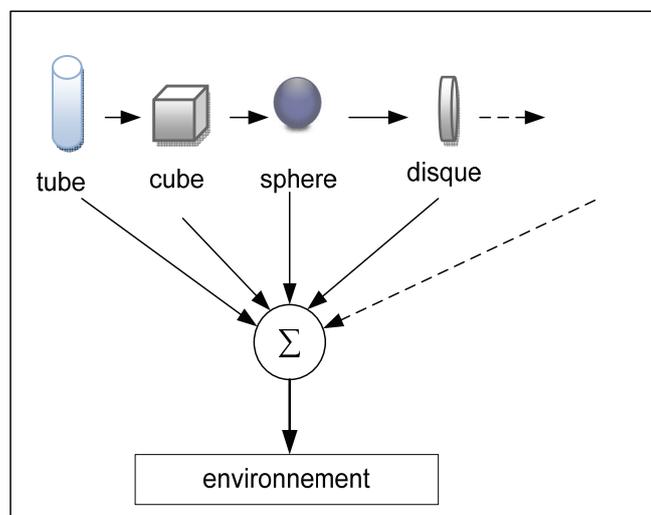


Figure 5.2 : Modélisation d'environnement par Massive.

- **Environnement étranger** : Massive donne la possibilité pour intégrer des environnements étrangers, créés et modélisés par des logiciels tiers comme « Maya ».

Les deux méthodes de modélisation d'environnement dans Massive sont montrées dans la figure ci-dessous.

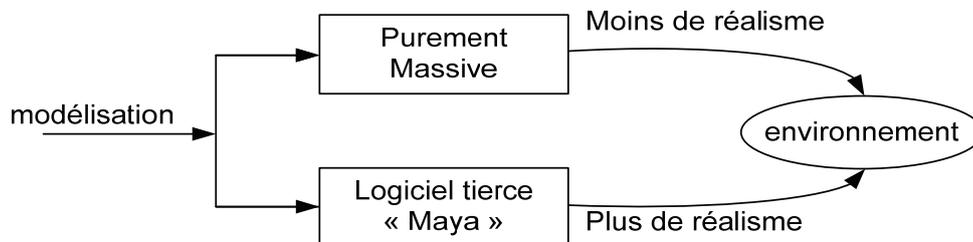


Figure 5.3 : modélisation de l'environnement dans Massive.

Nous nous choisissons la première méthode de modélisation par ce que c'est la plus simple. Même si l'environnement considéré contient des obstacles comme des boîtes simples, et bien sur les humains virtuels.

Nous avons choisi un environnement plat comme montré dans la figure 5.4.

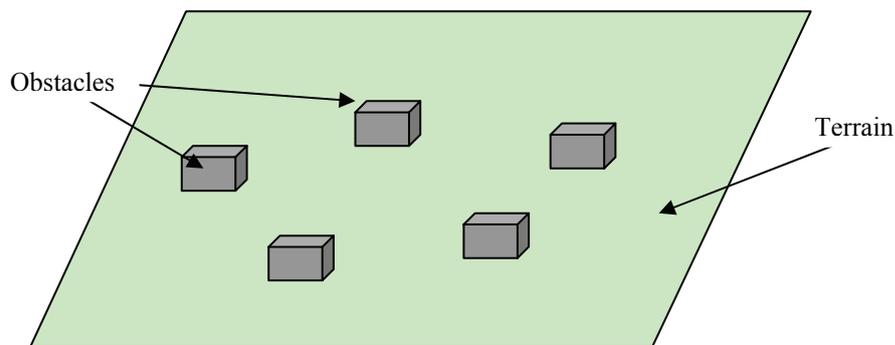


Figure 5.4 : L'environnement de notre simulation.

4.2 Hypothèses du problème

La simulation d'une foule d'humains virtuels, vient pour résoudre les problèmes de panic (l'évacuation des bâtiments et les catastrophes en générale, ...) ou pour étudier l'animation des phénomènes collectifs. Donc nous avons déterminé quelques hypothèses du problème pour notre simulation de foule virtuelle.

L'humain virtuel est confronté aux différents événements qui nécessitent des traitements spéciaux. Pour cela, on conçoit un ensemble des humains virtuels naviguant dans un environnement inconnu vérifiant les hypothèses suivantes :

- Les obstacles dynamiques sont les autres humains virtuels en circulation, ici nous avons nommé ce problème par l'évitement de collision.
- Chaque humain virtuel doit être capable tout seul de planifier lui-même et de suivre ses propres chemins, lui permettant de se rapprocher davantage de faire l'action souhaitée.
- L'environnement dans lequel évoluent les humains virtuels est supposé rectangulaire, plat et non libre (*avec obstacles*).

4.3 Description de l'architecture

L'architecture adoptée basée sur les systèmes multi-agents pour la navigation des humains virtuels est composée des modules suivants (voir la figure 5.5) :

- **Bases de connaissances** : elle permet de stocker toutes les informations utiles à la navigation (description de l'environnement, adresses et positions des autres humains virtuels, etc.)
- **Module de perception** : la fonction de perception permet à l'humain virtuel d'une part, de percevoir l'état de son environnement (les autres humains virtuels, les obstacles) et d'autre part, de percevoir ses états internes (les actions internes).
- **Module d'action** : les capacités de raisonnement de l'humain virtuel constituent le centre de décision des différentes compétences de l'humain virtuel.

En fonction du comportement déclenché, un module d'action associé à un comportement donné détermine une réponse adaptée à la situation dans laquelle se trouve l'humain virtuel.

La réponse de l'humain virtuel est fonction du savoir-faire et de la granularité du comportement. Dans notre application, ce module se compose des comportements suivants :

- Sélection chemin : il permet de trouver un chemin, plus ou moins optimal pour une animation réaliste.
- Eviter obstacle : il permet d'éviter les obstacles qui se trouvent dans le chemin de l'humain virtuel.
- Eviter collision : il permet d'éviter les collisions avec les autres humains virtuels.
- **Module de réaction** : c'est un module superviseur qui permet de contrôler l'action globale de l'humain virtuel.

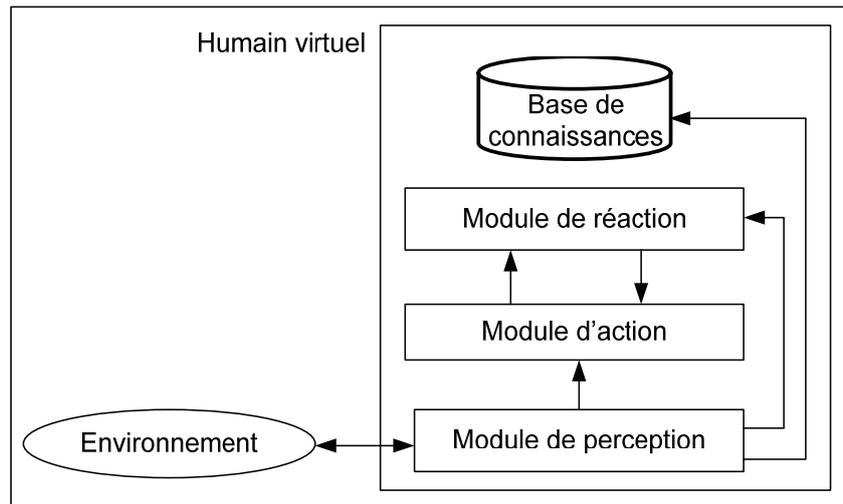


Figure 5.5 : Architecture de notre modèle.

4.3.1 Base de connaissances

Pour assurer une bonne connaissance sur l'environnement et l'évitement de certaines interactions. L'humain virtuel doit avoir certaines informations. Pour cela, l'architecture de contrôle proposée est dotée d'une structure de stockage appelée base de connaissance, et qui contient les informations suivantes : les adresses de tous les humains virtuels entrants dans le champ de perception (champ de capteur), la position de l'humain virtuel lui-même, et autres humains virtuels et les fonctions d'appartenances d'E/S du système flous.

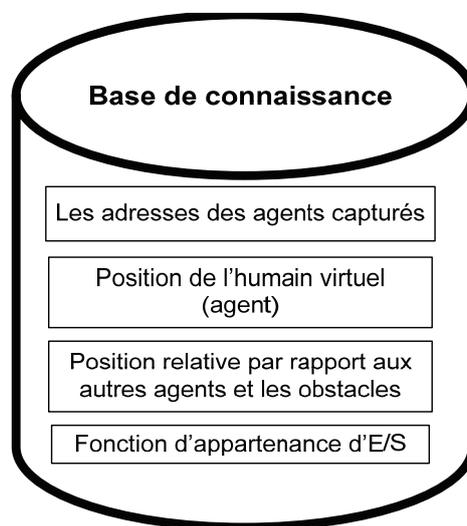


Figure 5.6 : La base de connaissance de l'agent.

La base de connaissances c'est la partie statique/dynamique dans le noyau du contrôleur, elle traduit le jugement de l'expert. Ce procédé est particulièrement adapté pour traduire des appréciations "à l'estime".

4.3.2 Module de perception

Le module de perception assure une fonction double :

- Détecter la présence des obstacles (obstacle, les autres humains virtuels) à l'aide de capteurs.
- Tester la nouvelle position de l'humain virtuel, qui se trouve dans la base de connaissances.

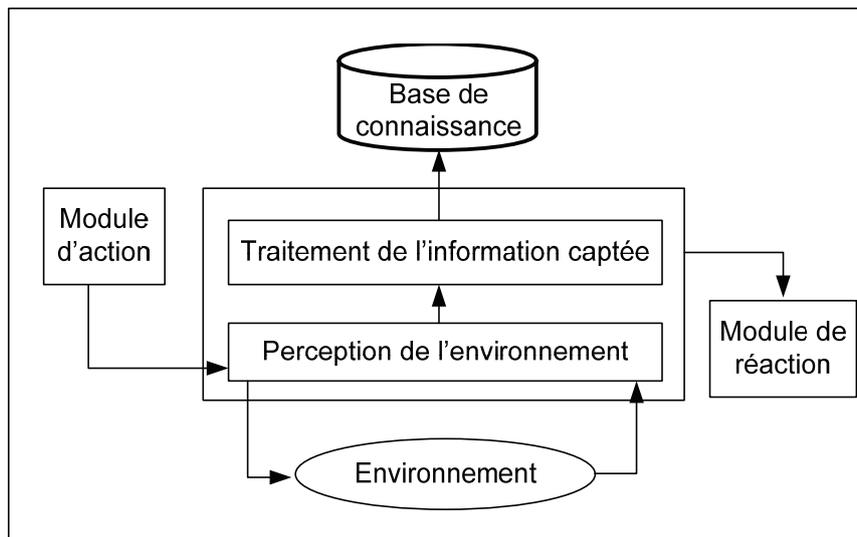


Figure 5.7 : Structure de module perception.

Le module de perception de l'humain virtuel extrait les informations de l'environnement, puis tester avec les informations stockées dans la base de connaissance. Ces informations captées sont utilisées ensuite, par les autres modules.

4.3.3 Module de réaction

Ce module représente le noyau de cette architecture de contrôle. Il permet à l'humain virtuel de réagir aux différents événements internes et externes en déclenchant un comportement correspondant à l'événement capté selon le mode stimulus/réponse comme illustré dans la figure 5.8.

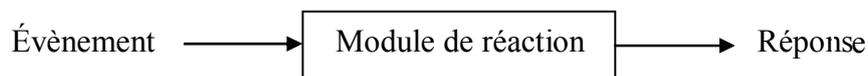


Figure 5.8 : Le module stimulus/réponse du module de réaction.

Le module de réaction effectue les tâches suivantes :

- Répondre aux messages qui étaient envoyés par l'humain virtuel lui-même : le module de réaction doit répondre aux comportements internes, généralement ces réactions individuelles capturées à l'aide de capteur de mouvement et stockés, interprètent le comportement psychologique/sociologique de l'individu.

- Traiter les événements signalés par le module de perception: Lorsque l'humain virtuel rencontre un nouvel obstacle, il doit avertir le module de réaction pour réagir aux changements de l'environnement. Cette réaction s'effectue en désactivant un comportement et activant un autre.

L'utilisation d'un module réactif permet de donner à l'humain virtuel une meilleure réaction avant les changements de l'environnement.

4.3.4 Module d'action

C'est une collection de comportements prédéfinis (voir la figure 5.9), tel que déplacement, arrêt, évitement d'obstacle. Ces comportements sont nécessaires pour le déplacement d'un humain virtuel.

Un comportement est défini comme évolution particulière observée d'un humain virtuel dans son monde réel (environnement, autres humains virtuels). Il correspond à un mécanisme reliant perception– raisonnement/décision-action de l'humain virtuel et son monde réel.

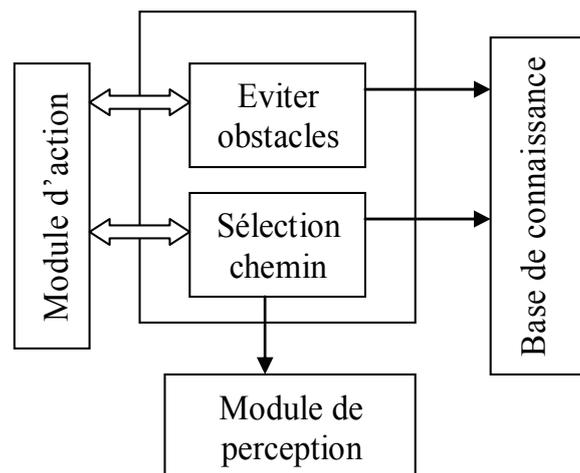


Figure 5.9 : Module d'action et ses Interactions.

Le logiciel Massive donne la possibilité pour capturer les mouvements à l'aide de capteur de mouvement, puis stocker ces mouvements (actions) dans une base.

4.3.4.1 Comportement pour sélection du chemin

Le comportement pour sélectionner un chemin il joue le rôle d'un contrôleur qui surveille le déplacement de l'humain virtuel. Ce comportement ne s'exécute que lorsque le module de réaction veut activer.

Le fonctionnement de ce comportement nécessite une configuration dans la fenêtre d'action qui est déterminé par le concepteur et l'humain virtuel lui-même.

4.4 Structure du contrôleur flou

Le contrôleur flou développé est une version classique ou traditionnelle, utilisé pour les processus de type multi-entrées, multi-sorties.

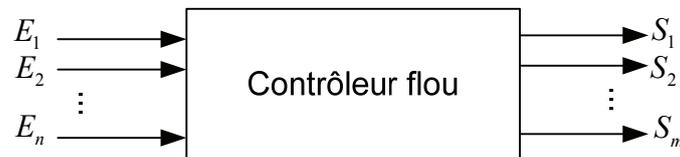


Figure 5.10 : Contrôleur flou multi-E/S.

Une fois les entrées sont connues (entrées capturées), les sorties sont déterminées de manière unique, dans laquelle chaque règle est représentée par une conjonction (minimum) et l'agrégation des règles par une disjonction (maximum). La figure 5.10 donne le principe de conception de notre contrôleur.

4.4.1 Variables linguistiques

La configuration du contrôleur est représentée par ensemble des variables et des fonctions comme illustré dans la figure suivante :

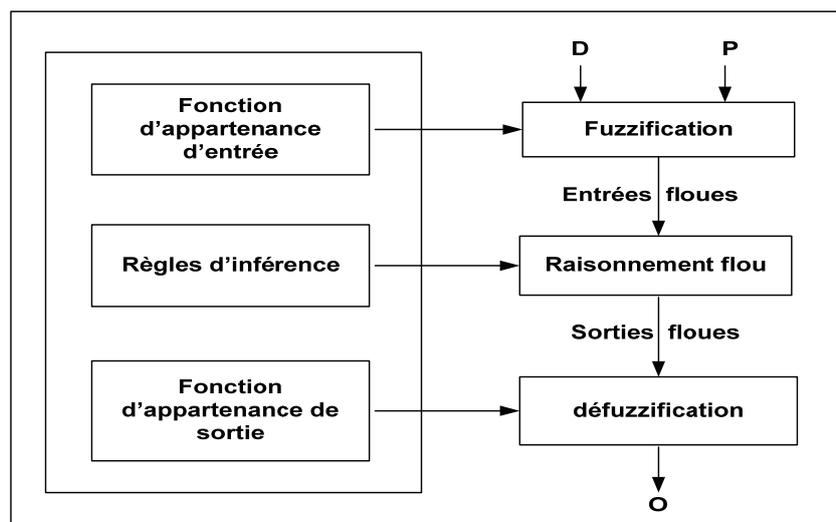


Figure 5.11 : Configuration du contrôleur flou.

Pour le contrôleur conçu, nous avons choisi comme variables d'entrées, la distance captée et la position de l'obstacle qui soit statique ou dynamique. L'univers de discours est réparti en sous ensemble flous. La figure 5.12 pour la position de l'obstacle (P) et la figure 5.13 pour la distance (D), et comme variables de sortie, l'orientation de l'humain virtuel (O), voir figure 5.14, l'humain virtuel comporte trois orientations : gauche (le sens négatif), droite (le sens positif) et en avant (nul).

Les fonctions d'appartenance de ses variables sont de forme triangulaire, où la somme des degrés d'appartenance de chaque variable aux différents sous ensembles flous est toujours égale à l'unité.

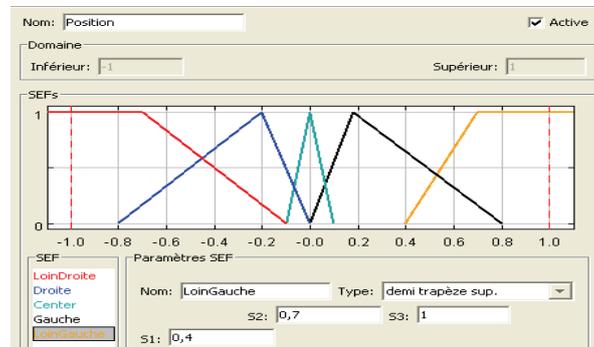


Figure 5.12 : Répartition floue de la variable Position (P).

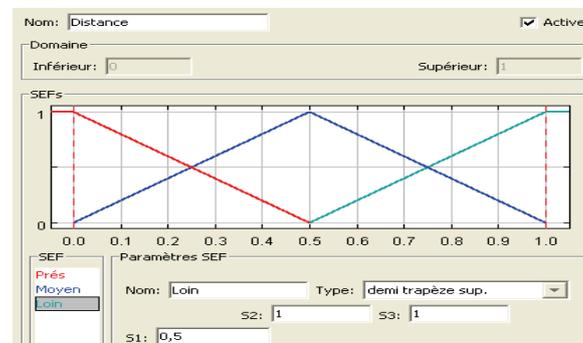


Figure 5.13 : Répartition floue de la variable Distance (D).

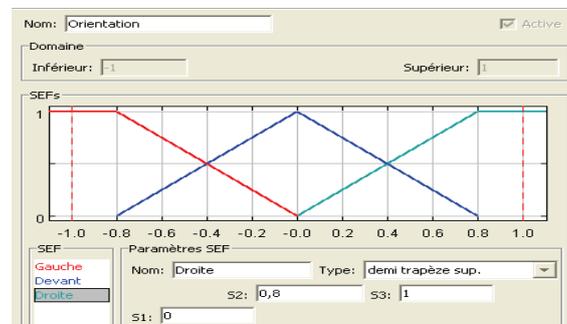


Figure 5.14 : Répartition de l'univers de la variable Orientation (O).

5 Etude de l'influence de l'environnement individuel

5.1 Comportement d'éviter la collision

Dans une scène, l'évitement de collisions entre les humains virtuels se fait de la même manière que l'évitement d'obstacles. Ceci est dû au fait qu'on traite toujours les acteurs et les obstacles comme des obstacles dynamiques et statiques, pour les deux le traitement se fait de la même façon.

L'évitement de collisions entre les individus d'une scène dynamique est un domaine très complexe. Il dépend de l'intelligence des individus mais dépend également des aspects sociologiques et psychologiques liés au comportement des êtres humains.

Le traitement de collisions dépend de la position et de la distance des individus. Dans notre simulation, nous avons opté les deux comme variables d'entrée. Pour acquérir le comportement d'évitement de collisions entre les humains virtuels, la logique floue est entraînée en utilisant un contrôleur flou (CF) basé sur des informations capturées pour déterminer le fonctionnement de ce dernier. Le comportement de l'évitement de collision est affecté par l'environnement individuel.

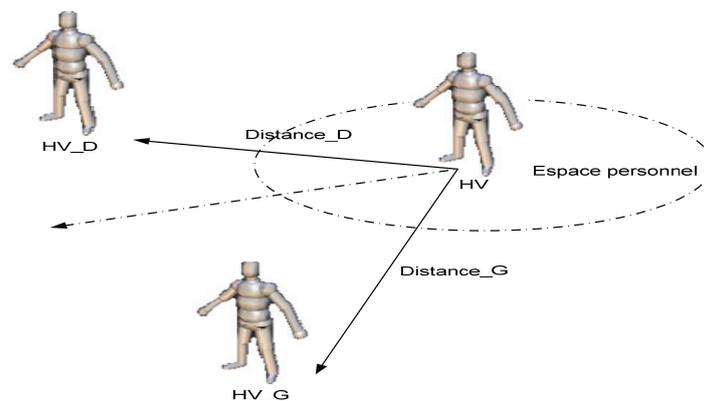


Figure 5.15 : Eviter la collision. Les distances capturées dépendent de la position des HV réparties dans le champ de vision.

5.2 Comportement d'évitement d'obstacles

Le comportement d'évitement d'obstacles est un comportement similaire au comportement décrit précédemment, avec moins de complexité car nos humains virtuels perçoivent les situations dans tels environnements comme des situations topologiques : murs, cubes, c'est-à-dire des objets statiques comme présentés dans la figure suivante :

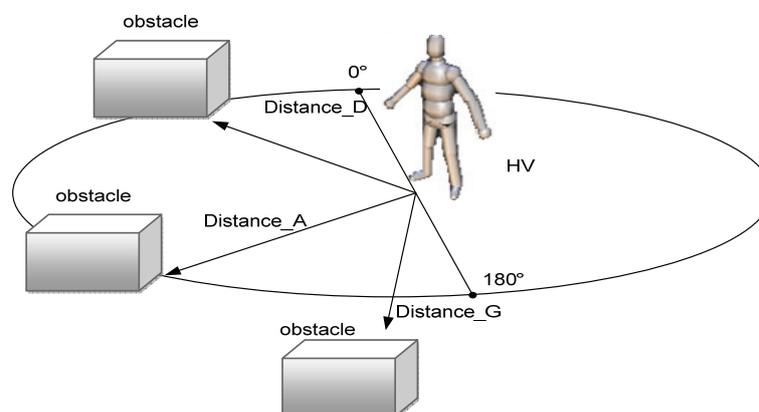


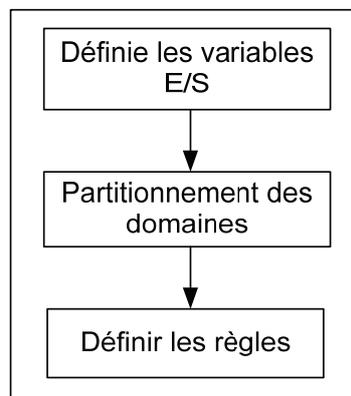
Figure 5.16 : Eviter les obstacles.

La technique utilisée dans Massive (la logique floue) permet de résoudre des situations complexes avec peu de règles. Ainsi le problème de l'évitement d'obstacles peut être résolu en posant quelques règles simples. Il faut pour cela identifier correctement et clairement le problème. Comme un exemple, nous allons voir comment un humain virtuel « agent » en état de mouvement simple ou complexe peut éviter les obstacles disposés dans un environnement comme présenté dans la figure précédente. Tout d'abord nous posons le problème comme suit :

Il s'agit ici d'éviter la collision d'humain virtuel avec les obstacles présents dans l'environnement. On ne se soucie pas de la direction que prend l'humain virtuel pour éviter l'obstacle ni de la mémorisation des obstacles rencontrés. Il n'y a pas de construction de carte, pas de navigation vers un but dans le comportement d'évitement d'obstacles.

La définition d'un problème en logique floue se décompose en trois parties principales :

- Définition des variables : quelles sont les entrées, les sorties disponibles et pertinentes.
- Partitionnement des domaines de définitions de ces variables, on crée alors des sous-ensembles flous (obstacle près, très près ou éloigné, ...).
- Définition des règles qui vont donner un comportement à suivre pour chaque situation.



5.2.1 Définition des variables

Le problème de l'évitement d'obstacles doit forcément s'appuyer sur des capteurs donnant des renseignements sur l'environnement. Nous utiliserons ici des capteurs préconfigurés dans l'humain virtuel. Deux capteurs nous permettent de relever les distances des obstacles situés dans les zones : **droite, gauche et devant.**

Nos capteurs travaillent sur le plan horizontal, c'est-à-dire l'axe (x) et l'axe (z) comme illustré dans la figure 5.17, cette vision représente la direction vers la cible de l'humain virtuel dans le plan plat.

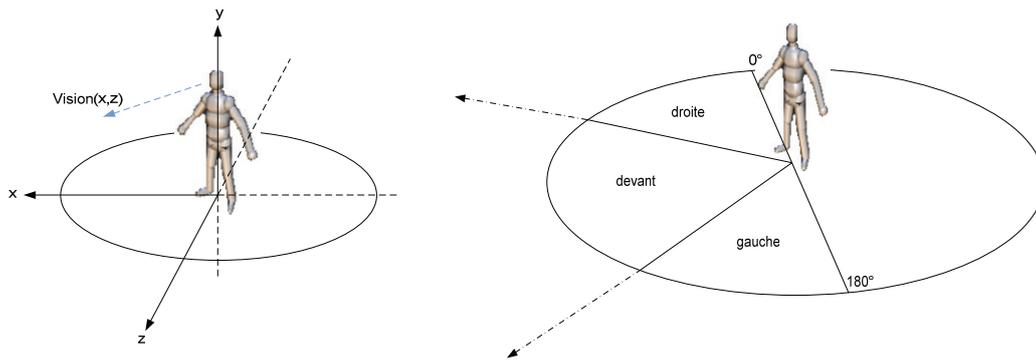


Figure 5.17 : Les champs de vision de l'agent.

Nos règles posséderont donc deux variables d'entrées, la première variable prend des valeurs de l'axe 'x' qui indiquent la position de l'obstacle. La deuxième variable prend des valeurs de l'axe 'z' pour indiquer la distance. Voyons maintenant les variables de sortie :

Pour déplacer l'humain virtuel « agent », on fournit à l'humain virtuel une valeur de position et une valeur de distance. Dans notre cas, on combine les deux, pour effectuer les actions (tourner à gauche, tourner à droite). Ces actions représentent les résultats des règles.

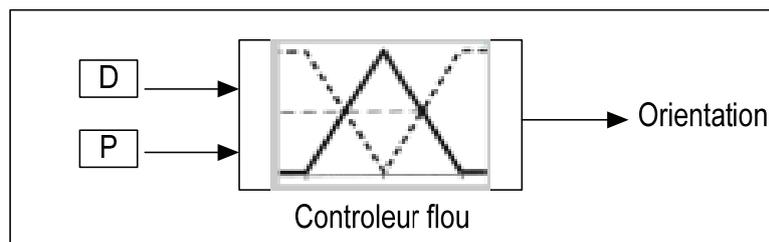


Figure 5.18 : Représentation de contrôleur flou (D : la distance. P : la position).

5.2.2 Partitionnement des domaines de variables

Pour le partitionnement des domaines de ces variables : les variables de **distance** et de **Position** seront comme suit :

Les éléments de vecteur **Position** possèdent les valeurs qualitatives : loin droite (far right), droite (right), center, gauche (left) et loin gauche (far left) définies par les fonctions d'appartenance comme illustré dans la figure suivante :

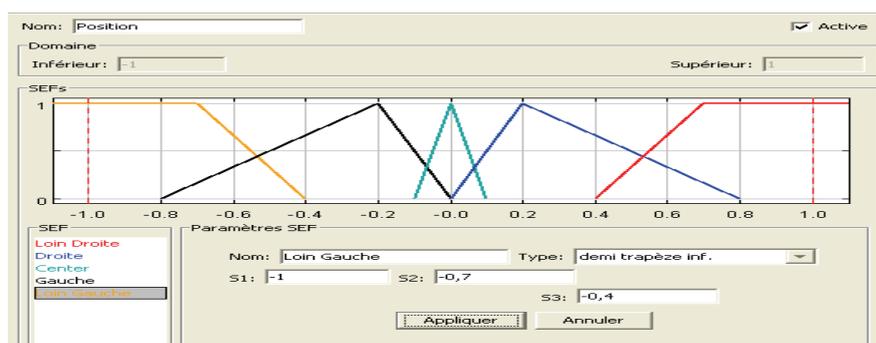


Figure 5.19 : fonction d'appartenance de Position.

La variable **Distance** possède les valeurs qualitatives : Prés (P), moyen (M) et Loin (L) définis par les fonctions d'appartenance comme illustré dans la figure 5.19.

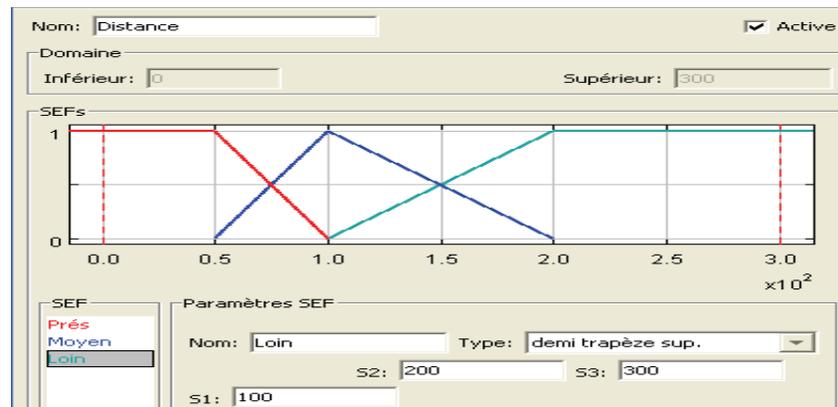


Figure 5.20 : fonction d'appartenance de distance.

L'orientation des humains virtuels, O_d (tourner droite) et O_g (tourner gauche), sont respectivement les sorties du contrôleur flou (CF) dont les variables floues correspondantes sont représentées dans la figure 5.14. Nous avons utilisé comme degré d'inclination 45° pour les deux sens de l'orientation (gauche et droite) comme illustré dans la figure suivante :

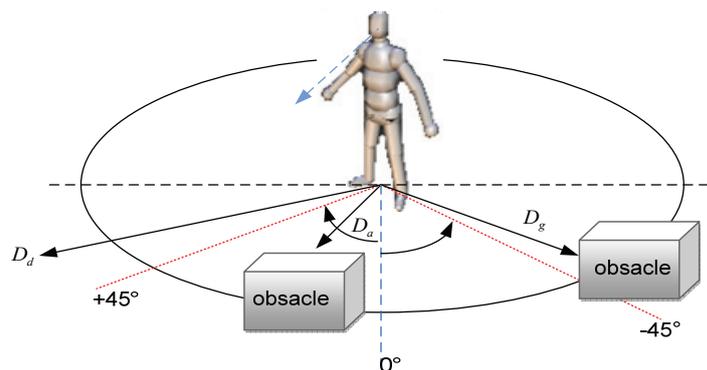


Figure 5.20 : Les deux sens de l'inclination.

La détection de la situation et l'évitement d'obstacle sont donc réalisées par un ensemble de règles floues de la forme (SI ALORS ...) comme suite :

- Si** (distance **est** loin) **et** (position d'obstacle est en droite loin) **alors** aller à droite
- Si** (distance **est** moyen) **et** (position d'obstacle **est** en droite) **alors** aller à gauche
- Si** (distance **est** moyen) **et** (position d'obstacle **est** en devant) **alors** aller à droite
- Si** (distance **est** moyen) **et** (position d'obstacle **est** en gauche moyen) **alors** aller à droite
- Si** (distance **est** loin) **et** (position d'obstacle **est** en gauche loin) **alors** aller à gauche

5.2.3 Définir l'ensemble des règles

Maintenant que chacune des variables et que le partitionnement de celle-ci est prêt, la définition va faire correspondre les entrées et les sorties. Pour cela il faut s'assurer que chaque sous ensemble de chacune des variables est utilisé au moins une fois. Si une situation correspondant à des "inputs" qui ne sont utilisés par aucune des règles, alors nous remarquons au niveau de la simulation, des erreurs comme la collision entre les humains virtuels et les obstacles ou situation de crash.

Les règles :

1. **Si** (D est loin) **et** (Po est en droite loin) **alors** O_d
2. **Si** (D est moyen) **et** (Po est en droite) **alors** O_g
3. **Si** (D est moyen) **et** (Po est en devant) **alors** O_d
4. **Si** (D est moyen) **et** (Po est en gauche moyen) **alors** O_d
5. **Si** (D est loin) **et** (Po est en gauche loin) **alors** O_g

D = la distance entre l'humain virtuel et l'obstacle (statique ou dynamique).

Po = la position de l'obstacle (localisation).

O = l'orientation de l'humain virtuel.

L'utilisation de ces règles dans un système flou se fait de la manière suivante :

- « fuzzification » des entrées.
- Combinaison des degrés d'appartenance.
- Combinaison avec les poids des règles.
- Agrégation des sorties des différentes règles.
- « défuzzification » de la sortie ainsi calculée.

La figure suivante illustre les étapes que nous avons pris dans notre travail.

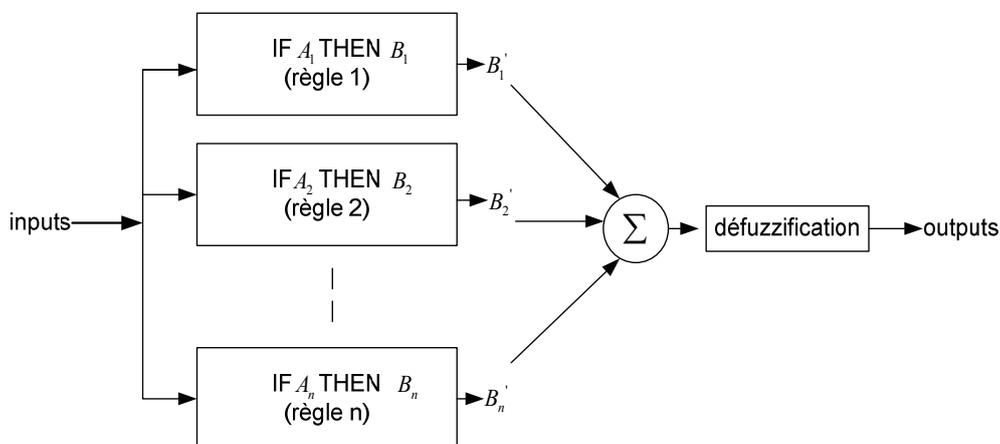


Figure 5.21 : Les étapes de fuzzification.

On a la possibilité pour représenter ces règles dans une table appelée table de décision.

Distance Position	Prés	Moyen	Loin
Droite loin			Aller D
Droite		Aller G	
Centre	Test sur les obstacles		
Gauche		Aller D	
Gauche loin			Aller G

Tableau 5.1 Table de décision.

La représentation des règles pour tous les comportements sera détaillée dans le chapitre suivant.

6. Conclusion

Nous avons essayé de mettre en évidence une modélisation d'animation de foule qui utilise le système multi-agents sur lequel repose notre étude, dont le but est de diminuer la complexité de navigation d'un ensemble d'humains virtuels dans un environnement totalement inconnu. Chaque humain virtuel est doté de faculté d'autonomie et d'intelligence de réagir aux événements perçus, et de s'interagir dans le groupe.

Pour Ces raisons, nous avons choisit d'adopter pour un humain virtuel une architecture sous forme de réseau flou (basé sur la logique floue), cette architecture est composée de quatre modules: module de perception, module de réaction, et un dernier module pour l'action.

CHAPITRE 6

Simulation et résultats

« Il est bien de changer un jour, puis de changer un autre jour. Il est bien de se dire aujourd'hui ceci s'est ajouté à moi, ceci s'est dégagé de moi, car c'est ainsi que l'homme devient un faisceau harmonieux dans une montée vers la simplicité »

Pierre Daco

Simulation et résultats

1. Introduction

La foule et les simulations des phénomènes collectifs deviennent de plus en plus importantes dans les sciences sociologiques et dans la simulation de secours (évacuation, manifestation,...). Elles peuvent être employées pour simuler la présence de vrais humains. Une telle technologie peut être employée dans des situations très dangereuses pour que des personnes réelles en soient les vrais acteurs. La recherche récente sur la simulation de foules a été en grande partie inspirée par les travaux de Craig Reynolds sur la simulation de groupes [Rey87] [Rey99]. Un élément principal de ce type d'animation est l'évitement de collision. Beaucoup de travaux ont abordé cette problématique en essayant d'apporter les solutions adéquates avec l'utilisation d'une nouvelle technique, qui est la logique floue. Au cours des années précédentes, l'utilisation de la logique floue confinée aux roboticiens. Aujourd'hui, l'utilisation de la logique floue dans l'intelligence artificielle, la cinématique, et les jeux vidéo, ...etc.

Dans les chapitres précédents, nous avons d'abord établi les contraintes et le cahier des charges d'un modèle de comportement, puis nous avons défini un modèle formel du comportement réalisé par Massive est montré par la technique de la logique floue. Puis nous avons présenté quelques résultats de simulation. Enfin, nous achèverons notre travail par rapport de simulation défini par l'outil « FisPro ».

2. Méthode de simulation

La figure 6.1 illustre les différents composants de notre méthode de simulation. Notre modèle de simulation est une scène composé de deux éléments principales « agents, environnement » déterminés par le réalisateur de l'animation.

Cette structure (voir la figure 6.1) représente les différentes unités principales de notre simulation. Elle détermine aussi les relations entre différents objets, qui permettent les échanges des données.

Notre simulation est une réflexion d'une séquence des images de scène, cette réflexion est une combinaison entre l'éclairage qui est engendré par les sources lumineuses et la projection qui capte et émis les images (les caméras) comme illustré dans la figure 6.2.

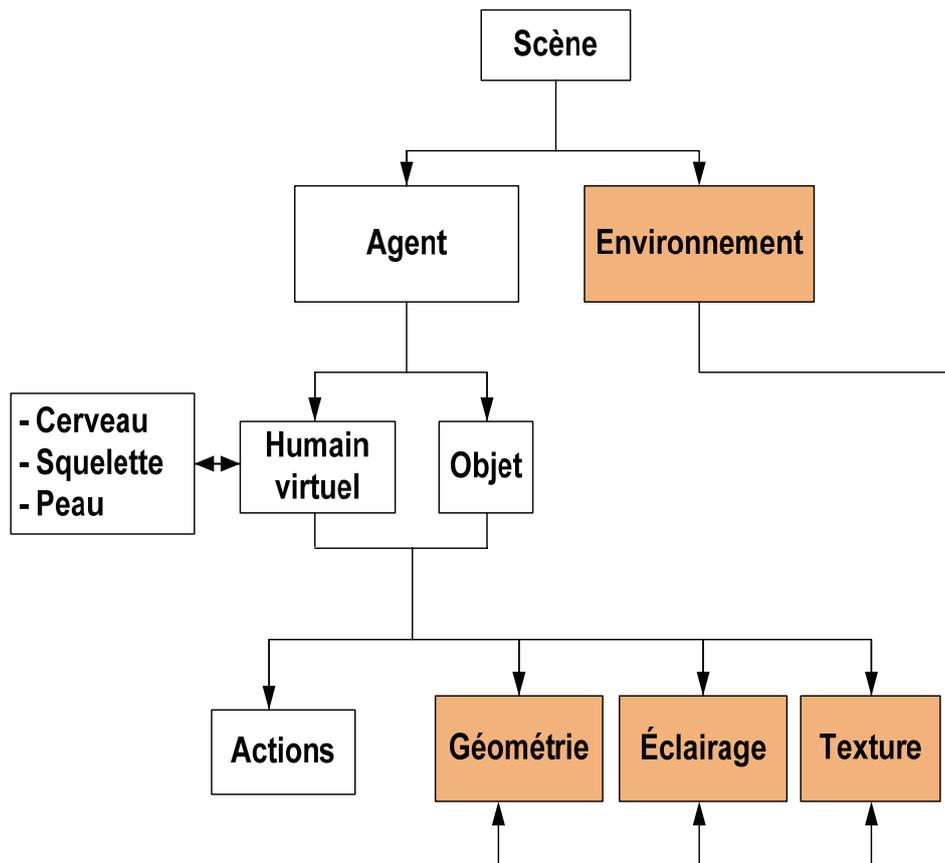


Figure 6.1 : Structure de simulation.

L'implémentation de notre scène en Massive est représentée dans la figure 6.2. Nous avons remarqué dans la fenêtre de scène qui est divisée en deux parties, une contient les éléments standards non visibles comme les caméras et les sources lumineuses, par contre la deuxième partie contient les éléments visibles au cours d'exécution comme les humains virtuels (agents) et les obstacles.

La réalisation de notre scène de simulation contient deux composants standards, généralement utilisés pour toutes les simulations telles que (les sources lumineuses, et les caméras) comme illustré dans la figure suivante :

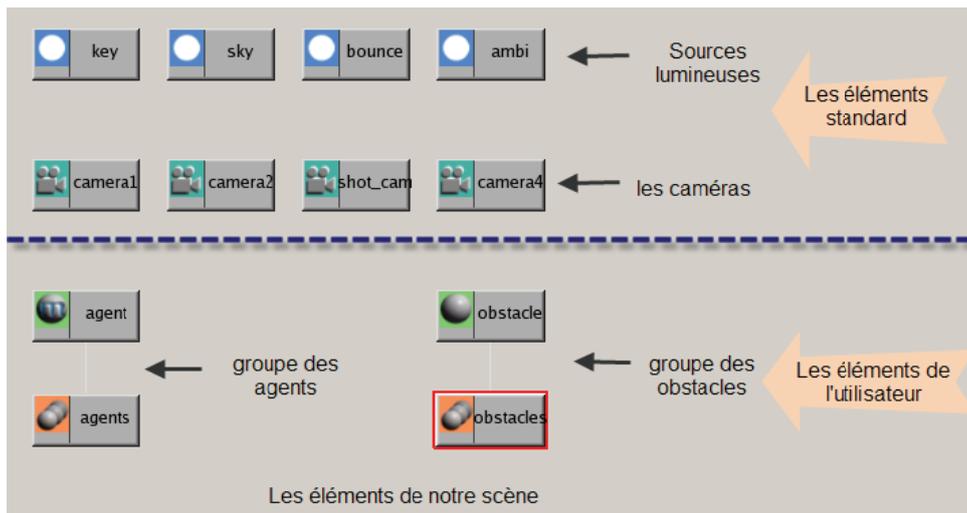


Figure 6.2 : Les éléments de notre scène défini en Massive.

Nous avons remplacé dans notre espace cartésien de la scène quatre sources lumineuses (voir la figure 6.3) et quatre caméras (voir la figure 6.5) pour obtenir une ambiance globale et pour projeter tous les coins possibles de la scène comme illustré dans la figure 6.3 et la figure 6.5.

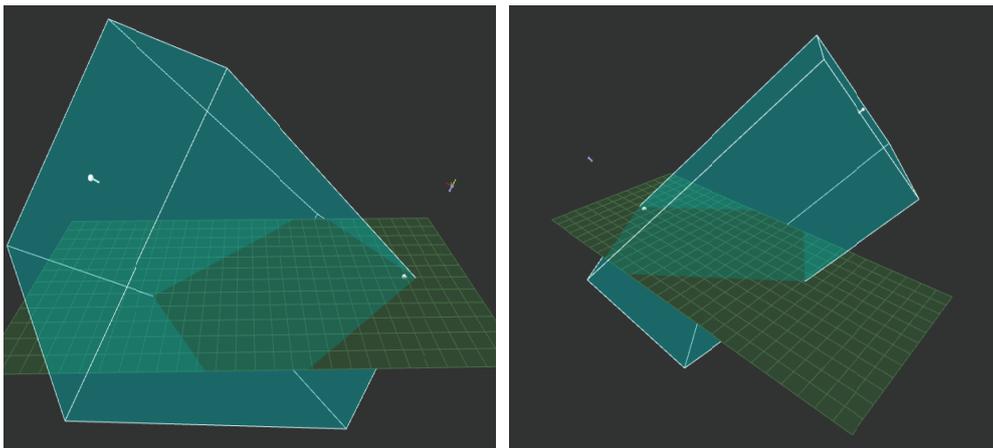


Figure 6.3 : Les sources lumineuses.

Massive offre la possibilité pour déterminer les propriétés des sources lumineuses, Où on a dans l'affichage par OpenGL, le rendu de lumière qui soit : ambiante, point, tache (spot) ou directionnel comme illustré dans la figure 6.4.



Figure 6.4 : Les types de rendu.

La meilleure façon pour déplacer la caméra autour est simple, en naviguant dans la fenêtre de vue, avec un glissement dans la fenêtre d'affichage en utilisant le bouton gauche de la souris provoquant la caméra tourner.

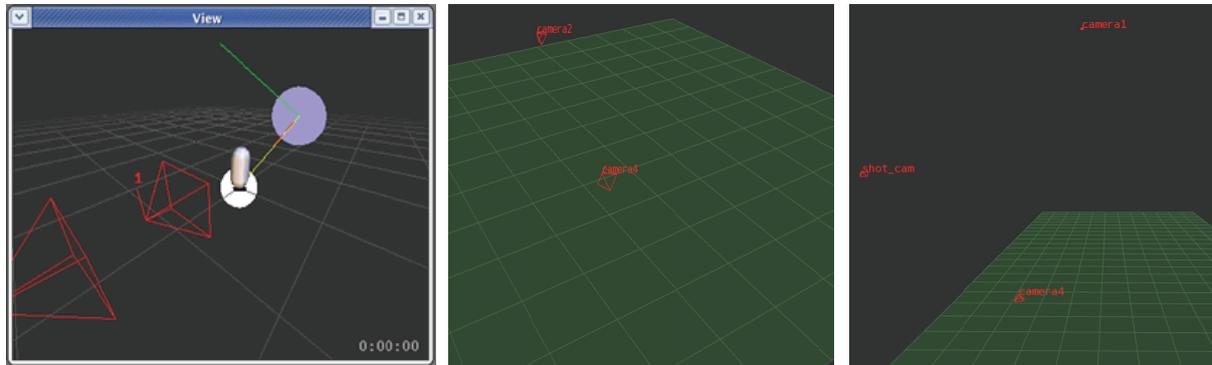


Figure 6.5 : Les positions des caméras.

Rappelons que notre but est de simuler le mouvement d'une foule dans un environnement inconnu. La scène est ainsi formée par un ensemble d'obstacles et un ensemble de groupes d'individus.

Nous avons choisi deux types d'objets à modéliser : l'environnement et l'humain virtuel.

2.1 Environnement

L'environnement U est un ensemble qui contient tous les objets de la simulation. Un objet o est constitué d'une forme géométrique s et d'un ensemble P de propriétés nommées. Une propriété est un couple $\langle \lambda, v \rangle$: λ est le label (unique dans l'ensemble des propriétés de l'objet) et v la valeur de la propriété. Pour simplifier le formalisme, cette valeur est typée; si elle prend une valeur symbolique, elle peut être liée aux autres propriétés du même objet par un connecteur. Parmi les propriétés, certaines sont inhérentes à tous les objets comme la localisation dans l'espace (les positions dans le repère euclidien (x, y, z) ainsi que son orientation suivant les trois axes (x, y, z) , les autres sont définies par l'utilisateur. Un objet est localisé par un triangle (voir figure 6.6). Un objet possède à tout moment une géométrie active. Ces géométries sont constituées des composantes classiquement utilisées dans les rendus 3d utilisant OpenGL (le matériau est pris en compte (sa couleur et sa réaction suivant la luminosité) ainsi que la définition de l'ensemble des faces triangulaires qui approximent sa forme).

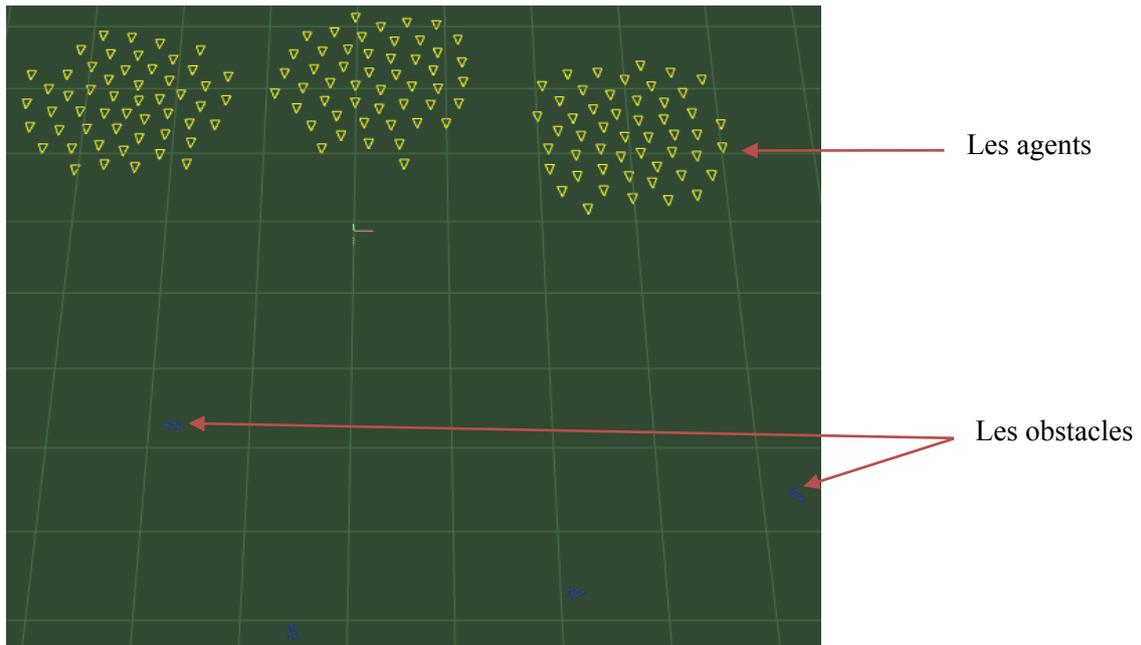


Figure 6.6 : Localisation des objets.

L'environnement est une imagination des besoins de l'utilisateur qui contient un ensemble des éléments statiques et dynamiques pour construire la scène tels que les obstacles. Dans notre simulation nous avons utilisé un terrain plat qui contient un ensemble d'obstacles, ces obstacles sont sous forme des boites (cubes), distribuées de façon aléatoire dans notre terrain, ces obstacles sont modélisés par le logiciel Massive comme illustré dans la figure suivante :

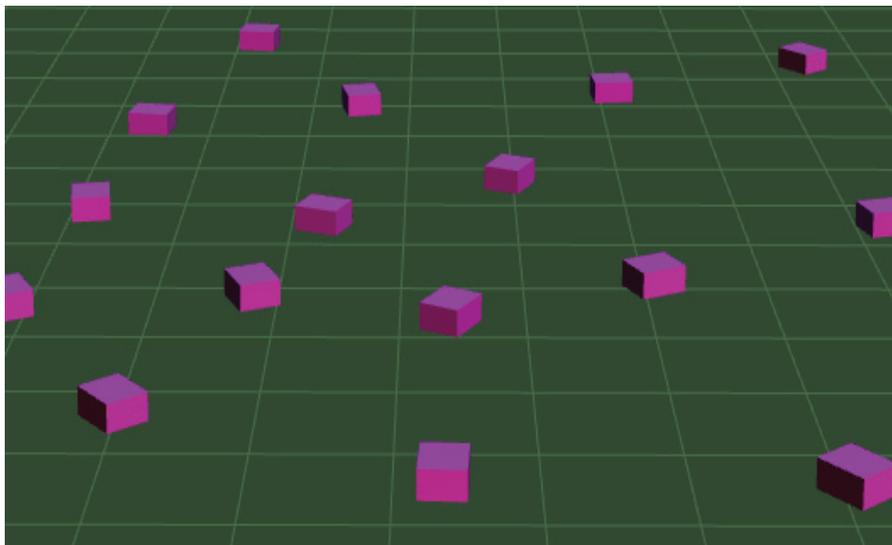


Figure 6.7 : Les obstacles statiques.

2.2 Humain virtuel (agent)

La modélisation de l'humain virtuel (agent) est un processus de liaison entre des formes géométriques (sphère, cube, tube, disque, ...), comme représenté dans la figure. Ceci nous donne à la fin une forme du corps humain acceptable. La connexion entre les segments par le connecteur ressort « spring » effectue une rigidité pour l'agent.

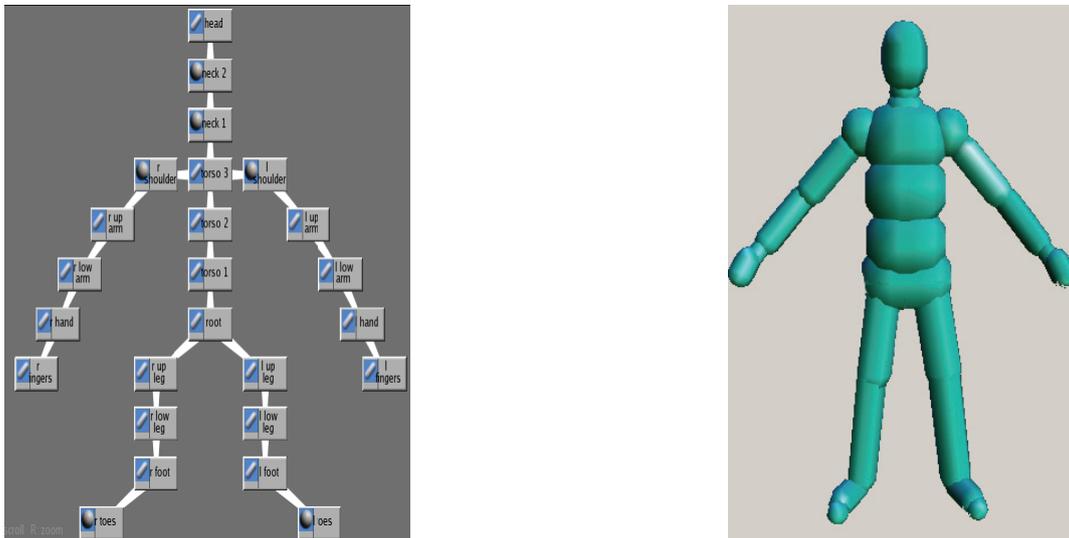


Figure 6.8 : Modélisation de l'humain virtuel.

Une fois la modélisation des humains virtuels « agents » est terminée, nous commençons à remplir notre base d'actions par des comportements comme illustré dans la figure.

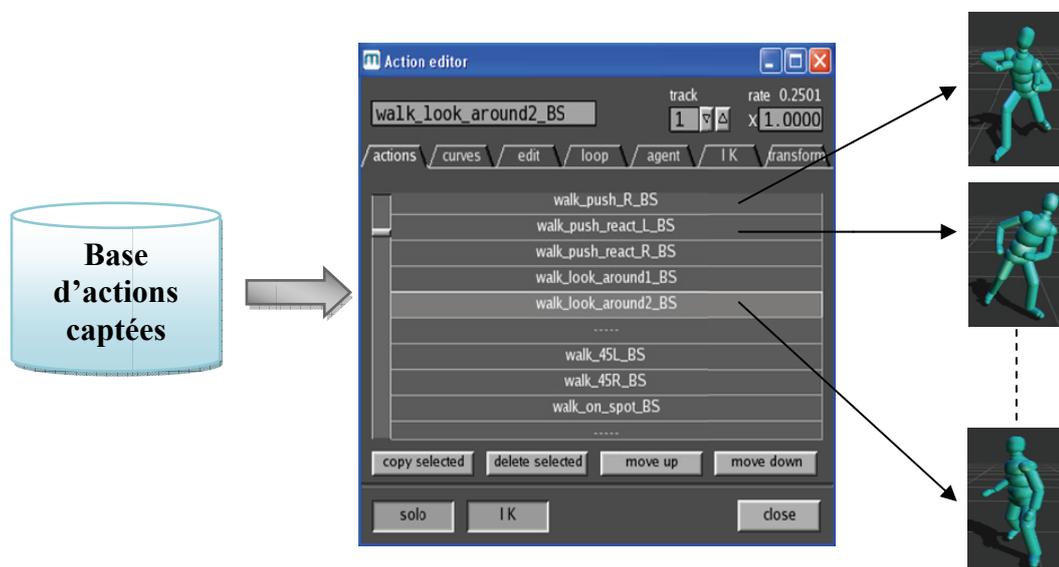


Figure 6.9 : L'éditeur d'actions.

Ainsi, d'une façon globale les éléments qui composent notre scène sont montrés dans la figure ci-dessous (voir figure 6.10). Nous utilisons comme obstacles statiques des boîtes sous forme de cube distribués de façon aléatoire sur un terrain plat. La foule est une composition de trois groupes comme illustré dans la figure ci-dessous, un groupe est une collection

d'humains virtuels « agents » réagit avec un environnement totalement non connu et orientés vers un seul sens.

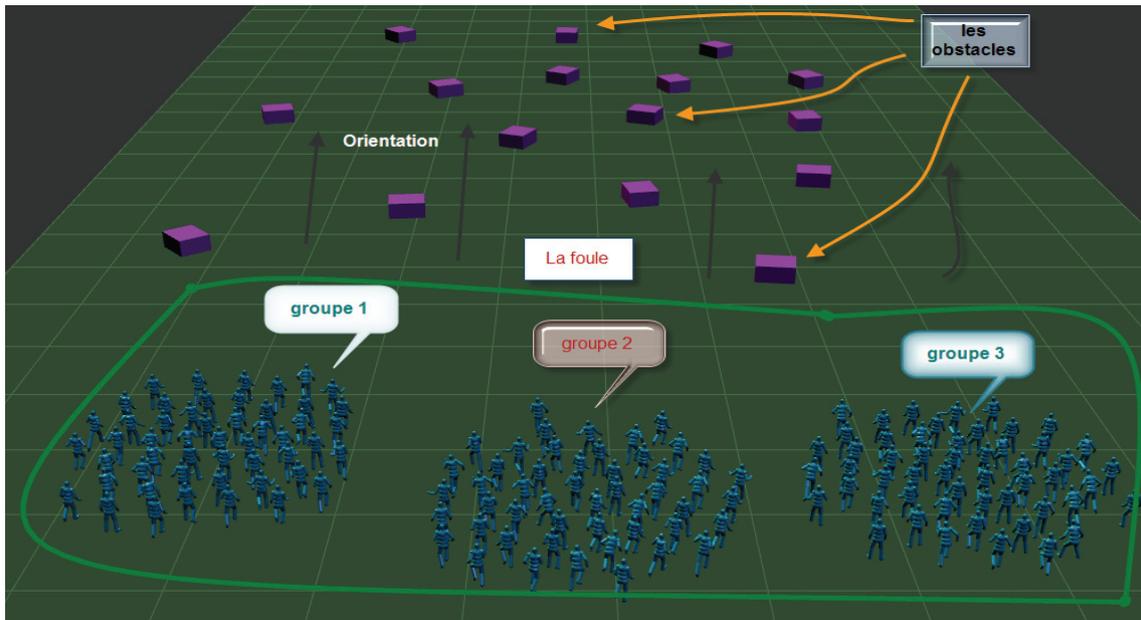


Figure 6.10 : Une vue globale.

3 modélisations des comportements

La complexité de l'animation des phénomènes collectifs d'une foule virtuelle située dans la modélisation des comportements des humains virtuels, nous présentons dans la figure ci-dessous les besoins d'un humain virtuel pour prendre un comportement avec un rendu réaliste.

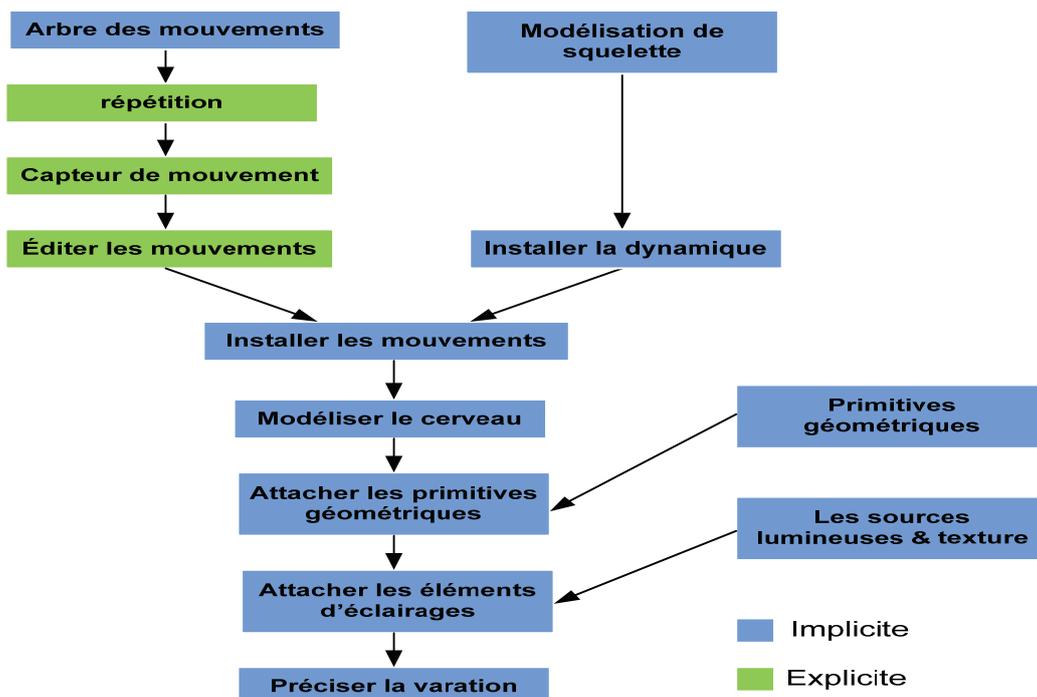


Figure 6.11 : Les besoins d'un humain virtuel.

3.1 Comportement d'évitement de collision

L'évitement de collisions entre les humains virtuels dans une scène se fait de la même manière que l'évitement d'obstacles comme est montré dans le chapitre précédent. Il dépend de l'intelligence des individus. Dans notre modèle de simulation le traitement de l'évitement de collisions dépend de la position et de la distance des individus comme illustré dans la figure 6.11 qui représente le réseau flou utilisé pour l'évitement de collision.

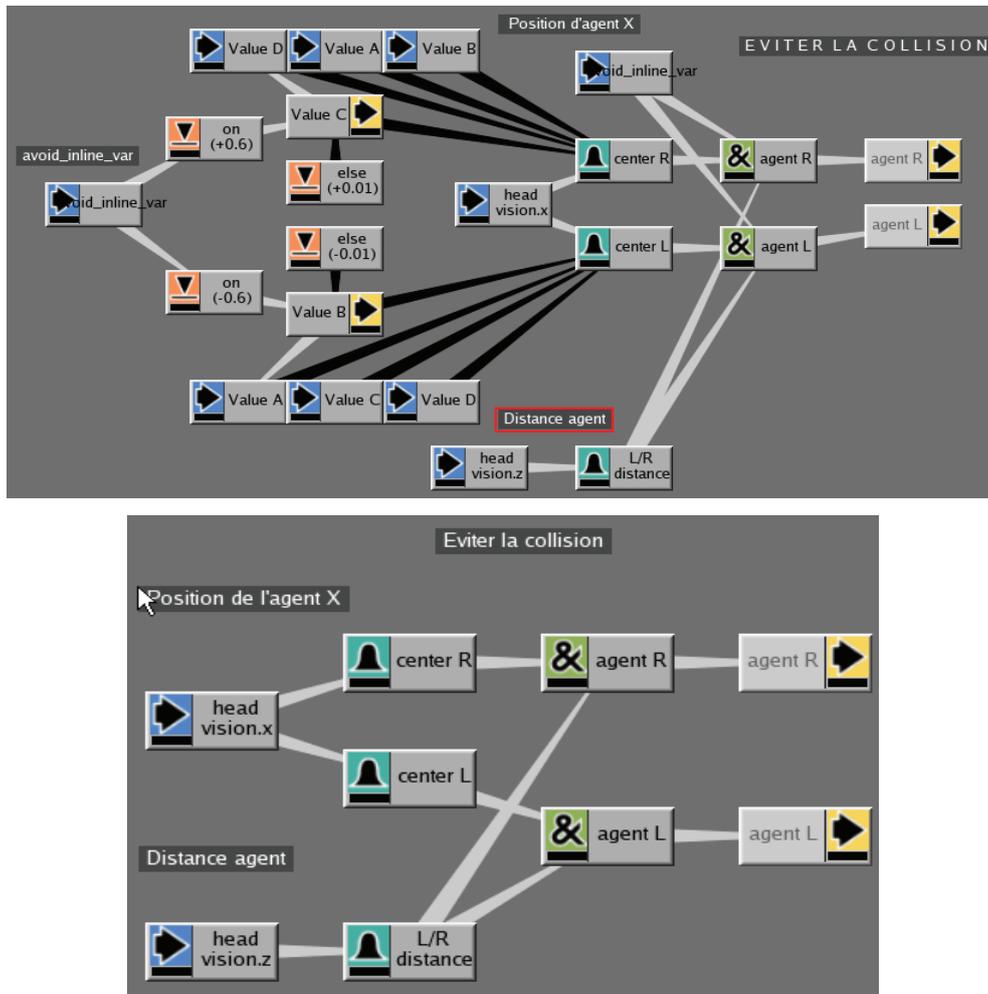


Figure 6.12 : Réseau flou « éviter la collision ».

Pour une bonne illustration du réseau flou ci-dessus nous traduisons ce dernier par l'algorithme suivant :

Programme Eviter_collision;

Inputs : Distance, Position;

Outputs : Orientation;

Fuzzy input values based on distance: far, center, near;

Fuzzy input values based on Position: right, left;

Fuzzy output values for orientation: go right, go left;

rules:

If (D is far) and (position agent is right) then go right;**If (D is center) and (position agent is right) then go left;****If (D is center) and (position agent is left) then go right;****If (D is far) and (position agent is left) then go left;**

Fin programme;

3.2 Comportement d'évitement d'obstacles

Le réseau flou utilisé pour l'évitement de collision est représenté dans la figure suivante :

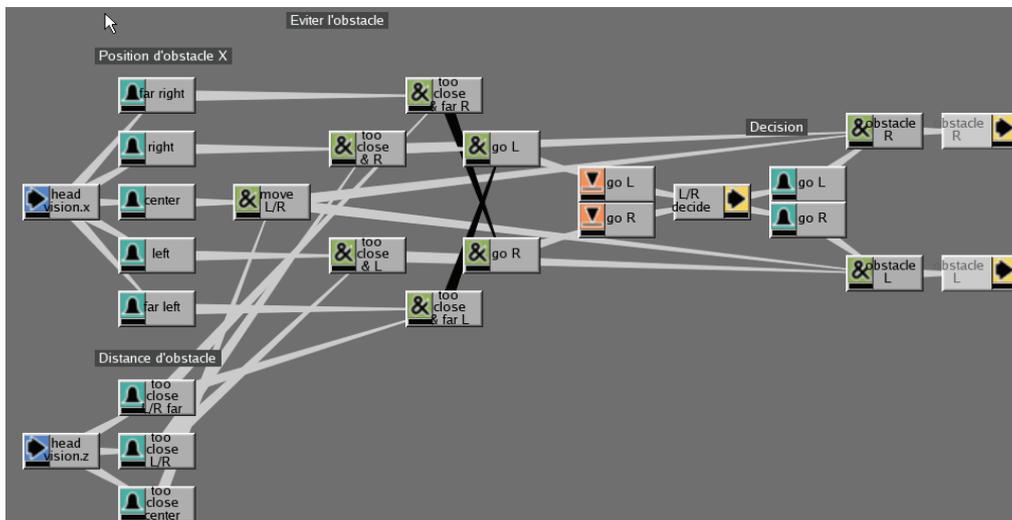


Figure 6.13 : Réseau flou « éviter les obstacles ».

Pour une bonne illustration du réseau flou ci-dessus nous traduisons ce dernier par l'algorithme suivant :

Programme Eviter_obstacle;

Inputs : Distance, Position;

Outputs : Orientation;

Fuzzy input values based on distance: far, center, near;

Fuzzy input values based on Position: far right, right, center, left, far left;

Fuzzy output values for orientation: go right, go left;

rules:

If (D is far) and (position obstacle is far right) then go right;**If (D is moyen) and (position obstacle is right) then go left;****If (D is center) and (position obstacle is center) then go right;****If (D is moyen) and (position obstacle is left) then go right;****If (D is far) and (position obstacle is far left) then go left;**

Fin programme;

4. Simulation d'un phénomène collectif « foule »

Dans cette partie, nous présentons quelques images qui montrent les comportements de nos humains virtuels en environnement plat non connu contient des obstacles. Cet environnement (imaginaire) a été modélisé avec Massive. Les figures ci-dessous organisées par un ordre chronologique depuis le point de départ jusqu'à la fin de simulation.

a) **Le point de départ** : c'est l'état initial pour notre simulation, il contient trois groupes d'humains virtuels, chaque groupe comporte 50 humain virtuel (agent), pour les obstacles nous utilisons 16 obstacles distribués de façon aléatoire comme illustré dans la figure suivante:

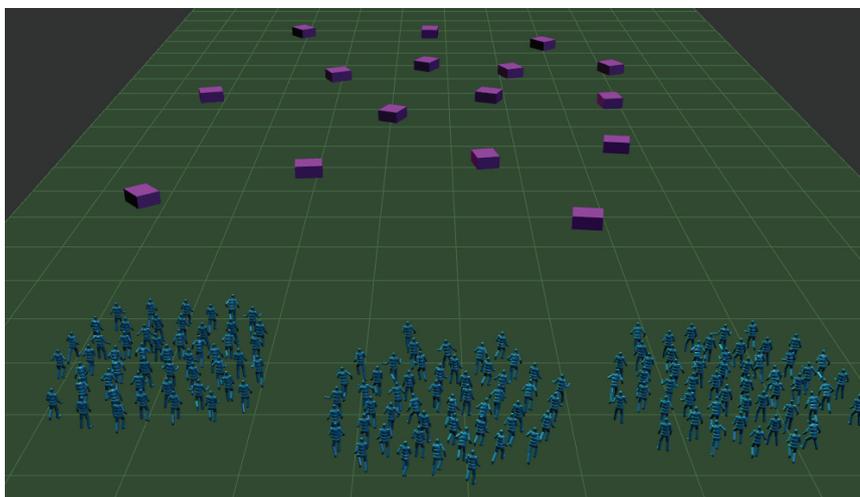


Figure 6.14 : Le point de départ.

- b) **Comportement d'évitement de collision** : car chaque humain virtuel contient un mécanisme donne une certaine intelligence au humain virtuel le traitement d'évitement de collision traite par le réseau flou comme est illustré dans la figure 6.11 et appliqué sur nos agents, la figure ci-dessous montre la représentation de comportement d'évitement de collision.

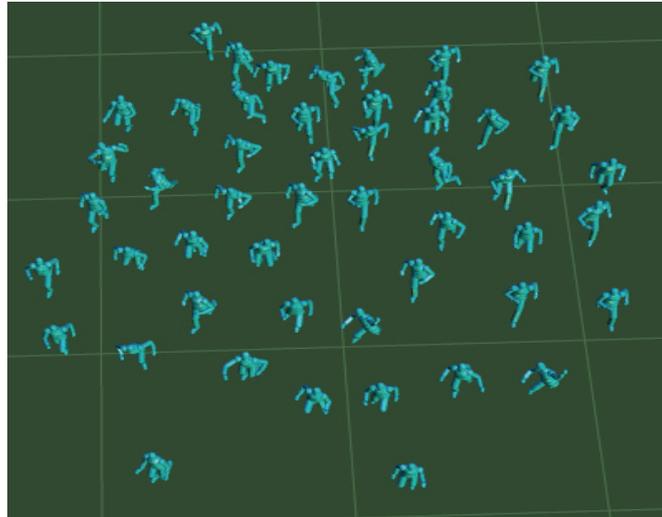


Figure 6.15 : Comportement d'évitement de collision.

- c) **Comportement d'évitement d'obstacles** : Comme nous l'avons vu plus haut, chaque action d'évitement de collision à besoin de différents comportements et de traitement différent, c'est le même processus pour le comportement d'évitement d'obstacles. Nous utilisons deux directions différentes (tourner 45° à gauche, tourner 45° à droite), où chaque action prendre les comportements à partir de la base d'actions comme nous montrons dans la figure 6.9. Le sens direct de l'humain virtuel est le comportement par défaut et les autres directions c'est les comportements traités par l'humain virtuel.

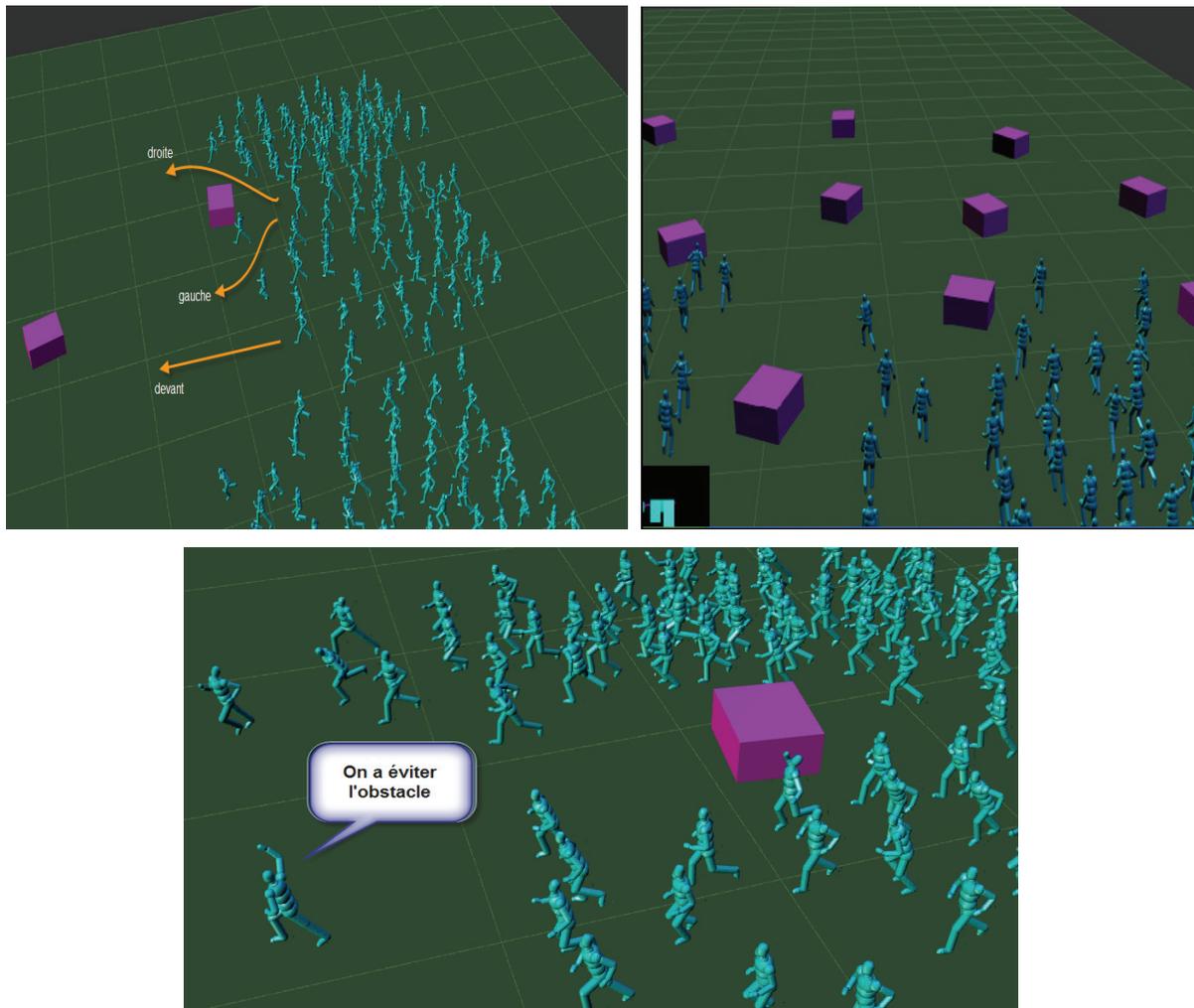


Figure 6.16 : Comportement d'évitement d'obstacles.

Maintenant, nous allons montrer quelques images par un ordre chronologique de deux situations, une pour évacuation simple dans une direction unique comme illustré dans les figures a, b et c

5. Rapport des résultats

Dans cette partie, nous allons présenter les différents résultats obtenus lors des simulations effectuées dans le cadre de FisPro.

FisPro (*Fuzzy Inference System Professional*) permet de créer des systèmes d'inférence floue, et de les utiliser à des fins de raisonnement, en particulier pour la simulation d'un système physique ou biologique. Ils fonctionnent à partir de règles de raisonnement floues, qui ont l'avantage de gérer la progressivité des phénomènes.

Ce logiciel est formé de deux parties distinctes : une bibliothèque de fonctions, écrite en C++, qui peut être utilisée de manière autonome et une interface utilisateur, écrite en Java, qui en implémente les principales fonctionnalités. Portable, il peut s'exécuter sur la majorité des plates-formes informatiques existantes.

Comme présenté dans le chapitre précédent, nous utilisons deux paramètres (distance, position) capturés par l'humain virtuel, ces paramètres entrants suffisants pour donner des connaissances au moins sur le champ de vision de l'humain virtuel pour éviter la collision par les obstacles statiques et dynamiques, c'est-à-dire pour prendre une direction convenable et similaire à la réalité.

La figure ci-dessous illustrée la représentation des données entrantes (capturés) par l'humain virtuel lesquels :

- Distance : représenté dans l'axe 'x'
- Position : représenté dans l'axe 'y'

La combinaison entre les deux valeurs capturés par l'humain virtuel prendre une décision finale pour orientation de l'agent et appelé Orientation comme présenter dans la figure 6.16.

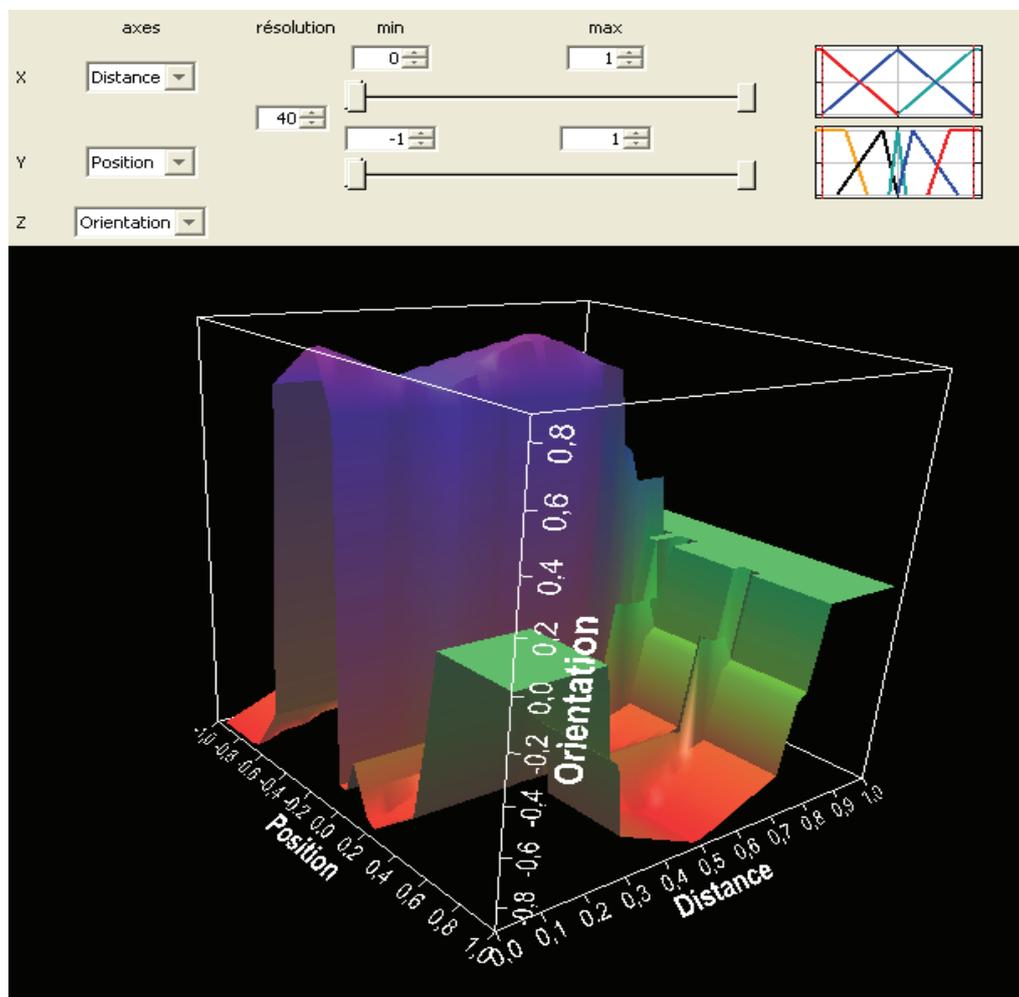


Figure 6.17 : Rapport de simulation.

La courbe 3 dimensionnels ci-dessus montre la relation entre les deux variables de distance et la position.

Pour bien comprendre la figure ci-dessus nous avons défini la courbe suivante :

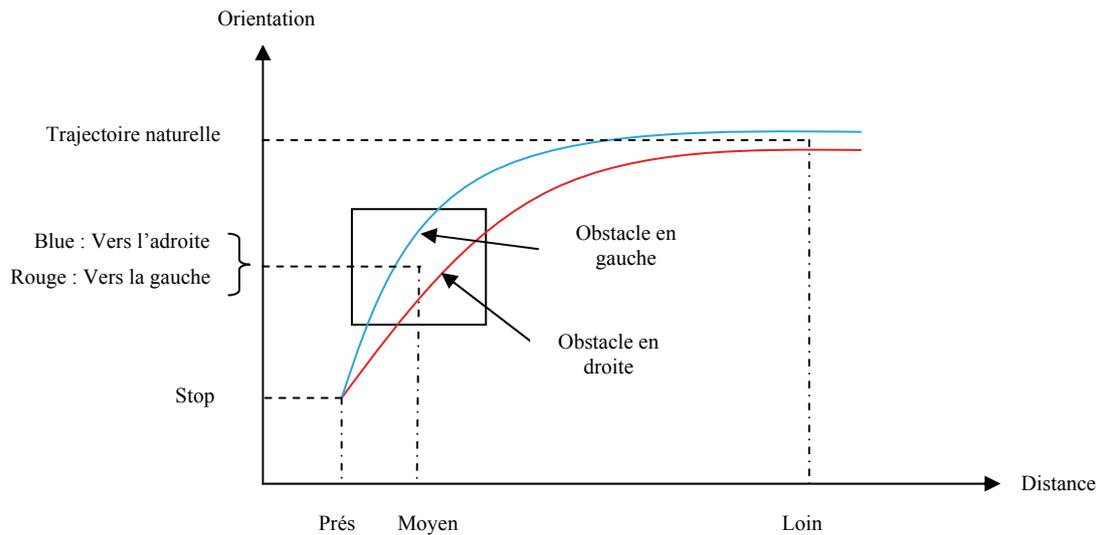


Figure 6.18 : L'orientation de l'humain virtuel.

La courbe ci-dessus représente la variation de trajectoire de l'humain virtuel dépend de la distance et la position de l'obstacle comme suite :

- Si la distance est très proche alors l'humain virtuel s'arrête.
- Si la distance augmente l'action suivante de l'humain virtuel dépend de la position de l'obstacle :
 - Si la position de l'obstacle est en droite alors le mouvement de l'humain virtuel sera vers la gauche.
 - Si la position de l'obstacle est en gauche alors le mouvement de l'humain virtuel sera vers la droite.
- Si la distance est très loin, l'humain virtuel continue son mouvement.

6. Conclusion

Nous avons décrit dans ce chapitre un modèle pour simuler le mouvement des foules autonome d'humains virtuels qui respecte le principe de la boucle perception-décision-action. Nous avons simulé une situation normale de foule en état de déplacement dans un environnement virtuel contient des obstacles. L'idée originale de ce travail est de présenter la possibilité de simuler plusieurs niveaux de réalisme comprenant des comportements d'individus et de groupes avec la technique « la logique floue ».

Conclusion générale

« Les braves gens ne savent pas ce qu'il en coûte de temps et de peine pour apprendre à lire. J'ai travaillé à cela quatre vingt ans et je ne peux pas dire encore que j'y suis arrivé »

Goethe

Conclusion générale

Le travail présenté dans ce mémoire a montré l'utilité d'une approche basée sur les concepts de la théorie de la logique floue pour aborder le problème complexe qui constitue l'animation des phénomènes collectifs. Il concerne la simulation des comportements innés d'humains virtuels dans leur environnement virtuel. La problématique que nous sommes confrontés, est celle de simuler les comportements d'entités autonomes « foule d'humains virtuels » en interaction dans un univers virtuel.

La simulation des phénomènes collectifs d'humains virtuels est un sujet très récent dans la recherche en infographie, et a pour but de générer facilement les mouvements, action et comportements d'un grand nombre d'individus « foule ». Il existe plusieurs types d'applications de simulation de foule. Ces travaux cherchent toujours à appliquer l'intelligence et l'autonomie aux humains virtuels pour obtenir des agents automatiquement animés.

La simulation macroscopique a été historiquement la première approche utilisée pour simuler les déplacements de personnes, du fait de son faible coût de calcul. En effet, dans cette approche, le piéton n'est pas traité individuellement mais en tant qu'élément constitutif d'une matière plus macroscopique. Cependant, les méthodes microscopiques ne peuvent pas remplacer les méthodes macroscopiques, mais elles sont considérées comme une analyse plus détaillée pour la conception et l'interaction de piétons considérant que des caractéristiques individuelles et des facteurs psychologiques et sociologiques.

Nous avons proposé une architecture adressant le problème de simulation des phénomènes collectifs d'humains virtuels autonomes qui se déplacent naturellement dans un environnement virtuel non connu, notre modèle étudie et simule les mouvements des agents dans un espace contenant des obstacles.

Les mouvements de foule virtuelle dans ce modèle sont décrits par les actions qui sont captées et stockées dans une base d'actions, nous obtenons ces actions par la capture de mouvement, ces actions réfléchissent un aspect socio-psychologiques.

Dans cette étude on a introduire la théorie des sous-ensembles flous, Zadeh a offert un outil puissant pour la modélisation des systèmes complexes, pour lesquels on ne dispose que d'une spécification approximative ou imprécise. Le but d'un modèle est de capturer la relation entre les entrées et les sorties d'un système (réseau flou). A l'encontre d'un modèle

conventionnel qui décrit cette relation par une loi mathématique, un modèle flou l'a décrit linguistiquement.

De façon générale, les ensembles flous peuvent intervenir efficacement dans la modélisation des systèmes complexes, principalement en raison de leur capacité à synthétiser des informations, à permettre une approche globale de certaines caractéristiques du système grâce à la gradualité qui leur est inhérente, et également, bien sur, en raison de leur aptitude à traiter des connaissances imparfaites, c'est-à-dire incomplètes, approximatives, vagues, soumises à des erreurs de mesure, ...

Les modèles flous ont deux propriétés essentielles :

- Le traitement se fait au niveau symbolique. En fait, ces modèles sont conçus pour manipuler des valeurs linguistiques, comme c'est le cas chez l'homme. Ceci est devenu possible grâce aux variables linguistiques et à la représentation des spécifications des valeurs linguistiques par des ensembles flous.
- Ils sont capables de représenter l'imprécision et l'incertitude d'un expert humain. Cette propriété est particulièrement intéressante, car les modèles flous sont souvent inspirés de la connaissance humaine. Il est donc nécessaire d'inclure l'imprécision et l'incertitude que contient cette connaissance dans le modèle flou.

Les comportements d'humains virtuels comme l'évitement de collision et l'évitement d'obstacles sont décrits par la technique de la logique floue qui s'applique sur les cerveaux des agents. afin de décrire les comportements d'humain virtuel de la foule dans des situations différentes.

Les résultats obtenus en simulation montrent la faisabilité de cette approche tels que :

- ✓ Le chemin de déplacement est proche de l'optimal.
- ✓ Le système produit en permanence le bon chemin autour des obstacles.
- ✓ Les techniques adoptées floues fournissent une stratégie de comportement flexible et donne la possibilité de construire une représentation de l'environnement nécessitant un espace mémoire réduit par rapport à l'utilisation d'une description géométrique exact.

Ses avantages viennent notamment de ses capacités à :

- formaliser et simuler l'expertise d'un opérateur ou d'un concepteur dans la conduite et le réglage d'un procédé.
- donner une réponse simple pour les procédés dont la modélisation est difficile.
- prendre en compte sans discontinuité des cas ou exceptions de natures différentes, et les intégrer au fur et à mesure dans l'expertise.

- prendre en compte plusieurs variables et effectuer de la « fusion pondérée » des grandeurs d'influence.

Perspectives

La réalisation de ce travail de mémoire de magister a ouvert différents axes de réflexion et offre plusieurs perspectives envisagées pour faire évoluer les systèmes d'animation de foule qui sont assez nombreuses. Celles qui nous apparaissent les plus importantes et qui, par conséquent, constituent la suite logique de cette étude, sont résumées par les points suivants :

- Il sera très intéressant d'ajouter la texture pour les obstacles et le sol.
- L'ajout d'habillage pour les humains virtuels.
- Réaliser une base d'actions pour les mouvements les plus fréquents comme (courir, marcher, sauter,...).
- Implémenter la technique utilisée dans notre travail (logique floue) par les langages de programmation (C++, Java, ...).

Pour conclure, il serait intéressant d'élargir le champ d'application de la méthodologie proposée à d'autres problèmes de l'animation des phénomènes collectifs et notamment à l'animation des phénomènes collectifs d'humains virtuels.

Bibliographie

- [Ahn06] Ahn, J, Oh, S. and Wohn, K. “*Optimized Motion Simplification for Crowd Animation. Computer Animation and Virtual Worlds*”. pp: 155-165. (2006).
- [Aie87] Aiello, J.R. Human Spatial Behaviour. In D. Stokols & I. Altman, “*Handbook of environmental Psychology*”: Volume 1. New York: Wiley & Sons, pp: 389-504. (1987).
- [All02] Allbeck, J. and Badler, N. “*Embodied Autonomous Agents. Handbook of Virtual Environments: Design, Implementation and Applications*”. K. Stanney. Lawrence Erlbaum Associates, Philadelphia, pp: 313-332. (2002).
- [All90] Allen Newell (1990). “*Unified theories of cognition*”. Harvard University Press, Cambridge, Massachusetts.
- [Ant98] Eric Darnell and Tim Johnson, PDI/DreamWorks, 1998. http://www.dreamworksanimation.com/dwa/opencms/movies/movie_antz.html
- [Arc79] J. Archea. The evacuation of non-ambulatory patients from hospital and nursing home fires: A framework for a model. Technical report. NBSIR 79-1906. 1.3.2 (11/1979)
- [Ash01] Ashida (K.), Lee (S.), Allbeck (J.), Sun (H.), Badler (N.) et Metaxas (D.). “*Pedestrians: Creating agent behaviors through statistical analysis of observation data*”. Dans : *Computer Animation*, éd. par Society (IEEE Computer), pp. 84-92. Seoul, Korea, (2001).
- [BCN97] Bouvier, Cohen et Najman. “*From crowd to airbag deployment: particle systems, a new paradigm of simulation*”. *Journal of Electronic Imaging*, vol. 6, n 1, pp. 94-107, (Janvier 1997).
- [Ben95] Belnesch Hellmuth. Atlas de la Psychologie, Encyclopédies d’Aujourd’hui, (1995).
- [Bez92] J. C. Bezdek et S. K. Pal, “*Fuzzy models for pattern recognition*”, *IEEE press*, New York, (1992).
- [BH97] Brogan (D.C.) et Hodgins, (J.). “*Group Behaviours for Systems with Significant Dynamics*”. *Autonomous Robots*, 4, pp: 137-153, (1997).
- [Bin00] Bindiganavale, R., Schuler, W., Allbeck, J., Badler, N., Joshi, A. and Palmer, M. “*Dynamically Altering Agent Behaviors Using Natural Language Instructions*”.

- In. Proc. Autonomous Agents, ACM New York, NY, USA: 293–300. (2000).
- [Bol81] Boles (W.). “*Planning pedestrian environments: A computer simulation model*”. *Man-Environment Systems*, vol. 11, pp. 41–56. (1981).
- [Bou95] Bouvier, E. and Cohen, E. “*Simulation Of Human Flow With Particles Systems*”. In. Proc. Simulators International XII, Phoenix. (1995).
- [Bow99] Bowers, J., Benford, S. D., Craven, M., Morphett, J., Wyver, J., Greenhalgh, C., “*Evaluating Out of This World: An Experiment in Inhabited Television*”, eRENA deliverable D7a.1, ESPRIT Long Term Research Project 25379. (1999).
- [Bra03] Braun (A.), Musse (S.R.), de Oliveira (L.P.L.) et Bodmann (B.E.J.). “*Modeling individual behaviors in crowd simulation*”. Dans : *CASA2003, 16th International Conference on Computer Animation and Social Agents*. pp. 143-148. Rutgers University, New Brunswick, New Jersey, USA, (05/2003).
- [Buc94] Buckmann, L. T. and Leather, J. “*Modelling Station Congestion the PEDROUTE Way. Traffic Engineering and Control*”. 35(6) pp : 373-377. (1994).
- [Bug99] John Lasseter, Andrew Stanton, “*A Bug’s Life*”. (1001 Pattes), Pixar, (1999). <http://www.pixar.com/featurefilms/abl/>
- [Che04] Stephen Cheney, “*Flow Tiles*”, Eurographics/ACM SIGGRAPH Symposium on Computer Animation R. Boulic, D. K. Pai (Editors) University of Wisconsin, Madison, (2004). <http://www.cs.wisc.edu/~schenney/>
- [Chi00] Chi, D., Costa, M., Zhao, L. and Badler, N. “*The EMOTE Model for Effort and Shape*”. In. Proc. ACM SIGGRAPH, New Orleans, LA, ACM Press. New York, USA, pp: 173-183. (2000).
- [Dav98] David C. Brogan, Ronald A. Metoyer et Jessica K. Hodgins, “*Dynamically Simulated Characters in Virtual Environments*”, Georgia Institute of Technology, Atlanta, (1998).
- [Dij00] Dijkstra, J., Timmermans, H. J. P. and Jessurun, A. J. A “*Multi-Agent Cellular Automata System for Visualizing Simulated Pedestrian Activity*”. In. Proc. Theoretical and Practical Issues on Cellular Automata. Cellular Automata for research and Industry, Springer-Verlag. Berlin, pp: 29-36. (2000).
- [Don04] Stéphane Donikian, “*Modélisation, contrôle et animation d’agents virtuels autonomes évoluant dans des environnements informés et structurés*”, Université de Rennes 1. (26/08/2004).
- [Dub80] D. Dubois et H. Prade, “*Fuzzy sets and systems: theory and applications*”,

- Academy Press, New York, (1980).
- [Fel84] Feldman, D. C. “*The development and enforcement of groups norms. Academy of management Review*”, pp. 47-53. (1984).
- [Fou06] Cherif FOUJIL, “*Animation comportementale : simulation de foule d’humains virtuels*”; université de Biskra. (12/12/2006).
- [Gal93] Galea, E. R., Perez Galparsoro, J. M. and Pearce, J. A “*Brief Description of the EXODUS Evacuation Model*”. In. Proc. International Conference on Fire Safety, San Francisco, USA, pp: 149-162. (1993).
- [Gol01] Goldenstein (S.), Karavelas (M.), Metaxas (D.), Guibas (L.), Aaron (E.) et Goswami (A.). “*Scalable nonlinear dynamical systems for agent steering and crowd simulation*”. *Computer & Graphics*, vol. 25, pp. 983-998. (2001)
- [Hal98] Hall, E. T. The Silent Language. Garden City: Anchor Books. “*Emerging Social Conventions: personal Space and Privacy in Shared Virtual Worlds*”, pp: 27-430 Project, (12/1998).
- [Har00] O’Hara (N.). “*K-d tree neighbour finding for the flocking algorithm*”. Dans: *Pacific Graphics*. (2000).
- [Hel00] Helbing, D., Farkas, I. and Vicsek, T. “*Simulating Dynamical Features of Escape Panic*”. *Nature*. 407, pp: 487-490. (2000).
- [Hen71] Henderson, L. F. “*The Statistics of Crowd Fluids*”. *Nature*. 229, pp: 381-383. (1971).
- [Hoo03] Hoogendoorn, S. P. “*Pedestrian Travel Behavior Modeling*”. In. Proc. Travel Behavior Research Lucerne, Elsevier, pp: 10-15. (2003).
- [Jef98a] Jeffrey, P., Mark, G., “*Constructing Social Spaces in Virtual Environments: A Study of Navigation and Interaction*”, Hook, K., Munro, A., Benyon, D., Workshop on Personalised and Social Navigation in Information Spaces, pp: 16-17, (03/1998).
- [Jef98b] Jeffrey, P. “*Emerging Social Conventions: personal Space and Privacy in Shared Virtual Worlds*”, CS, pp: 27-430 Project, (12/1998).
- [Kau98] Kauppinen, K.; Kivimäki, A.; Era, T. and Robinson, M. “*Producing identity in Collaborative Virtual Environments*”. In Proceedings of ACM Symposium on Virtual Reality Software and Technology, pp: 35-42. (1998).
- [Kir03] Kirchner, A., Namazi, A., Nishinari, K. and Schadschneider, A. “*Role of Conflicts in the Floor Field Cellular Automaton Model for Pedestrian*

- Dynamics*". In. Proc. 2nd International Conference on Pedestrians and Evacuation Dynamics. (*PED*), London, UK, pp: 51-62. (2003).
- [Lam03] LAMARCHE Fabrice, "*Humanoïdes virtuels, réaction et cognition : une architecture pour leur autonomie*" ; Université de Rennes 1. (19/12/2003).
- [Los03] Loscos, C., Marchal, D. and Meyer, A. "*Intuitive Crowd Behaviour in Dense Urban Environments using Local Laws*". IEEE Theory and Practice of Computer Graphics, pp: 122. (2003).
- [Lov94] Lovas, G. C. "*Modeling and Simulation of Pedestrian Traffic Flow*". *Transportation Research* **28**(6), pp: 429-443. (1994).
- [Mal87] Maletic, V. "*Body, Space, Expression: The development of Rudolf Labans Movement and Dance Concepts*". Mouton de Gruyter, New York. (1987).
- [Mal97] Mallot (H.A). "*Behavior-oriented approaches to cognition: theoretical perspectives*". Theory in biosciences, pp: 196-200. (1997).
- [Mal99] Hanspeter A. Mallot. "*Spatial cognition: Behavioral competences, neural mechanisms and evolutionary scaling*". pp 40-48. (1999).
- [Mas08] <http://www.geek.com/articles/news/massive-software-turns-talents-to-architecture-and-robotics-20080310/>
- [Mat94] Mataric (M. J.). "*Learning to Behave Socially*". Dans: D. Cliff, P. Husbands, J.A. Meyer & S.Wilson, eds, From Animals to Animats: International Conference on Simulation of Adaptive Behavior, pp: 453-462. (1994).
- [Mil69] Milgram (S.) et Toch (H.). *The handbook of social psychology*, chap. Collective Behavior: Crowds and social movements, pp. 507-610. Addison-Wesley, 1969.
- [Mil98] Milazzo, J. S., Roupail, N. M., Hummer, J. E. and Allen, D. P. "*The Effect of Pedestrians on the Capacity of Signalized Intersections*". Transportation Research Record. pp: 37-46. (1998).
- [Mot03] Mott-MacDonald. "*STEPS Simulation of Transient Evacuation and Pedestrian Movements User Manual, Mott MacDonald*". (2003).
- [Mou03] Moulin (B.), Chaker (W.), Perron (J.) et Pelletier (P.). "*Mags project: Multi-agent geosimulation and crowd simulation*". Dans : *COSIT 2003*, éd. par Kuhn (W.), Worboys (M.F.) et Timpf (S.), pp. 151-168. Springer-Verlag, 2003.
- [Mus, al 98] Musse, S. R., Babski, C., Capin, T. and Thalmann, D. "*Crowd Modelling in Collaborative Virtual Environments*". In. Proc. ACM Virtual Reality Software and Technology (*VRST*), Taipei, Taiwan, pp: 115-123. (1998).

- [Mus00a] Soraia Raupp Musse et Daniel Thalmann, “*Hierarchical Model for Real Time Simulation of Virtual Human Crowds*”, Computer Graphics Lab, Federal Institute of Technology, Lausanne, (2000).
- [Mus00b] Musse, S. R. and Thalmann, D. “*From One Virtual Actor to Virtual Crowds: Requirements and Constraints*”. In. Proc. (2000).
- [Mus00c] Musse (S. Raupp). “*Human Crowd Modelling with Various Levels of Behaviour Control*”. Lausanne, Suisse, Thèse de doctorat, EPFL, (janvier 2000).
- [Nur05] Nuria Pelechano, Kevin O’Brien, Barry Silverman et Norman Badler, “*Crowd Simulation Incorporating Agent Psychological Models, Roles and Communication*”, Center for Human Modeling and Simulation, University of Pennsylvania, Philadelphia, (2005). <http://www.seas.upenn.edu/~npelecha/>
- [Nur08] Nuria Pelechano, Jan M. Allbeck, and Norman I. Badler. “*Virtual crowds: methods, simulation, and control*”. Book. University of California–Berkeley. www.morganclaypool.com. (2008).
- [Owe98] Owen, M., Galea, E. R., Lawrence and Filippidis, L. “*The Numerical Simulation of Aircraft Evacuation and its Applications to Aircraft Design and Certification*”. The Aeronautical Journal. 102(1016), pp: 301-312. (1998).
- [Pan08] PANZOLI David. “*Proposition de l’architecture « Cortexionist » pour l’intelligence comportementale de créatures artificielles.* ”, Thèse de Doctorat, Université de Toulouse, (30/09/2008).
- [Par07] Paris (S.), “*Caractérisation des niveaux de services et modélisation des circulations de personnes dans les lieux d’échanges*”, Thèse de Doctorat, Université de Rennes I, (10/10/2007).
- [Ped90] W. Pedrycz, “*Fuzzy sets in pattern recognition: methodology and methods*”, Pattern Recognition, vol. 23, n° 12, pp: 121-146, (1990).
- [Pel06] Pelechano, N. and Badler, N. “*Modeling Crowd and Trained Leader Behavior during Building Evacuation*”. IEEE Computer Graphics and Applications. 26(6), pp: 80-86. (2006).
- [Pel07] Pelechano, N., Allbeck, J. and Badler, N. “*Controlling Individual Agents in High-Density Crowd Simulation*”. In. Proc. ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA), San Diego, CA, ACM Press. New York, USA, pp: 99-108. (2007).
- [Pes71] Peschl (I.). “*Flow capacity of door openings in panic situations*”. BAUN, vol.

- 26, n2, pp. 62-67. (1971)
- [Rey99] Reynolds, C. “*Steering Behaviors for Autonomous Characters*”. In. Proc. Game Developers Conference, pp: 763-782. (1999).
- [Rey87] Craig Reynolds, “*Flocks, Herds and Schools: A Distributed Behavioral Model*”, Proc. SIGGRAPH, Computer Graphics, v.21, n.4, Symbolics Graphics Division, Los Angeles, (1987). <http://www.red3d.com/cwr/>
- [Rob94] Robert G. Lord et P. E. Levy. “*Moving from cognition to action: A control theory perspective*”. Applied Psychology: an international review, pp: 335-398. (1994).
- [Sha05] Shao, W. and Terzopoulos, D. “*Autonomous Pedestrians*”. In. Proc. ACM SIGGRAPH / Eurographics Symposium on Computer Animation, Los Angeles, California ACM Press. New York, USA, pp: 19-28. (2005).
- [Sha07] Shao, W. and Terzopoulos, D. “*Autonomous Pedestrians*”. Graphical Models. 69, pp: 246-274. (2007).
- [Ste94] Steels (L.). “*The artificial life roots of artificial intelligence*”. In: Artificial Life Journal, Vol. 1(1), pp: 75-110, MIT Press, (1994).
- [Tec01] Tecchia, F., Loscos, C., Conroy, R. and Chrysanthou, Y. “*Agent Behavior Simulator (ABS): A Platform for Urban Behavior Development*”. In. Proc. ACM/EG Games Technology Conference. (2001).
- [Tha98] D.Thalmann, “*Using Virtual Reality Techniques in the Animation Process in: Virtual Reality Systems*”. (Earnshaw et al. eds), Academic Press, pp:143-159. (01/12/1998)
- [Tha96] D.Thalmann, “*A New Generation of Synthetic Actors: the Interactive Perceptive Actors*”, Proc. Pacific Graphics'96 Taipei, Taiwan, pp: 200-219. (1996).
- [Tho00] Thomas, G. and Donikian, S. “*Virtual Humans Animation in Informed Urban Environments*”. In. Proc. Computer Animation, IEEE Computer Society. Washington, DC, USA, pp: 112-121. (2000).
- [Tho95] Thompson, P. A. and Marchant, E. W. “*Testing and Application of the Computer Model Simulex*”. Fire Safety Journal 25, pp: 149-166. (1995).
- [TM95] Thompson (P.A.) et Marchant, (E.W.), “*Simulex: Developing new computer modeling techniques for evacuation*”. Unit of Fire Safety Engineering, University of Edhimburg. (1995).
- [Tor07] Torrens, P. M. “*Behavioral intelligence for geospatial agents in urban*

- environments*". In. Proc. IEEE Intelligent Agent Technology, Los Alamitos, CA, IEEE, pp: 63-66. (2007).
- [Tsu00] Tsutsuguchi (Ken), Sugiyama (Kazuhiro) et Sonehara (Noboru). "*The motion generation of pedestrians as avatars and crowds of people*". Dans : *Digital Cities*. pp. 275-287. Springer-Verlag, (2000).
- [Tu94] Tu X. et Terzopoulos D. "*Artificial Fishes: physics, locomotion, perception, behaviour*". In: SIGGRAPH 94 Conference Proceedings, pp: 43-50, Orlando, FL, USA, (1994).
- [Tur02] Turner, A. and Penn, A. "*Encoding Natural Movement as an Agent-based System: an Investigation into Human Pedestrian Behaviour in the Built Environment*". Environment and Planning B: Planning and Design. Pion Limited, London. 29, pp: 473-490. (2002).
- [Tur64] Turner (R.H.). *Handbook of modern sociology*, chap. Collective Behavior, pp. 382-425. Rand McNally, 1964.
- [Uli,Th01] Ulicny, B. and Thalmann, D. "*Crowd Simulation for Interactive Virtual Environments and VR Training Systems*". In. Proc. Eurographics Workshop on Animation and Simulation. Springer-Verlag. Berlin, pp: 163-170. (2001).
- [Uli02] Ulicny (B.) et Thalmann (D.). – Towards interactive real-time crowd behavior simulation. *Computer Graphics Forum*, vol. 21, n4, pp. 767-775. (2002)
- [Wol01] Tecchia, F., Loscos, C., Conroy, R. and Chrysanthou, Y. "*Agent Behavior Simulator (ABS): A Platform for Urban Behavior Development*". In. Proc. ACM/EG Games Technology Conference 55(3), pp: 601-644. (2001).
- [Yam98] Yamori (K.). "*Going with the flow: Micro-macro dynamics in the macrobehavioral patterns of pedestrian crowds*". *Psychological Review*, vol. 105, n3, pp. 530-557.(1998)
- [Yu07] Yu, Q. and Terzopoulos, D. "*A Decision Network Framework for the Behavioral Animation of Virtual Humans*". In. Proc. ACM SIGGRAPH/Eurographics, symposium on Computer animation, San Diego, California Eurographics Association, pp: 119-128. (2007).
- [Zad65] L. A. Zadeh. "*Fuzzy sets*", Information and Control, pp: 338-353, (1965).
- [Zad78] L. A. Zadeh, "*Fuzzy sets as a basis for theory of possibility*", Int. J. Fuzzy Sets Syst, vol. 1, n° 1, pp: 3-28, (1978).
- CASA10 CHATRA Mohamed, CHERIF Foudil, "Virtual crowd simulation using fuzzy

logic”; Workshop on crowd simulation, conference international CASA 2010, Rennes France.

Résumé

Ce travail s'inscrit dans le cadre de l'animation des phénomènes collectifs. Ces phénomènes sont classés dans les effets complexes, et ils sont un sujet de prédilection pour le concept d'émergence. Une grande effervescence règne aujourd'hui sur la modélisation et la simulation de ces modèles, où dans des domaines applicatifs aussi divers que l'architecture, la spécification des nuisances urbaines, le trafic autoroutier ou la biologie. La modélisation des comportements collectifs se divise en deux classes : la première classe concerne les comportements collectifs non délibératifs. Et la classe concerne les comportements collectifs coopératifs qui sont désignés par « action à plusieurs ».

Nous nous intéressons dans notre étude à la première classe qui est celui d'un ensemble d'individus ayant des buts individuels se trouvant en interaction. Apparaissent alors des évolutions structurées de ce collectif. C'est cet effet qui est strictement nommé « foule ».

Les sciences sociales ne disposent pas encore d'une typologie précise de ces formes. La simulation joue, alors, le rôle d'outil de spécification de comportements collectifs. Elle permet, ensuite, l'exploration des conditions de leur émergence. Enfin, elle permet, aussi, d'agir sur elles dans le retour au réel.

Nous proposons un modèle microscopique d'une foule d'humains virtuels. Principalement, nous abordons deux problématiques : d'une part, nous proposons un modèle basé sur un système multi-agents, et d'autre part nous utilisons les réseaux flous pour la description des comportements. Les réseaux flous permettant à l'humain virtuel d'effectuer des engagements dans un environnement virtuel inconnu. Nous avons utilisé l'outil *Massive* qui facilite les tâches de modélisation des comportements et la simulation.

Les résultats obtenus sont très satisfaisants dans le cas d'évitement de collision et l'évitement d'obstacles.